MIPS16bits

Generated by Doxygen 1.8.11

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

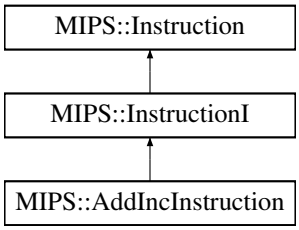
Chapter 4

Class Documentation

4.1 MIPS::AddIncInstruction Class Reference

`#include <addinc.hpp>`

Inheritance diagram for MIPS::AddIncInstruction:



Public Member Functions

- AddIncInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.1.1 Detailed Description

Classe que faz a operação de ADDINC no processador.

Author

Felipe Dias

4.1.2 Constructor & Destructor Documentation

4.1.2.1 MIPS::AddIncInstruction::AddIncInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.1.3 Member Function Documentation

4.1.3.1 bit16_t MIPS::AddIncInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

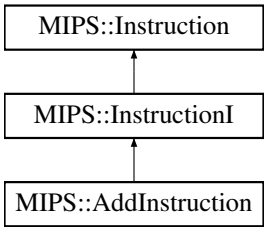
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/addinc.hpp

4.2 MIPS::AddInstruction Class Reference

`#include <add.hpp>`

Inheritance diagram for MIPS::AddInstruction:



Public Member Functions

- AddInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.2.1 Detailed Description

Classe que faz a operação de ADD no processador.

Author

Felipe Dias

4.2.2 Constructor & Destructor Documentation

4.2.2.1 MIPS::AddInstruction::AddInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.2.3 Member Function Documentation

4.2.3.1 bit16_t MIPS::AddInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

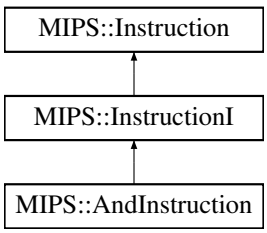
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/add.hpp

4.3 MIPS::AndInstruction Class Reference

`#include <and.hpp>`

Inheritance diagram for MIPS::AndInstruction:



Public Member Functions

- AndInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.3.1 Detailed Description

Classe que faz a operação de and no processador.

Author

Lucas Fonseca dos Santos

4.3.2 Constructor & Destructor Documentation

4.3.2.1 MIPS::AndInstruction::AndInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.3.3 Member Function Documentation

4.3.3.1 bit16_t MIPS::AndInstruction::execute ( ) [virtual]

Função que executa a operação de AndInstruction.

Implements MIPS::InstructionI.

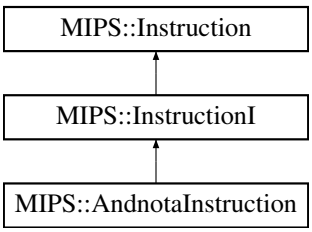
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/and.hpp

4.4 MIPS::AndnotaInstruction Class Reference

`#include <andnota.hpp>`

Inheritance diagram for MIPS::AndnotaInstruction:



Public Member Functions

- AndnotaInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.4.1 Detailed Description

Classe que faz a operação de andnota no processador.

Author

Lucas Pereira

4.4.2 Constructor & Destructor Documentation

4.4.2.1 MIPS::AndnotaInstruction::AndnotaInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.4.3 Member Function Documentation

4.4.3.1 bit16_t MIPS::AndnotaInstruction::execute ( ) [virtual]

Função que executa a operação de AndnotaInstruction.

Returns

resultado da operação

Implements MIPS::InstructionI.

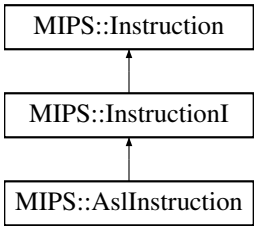
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/andnota.hpp

4.5 MIPS::AslInstruction Class Reference

```
#include <asl.hpp>
```

Inheritance diagram for MIPS::AslInstruction:



Public Member Functions

- AslInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.5.1 Detailed Description

Classe que faz a operação de ASR no processador.

Author

Felipe Dias

4.5.2 Constructor & Destructor Documentation

4.5.2.1 MIPS::AslInstruction::AslInstruction (bit8_t *opcode*, Register * *rs*, Register * *rt*, bit8_t *shamt*, bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.5.3 Member Function Documentation

4.5.3.1 bit16_t MIPS::AslInstruction::execute ( ) [virtual]

Função que executa a operação de shift a esquerda.

Returns

resultado da operação

Implements MIPS::InstructionI.

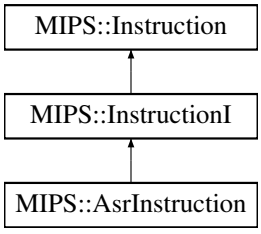
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/asl.hpp

4.6 MIPS::AsrInstruction Class Reference

`#include <asr.hpp>`

Inheritance diagram for MIPS::AsrInstruction:



Public Member Functions

- AsrInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.6.1 Detailed Description

Classe que faz a operação de ASR no processador.

Author

Felipe Dias

4.6.2 Constructor & Destructor Documentation

4.6.2.1 MIPS::AsrInstruction::AsrInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.6.3 Member Function Documentation

4.6.3.1 bit16_t MIPS::AsrInstruction::execute ( ) [virtual]

Função que executa a operação de shift a direita.

Returns

resultado da operação

Implements MIPS::InstructionI.

The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/asr.hpp

4.7 MIPS::ControlUnit Class Reference

`#include <control.hpp>`

Public Member Functions

- ControlUnit ()
- ~ControlUnit ()

4.7.1 Detailed Description

Unidade de controle do processador. Essa unidade é responsável por setar todas as flags que serão utilizadas para executar uma instrução.

Author

Matheus Nogueira

4.7.2 Constructor & Destructor Documentation

4.7.2.1 MIPS::ControlUnit::ControlUnit ( )

Cria uma nova unidade de controle.

4.7.2.2 MIPS::ControlUnit::∼ControlUnit ( )

Destroi a unidade de controle.

The documentation for this class was generated from the following file:

- include/mips/units/control.hpp

4.8 MIPS::CPU Class Reference

`#include <cpu.hpp>`

Public Member Functions

- CPU ()
- ∼CPU ()
- void loadProgram (const char ∗program)
- void execute ()

4.8.1 Detailed Description

Classe que representa o processador m-RISC, esta é responsável por gerenciar toda a execução das instruções no processador.

Author

Matheus Nogueira

4.8.2 Constructor & Destructor Documentation

4.8.2.1 MIPS::CPU::CPU ( )

Cria uma nova CPU.

4.8.2.2 MIPS::CPU::~CPU ( )

Destroi a CPU.

4.8.3 Member Function Documentation

4.8.3.1 void MIPS::CPU::execute ( )

Executa as instruções carregadas previamente pelo método loadProgram.

4.8.3.2 void MIPS::CPU::loadProgram (const char * *program*)

Carrega um programa na memória de instruções do processador.

Parameters

program	caminho para o arquivo contendo o programa.

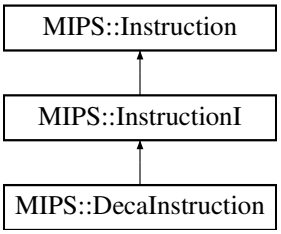
The documentation for this class was generated from the following file:

- include/mips/cpu.hpp

4.9 MIPS::DecaInstruction Class Reference

`#include <deca.hpp>`

Inheritance diagram for MIPS::DecaInstruction:



Public Member Functions

- DecaInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.9.1 Detailed Description

Classe que faz a operação de DECA no processador.

Author

Felipe Dias

4.9.2 Constructor & Destructor Documentation

4.9.2.1 MIPS::DecaInstruction::DecaInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.9.3 Member Function Documentation

4.9.3.1 bit16_t MIPS::DecaInstruction::execute ( ) [virtual]

Função que executa a operação de decremento.

