### 12.1 Planejamento:

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 interrupção.**

Para isso

* 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.
* 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 provém de fonte diferente (veja: <http://aelmahmoudy.users.sourceforge.net/electronix/arm/chapter3.htm>)

* - CPSR/supervisor está salvo em SPSR/IRQ.
* - 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 código em casa. Observe os 1s e 2s sendo impressos.