ipPROCESS

Documento de Casos de Uso

Core-MUSA

Universidade Estadual de Feira de Santana

Build 3

Histórico de Revisões

Date	Descrição	Autor(s)
08/10/2014	Concepção do documento	• bezourokq; • wsbittencourt; • fmbboaventura;
13/10/2014	Build 2: Novo modelo de caso de uso	• wsbittencourt; • jadsonfirmo; • fmbboaventura;
16/10/2014	Build 3: Novo modelo de caso de uso	• wsbittencourt;
20/10/2014	Adição caso de uso LW e SW	• kelvincarmo;
23/10/2014	Revisão	• jadsonfirmo;
29/10/2014	Inclusão Casos de Uso: JPC	• di3goleite;
29/10/2014	Inclusão Casos de Uso: RET e NOP	• mtcastro;
30/10/2014	Refatoração do documento	• di3goleite;
30/10/2014	Inclusão Casos de Uso: HALT	• mtcastro;
30/10/2014	Refatoração dos fluxos e tamanho das Imagens	• Odivio Caio;

SUMÁRIO

1. Introdução

Este documento tem como objetivo a especificação dos casos de uso do projeto Core Musa (concepção de um processador simples de propósito geral). O documento detalha cada caso de uso indicando os atores, os eventos (ações) e as condições de cada caso, além dos diagramas de casos de uso.

1.1. Objetivo

1.2. Visão Geral do Documento

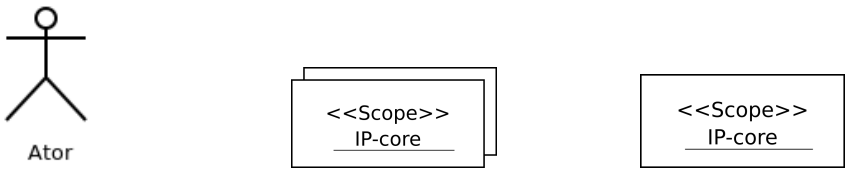
- Sessão 2: Lista todos os possíveis atores do sistema.
- Sessão 3: Relata a lista dos casos de uso do projeto.

1.3. Representação Simbólica

A Figura 1 ilustra a simbologia utilizada para representar operações que devem ser realizadas pelo sistema. A Figura 2 apresenta os modelos de ilustração utilizados 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, utiliza-se as representações ilustradas nas Figuras 2b e 2c, definidas por convenção. Este elemento, em geral, está associado aos módulos do sistema, ou IP cores de terceiros incorporados ao mesmo. Esta simbologia foi divida, com o objetivo de representar instâncias únicas (Figura 2c), ou múltiplas (Figura 2b) de um determinado componente.



(a)**Ator do Sistema.** (b)**Instância múltipla de um IP.** (c)**Instância de um IP.**

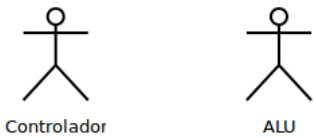
Figura 2: Simbologia utilizada na implementação dos Casos de Uso.

O projetista responsável por interpretar os diagramas não deve confundir-se no momento de analisar 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
ALU	Unidade Lógica e Aritmética
SB	Sub-fluxo
FS	Fluxo Secundário
NFR	Requisito Não Funcional
FR	Requisito Funcional
BT	Botão Direcional
PC	*Program Counter*

2. Atores do Sistema



Controlador – Unidade que controla a execução das operações.

ALU – Unidade Lógica e Aritmética.

3. Casos de Usos

Esta sessão apresenta o conjunto de UC realizados para a implementação do projeto *Core* MUSA (Núcleo de processamento de instruções do processador de propósito geral MUSA). As sessões a seguir foram divididas e nomeada utilizando a nomenclatura abreviada [UC (NÚMERO DO UC)] seguido de uma breve descrição em forma de título.

3.1. [UC 001] Execução de instruções

O controlador é responsável por decodificar instrução, solicitar operações na ALU e garantir o armazenamento dos resultados das operações no banco registradores.

Atores

Controlador – Unidade que controla a execução das operações.

ALU – Unidade Lógica e Aritmética.