Returns

resultado da operação

Implements MIPS::InstructionI.

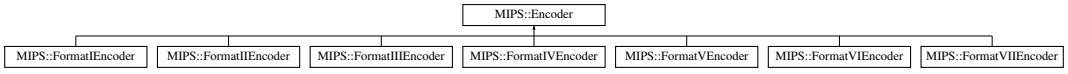
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/deca.hpp

4.10 MIPS::Encoder Class Reference

`#include <encoder.hpp>`

Inheritance diagram for MIPS::Encoder:



Public Member Functions

- Encoder ()
- virtual ~Encoder ()
- virtual instruction_t encode ()=0
- virtual void parse (std::vector< char * > &params)=0

Protected Member Functions

- bit8_t getRegisterNumber (const char *name)

Protected Attributes

- bit8_t opcode

4.10.1 Detailed Description

Classe abstrata que cria uma interface para todos os codificadores de instruções MIPS 32.

Author

Matheus Nogueira

4.10.2 Constructor & Destructor Documentation

4.10.2.1 MIPS::Encoder::Encoder ( )

Cria um novo codificador.

Parameters

labels	tabela de labels extraídos do código.
type	tipo de codificador.

4.10.2.2 virtual MIPS::Encoder::~Encoder ( ) [inline], [virtual]

Destroi o codificador.

4.10.3 Member Function Documentation

4.10.3.1 virtual instruction_t MIPS::Encoder::encode ( ) [pure virtual]

Codifica a ultima instrução que foi analisada pelo parser do codificador.

Returns

instrução 16 bits.

Implemented in MIPS::FormatIEncoder, MIPS::FormatIIEncoder, MIPS::FormatIIIEncoder, MIPS::FormatIV↩Encoder, MIPS::FormatVEncoder, MIPS::FormatVIEncoder, and MIPS::FormatVIIEncoder.

4.10.3.2 bit8_t MIPS::Encoder::getRegisterNumber (const char * *name*) [protected]

Retorna o número do registrador solicitado.

Parameters

name	nome do registrador.

Returns

número do registrador

4.10.3.3 virtual void MIPS::Encoder::parse (std::vector< char * > & *params*) [pure virtual]

Percorre a instrução em assembly e extraí os dados dela para poder montar uma instrução binária.

Parameters

params	parametros da instrução.

Implemented in MIPS::FormatIEncoder, MIPS::FormatIIEncoder, MIPS::FormatIIIEncoder, MIPS::FormatIV↩Encoder, MIPS::FormatVEncoder, MIPS::FormatVIEncoder, and MIPS::FormatVIIEncoder.

4.10.4 Member Data Documentation

4.10.4.1 bit8_t MIPS::Encoder::opcode [protected]

Opcode da instrução.

The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/encoder.hpp

4.11 MIPS::EncoderFactory Class Reference

`#include <encoder_factory.hpp>`

Public Member Functions

- EncoderFactory ()
- Encoder * produce (const char *instruction)

4.11.1 Detailed Description

Fábrica responsável por referenciar os codificadores corretos para cada instrução da linguagem de montagem do micro-RISC.

Author

Matheus Nogueira

4.11.2 Constructor & Destructor Documentation

4.11.2.1 MIPS::EncoderFactory::EncoderFactory ( ) [inline]

Cria uma nova fábrica de codificadores.

4.11.3 Member Function Documentation

4.11.3.1 Encoder* MIPS::EncoderFactory::produce (const char * *instruction*)

Produz um codificador para determinada instrução.

Parameters

instruction	nome da instrução que deve ser codificada.

The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/encoder_factory.hpp

4.12 MIPS::Event Class Reference

```
#include <event.hpp>
```

Public Member Functions

- Event (EventType type, void *data, bool autodestroy=false)
- virtual ~Event ()
- bool shouldDestroy ()

Public Attributes

- const EventType event_type
- void * data_ptr

4.12.1 Detailed Description

Classe responsável por representar um evento que pode ser despachado pelo despachante de eventos do emulador.

Author

Matheus Nogueira

4.12.2 Constructor & Destructor Documentation

4.12.2.1 MIPS::Event::Event (EventType *type*, void * *data*, bool *autodestroy* = `false`)

Cria um novo evento.

Parameters

type	tipo do evento
data	dados associados ao evento. Esses dados associados devem ser alocados de forma dinâmica pelo programador, utilizando a função malloc. O despachante de eventos é responsável por destruir esses dados após seu uso caso a flag autodestroy seja ativada.
autodestroy	indica que os dados do evento devem ser destruídos após que o evento seja despachado para todos seus ouvintes.

4.12.2.2 virtual MIPS::Event::~Event ( ) `[virtual]`

Destroi o evento e os dados relacionados ao mesmo.

4.12.3 Member Function Documentation

4.12.3.1 bool MIPS::Event::shouldDestroy ( )

Verifica se o evento deve se auto destruir.

Returns

true se o evento deve se auto destruir.

4.12.4 Member Data Documentation

4.12.4.1 void* MIPS::Event::data_ptr

Ponteiro para os dados do evento.

4.12.4.2 const EventType MIPS::Event::event_type

Tipo de evento.

The documentation for this class was generated from the following file:

- include/mips/util/event/event.hpp

4.13 MIPS::EventDispatcher Class Reference

`#include <event_dispatcher.hpp>`

Classes

- struct ListenerMap

Public Member Functions

- EventDispatcher ()
- virtual ~EventDispatcher ()
- void dispatch (Event &event)
- void addEventListener (EventListener *listener, EventType event)

4.13.1 Detailed Description

Classe responsável por permitir que componentes do emulador possam se comunicar por troca de mensagens transmitidas por eventos.

Author

Matheus Nogueira

4.13.2 Constructor & Destructor Documentation

4.13.2.1 MIPS::EventDispatcher::EventDispatcher ( )

Cria um novo despachante de eventos.

4.13.2.2 virtual MIPS::EventDispatcher::~EventDispatcher ( ) [virtual]

Destroi o despachante de eventos.

4.13.3 Member Function Documentation

4.13.3.1 void MIPS::EventDispatcher::addEventListener (EventListener * *listener,* EventType *event*)

Adiciona um ouvinte de eventos nesse despachante.

Parameters

listener	ouvinte de eventos
event	tipo de evento que o ouvinte irá escutar.

4.13.3.2 void MIPS::EventDispatcher::dispatch (Event & *event*)

Despacha um evento para todos os seus ouvintes. Caso o evento esteja marcado com a flag autodestroy, o evento será destruido logo após a chamada desse método.

Parameters

event	evento a ser despachado.

The documentation for this class was generated from the following file:

- include/mips/util/event/event_dispatcher.hpp

4.14 MIPS::EventListener Class Reference

`#include <event_listener.hpp>`

Public Member Functions

- virtual void notify (Event &event)

4.14.1 Detailed Description

Classe abstrata que permite um objeto ouvir eventos vindos de outra classe.

Author

Matheus Nogueira

4.14.2 Member Function Documentation

4.14.2.1 virtual void MIPS::EventListener::notify (Event & *event*) [virtual]

Função utilizada para notificar o objeto que algum evento o qual ele está esperando, ocorreu.

Parameters

event	evento que ocorreu.

The documentation for this class was generated from the following file:

- include/mips/util/event/event_listener.hpp

4.15 MIPS::FileReader Class Reference

`#include <file_reader.hpp>`

Public Member Functions

- FileReader (const char *filename)
- FileReader (const char *filename, Filter &filter)
- ~FileReader ()
- char * next ()
- void rewind ()
- bool hasNext ()

4.15.1 Detailed Description

Classe responsável por ler o conteúdo de um arquivo e salvá-lo internamente, fazendo com que seja simples para obter as informações do mesmo.

