

Computação Embarcada-Interrupção e Exceções

Felipe Frid Buniac

March 29, 2017

1 Introdução

1.1 Embarcados

2 Exceções

Questão 3.1: NMI vs IRQ

Qual a diferença entre as exceções NMI e IRQ ?

NMI:Um NMI pode ser sinalizado por um periférico ou acionado por software. Esta é a exceção de prioridade mais alta, com exceção da de redefinição. Este é permanentemente habilitado e tem uma prioridade fixa de -2. NMIs que não podem ser: Mascarado ou impedido de ativação por qualquer outra exceção diferente da de Reset.[1]

IRQ:Uma interrupção, ou IRQ, é uma exceção sinalizada por um periférico, ou gerada por uma solicitação de software. Todas as interrupções são assíncronas à execução da instrução. No sistema, os periféricos utilizam interrupções para se comunicar com o processador.[1]

3 Interrupção

Questão 3.2: IRQ vs ISR

Qual a diferença entre as exceções IRQ e ISR ?

IRQ:Uma interrupção, ou IRQ, é uma exceção sinalizada por um periférico, ou gerada por uma solicitação de software. Todas as interrupções são assíncronas à execução da instrução. No sistema, os periféricos utilizam interrupções para se comunicar com o processador.[1]

ISR:ISR é a Interruption Service Routine. Esta mostra se as exceções IRQ, FIQ ou um aborto externo estão pendente e é responsável por limpar uma fonte de interrupção.[2] [3][4]

3.1 Prioridades

Questão 3.2: SAME70

No ARM que utilizamos no curso, quantas são as interrupções suportadas e qual a sua menor prioridade ?

No ARM, são suportadas 72 interrupções com a menor prioridade sendo igual a 8. (Pg 73 [7])

Questão 3.3: FIQ

Descreva o uso do FIQ. As Fast Interrupt Requests (FIQs) são um tipo especializado de solicitação de interrupção, uma técnica padrão usada em CPUs de computadores para lidar com eventos que precisam ser processados à medida que ocorrem, como, por exemplo, receber dados de uma placa de rede ou ações de teclado ou mouse. FIQs são específicos para a arquitetura da CPU ARM, que suporta dois tipos de interrupções; FIQs para processamento de interrupções de latência rápida e de baixa latência e solicitações de interrupção (IRQs), para interrupções mais gerais.[5][6][7]

Questão 3.4: IRQ vs FIQ

No diagrama anterior, quem possui maior prioridade IRQ ou FIQ ?
Quem possui maior prioridade é a **FIQ**.

3.2 Interrupt Requests - IRQ

Questão 3.5: SAME70 identificador (ID) da interrupção dos periféricos

No datasheet, secção 13.1 informa o ID do periférico que está associado com a sua interrupção. Busque a informação e liste o ID dos seguintes periféricos :

- PIOA ID:10
- PIOB ID:11
- TC0 ID:23

3.2.1 Sinais de interrupção vindo dos periféricos

3.2.2 Input Edge/Level Interrupt

Questão 3.6: Limpando interrupção

O que aconteceria caso não limpemos a interrupção ?

Caso a interrupção não seja limpada o programa se mantém na interrupção até que ela seja limpada.

3.3 Interrupt Service Routine - ISR

Questão 3.7: Latência da interrupção.

O que é latência na resolução de uma interrupção, o que é feito nesse tempo ? (Interrupt latency).

A latência de interrupção é o tempo que decorre do momento em que uma interrupção é gerada até o momento que a fonte da interrupção é atendida. Para muitos sistemas operacionais, os dispositivos são atendidos assim que o manipulador de interrupção do dispositivo é executado. A latência de interrupção pode ser afetada pelo design do microprocessador

4 Software - CMSIS

5 PIO - Interrupção

Questão 5.1: PIO - Interrupção Botão

Qual deve ser a configuração para operarmos com interrupção no botão do kit SAME70-EK2 ?

Para que o LED central pisque ao clicar o botão, devemos setar Low-level detection.

5.0.1 Input Edge/Level Interrupt

Questão 5.2: PIO - Interrupção

Com base no texto anterior e nos diagramas de blocos descreva o uso da interrupção e suas opções. Uma interrupção é um sinal de um dispositivo que tipicamente resulta em uma troca de contextos, isto é, o processador para de fazer o que está fazendo para atender o dispositivo que pediu a interrupção. A inicialização de rotinas de software em resposta a eventos eletrônicos assíncronos são sinalizadas para o processador através de pedidos de interrupção (IRQs). O processamento da interrupção compõe uma troca de contexto para uma rotina de software especificamente escrita para tratar a interrupção. Essa rotina é chamada rotina de serviço de interrupção, ou tratador de interrupção (interrupt handler). Modos adicionais de interrupção são: Rising edge detection, Falling edge detection, Low-level detection e High-level detection.

Questão 5.3: Registradores Interrupção

Descreva as funções dos registradores :

- *PIO_IER/PIO_IDR*: A interrupção de Edge / Level é controlada pela gravação do Interrupt Enable Register (PIO_IER) e do Interrupt Disable Register (PIO_IDR), que habilitam e desabilitam a interrupção de mudança de entrada respectivamente definindo e eliminando o bit correspondente no Registro de Máscara de Interrupção (PIO_IMR) .(Pg 351 [7])
- *PIO_AIMER/PIO_AIDR*: Por padrão, uma interrupção pode ser gerada a qualquer momento que uma borda for detectada na entrada. Alguns modos de interrupção adicionais podem ser ativados / desativados gravando no Registro de Ativação de Modos de Interrupção Adicionais (PIO_AIMER) e no Modo de Interrupção Adicional (PIO_AIMDR). O estado atual desta seleção pode ser lido através do Registro de Máscara de Modos de Interrupção Adicionais (PIO_AIMMR). Estes modos adicionais são: Detecção de borda crescente, Detecção de borda de queda, Detecção de baixo nível, Detecção de alto nível. (Pg 352 [7])
- *PIO_ELSR*: O tipo de detecção de evento (edge ou level) deve ser selecionado por escrito no Edge Select Register (PIO_ESR) e Level Select Register (PIO_LSR) que selecionam, respectivamente, a detecção de borda e nível. O status atual desta seleção é acessível através do Registro de Status de Borda e Nível (Edge e Level Status Register) (PIO_ELSR). (Pg 353 [7])
- *PIO_FRLHSR*: A polaridade da detecção de eventos (rising e falling edge or high e low-level) deve ser selecionada escrevendo no Falling Edge e Low-Level Select Register (PIO_FELLSR) e Rising Edge e High Level Select Register (PIO_REHLSR) que permitem selecionar borda de descida ou subida (se a borda estiver selecionada em PIO_ELSR) ou detecção de nível alto ou baixo (se o nível estiver selecionado em PIO_ELSR). O status atual desta seleção é acessível através do Fall e Rise - Low e High Status Register (PIO_FRLHSR). (Pg 353 [7])

References

- [1] <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.dui0497a/BABBGBEC.html>
- [2] <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.dai0179b/CHDHCEDC.html>
- [3] <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ddi0337e/Babefdj.html>
- [4] <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ddi0500g/CIHDAHJG.html>
- [5] https://en.wikipedia.org/wiki/Fast_interrupt_request
- [6] <http://stackoverflow.com/questions/973933/what-is-the-difference-between-fiq-and-irq-interrupt-system>
- [7] http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-11296-32-bit-Cortex-M7-Microcontroller-SAM-E70Q-SAM-E70N-SAM-E70J_Datasheet.pdf

6 GITHUB

<https://github.com/febuniac/EmbeddedComputing>