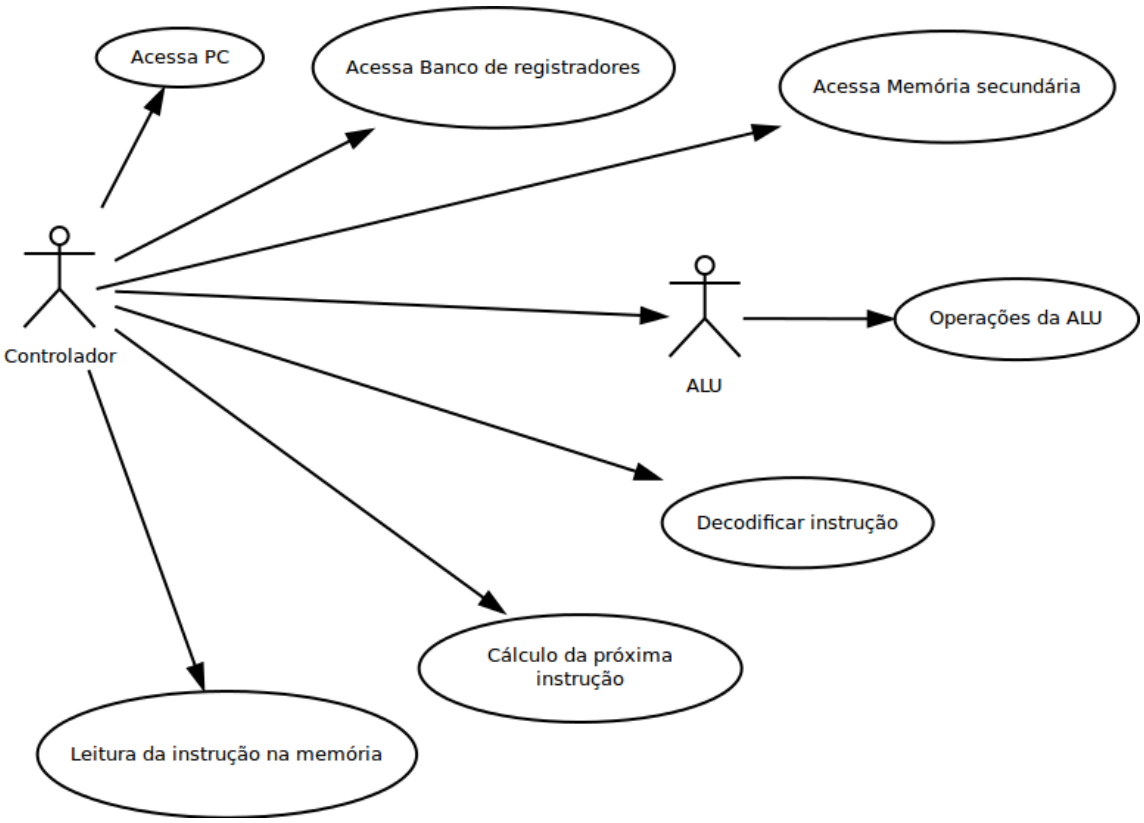
Pré-condições

- Atender aos requisitos funcionais [FR01 e FR02];
- Leitura do PC;
- Realizar operações lógicas e aritméticas na ALU;

Pós-condições

- Os resultados devem ser expressos nos registradores.

Diagrama de Caso de Uso



3.1.1. *Fluxo Principal de Eventos*

P1. Controlador faz a decodificação do OPcode da instrução recebida;

P2. Controlador comanda a execução da leitura dos registradores recebidos;

P3. A ULA executa as operação relacionada ao Function;

P4. Controlador comanda a execução da escrita do valor no registrador destino;

3.2. [UC 002] **BRFL**

O Processador tem a capacidade de fazer desvios condicionais através da utilização das flags do sistema.

Atores

Controlador – Unidade que controla a execução das operações.

ALU – Unidade Lógica e Aritmética.