Author

Matheus Nogueira

4.15.2 Constructor & Destructor Documentation

4.15.2.1 MIPS::FileReader::FileReader (const char * *filename*)

Cria um novo leitor de arquivo.

Parameters

filename	nome do arquivo a ser lido.

4.15.2.2 MIPS::FileReader::FileReader (const char * *filename,* Filter & *filter*)

Cria um novo leitor de arquivo que filtra determinadas entradas.

Parameters

filename	nome do arquivo a ser lido.
filter	filtro a ser utilizado.

4.15.2.3 MIPS::FileReader::~FileReader ( )

Destroi o leitor de arquivo e o conteúdo do buffer interno.

4.15.3 Member Function Documentation

4.15.3.1 bool MIPS::FileReader::hasNext ( )

Checa se o arquivo ainda tem conteúdo para ser lido.

Returns

booleano que se true, indica que ainda existe linhas a ser lidas.

4.15.3.2 char* MIPS::FileReader::next ( )

Retorna a proxima linha do arquivo.

Returns

proxima linha do arquivo.

4.15.3.3 void MIPS::FileReader::rewind ( )

Retorna o ponteiro da linha atual para o inicio do arquivo.

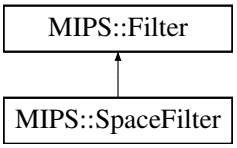
The documentation for this class was generated from the following file:

- include/mips/util/file_reader.hpp

4.16 MIPS::Filter Class Reference

`#include <filter.hpp>`

Inheritance diagram for MIPS::Filter:



Public Member Functions

- virtual std::string filter (std::string &input)=0

4.16.1 Detailed Description

Classe abstrata que representa um filtro de texto.

Author

Matheus Nogueira

4.16.2 Member Function Documentation

4.16.2.1 virtual std::string MIPS::Filter::filter (std::string & *input*) [pure virtual]

Filtra uma string e retorna uma nova instância da mesma.

Parameters

input	string a ser filtrada.

Returns

string filtrada.

Implemented in MIPS::SpaceFilter.

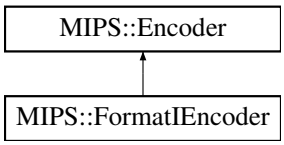
The documentation for this class was generated from the following file:

- include/mips/util/filter/filter.hpp

4.17 MIPS::FormatIEncoder Class Reference

`#include <format_I_encoder.hpp>`

Inheritance diagram for MIPS::FormatIEncoder:



Public Member Functions

- instruction_t encode ()
- void parse (std::vector< char * > &params)

Additional Inherited Members

4.17.1 Detailed Description

Codificador responsável por transformar uma instrução assembly em uma instrução binária, seguindo o formato de instrução no formato I.

Author

Matheus Nogueira

4.17.2 Member Function Documentation

4.17.2.1 instruction_t MIPS::FormatIEncoder::encode ( ) [virtual]

Codifica uma instrução assembly do formato I para uma instrução binária.

Returns

instrução binária de 16 bits

Implements MIPS::Encoder.

4.17.2.2 void MIPS::FormatIEncoder::parse (std::vector< char * > & *params*) [virtual]

Realiza uma varredura na instrução assembly e define seus campos binários.

Parameters

params	vector de parâmetros da instrução

Implements MIPS::Encoder.

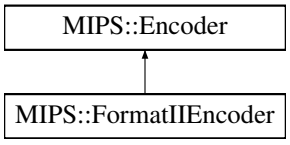
The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/format_I_encoder.hpp

4.18 MIPS::FormatIIEncoder Class Reference

`#include <format_II_encoder.hpp>`

Inheritance diagram for MIPS::FormatIIEncoder:



Public Member Functions

- instruction_t encode ()
- void parse (std::vector< char * > &params)

Additional Inherited Members

4.18.1 Detailed Description

Codificador responsável por transformar uma instrução assembly em uma instrução binária, seguindo o formato de instrução no formato II.

Author

Matheus Nogueira

4.18.2 Member Function Documentation

4.18.2.1 instruction_t MIPS::FormatIIEncoder::encode ( ) [virtual]

Codifica uma instrução assembly do formato II para uma instrução binária.

Returns

instrução binária de 16 bits

Implements MIPS::Encoder.

4.18.2.2 void MIPS::FormatIIEncoder::parse (std::vector< char * > & *params*) [virtual]

Realiza uma varredura na instrução assembly e define seus campos binários.

Parameters

params	vector de parâmetros da instrução

Implements MIPS::Encoder.

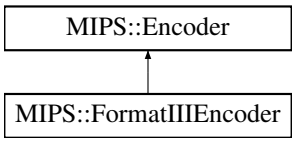
The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/format_II_encoder.hpp

4.19 MIPS::FormatIIIEncoder Class Reference

`#include <format_III_encoder.hpp>`

Inheritance diagram for MIPS::FormatIIIEncoder:



Public Member Functions

- instruction_t encode ()
- void parse (std::vector< char * > &params)

Additional Inherited Members

4.19.1 Detailed Description

Codificador responsável por transformar uma instrução assembly em uma instrução binária, seguindo o formato de instrução no formato III.

Author

Matheus Nogueira

4.19.2 Member Function Documentation

4.19.2.1 instruction_t MIPS::FormatIIIEncoder::encode ( ) [virtual]

Codifica uma instrução assembly do formato III para uma instrução binária.

Returns

instrução binária de 16 bits

Implements MIPS::Encoder.

4.19.2.2 void MIPS::FormatIIIEncoder::parse (std::vector< char * > & *params*) [virtual]

Realiza uma varredura na instrução assembly e define seus campos binários.

Parameters

params	vector de parâmetros da instrução

Implements MIPS::Encoder.

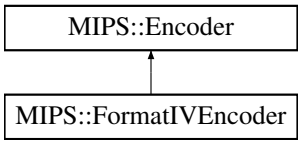
The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/format_III_encoder.hpp

4.20 MIPS::FormatIVEncoder Class Reference

`#include <format_IV_encoder.hpp>`

Inheritance diagram for MIPS::FormatIVEncoder:



Public Member Functions

- instruction_t encode ()
- void parse (std::vector< char * > &params)

Additional Inherited Members

4.20.1 Detailed Description

Codificador responsável por transformar uma instrução assembly em uma instrução binária, seguindo o formato de instrução no formato IV.

Author

Matheus Nogueira

4.20.2 Member Function Documentation

4.20.2.1 instruction_t MIPS::FormatIVEncoder::encode ( ) [virtual]

Codifica uma instrução assembly do formato IV para uma instrução binária.

Returns

instrução binária de 16 bits

Implements MIPS::Encoder.

4.20.2.2 void MIPS::FormatIVEncoder::parse (std::vector< char * > & *params*) [virtual]

Realiza uma varredura na instrução assembly e define seus campos binários.

Parameters

params	vector de parâmetros da instrução

Implements MIPS::Encoder.

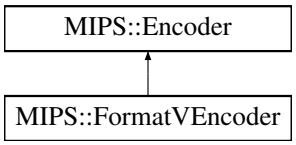
The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/format_IV_encoder.hpp

4.21 MIPS::FormatVEncoder Class Reference

`#include <format_V_encoder.hpp>`

Inheritance diagram for MIPS::FormatVEncoder:



Public Member Functions

- instruction_t encode ()
- void parse (std::vector< char * > &params)

Additional Inherited Members

4.21.1 Detailed Description

Codificador responsável por transformar uma instrução assembly em uma instrução binária, seguindo o formato de instrução no formato V.

Author

Matheus Nogueira

4.21.2 Member Function Documentation

4.21.2.1 instruction_t MIPS::FormatVEncoder::encode ( ) [virtual]

Codifica uma instrução assembly do formato V para uma instrução binária.

