

# Entrega 2

O que deve ser entregue? - Pasta `Entrega-1` no git - Demonstração em sala

Nessa entrega iremos ter a mesma funcionalidade que a **Entrega 1** porém com os LEDs e botões sendo acionados e lidos pelo NIOS (soft processor).

Para isso será necessário modificar o projeto criado no tutorial para possuir ao menos mais um periférico PIO (que será responsável por ler os botões). Além de adicionar esse novo periférico, nessa entrega iremos aprimorar nosso sistema com:

- Periférico JTAG deve gerar interrupção
- Periférico PIO que lida com o botão deve gerar interrupção
- Memória de programa separada da de dados

Uma vez que o JTAG começa a gerar interrupções não será mais necessário usar o *small driver* do JTAG, lembre de alterar isso no **bsp**.

Comece por ler os botões sem interrupção, uma vez que estiver funcionando, utilize os sites a seguir como referência para implementar interrupção no NIOS:

Dicas:

- <http://www.johnloomis.org/NiosII/interrupts/interrupt/interrupt.html>
- [https://www.altera.com/en\\_US/pdfs/literature/hb/nios2/n2sw\\_nii52006.pdf](https://www.altera.com/en_US/pdfs/literature/hb/nios2/n2sw_nii52006.pdf)

Rubrica:

- I
  - Não entregou nada
- D
  - Entregou somente tutorial

- C
  - Memória de dados separada da de programa
  - JTAG gerando interrupção.
  - PIO dedicado a ler botões (SWx e KEYx)
  - Adicione um novo PIO para ler o valor dos botões
  - Interrupção na leitura do botão
- B
  - Crie um driver para controlar os LEDs e Ler as chaves (criar uma biblioteca com arquivos .c e .h separados e funções), ou :
  - Adicione e faça uso de um timer (Platform Design)
- A
  - Insira um RTOS no NIOS para fazer o controle da aplicação