



Documento de Casos de Uso
Processador de Propósito Geral Simples

Universidade Estadual de Feira de Santana
Build 1.1a

Histórico de Revisões

Date	Descrição	Autor(s)
03/10/2014	Document conception	manuellemacedo
07/10/2014	Adição do caso de uso da unidade lógica e aritmética	manuellemacedo

SUMÁRIO

1	Introdução	3
1.1	Objetivo	3
1.2	Visão Geral do Documento	3
1.3	Representação Simbólica	3
1.4	Definições, Acrônimos e Abreviações	4
2	Atores do Sistema	4
3	Casos de Usos	4
3.1	[UC 001] Unidade de Controle	4
3.1.1	Fluxo Principal de Eventos	8
3.1.2	Fluxo Secundário: Alternativo	8
3.2	[UC 002] Unidade Lógica Aritmética	8
3.2.1	Fluxo Principal de Eventos	8
3.2.2	Fluxo Secundário: Alternativo	9

1. Introdução

1.1. Objetivo

O objetivo desse documento é especificar os casos de uso do projeto Processador de Propósito Geral Simples. O documento contempla as seguintes informações: descrição dos Atores envolvidos no processo; definição dos fluxos de eventos principal e secundário; lista de requisitos essenciais, funcionais e não funcionais; estabelecimento de pré-condições e pós-condições.

1.2. Visão Geral do Documento

- Sessão 2: lista todos os possíveis atores do sistema.
- Sessão 3: relata a lista dos casos de uso do projeto.

1.3. Representação Simbólica

A Figura 1 ilustra a simbologia utilizada para representar operações que devem ser realizadas pelo sistema. A Figura 2 ilustra as duas simbologias utilizadas para representar os Atores do sistema. Um ator, dentro do escopo desta descrição, pode ser identificado como um módulo *top level*, ou como um elemento de entrada e saída (botões, sensores, displays, etc).

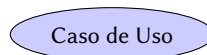


Figura 1: Exemplo de Caso de Uso.

A simbologia usual para representação de um Ator é apresentada na Figura 2a, no entanto, para representar módulos incorporados que outrora deveriam utilizar a mesma simbologia, utiliza-se a representação ilustrada nas Figuras 2b e 2c, definida por convenção. Este elemento, em geral, está associado aos módulos do sistema, ou IP-cores que de terceiros incorporados ao mesmo. Esta simbologia ainda foi dividida, tendo em vista representar instâncias únicas (Figura 2c), ou múltiplas (Figura 2b) de um determinado componente.

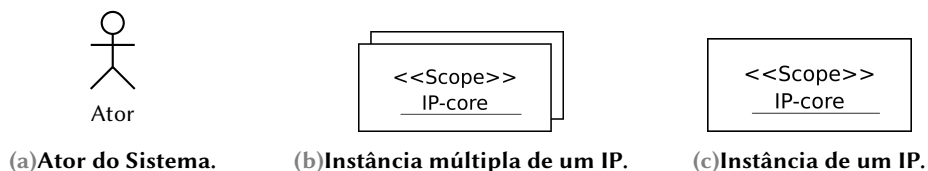


Figura 2: Simbologia utilizada na implementação dos Casos de Uso.

O projetista responsável por interpretar os diagramas não deve confundir-se no momento de interpretar as simbologias de atores. A representação alternativa, não implica que o módulo será instanciado no subsistema em questão, mas sim que os recursos providos por este *core* são necessários para garantir o seu funcionamento.

1.4. Definições, Acrônimos e Abreviações

Termo	Descrição
UC	Caso de Uso
SF	Fluxo Secundário
FR	Requisito Funcional
ID	Decodificação de Instrução
IF	Busca de Instrução
SC	Sinal de Controle
OLA	Operação Lógica Aritmética
DM	Data Memory

2. Atores do Sistema



Unidade de Controle – Unidade que controla a execução das operações.

ULA – Unidade que realiza as operações lógicas e aritméticas.

3. Casos de Usos

3.1. [UC 001] Unidade de Controle

A **Unidade de Controle** é responsável por definir a partir de uma instrução decodificada o comportamento de outros módulos, por exemplo: ULA.

Atores

Controle – Unidade que controla a execução das operações.