Returns

instrução binária de 16 bits

Implements MIPS::Encoder.

4.21.2.2 void MIPS::FormatVEncoder::parse (std::vector< char * > & *params*) [virtual]

Realiza uma varredura na instrução assembly e define seus campos binários.

Parameters

params	vector de parâmetros da instrução

Implements MIPS::Encoder.

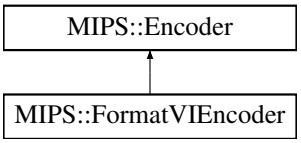
The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/format_V_encoder.hpp

4.22 MIPS::FormatVIEncoder Class Reference

`#include <format_VI_encoder.hpp>`

Inheritance diagram for MIPS::FormatVIEncoder:



Public Member Functions

- instruction_t encode ()
- void parse (std::vector< char * > &params)

Additional Inherited Members

4.22.1 Detailed Description

Codificador responsável por transformar uma instrução assembly em uma instrução binária, seguindo o formato de instrução no formato VI.

Author

Matheus Nogueira

4.22.2 Member Function Documentation

4.22.2.1 instruction_t MIPS::FormatVIEncoder::encode ( ) [virtual]

Codifica uma instrução assembly do formato V para uma instrução binária.

Returns

instrução binária de 16 bits

Implements MIPS::Encoder.

4.22.2.2 void MIPS::FormatVIEncoder::parse (std::vector< char * > & *params*) [virtual]

Realiza uma varredura na instrução assembly e define seus campos binários.

Parameters

params	vector de parâmetros da instrução

Implements MIPS::Encoder.

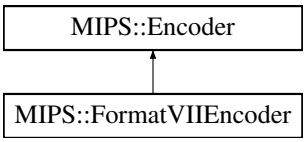
The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/format_VI_encoder.hpp

4.23 MIPS::FormatVIIEncoder Class Reference

`#include <format_VII_encoder.hpp>`

Inheritance diagram for MIPS::FormatVIIEncoder:



Public Member Functions

- instruction_t encode ()
- void parse (std::vector< char * > &params)

Additional Inherited Members

4.23.1 Detailed Description

Codificador responsável por transformar uma instrução assembly em uma instrução binária, seguindo o formato de instrução no formato VII.

Author

Matheus Nogueira

4.23.2 Member Function Documentation

4.23.2.1 instruction_t MIPS::FormatVIIEncoder::encode ( ) [virtual]

Codifica uma instrução assembly do formato V para uma instrução binária.

Returns

instrução binária de 16 bits

Implements MIPS::Encoder.

4.23.2.2 void MIPS::FormatVIIEncoder::parse (std::vector< char * > & *params*) [virtual]

Realiza uma varredura na instrução assembly e define seus campos binários.

Parameters

params	vector de parâmetros da instrução

Implements MIPS::Encoder.

The documentation for this class was generated from the following file:

- include/mips/interpreter/encoder/format_VII_encoder.hpp

4.24 MIPS::FullAdder Class Reference

`#include <full_adder.hpp>`

Public Member Functions

- FullAdder ()
- bit16_t add (bit16_t a, bit16_t b, bit8_t c=0)
- bool overflow ()

4.24.1 Detailed Description

Classe responsável por realizar as operações de um somador de 16 bits.

Author

Matheus Nogueira

4.24.2 Constructor & Destructor Documentation

4.24.2.1 MIPS::FullAdder::FullAdder ( )

Cria um novo somador.

4.24.3 Member Function Documentation

4.24.3.1 bit16_t MIPS::FullAdder::add (bit16_t *a*, bit16_t *b*, bit8_t *c* = 0)

Soma dois números de 16 bits.

Parameters

a	primeiro parametro da soma
b	segundo parametro da soma
c	carry de entrada (padrão: 0)

Returns

resultado da soma entre a e b

4.24.3.2 bool MIPS::FullAdder::overflow ( )

Verifica se houve overflow na operação de adição.

Returns

true se houve overflow.

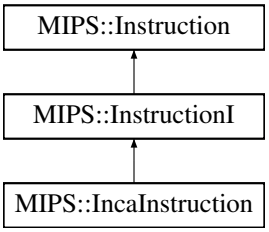
The documentation for this class was generated from the following file:

- include/mips/circuits/full_adder.hpp

4.25 MIPS::IncaInstruction Class Reference

`#include <inca.hpp>`

Inheritance diagram for MIPS::IncaInstruction:



Public Member Functions

- IncaInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.25.1 Detailed Description

Classe que faz a operação de INCA no processador.

Author

Felipe Dias

4.25.2 Constructor & Destructor Documentation

4.25.2.1 MIPS::IncaInstruction::IncaInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.25.3 Member Function Documentation

4.25.3.1 bit16_t MIPS::IncaInstruction::execute ( ) [virtual]

Função que executa a operação de incremento.

Returns

resultado da operação

Implements MIPS::InstructionI.

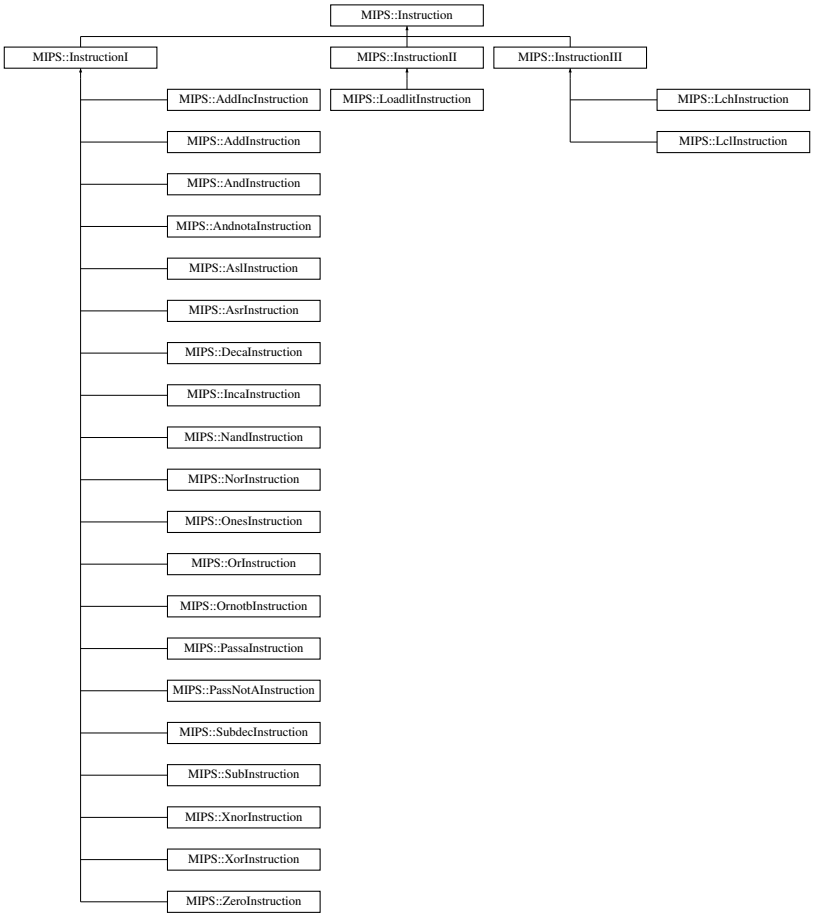
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/inca.hpp

4.26 MIPS::Instruction Class Reference

`#include <instruction.hpp>`

Inheritance diagram for MIPS::Instruction:



Public Member Functions

- ~Instruction ()
- virtual bit16_t execute ()=0

Protected Attributes

- bit8_t opcode
- bit8_t zero
- bit8_t neg
- bit8_t carry
- bit8_t overflow

4.26.1 Detailed Description

Classe abstrata responsável por representar qualquer instrução em uma arquitetura de 16 bits.

