



## **Especificação de Requisitos**

Processador de Propósito Geral Simples

Universidade Estadual de Feira de Santana

**Compilação 2.2**

## Histórico de Revisões

Data	Descrição	Autor(es)
26/09/2014	Inicialização do documento	patrickecomp
26/09/2014	Inclusão dos requisitos [FR1] à [FR8]	patrickecomp
28/09/2014	Inclusão dos requisitos [FR20] à [FR24]	santana22 e gabri4el
28/09/2014	Concepção da introdução e inclusão dos requisitos [FR10] e [FR11]	manuellemacedo
29/09/2014	Inclusão dos requisitos [FR17] à [FR20], Inclusão dos requisitos [NFR3] à [NFR8]	patrickecomp, mirelarios e tarles-w7
29/09/2014	Inclusão dos requisitos [FR21] e [FR22]	ardersonqdv, lucas
29/09/2014	Inclusão do requisito [FR7]	patrickecomp
30/09/2014	Refinamento da introdução, revisão preliminar do documento e revisão da [FR16]	santana22
30/09/2014	Revisão do requisito [FR13]	santana22 e gabri4el
30/09/2014	Adição do requisito [NFR1]	santana22
30/09/2014	Revisão dos requisitos [FR21] e [FR22]	lasilva
30/09/2014	Adição dos requisitos [FR23] à [FR26], adição dos requisitos [NFR2], [NFR9], [NFR10]	manuellemacedo
05/10/2014	Revisão dos requisitos [FR12] à [FR16]	santana22 e gabri4el
06/10/2014	Adição dos requisitos [FR12], [FR14] [FR2], [FR4]	manuellemacedo
06/10/2014	Adição dos requisitos [FR26] e revisão dos requisitos [NFR3], [NFR4], [NFR5] e [NFR8].	patrickecomp e mirelarios
06/10/2014	Adição dos requisitos [FR17] à [FR19]	manuellemacedo
09/10/2014	Revisão final do documento	manuellemacedo

## Histórico de Revisões

<b>Data</b>	<b>Descrição</b>	<b>Autor(es)</b>
08/10/2014	Adicionado a dependência do documento de requisitos da Processing Unit	Anderson e Lucas
08/10/2014	Mudança dos requisitos [FR29], [FR30] de NFR para FR, revisão dos requisitos [FR28], [NFR3], [NFR4] e [NFR5]	patrickecomp e mirelarios
08/10/2014	Ajustes da introdução e das dependências. Remoção do NFR sobre Interfaceamento.	santana22

## SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>3</b>
1.1	Visão Geral do Documento . . . . .	3
1.2	Definições . . . . .	3
1.3	Acrônimos e Abreviações . . . . .	4
1.4	Prioridades dos Requisitos . . . . .	4
<b>2</b>	<b>Requisitos Funcionais</b>	<b>4</b>
2.1	Conjunto de Operações Aritméticas . . . . .	4
2.2	Conjunto de Operações Lógicas . . . . .	6
2.3	Conjunto de Operações de Desvio . . . . .	7
2.4	Conjunto de Operações de Memória . . . . .	8
2.5	Conjunto de Operações de Controle . . . . .	9
<b>3</b>	<b>Requisitos não Funcionais</b>	<b>9</b>
<b>4</b>	<b>Dependências</b>	<b>11</b>

## 1. Introdução

Em um ambiente empresarial competitivo e dinâmico, possuir o domínio de técnicas que possam agilizar e minimizar as atividades de produção são necessárias para se obter êxito. Principalmente, quando o segmento de *Systems on Chip* cresce rapidamente em todo o mundo e, essa conjuntura, necessita de um rápido desenvolvimento de novas e mais acessíveis tecnologias. Dessa maneira, torna-se quase uma urgência a formação de mão de obra especializada nesse (e em outros) segmentos. Nesse contexto, a **Fazemos Qualquer Negócio Inc.** em parceria com a **Universidade Estadual de Feira de Santana**, foi contratada para o desenvolvimento de um *IP-Core* de um micro processador de propósito geral que será utilizado em escolas da África com o intuito de impulsionar o desenvolvimento deste continente.

A seguir, como parte da fase **Concepção**, este documento apresenta o levantamento de diversos requisitos identificados a partir das necessidades dos nossos clientes para produção deste *IP-Core*.

### 1.1. Visão Geral do Documento

- **Requisitos funcionais** - lista de todos os requisitos funcionais.
- **Requisitos não funcionais** - lista de todos os requisitos não funcionais.
- **Dependências** - conjunto de dependências de IP-cores previstos.