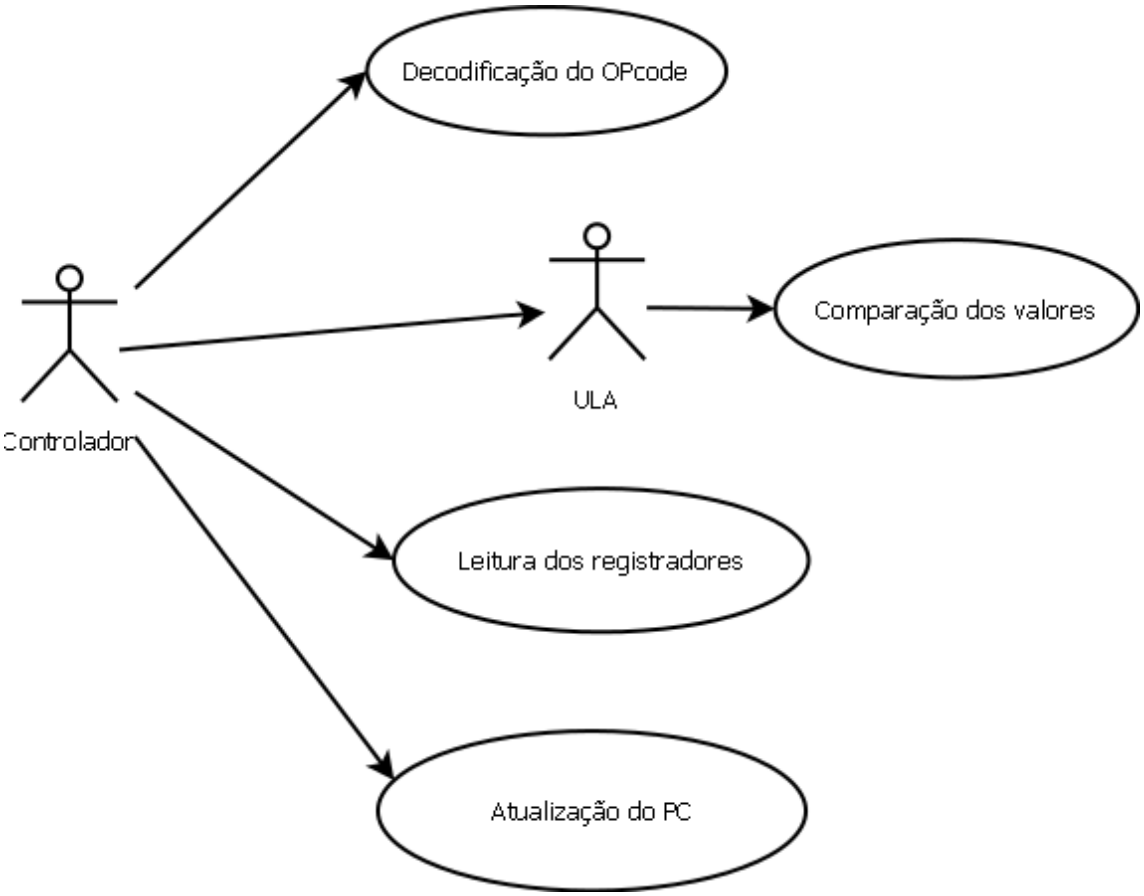
Pré-condições

- Atender ao requisito funcional [FR14];
- Leitura do PC;
- Leitura do Banco de Registrador;
- Realizar operações lógicas na ALU;

Pós-condições

- Alteração do PC caso verdadeira.

Diagrama de Caso de Uso



3.2.1. Fluxo Principal de Eventos

P1. Controlador faz a decodificação do OPcode da instrução recebida;

P2. Controlador executa a leitura da *Flag* e da constante contidas no bando de registrador;

P3. A ULA executa a operação lógica dos operandos;

P4. Controlador executa a atualização do PC dependendo do resultado;

3.3. [UC 003] Instrução LW

O processador é capaz de carregar dados da memória pro banco de registradores.

Atores

Controlador – Unidade que controla a execução das operações.

ALU – Unidade Lógica e Aritmética.