Author

Matheus Nogueira

4.26.2 Constructor & Destructor Documentation

4.26.2.1 MIPS::Instruction::~Instruction ( ) [inline]

Destroi a instrução.

4.26.3 Member Function Documentation

4.26.3.1 virtual bit16_t MIPS::Instruction::execute ( ) [pure virtual]

Método abstrato que deverá ser invocado para que uma instrução seja executada pelo emulador.

Parameters

resultado	de saída da instrução.

Implemented in MIPS::InstructionI, MIPS::InstructionII, MIPS::InstructionIII, MIPS::LclInstruction, MIPS::LoadlitInstruction, MIPS::LchInstruction, MIPS::AddInstruction, MIPS::AddIncInstruction, MIPS::AndnotaInstruction, MIPS::AslInstruction, MIPS::AsrInstruction, MIPS::DecaInstruction, MIPS::IncaInstruction, MIPS::NandInstruction, MIPS::NorInstruction, MIPS::OnesInstruction, MIPS::PassaInstruction, MIPS::PassNotAInstruction, MIPS::SubInstruction, MIPS::XnorInstruction, MIPS::XorInstruction, MIPS::ZeroInstruction, MIPS::SubdecInstruction, MIPS::AndInstruction, MIPS::OrInstruction, and MIPS::OrnotbInstruction.

4.26.4 Member Data Documentation

4.26.4.1 bit8_t MIPS::Instruction::carry [protected]

Flag de carry.

4.26.4.2 bit8_t MIPS::Instruction::neg [protected]

Flag de negativo

4.26.4.3 bit8_t MIPS::Instruction::opcode [protected]

Código da operação (opcode) da instrução.

4.26.4.4 bit8_t MIPS::Instruction::overflow [protected]

flag de overflow

4.26.4.5 bit8_t MIPS::Instruction::zero [protected]

Flag de zero.

The documentation for this class was generated from the following file:

- include/mips/instructions/instruction.hpp

4.27 MIPS::InstructionDecoder Class Reference

`#include <instruction_decoder.hpp>`

Public Member Functions

- InstructionDecoder (RegisterBank &bank)
- ~InstructionDecoder ()
- Instruction * decode (instruction_t instruction)
- bit8_t getOPCode (instruction_t instruction)
- bit8_t getRs (instruction_t instruction)
- bit8_t getRt (instruction_t instruction)
- bit8_t getRd (instruction_t instruction)
- bit8_t getFunct (instruction_t instruction)
- bit16_t getOffset (instruction_t instruction, bit8_t size=8)

Protected Attributes

- RegisterBank & registerBank

4.27.1 Detailed Description

Classe responsável por receber uma instrução em binário e instanciar uma instrução do emulador que pode executar a instrução equivalente.

Author

Matheus Nogueira

4.27.2 Constructor & Destructor Documentation

4.27.2.1 MIPS::InstructionDecoder::InstructionDecoder (RegisterBank & *bank*)

Cria um novo decodificador de instruções.

Parameters

bank	banco de registradores usado.

4.27.2.2 MIPS::InstructionDecoder::~InstructionDecoder ( )

Destroi o decodificador de instruções.

4.27.3 Member Function Documentation

4.27.3.1 Instruction* MIPS::InstructionDecoder::decode (instruction_t *instruction*)

Decodifica uma instrução em binário e cria uma instrução do emulador que realize a operação equivalente.

Parameters

instruction	instrução 16 bits em binário.

Returns

ponteiro para a instrução criada pelo emulador.

4.27.3.2 bit8_t MIPS::InstructionDecoder::getFunct (instruction_t *instruction*)

Função que recupera o valor do funct da instrução.

Parameters

instruction	instrução binária de 16 bits.

Returns

valor do funct

4.27.3.3 bit16_t MIPS::InstructionDecoder::getOffset (instruction_t *instruction*, bit8_t *size* = 8)

Função que recupera o valor do offset da instrução.

Parameters

instruction	instrução binária de 16 bits.
size	número de bits de offset

Returns

valor do offset.

4.27.3.4 bit8_t MIPS::InstructionDecoder::getOPCode (instruction_t *instruction*)

Método responsável por recuperar o código de operação (opcode) de uma instrução.

Parameters

instruction	instrução de onde o opcode deve ser extraído.

Returns

opcode da instrução.

4.27.3.5 bit8_t MIPS::InstructionDecoder::getRd (instruction_t *instruction*)

Função que recupera o endereço do registrador destination (Rd) da instrução.

Parameters

instruction	instrução binária de 16 bits.

Returns

endereço do registrador destination.

4.27.3.6 bit8_t MIPS::InstructionDecoder::getRs (instruction_t *instruction*)

Função que recupera o endereço do registrador source (Rs) da instrução.

Parameters

instruction	instrução binária de 16 bits.

Returns

endereço do registrador source.

4.27.3.7 bit8_t MIPS::InstructionDecoder::getRt (instruction_t *instruction*)

Função que recupera o endereço do registrador target (Rt) da instrução.

Parameters

instruction	instrução binária de 16 bits.

Returns

endereço do registrador target.

4.27.4 Member Data Documentation

4.27.4.1 RegisterBank& MIPS::InstructionDecoder::registerBank [protected]

Banco de registradores do decodificador.

The documentation for this class was generated from the following file:

- include/mips/decoder/instruction_decoder.hpp

4.28 MIPS::InstructionFinder Class Reference

`#include <instruction_finder.hpp>`

Public Member Functions

- InstructionFinder (Memory &memoryUnit, RegisterBank &bank)
- ~InstructionFinder ()
- instruction_t getNext ()

4.28.1 Detailed Description

Classe que representa a unidade de busca de instruções na memória.

Author

Matheus Nogueira

4.28.2 Constructor & Destructor Documentation

4.28.2.1 MIPS::InstructionFinder::InstructionFinder (Memory & *memoryUnit,* RegisterBank & *bank*)

Cria uma nova unidade para busca de instruções.

4.28.2.2 MIPS::InstructionFinder::~InstructionFinder ( )

Destroi a unidade de busca de instruções.

4.28.3 Member Function Documentation

4.28.3.1 instruction_t MIPS::InstructionFinder::getNext ( )

Retorna a próxima instrução a ser executada, utilizando o registrador PC para identificá-la.

Returns

próxima instrução a ser executada.

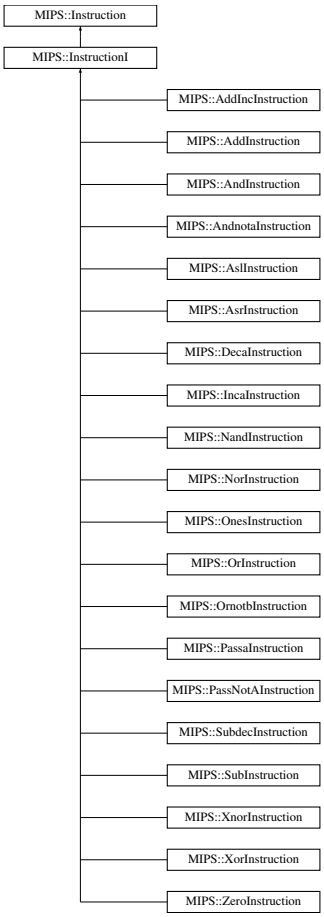
The documentation for this class was generated from the following file:

- include/mips/units/instruction_finder.hpp

4.29 MIPS::InstructionI Class Reference

```
#include <instruction_I.hpp>
```

Inheritance diagram for MIPS::InstructionI:



Public Member Functions

- InstructionI (bit8_t opcode, Register ∗rs, Register ∗rt, bit8_t shamt, bit8_t funct)
- virtual ∼InstructionI ()
- virtual bit16_t execute ()=0

Protected Attributes

- Register ∗ rs
- Register ∗ rt
- bit8_t shamt
- bit8_t funct

4.29.1 Detailed Description

Classe que representa uma instrução do tipo (R)egister.