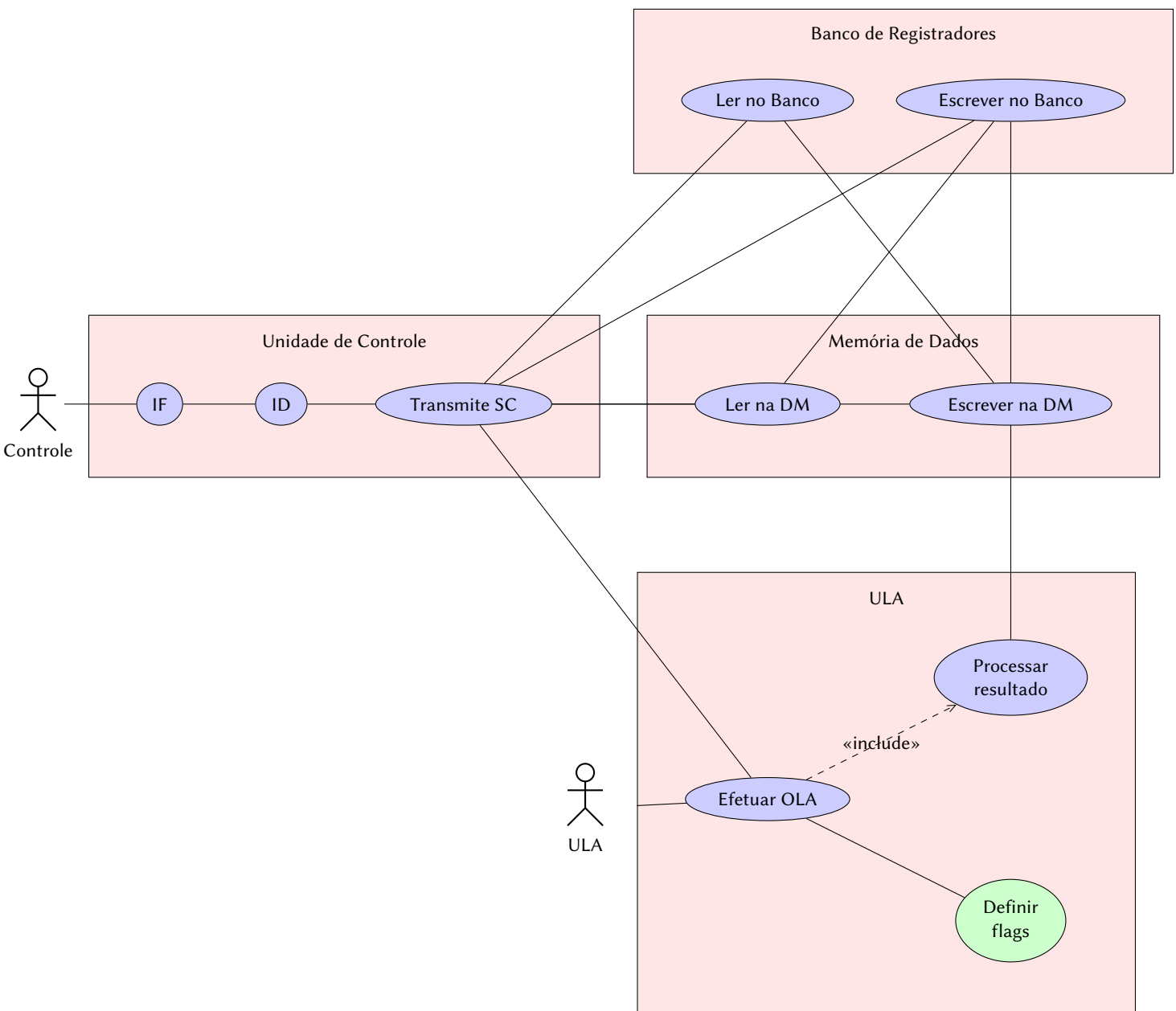
Pré-condições

- Atender aos requisitos funcionais [FR27], [FR29-30];
- Identificação das unidades sequenciais e combinacionais;

Pós-condições

- Analisar

Diagrama de Caso de Uso



3.1.1. Fluxo Principal de Eventos

- P1. Decodificar o indicador operação;
- P2. Realizar operação aritmética ou lógica;
- P3. Armazenar o resultado em um registrador;

3.1.2. Fluxo Secundário: Alternativo

[SF1] Valor do resultado excede o suportado

- 1. Habilitar sinal de overflow;

3.2. [UC 002] Unidade Lógica Aritmética

A **Unidade Lógica Aritmética** é responsável por realizar as operações aritméticas e lógicas, de acordo com o código da operação.

Atores

ULA – Unidade que realiza as operações lógicas e aritméticas.

Pré-condições

- Atender aos requisitos funcionais [FR7] e [FR30-31];
- Codificação das operações deve ser definida;
- Identificação das unidades sequenciais e combinacionais;

Pós-condições

- O módulo deve ser capaz de detectar overflow e underflow aritmético;
- O módulo deve ser capaz de ativar flags;
- O resultado deve estar armazenada no banco de registradores;

3.2.1. Fluxo Principal de Eventos

- P1. Decodificar o indicador operação;
- P2. Realizar operação aritmética ou lógica;
- P3. Armazenar o resultado no banco de registradores;

3.2.2. Fluxo Secundário: Alternativo

[SF1] Valor do resultado excede o suportado

1. Habilitar sinal de overflow ou underflow;

[SF2] Operação que ativa flags

1. Habilitar um dos sinais de flags;