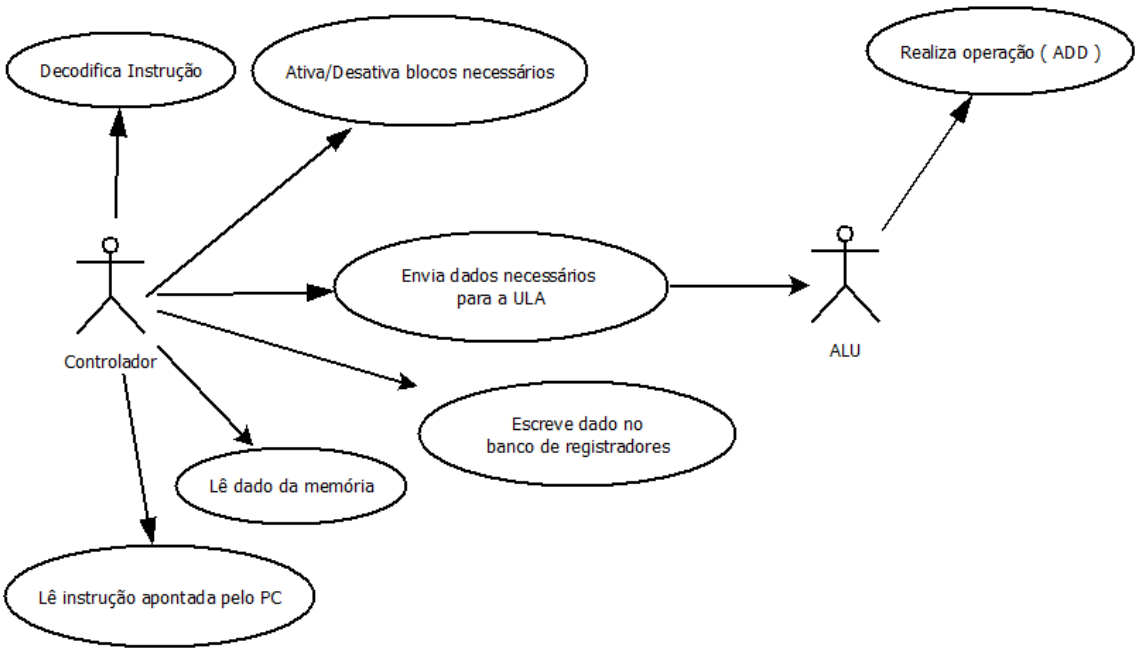
Pré-condições

- Leitura da Memória de instrução;
- A instrução necessita ter o *opcode* específico para a instrução LW;

Pós-condições

- O dado deverá ser salvo no registrador de escrita (RD) do banco de registradores;

Diagrama de Caso de Uso



3.3.1. *Fluxo Principal de Eventos*

P1. Decodificação da instrução lida na Memória de Programa;

P2. Acesso aos respectivos registradores;

P3. Leitura do endereço base, a partir do registrador RT;

P4. Extensão do valor de 16 bits lido na instrução para 32 bits;

P5. Operação de soma com os dados de 32 bits, dado lido no registrador, com o valor de 16 bits estendido para 32 bits;

P6. Resultado da operação será lido pela memória de dados, que servirá como endereço de memória para ler o dado;

P7. O dado será enviado para o banco de registradores, e será escrito no registrador de escrita RD;

3.4. [UC 004] **Instrução SW**

O processador é capaz de escrever dados na memória.

Atores

Controlador – Unidade que controla a execução das operações.

ALU – Unidade Lógica e Aritmética.