Author

Matheus Nogueira

4.29.2 Constructor & Destructor Documentation

4.29.2.1 MIPS::InstructionI::InstructionI (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Cria uma nova instrução do formato I.

Parameters

opcode	codigo da operação
rs	registrador source
rt	registrador target
rd	registrador destination
shamt	quantidade de shift
funct	bits para escolha da função da instrução

4.29.2.2 virtual MIPS::InstructionI::~InstructionI ( ) [inline], [virtual]

Destroi a instrução.

4.29.3 Member Function Documentation

4.29.3.1 virtual bit16_t MIPS::InstructionI::execute ( ) [pure virtual]

Executa a instrução.

Returns

resultado da instrução

Implements MIPS::Instruction.

Implemented in MIPS::AddInstruction, MIPS::AddIncInstruction, MIPS::AndnotaInstruction, MIPS::AslInstruction, MIPS::AsrInstruction, MIPS::DecaInstruction, MIPS::IncaInstruction, MIPS::NandInstruction, MIPS::NorInstruction, MIPS::OnesInstruction, MIPS::PassaInstruction, MIPS::PassNotAInstruction, MIPS::SubInstruction, MIPS::Xnor↩Instruction, MIPS::XorInstruction, MIPS::ZeroInstruction, MIPS::SubdecInstruction, MIPS::AndInstruction, MIPS↩::OrInstruction, and MIPS::OrnotbInstruction.

4.29.4 Member Data Documentation

4.29.4.1 bit8_t MIPS::InstructionI::funct [protected]

Valor da funct da instrução.

4.29.4.2 Register* MIPS::InstructionI::rs [protected]

Registrador source (Rs) da instrução.

4.29.4.3 Register* MIPS::InstructionI::rt [protected]

Registrador target (Rt) da instrução.

4.29.4.4 bit8_t MIPS::InstructionI::shamt [protected]

Valor do shamt (shift amount) da instrução.

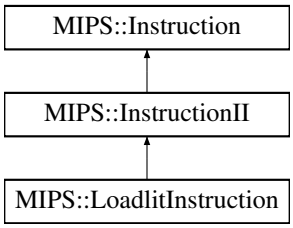
The documentation for this class was generated from the following file:

- include/mips/instructions/instruction_I.hpp

4.30 MIPS::InstructionII Class Reference

`#include <instruction_II.hpp>`

Inheritance diagram for MIPS::InstructionII:



Public Member Functions

- InstructionII (bit8_t opcode, Register *rd, bit16_t offset)
- virtual bit16_t execute ()=0

Protected Attributes

- Register * rd
- bit16_t offset

4.30.1 Detailed Description

Classe que representa uma instrução do formato II do trabalho.

Author

Matheus Nogueira

4.30.2 Constructor & Destructor Documentation

4.30.2.1 MIPS::InstructionII::InstructionII (bit8_t *opcode*, Register * *rd*, bit16_t *offset*) [inline]

Cria uma nova instrução do formato II

Parameters

opcode	código da operação
rd	registrador de destino
offset	offset de 11 bits.

4.30.3 Member Function Documentation

4.30.3.1 virtual bit16_t MIPS::InstructionII::execute ( ) [pure virtual]

Executa a instrução.

Returns

resultado da instrução

Implements MIPS::Instruction.

Implemented in MIPS::LoadlitInstruction.

4.30.4 Member Data Documentation

4.30.4.1 bit16_t MIPS::InstructionII::offset [protected]

Offset utilizado pela instrução.

4.30.4.2 Register* MIPS::InstructionII::rd [protected]

Registrador de destino da instrução.

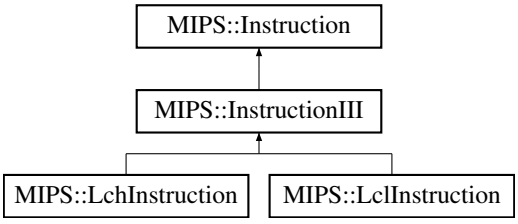
The documentation for this class was generated from the following file:

- include/mips/instructions/instruction_II.hpp

4.31 MIPS::InstructionIII Class Reference

`#include <instruction_III.hpp>`

Inheritance diagram for MIPS::InstructionIII:



Public Member Functions

- InstructionIII (bit8_t opcode, Register *rd, bit8_t offset)
- virtual bit16_t execute ()=0

Protected Attributes

- Register * rd
- bit8_t offset

4.31.1 Detailed Description

Classe que representa uma instrução do formato III do trabalho.

Author

Matheus Nogueira

4.31.2 Constructor & Destructor Documentation

4.31.2.1 MIPS::InstructionIII::InstructionIII (bit8_t *opcode*, Register * *rd*, bit8_t *offset*) [inline]

Cria uma nova instrução do formato III

Parameters

opcode	código da operação
rd	registrador de destino
offset	offset de 8 bits

4.31.3 Member Function Documentation

4.31.3.1 virtual bit16_t MIPS::InstructionIII::execute ( ) [pure virtual]

Executa a instrução.

Returns

resultado da instrução

Implements MIPS::Instruction.

Implemented in MIPS::LclInstruction, and MIPS::LchInstruction.

4.31.4 Member Data Documentation

4.31.4.1 bit8_t MIPS::InstructionIII::offset [protected]

Offset utilizado pela instrução.

4.31.4.2 Register* MIPS::InstructionIII::rd [protected]

Registrador de destino da instrução.

The documentation for this class was generated from the following file:

- include/mips/instructions/instruction_III.hpp

4.32 MIPS::Interpreter Class Reference

`#include <interpreter.hpp>`

Public Member Functions

- Interpreter (const char *file)
- ~Interpreter ()
- void compile (const char *output="out.mips")
- bool ok ()

4.32.1 Detailed Description

Classe responsável por receber um texto e criar instruções 16 bits correspondentes para a arquitetura MIPS 32.

Author

Matheus Nogueira

4.32.2 Constructor & Destructor Documentation

4.32.2.1 MIPS::Interpreter::Interpreter (const char * *file*)

Cria uma nova instância do interpretador.

Parameters

file	arquivo que será interpretado.

4.32.2.2 MIPS::Interpreter::∼Interpreter ( )

Destroi o interpretador.

4.32.3 Member Function Documentation

4.32.3.1 void MIPS::Interpreter::compile (const char * *output* = "out.mips")

Processa o arquivo de entrada para que ele possa ser interpretado.

Parameters

output	arquivo que será escrito com as instruções.

4.32.3.2 bool MIPS::Interpreter::ok ( ) [inline]

Checa se o interpretador encontrou algum erro durante a sua execução. Se sim, retorna false.

Returns

true se o interpretador realizou sem papel sem erros.

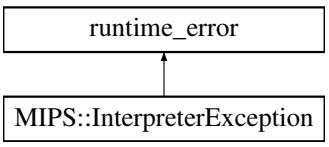
The documentation for this class was generated from the following file:

- include/mips/interpreter/interpreter.hpp

4.33 MIPS::InterpreterException Class Reference

`#include <interpreter_exception.hpp>`

Inheritance diagram for MIPS::InterpreterException:



Public Member Functions

- InterpreterException (const char *msg, bit8_t opcode)
- virtual const char * what () const throw ()
- bit8_t getCode ()

4.33.1 Detailed Description

Exceção que é lançada pelo interpretador quando um erro é encontrado no arquivo fonte.

Author

Matheus Nogueira

4.33.2 Constructor & Destructor Documentation

4.33.2.1 MIPS::InterpreterException::InterpreterException (const char * *msg*, bit8_t *opcode*) [inline]

Cria uma nova exceção.

