<http://www.ic.unicamp.br/~edson/disciplinas/mc404/2014-1s/ab/labs/trab03.html>

Escreva o codigo de um programa numa folha de papel a mao que salva e recupera todos os registradores, inclusive PC, SP, LR e CPSR do processo que sofreu a interrupÃ§Ã£o. Para isso,

* caso seu grupo nao tenha rodado a experiencia da aula passada, continue ateh ver os "1"s (da interrupacao) e "2"s (do programa principal) intercalados.

Terminamos na aula

* Crie uma estrutura de dados (um espaco na memoria), linhaA, onde voce deverah salvar todos os registradores r0-r12 ao entrar e sair da rotina de interrupcao. Rode e teste isso em casa.

linhaA:

.space 1024

* Aumente essa estrutura de dados para armazenar os outros registradores: PC do programa principal, LR do programa principal, SP do programa principal e CPSR do programa principal. Os outros registradores LR, SP, PC e CPSR provÃ©m de fonte diferente (veja: <http://aelmahmoudy.users.sourceforge.net/electronix/arm/chapter3.htm>)

In addition there are also five SPSRs (Saved Program Status Registers) that are loaded with the CPSR when an exception occurs. There is one SPSR for each privileged mode.

* - CPSR/supervisor está salvo em SPSR/IRQ. Saved Program Status Register do modo IRQ (Interrupt ReQuest)
* - PC/supervisor está salvo em LR/IRQ; melhor dizendo: PC/supervisor = LR/irq - 4

**Obtenha esses registradores e armazena em linhaA**. Uma ideia eh, logo ao entrar na interrupcao de relogio, salvar o PC do processo, ou seja, salvar LR/IRQ -4 (pode ser em memoria numa variavel declarada por voce), liberando LR para outros usos.

1. - LR e SP sÃo backed registeres (ver pag 1.7 da apostila Lab Manual).

**Para obter o LR e o SP do modo supervisor** use as **instrucoes que alteram o modo do processador MRS R0,CPSR** ; or **SPSR MSR CPSR,R0** ; or **SPSR**

Tome cuidado ao usar MSR e MRS pois, ao chavear de modo, vc. pode acabar sujando registradores como o spsr; alem disso, lembre-se que o CPSR contem o bit I onde zero em I habilita as interrupcoes. **Voce deverah alterar os modos para pegar LR e SP com as interrupcoes desabilitadas**.

* Reescreva o código de forma que na interrupcao de relogio, **salve todos os registradores (incluindo LR, SP, PC, CPSR) e os recupere (como se fosse o chaveamento de um processo apenas).**
* Compile e execute o cÃ³digo em casa. Observe os 1s e 2s sendo impressos.

Para salvar os registradores pensei nisso: ainda falta recuperar os registradores

do\_irq\_interrupt: @Rotina de interrupções IRQ

SUB r14, r14, #4

stmfd sp!, {r14} @ empilha o (lr\_irq - 4), pc do programa principal

e se usar aqui mrs registradordado, cpsr

msr cpsr\_c, 0xd3 @ vai para o modo supervisor

stmfd sp!, {r14} @empilha LR link register do programa principal

adr r14, linhaA @link register recebe inicio do endereco da pilha

stmfa r14!, {r0-r12, sp, spsr} @empilha todos registradores na area de dados LinhaA

mov r0, r14

ldmfd sp!, {r14} @ desempilha o Link Register

stmfa r0!, {r14} @ empilho Link Register na area de dados LinhaA

msr cpsr\_c, 0x53 @ volta para o modo de interrupcao

e se usar aqui msr cpsr\_c, 0x53

acabei de ver mrs e msr, o mrs copia o valor do cprs para o registrador dado e o msr copia o valor dado para o registrador

<http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.dui0204g/Bcfbdihi.html>

@STMFD sp!, {r0-r12} @Empilha os registradores, Não sei se precisa

LDR r0, INTPND @Carrega o registrador de status de interrupção

LDR r1, [r0]

TST r1, #0x0010 @verifica se é uma interupção de timer

BNE handler\_timer @vai para o rotina de tratamento da interupção de timer

handle r\_timer:

LDR r0, TIMER0X

MOV r1, #0x0

STR r1, [r0] @Escreve no registrador TIMER0X para limpar o pedido de interrupção

@ Inserir código que sera executado na interrupção de timer aqui (chaveamento de processos, ou alternar LED por exemplo)

bl print\_2

adr r0, linhaA @r0 recebe EstruturaDado

ldmfa r0!, {r14} @ recupera o lr do programa principal

stmfd sp!, {r14} @ empilha o lr do programa principal

mov r14, r0 @volta r0 no lr

ldmfa r14!, { r0-r12, sp, spsr } @ recupera os registradores do programa principall

ldmfd sp!, { r14 } @ recupera o lr do programa principal

ldmfa r14!, { pc }^ @ recupera o pc empilhado primeiro e volta pro modo supervisor