### 1.2. Definições

Termo	Descrição
Requisito Funcional	Requisitos de hardware que compõem os módulos, descrevendo as ações que o mesmo deve estar apto a executar. Estas informações são capturadas a partir do desenvolvimento dos casos de uso, que documentam as entradas, os processos e as saídas geradas.
Requisito Não Funcional	Requisitos de hardware que compõem os módulos, representando as características que o mesmo deve ter, ou restrições que o mesmo deve operar. Estas características referem-se a técnicas, algoritmos, tecnologias e especificidades do Sistema como um todo.
Dependências	Requisitos de reuso de IP-cores, descrevendo as funções que cada um deve exercer.

### 1.3. Acrônimos e Abreviações

Sigla	Descrição
FR	Requisito Funcional
NFR	Requisito Não Funcional
D	Dependência
PC	Program Counter

### 1.4. Prioridades dos Requisitos

Prioridade	Característica
Importante	Requisito sem o qual o sistema funciona, porém não como deveria.
Essencial	Requisito que deve ser implementado para que o sistema funcione.
Desejável	Requisito que não compromete o funcionamento do sistema.

## 2. Requisitos Funcionais

### 2.1. Conjunto de Operações Aritméticas

#### [FR1] Operação de soma

**Descrição:** O módulo deve ser capaz de realizar a operação de soma de dois valores de 32 bits.

**Nível de Prioridade:** Essencial

#### [FR2] Operação de soma imediata

**Descrição:** O módulo deve ser capaz de realizar a operação de soma de dois valores de 32 bits, sendo um destes valor imediato.

**Nível de Prioridade:** Essencial

#### [FR3] Operação de subtração

**Descrição:** O módulo deve ser capaz de realizar a operação de subtração de dois valores de 32 bits.

**Nível de Prioridade:** Essencial

---

**[FR4] Operação de subtração imediata**

---

**Descrição:** O módulo deve ser capaz de realizar a operação de subtração de dois valores de 32 bits, sendo um destes valor imediato.

**Nível de Prioridade:** Essencial

---

**[FR5] Operação de multiplicação**

---

**Descrição:** O módulo deve ser capaz de realizar a operação de multiplicação de dois valores de 16 bits.

**Nível de Prioridade:** Essencial

---

**[FR6] Operação de divisão**

---

**Descrição:** O módulo deve ser capaz de realizar a operação de divisão de dois valores de 32 bits.

**Nível de Prioridade:** Essencial

---

**[FR7] Tamanho da palavra de saída**

---

**Descrição:** O componente deve apresentar uma saída única de 32 bits para todas as operações aritméticas.

**Nível de Prioridade:** Importante

---

**[FR8] Detecção de overflow aritmético**

---

**Descrição:** O módulo deve ser capaz de detectar *overflow* aritmético.

**Nível de Prioridade:** Importante

---

**[FR9] Detecção de underflow aritmético**

---

**Descrição:** O módulo deve ser capaz de detectar *underflow* aritmético.

**Nível de Prioridade:** Importante

---

**[FR10] Detecção de zero**

---

**Descrição:** O módulo deve ser capaz de detectar no resultado da operação o valor zero e habilitar uma *flag*.

**Nível de Prioridade:** Importante

## 2.2. Conjunto de Operações Lógicas

### [FR11] Operação AND

---

**Descrição:** O módulo deve ser capaz de realizar a operação AND lógico de dois operandos de 32 bits.

**Nível de Prioridade:** Essencial

### [FR12] Operação ANDI

---

**Descrição:** O módulo deve ser capaz de realizar a operação AND lógico de dois operandos de 32 bits, sendo um destes valor imediato.

**Nível de Prioridade:** Essencial

### [FR13] Operação OR

---

**Descrição:** O módulo deve ser capaz de realizar a operação OR lógico de dois operandos de 32 bits.

**Nível de Prioridade:** Essencial

### [FR14] Operação ORI

---

**Descrição:** O módulo deve ser capaz de realizar a operação OR lógico de dois operandos de 32 bits, sendo um destes valor imediato.