Parameters

msg	mensagem de erro.
opcode	código do erro.

4.33.3 Member Function Documentation

4.33.3.1 bit8_t MIPS::InterpreterException::getCode ( ) [inline]

Retorna o codigo de operação da exceção.

Returns

codigo de operação.

4.33.3.2 virtual const char* MIPS::InterpreterException::what ( ) const throw) [inline], [virtual]

Retorna a mensagem de erro.

Returns

mensagem de erro.

The documentation for this class was generated from the following file:

- include/mips/interpreter/exception/interpreter_exception.hpp

4.34 MIPS::Label Struct Reference

`#include <label.hpp>`

Public Attributes

- char **label** [64]
- unsigned long **line**

4.34.1 Detailed Description

Estrutura que armazena o nome do label e a linha que ele se encontra.

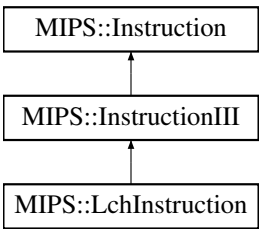
The documentation for this struct was generated from the following file:

- include/mips/interpreter/label.hpp

4.35 MIPS::LchInstruction Class Reference

`#include <lch.hpp>`

Inheritance diagram for MIPS::LchInstruction:



Public Member Functions

- LchInstruction (bit8_t opcode, Register *rd, bit8_t offset)
- bit16_t execute ()

Additional Inherited Members

4.35.1 Detailed Description

Instrução utilizada para carregar 8 bits e carregá-los em um no bit mais significativo do registrador definido pelo programador.

Author

Matheus Nogueira

4.35.2 Constructor & Destructor Documentation

4.35.2.1 MIPS::LchInstruction::LchInstruction (bit8_t *opcode*, Register * *rd*, bit8_t *offset*) [inline]

Cria uma instrução de lhc.

Parameters

opcode	código da operação
offset	offset de 11 bits

4.35.3 Member Function Documentation

4.35.3.1 bit16_t MIPS::LchInstruction::execute ( ) [virtual]

Executa a instrução.

Returns

resultado da operação

Implements MIPS::InstructionIII.

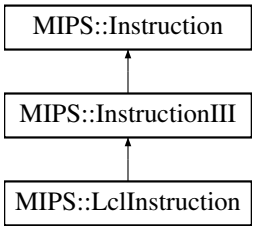
The documentation for this class was generated from the following file:

- include/mips/instructions/format_III/lch.hpp

4.36 MIPS::LclInstruction Class Reference

`#include <lcl.hpp>`

Inheritance diagram for MIPS::LclInstruction:



Public Member Functions

- LclInstruction (bit8_t opcode, Register *rd, bit8_t offset)
- bit16_t execute ()

Additional Inherited Members

4.36.1 Detailed Description

Instrução utilizada para carregar 8 bits e carregá-los em um no bit menos significativo do registrador definido pelo programador.

Author

Matheus Nogueira

4.36.2 Constructor & Destructor Documentation

4.36.2.1 MIPS::LclInstruction::LclInstruction (bit8_t *opcode,* Register * *rd,* bit8_t *offset*) [inline]

Cria uma instrução de lhc.

Parameters

opcode	código da operação
rd	registrador de destino
offset	offset de 11 bits

4.36.3 Member Function Documentation

4.36.3.1 bit16_t MIPS::LclInstruction::execute ( ) [virtual]

Executa a instrução.

Returns

resultado da operação

Implements MIPS::InstructionIII.

The documentation for this class was generated from the following file:

- include/mips/instructions/format_III/lcl.hpp

4.37 MIPS::EventDispatcher::ListenerMap Struct Reference

`#include <event_dispatcher.hpp>`

Public Attributes

- EventType type

  Tipo de evento.
- Queue< EventListener * > * listeners

  Fila de ouvintes.

4.37.1 Detailed Description

Classe responsável por representar um tipo de evento ligado a uma fila de ouvintes que devem ser notificados.

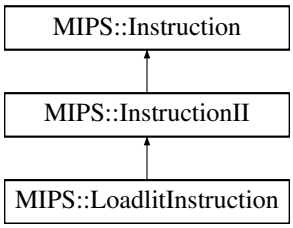
The documentation for this struct was generated from the following file:

- include/mips/util/event/event_dispatcher.hpp

4.38 MIPS::LoadlitInstruction Class Reference

`#include <loadlit.hpp>`

Inheritance diagram for MIPS::LoadlitInstruction:



Public Member Functions

- LoadlitInstruction (bit8_t opcode, Register *rd, bit16_t offset)
- bit16_t execute ()

Additional Inherited Members

4.38.1 Detailed Description

Instrução utilizada para carregar 11 bits e carregá-los em um registrador definido pelo programador.

Author

Matheus Nogueira

4.38.2 Constructor & Destructor Documentation

4.38.2.1 MIPS::LoadlitInstruction::LoadlitInstruction (bit8_t *opcode*, Register * *rd*, bit16_t *offset*) [inline]

Cria uma instrução de loadlit.

Parameters

opcode	código da operação
rd	registrador de destino
offset	offset de 11 bits

4.38.3 Member Function Documentation

4.38.3.1 bit16_t MIPS::LoadlitInstruction::execute ( ) [virtual]

Executa a instrução.

Returns

resultado da operação

Implements MIPS::InstructionII.

The documentation for this class was generated from the following file:

- include/mips/instructions/format_II/loadlit.hpp

4.39 MIPS::Memory Class Reference

`#include <memory.hpp>`

Public Member Functions

- Memory ()
- ~Memory ()
- void setInstructionSize (size_t size)
- void setDataSize (size_t size)
- void write (bit16_t data, bit32_t offset, bit8_t iOrD=1)
- bit16_t read (bit32_t offset, bit8_t iOrD=1)

4.39.1 Detailed Description

Classe responsável por iteragir com a memória do processador, para assim, criar uma interface de maior facilidade para acessar a memória.

Author

Matheus Nogueira

4.39.2 Constructor & Destructor Documentation

4.39.2.1 MIPS::Memory::Memory ( )

Cria uma nova unidade de memória.

4.39.2.2 MIPS::Memory::~Memory ( )

Destroi a unidade de memória.

4.39.3 Member Function Documentation

4.39.3.1 bit16_t MIPS::Memory::read (bit32_t *offset*, bit8_t *iOrD* = `1`)

L uma palavra que está na posição de memória especificada.

Parameters

offset	posição da memória que será lida.
type	tipo de dado que será lido (instruçao ou dado) (padrão: dado)

Returns

palavra armazenada na posição de memória especificada.

4.39.3.2 void MIPS::Memory::setDataSize (size_t *size*)

Define o tamanho da memória de dados.

Parameters

size	tamanho da memória em número de palavras.

4.39.3.3 void MIPS::Memory::setInstructionSize (size_t *size*)

Define o tamanho da memória de instruções.

Parameters

size	tamanho da memória em número de palavras.

4.39.3.4 void MIPS::Memory::write (bit16_t *data*, bit32_t *offset*, bit8_t *iOrD* = `1`)

Escreve uma palavra na posição de memória especificada.

Parameters

data	palavra que será escrita.
offset	posição da memória em que a palavra será escrita.
type	tipo de dado que será escrito (instrução ou dados) (padrão: dado)

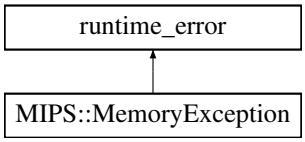
The documentation for this class was generated from the following file:

- include/mips/memory/memory.hpp

4.40 MIPS::MemoryException Class Reference

`#include <memory_exception.hpp>`

Inheritance diagram for MIPS::MemoryException:



Public Member Functions

- MemoryException (const char ∗msg)
- virtual const char ∗ what () const throw ()

4.40.1 Detailed Description

Exceção que é lançada pelo interpretador quando um erro é encontrado durante o acesso à memória.

