pio interrupção

# 2.1 – NMI vs irq

Uma interrupção não mascarável é um tipo de interrupção que não pode ser normalmente ignorada pelo sistema. Por isso, ela é usada quando o tempo de resposta é crítico, como quando há um erro irreversível de hardware ou eventos como resets do sistema.

Já o IRQ é um pedido de interrupção que para o programa que está rodando para então permitir que um handler assuma seu lugar até que ele termine, assim como vimos no código da aula 10.

# 3.1 – IRQ VS ISR

A relação existente entre o IRQ e o ISR é como se fosse a de caller-callee, no sentido que quando um IRQ ocorre, ele chama a função ISR de call-back no microcontrolador para que seja tratada a interrupção.

# 3.2 – SAME70

No ARM que utilizamos existem 72 interrupções possíveis com 8 níveis diferentes de prioridade entre elas.

# 3.3 – FIQ

O FIQ é uma interrupção de prioridade maior que qualquer IRQ. Assim, quando uma FIQ ocorre, todas IRQs são mascaradas para zero e não são chamadas. Além disso, ela possui algumas otimizações que a permitem acontecer mais rápido que uma IRQ.

# 3.4 – IRQ VS FIQ

A interrupção FIQ é a de maior prioridade.

# 3.5 – id das iterrupções

ID TC0 – 23

ID PIOA – 10

ID PIOC – 12

# 3.6 – LIMPANDO INTERRUPÇÕES

Caso não limpássemos a interrupção, o programa que estava rodando antes não iria resumir.

# 3.7 – Latência de interrupção

A latência de interrupção é o tempo que demora entre o aviso de interrupção até a execução do código relacionado a tal interrupção.

# 5.1 – PIO – INTERRUPÇÃO BOTÃO

|  |
| --- |
|  |
|  | /\* indica funcao (but\_Handler) a ser chamada quando houver uma interrupção \*/ |
|  | pio\_enable\_interrupt(pio, pin\_mask); |
|  | pio\_handler\_set(pio, pio\_id, pin\_mask, PIO\_IT\_FALL\_EDGE, (\*handler)); |
|  |  |
|  | /\* habilita interrupçcão do PIO que controla o botao \*/ |
|  | /\* e configura sua prioridade \*/ |
|  | NVIC\_EnableIRQ(pio\_id); |
|  | NVIC\_SetPriority(pio\_id, 1); |

# 5.1 – pio – interrupção

O uso da interrupção ocorre quando um periférico muda de estado. É então trabalho do NVIC perceber essa mudança e checa-la para ver se é a mudança configurada. Essas mudanças podem ser configuradas como detecção de borda ou de nível, através de registradores dedicados ao NVIC, podendo ser também de subida/alto ou descida/baixo. Assim, o NVIC só gera uma interrupção de fato quando acontecer o caso configurado. Após tratado a interrupção, é necessário avisar que ela foi tratada. Isso é feito através da leitura de status do registrador PIO\_ISR. Isso automaticamente limpa todas as interrupções.