**Nível de Prioridade:** Essencial

### [FR15] Operação NOT

---

**Descrição:** O módulo deve ser capaz de realizar a operação NOT lógico de um operando de 32 bits.

**Nível de Prioridade:** Essencial

### [FR16] Operação CMP

---

**Descrição:** O módulo deve ser capaz de comparar dois registradores de 32 bits cada e ativar um *flag*.

**Nível de Prioridade:** Essencial



---

**[FR17] Ativação do flag *above***

---

**Descrição:** O módulo deve ser capaz de ativar o flag *above* após uma comparação entre duas entradas.

**Nível de Prioridade:** Importante

---

**[FR18] Ativação do flag *equals***

---

**Descrição:** O módulo deve ser capaz de ativar o flag *equals* após uma comparação entre duas entradas.

**Nível de Prioridade:** Importante

---

**[FR19] Ativação do flag *error***

---

**Descrição:** O módulo deve ser capaz de ativar o flag *error* após um erro ser detectado.

**Nível de Prioridade:** Importante

### 2.3. Conjunto de Operações de Desvio

---

**[FR20] Operação JR**

---

**Descrição:** O módulo deve ser capaz de realizar desvios para endereços específicos na memória.

**Nível de Prioridade:** Essencial

---

**[FR21] Operação JPC**

---

**Descrição:** O módulo deve ser capaz de realizar desvios para endereços relativos ao PC.

**Nível de Prioridade:** Essencial

---

**[FR22] Operação BRFL**

---

**Descrição:** O módulo deve ser capaz de realizar desvios condicionais com base em um *flag*.

**Nível de Prioridade:** Essencial

---

**[FR23] Operação CALL**

---

**Descrição:** O módulo deve ser capaz de desviar a execução para uma sub-rotina.

**Nível de Prioridade:** Essencial

---

#### [FR24] Operação RET

**Descrição:** O módulo deve ser capaz de retornar de uma sub-rotina.

**Nível de Prioridade:** Essencial

### 2.4. Conjunto de Operações de Memória

---

#### [FR25] Operação LW

**Descrição:** O módulo deve ser capaz de realizar leitura de dados na memória de dados.

**Nível de Prioridade:** Essencial

---

#### [FR26] Operação SW

**Descrição:** O módulo deve ser capaz de armazenar dados na memória de dados.

**Nível de Prioridade:** Essencial

---

#### [FR27] Ler na memória de instruções

**Descrição:** O módulo deve ser capaz de ler instruções na memória de instruções.

**Nível de Prioridade:** Essencial

---

#### [FR28] Organização de memória

**Descrição:** O módulo deve possuir duas memórias diferentes e independentes (Arquitetura de Harvard). Sendo elas: RAM para dados e ROM para instruções.

**Nível de Prioridade:** Essencial

---

#### [FR29] Modo de endereçamento

**Descrição:** O módulo deve possuir três modos de endereçamento: imediato, por registrador e base-deslocamento.

**Nível de Prioridade: Essencial**

---

**[FR30] Tamanho da palavra de instrução**

---

**Descrição:** O tamanho da palavra de instrução será de 32 bits.

**Nível de Prioridade: Essencial**

---

**[FR31] Padrão de conversão de tamanho de palavra**

---

**Descrição:** O módulo deve converter dados de 16 bits para 32 bits utilizando o padrão zero-extend ou o padrão sign-extend.

**Nível de Prioridade: Essencial**

## **2.5. Conjunto de Operações de Controle**

---

**[FR32] Operação NOP**

---

**Descrição:** O módulo não realiza operação. *Refresh* nas memórias e incrementar o contador de programa. Reserva espaço de memória para carregar novas instruções.

**Nível de Prioridade: Essencial**

---

**[FR33] Operação HALT**

---

**Descrição:** O módulo paralisa a execução de um programa e continua a realizar operação HALT até que uma requisição de interrupção seja aceita. Não incrementa o contador de programa. *Refresh* das memórias é realizado.

**Nível de Prioridade: Essencial**

## **3. Requisitos não Funcionais**

Esta seção apresenta a lista de Requisitos não Funcionais do projeto.

---

**[NFR1] Modo de Armazenamento**

---

**Descrição:** O tratamento e o armazenamento na memória de dados será feito no formato big-endian.

**Nível de Prioridade: Importante**

---

**[NFR2] Tamanho da memória de dados**

---

**Descrição:** A memória de dados deve ser de 32 Kbytes de memória de block (M9k).

**Nível de Prioridade:** Essencial

---

**[NFR3] Tamanho da memória de instruções**

---

**Descrição:** A memória de instruções deve ser de 64 Kbytes de memória de block (M9k).

**Nível de Prioridade:** Essencial

---

**[NFR4] Tamanho do banco de registradores**

---

**Descrição:** O banco de registradores possui 32 registradores cada um com 32 bits.

**Nível de Prioridade:** Essencial

---

**[NFR5] Formato dos Números**

---

**Descrição:** Na representação dos números inteiros para a soma e subtração será adotado números com sinal e o complemento  $a_2$  para representação de valores negativos

**Nível de Prioridade:** Importante

---

**[NFR6] Arquitetura multi-ciclo**

---

**Descrição:** As instruções serão executadas em cinco ciclos de clock.

**Nível de Prioridade:** Essencial

## 4. Dependências

---

**[D1] Process Unit**

---

Módulo responsável por realizar operações lógicas e aritméticas desenvolvido no *Warm-Up*.