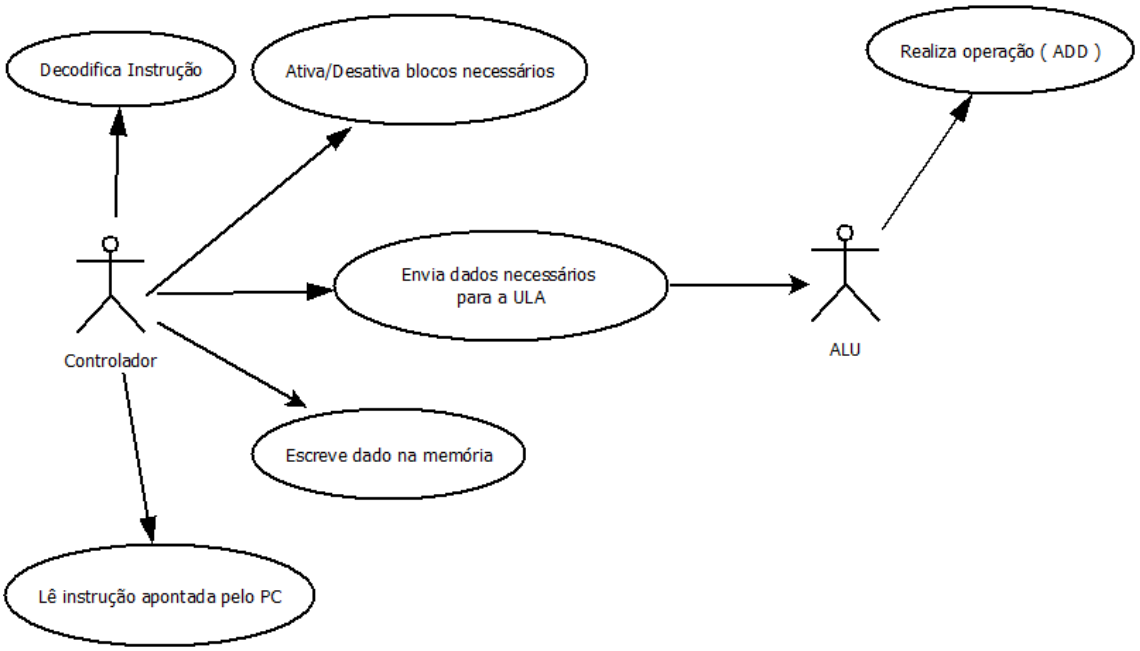
Pré-condições

- Leitura da Memória de instrução;
- A instrução necessita ter o *opcode* específico para a instrução SW;

Pós-condições

- O dado deverá ser escrito na memória de dados.

Diagrama de Caso de Uso



3.4.1. *Fluxo Principal de Eventos*

P1. Decodificação da instrução lida na Memória de Programa;

P2. Acesso aos respectivos registradores;

P3. Leitura do endereço base, a partir do registrador RT;

P4. Leitura do dado a ser escrito no registrador RS;

P5. Extensão do valor de 16 bits, lido na instrução, para 32 bits;

P6. Operação de soma com os dados de 32 bits com os seguintes valores: endereço lido do registrador RT e valor de 32 bits estendido;

P7. Resultado da operação será lido pela Memória de Dados, que servirá como endereço de memória para escrever o dado.

P8. O dado será escrito na Memória de Dados.

3.5. [UC 005] CALL

O Processador deve ser capaz de desviar um programa em execução para uma subrotina.

Atores

Controlador – Unidade que controla a execução das operações.

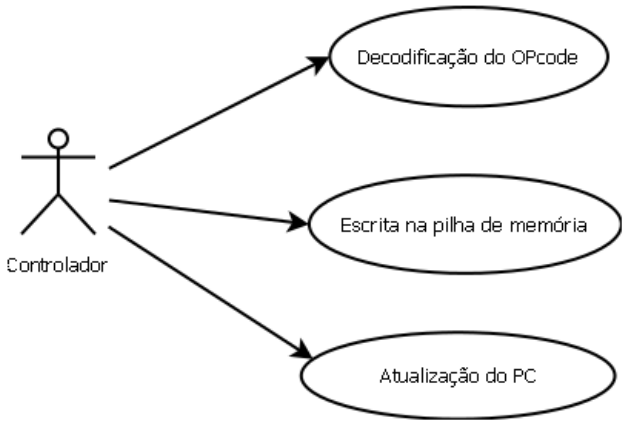
Pré-condições

- Atender ao requisito funcional [FR15];
- Leitura do PC;
- Ter espaço disponível na pilha de memória;

Pós-condições

- Alteração do PC.

Diagrama de Caso de Uso



3.5.1. Fluxo Principal de Eventos

P1. Controlador faz a decodificação do OPcode da instrução apontada pelo PC;

P2. Controlador executa a escrita do endereço atual do PC na pilha de memória;

P3. Controlador executa a atualização do PC para o endereço recebido pela instrução;

3.6. [UC 006] JR

O Processador deve ser capaz de desviar um programa em execução para um endereço de destino.

Atores

Controlador – Unidade que controla a execução das operações.

