# O que são interrupções, qual importância tem

Interrupções de software são um tipo de interrupções causadas por uma instrução específica da ISA ou por condições excepcionais na CPU, e servem para comunicar com o kernel ou criar chamadas de sistema, e acontecem geralmente em erros ou gerenciamento de exceções.

# Quais registradores em RISC V são pertinentes

Na ISA RISC V, os registradores necessários para todo o ciclo de vida de uma interrupção são [para privilegio M]:

1. **MSTATUS**: nele, deve ser acionado o bit MIE (interrupt enable) para que interrupções sejam habilitadas; também nele estão os bits MPP e MPIE
2. **MEPC**: Registrador usado para salvar o valor de PC onde ocorreu uma exceção.
3. **MCAUSE**: Registrador de status que indica se ocorreu uma interrupção ou exceção, junto com um código para distinguir a causa da interrupção/exceção.
4. **MTVEC**: define o modo de interrupção e o endereço base da tabela de vetores de interrupção, tabela essa que possui os endereços para os handlers de interrupções de cada tipo.
   1. A tabela de vetores de interrupção serve para definir handlers específicos para interrupções, quando o modo é *vectored* ao invés de *direct*.

Mostrar tabela de endereços dos registradores

Layout do MSIP

citar que, apesar do numero de bits ser alterado quando migramos de rv32 pra rv64, o endereço não muda, ele apenas é extendido com zeros a esquerda

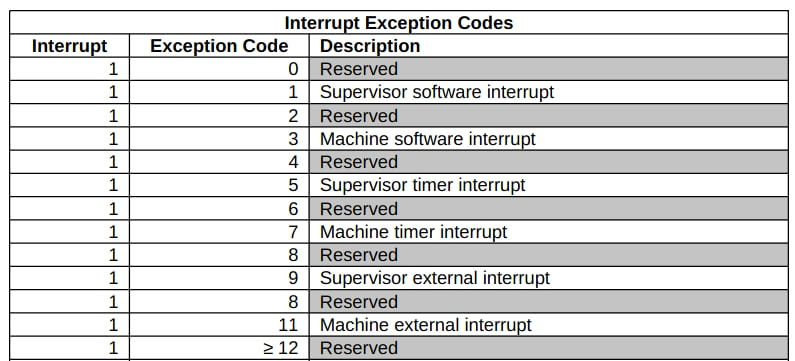
layout dos registradores relevantes da CLINT

# Ciclo de vida de uma interrupção

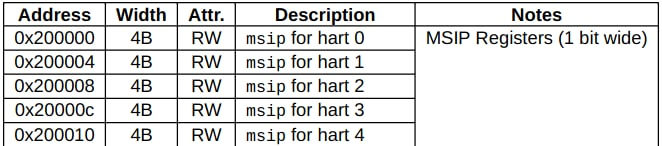
Em RISC V, podemos definir o ciclo de vida de uma interrupção da seguinte forma:

* Geração da interrupção

1. Antes de tudo, o registrador *mtvec* - que guarda o endereço base da tabela de vetores de interrupção - deve ser configurado:
   1. Se os interrupts vetorizados estiverem desabilitados (ou seja, mtvec.MODE = 0 - como é no EPOS) sempre que houver um interrupt, o PC vai ser redirecionado para mtvec.BASE (ou seja, sempre que houver uma interrupção, independente de qual for ela, o PC vai ser redirecionado pro mesmo lugar). Caso mtvec.MODE = 1, ou seja, for vectored, o PC vai ser redirecionado para BASE + 4 \* *cause*, possibilitando um handler específico para cada causa de interrupção (definido em *mcause*), onde os bits são:



1. O core que for gerar uma interrupção deverá setar um bit específico no registrador de interrupções no espaço de memória do CLINT - o MSIP - onde cada registrador cria um interrupt em um hart específico



* **Gerenciamento da interrupção**

1. Salvar o contexto do processo em execução
2. Coleta do PC interrompido no CSR *mepc*
3. Coleta do nível de privilégio (também em um CSR)
4. Setar a causa da interrupção em *mcause*
5. Desligar outras interrupções, trocando o bit de *mie*
6. Olhar o vetor de interrupções, endereçado por *mtvec*
7. Restaurar o contexto do processo que estava em execução
8. Transferir de volta o controle, setando PC

Comentar que, caso o mesmo processador tente passar duas interrupçoes sem que o processador alvo tenha avisado que lidou com a primeira (limpado o sip), a segunda interrupção é perdida. [aqui podemos fazer um teste escrevendo varias vezes seguidas no registrador e depois checando quantas interrupções foram geradas]

# Como gerar uma interrupção

## codigo

Pra evitar concorrencia, os processadores devem usar a instrucao *tsl* (atomico) pra gerar uma IPI

Deve ser limpado o bit sip no registrador que recebe a interrupção

# Como tratar uma interrupção

## Aonde/como o outro core recebe a interrupção

* Em um sistema com várias CPUs, outras CPUs podem setar msip para acionar uma interrupção de software em qualquer outra CPU do sistema. Isso permite uma eficiente comunicação entre processadores.

## O que são e pra que servem os vetores de interrupções

* Vetores de interrupção são vetores que contém instruções de salto para endereços base de cada rotina de tratamento de interrupção.
* Elas servem para desviar o fluxo de execução para um tratador de interrupção especializado para o tratamento da interrupção que ocorreu.
* ajudam a reduzir a latência no tratamento de interrupções

Referências

<https://www.techopedia.com/definition/22195/software-interrupt#:~:text=A%20software%20interrupt%20is%20invoked,during%20error%20or%20exception%20handling>.

<https://sites.google.com/site/proffernandosiqueiraso/aulas/3-concorrencia>

<https://starfivetech.com/uploads/sifive-interrupt-cookbook-v1p2.pdf>