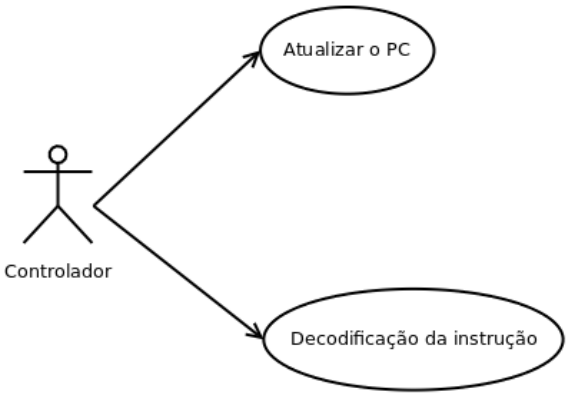
Pré-condições

- Atender ao requisito funcional [FR12];
- Leitura do PC;

Pós-condições

- Alteração do PC.

Diagrama de Caso de Uso



3.6.1. Fluxo Principal de Eventos

P1. Controlador faz a decodificação do OPcode da instrução apontada pelo PC;

P2. Controlador executa a atualização do PC para o endereço recebido pela instrução;

3.7. [UC 007] JPC

O processador deve ser capaz de desviar um programa em execução para um endereço relativo ao PC.

Atores

Controlador – Unidade que controla a execução das operações.

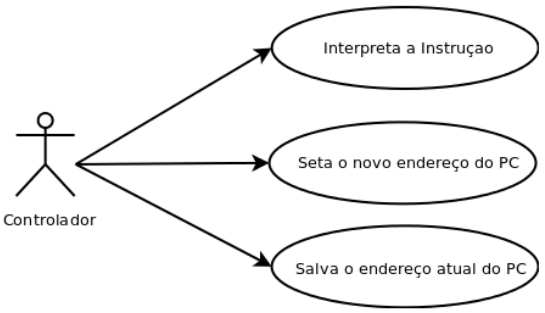
Pré-condições

- Leitura do PC;

Pós-condições

- Alteração do PC.

Diagrama de Caso de Uso



3.7.1. *Fluxo Principal de Eventos*

P1. Controlador faz a decodificação do OPcode da instrução apontada pelo PC;

P2. Controlador executa a leitura da instrução na pilha;

P3. Modifica o valor do PC pro endereço recebido na instrução;

3.8. [UC 008] **Instruções Lógicas e Aritméticas.**

O controlador é responsável por decodificar instrução, solicitar operações na ALU e garantir o armazenamento dos resultados das operações no banco registradores.

Atores

Controlador – Unidade que controla a execução das operações.

ALU – Unidade Lógica e Aritmética.

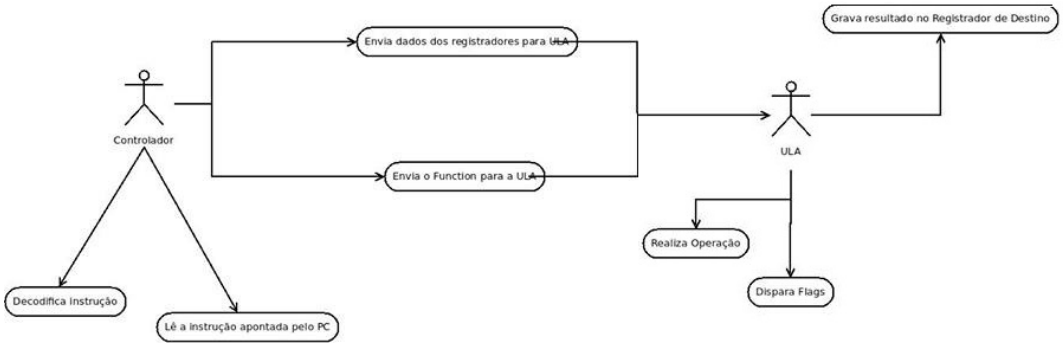
Pré-condições

- Atender aos requisitos funcionais [FR03 a FR10];
- Leitura do PC;
- Realizar operações lógicas e aritméticas na ALU;

Pós-condições

- Os resultados devem ser expressos nos registradores.

Diagrama de Caso de Uso



3.8.1. *Fluxo Principal de Eventos*

P1. Controlador faz a decodificação do OPcode da instrução apontada pelo PC;

P2. Acesso aos respectivos registradores;

P3. A ULA realiza as operações;

P4. *Flags* são disparadas, caso seja necessário;

P5. Envia o resultado pro registrador de destino;

3.9. [UC 009] **RET**

O processador deve ser capaz de retornar do último desvio tomado pelo programa.

Atores

Controlador – Unidade que controla a execução das operações.

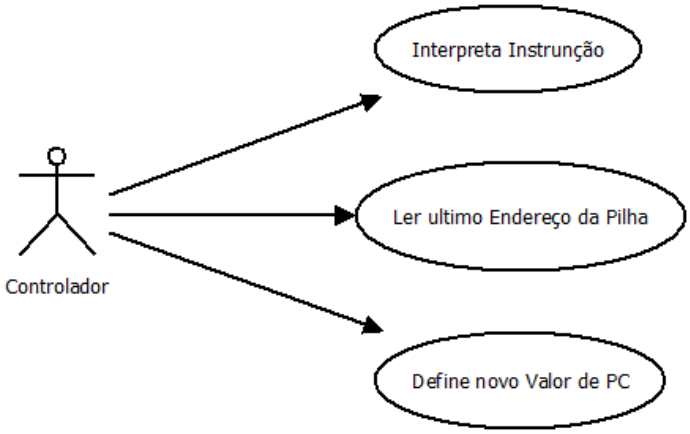
Pré-condições

- Atender ao requisito funcional [FR15];
- Leitura do PC;

Pós-condições

- Alteração do PC.

Diagrama de Caso de Uso



3.9.1. Fluxo Principal de Eventos

P1. Controlador faz a decodificação do OPcode da instrução apontada pelo PC;

P2. Leitura do endereço do topo da pilha;

P3. Remoção do endereço do topo da pilha;

P4. Incrementa o endereço removido do topo da pilha;

P5. Modifica o valor do PC pro valor incrementado;

3.10. [UC 010] NOP

O processador deve ser capaz de não realizar operações durante cinco ciclos de *clock*.

Atores

Controlador – Unidade que controla a execução das operações.

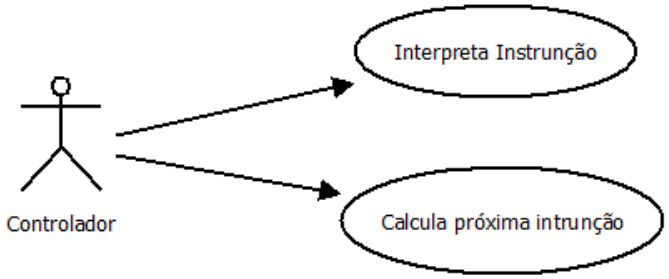
Pré-condições

- Atender ao requisito funcional [FR15];
- Leitura do PC;

Pós-condições

- Alteração do PC.

Diagrama de Caso de Uso



3.10.1. Fluxo Principal de Eventos

P1. Controlador faz a decodificação do OPcode da instrução apontada pelo PC;

P2. Modifica o valor do PC pro endereço da próxima instrução.

3.11. [UC 011] HALT

O processador deve ser capaz de encerra o programa.

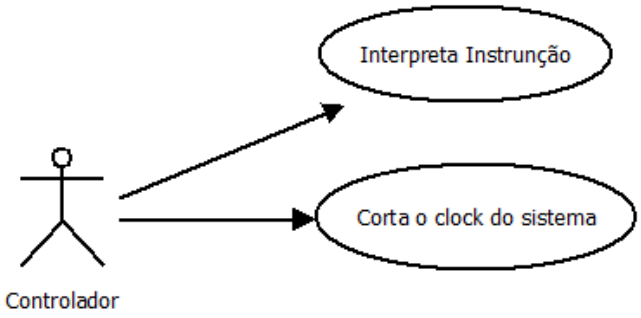
Atores

Controlador – Unidade que controla a execução das operações.

Pré-condições

- Atender ao requisito funcional [FR15];
- Leitura do PC;

Diagrama de Caso de Uso



3.11.1. Fluxo Principal de Eventos

P1. Controlador faz a decodificação do OPcode da instrução apontada pelo PC;

P2. Desliga o *clock* do processador.