Author

Matheus Nogueira

4.40.2 Constructor & Destructor Documentation

4.40.2.1 MIPS::MemoryException::MemoryException (const char ∗ *msg*) [inline]

Cria uma nova exceção.

Parameters

msg	mensagem de erro.

4.40.3 Member Function Documentation

4.40.3.1 virtual const char∗ MIPS::MemoryException::what ( ) const throw) [inline], [virtual]

Retorna a mensagem de erro.

Returns

mensagem de erro.

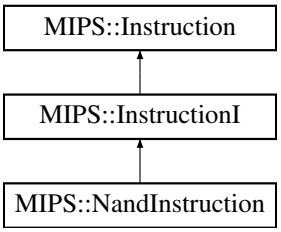
The documentation for this class was generated from the following file:

- include/mips/memory/memory_exception.hpp

4.41 MIPS::NandInstruction Class Reference

`#include <nand.hpp>`

Inheritance diagram for MIPS::NandInstruction:



Public Member Functions

- NandInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.41.1 Detailed Description

Classe que faz a operação de NAND no processador.

Author

Matheus Nogueira

4.41.2 Constructor & Destructor Documentation

4.41.2.1 MIPS::NandInstruction::NandInstruction (bit8_t *opcode*, Register * *rs*, Register * *rt*, bit8_t *shamt*, bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.41.3 Member Function Documentation

4.41.3.1 bit16_t MIPS::NandInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/nand.hpp

4.42 MIPS::Queue< T >::Node Struct Reference

`#include <queue.hpp>`

Public Attributes

- T **content**
- struct Node * **next**

4.42.1 Detailed Description

template<typename T>
struct MIPS::Queue< T >::Node

Nó da fila.

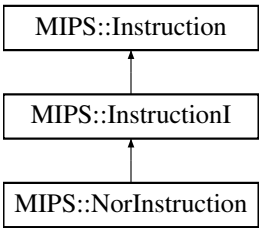
The documentation for this struct was generated from the following file:

- include/mips/util/structure/queue.hpp

4.43 MIPS::NorInstruction Class Reference

`#include <nor.hpp>`

Inheritance diagram for MIPS::NorInstruction:



Public Member Functions

- NorInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.43.1 Detailed Description

Classe que faz a operação de NOR no processador.

Author

Felipe Dias

4.43.2 Constructor & Destructor Documentation

4.43.2.1 MIPS::NorInstruction::NorInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.43.3 Member Function Documentation

4.43.3.1 bit16_t MIPS::NorInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

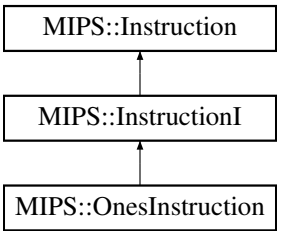
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/nor.hpp

4.44 MIPS::OnesInstruction Class Reference

`#include <ones.hpp>`

Inheritance diagram for MIPS::OnesInstruction:



Public Member Functions

- OnesInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.44.1 Detailed Description

Classe que faz a operação de ONES no processador.

Author

Felipe Dias

4.44.2 Constructor & Destructor Documentation

4.44.2.1 MIPS::OnesInstruction::OnesInstruction (bit8_t *opcode*, Register * *rs*, Register * *rt*, bit8_t *shamt*, bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.44.3 Member Function Documentation

4.44.3.1 bit16_t MIPS::OnesInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

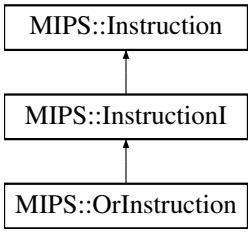
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/ones.hpp

4.45 MIPS::OrInstruction Class Reference

`#include <or.hpp>`

Inheritance diagram for MIPS::OrInstruction:



Public Member Functions

- OrInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.45.1 Detailed Description

Classe que faz a operação de OR no processador.

Author

Lucas Fonseca dos Santos

4.45.2 Constructor & Destructor Documentation

4.45.2.1 MIPS::OrInstruction::OrInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.45.3 Member Function Documentation

4.45.3.1 bit16_t MIPS::OrInstruction::execute ( ) [virtual]

Função que executa a operação de OrInstruction.

Implements MIPS::InstructionI.

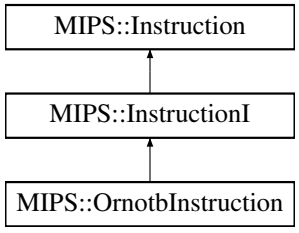
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/or.hpp

4.46 MIPS::OrnotbInstruction Class Reference

`#include <ornotb.hpp>`

Inheritance diagram for MIPS::OrnotbInstruction:



Public Member Functions

- OrnotbInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.46.1 Detailed Description

Classe que faz a operação de ornotb no processador.

Author

Lucas Pereira

4.46.2 Constructor & Destructor Documentation

4.46.2.1 MIPS::OrnotbInstruction::OrnotbInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.46.3 Member Function Documentation

4.46.3.1 bit16_t MIPS::OrnotbInstruction::execute ( ) [virtual]

Função que executa a operação de OrnotbInstruction.

Implements MIPS::InstructionI.

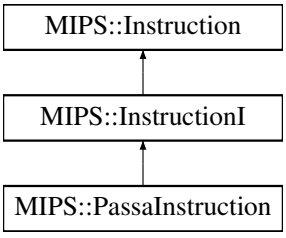
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/ornotb.hpp

4.47 MIPS::PassaInstruction Class Reference

```
#include <passa.hpp>
```

Inheritance diagram for MIPS::PassaInstruction:



Public Member Functions

- PassaInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.47.1 Detailed Description

Classe que faz a operação de PASSA no processador.

Author

Matheus Nogueira

4.47.2 Constructor & Destructor Documentation

4.47.2.1 MIPS::PassaInstruction::PassaInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.47.3 Member Function Documentation

4.47.3.1 bit16_t MIPS::PassaInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

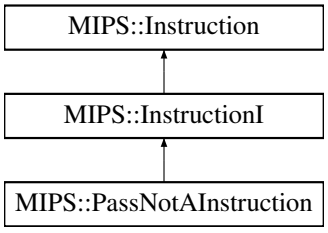
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/passa.hpp

4.48 MIPS::PassNotAInstruction Class Reference

`#include <passnota.hpp>`

Inheritance diagram for MIPS::PassNotAInstruction:



Public Member Functions

- PassNotAInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.48.1 Detailed Description

Classe que faz a operação de PASSNOTA no processador.

Author

Felipe Dias

4.48.2 Constructor & Destructor Documentation

4.48.2.1 MIPS::PassNotAInstruction::PassNotAInstruction (bit8_t *opcode*, Register * *rs*, Register * *rt*, bit8_t *shamt*, bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.48.3 Member Function Documentation

4.48.3.1 bit16_t MIPS::PassNotAInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/passnota.hpp

4.49 MIPS::Queue< T > Class Template Reference

`#include <queue.hpp>`

Classes

- struct Node

Public Member Functions

- Queue ()
- ~Queue ()
- void push (T &elem)
- T pop ()
- T get (size_t pos)
- size_t size ()

4.49.1 Detailed Description

template<typename T>
class MIPS::Queue< T >

Fila genérica que utiliza template para auxiliar em seu reuso.

Parameters

Matheus	Nogueira

4.49.2 Constructor & Destructor Documentation

4.49.2.1 template<typename T> MIPS::Queue< T >::Queue ( ) [inline]

Cria uma nova fila encadeada.

4.49.2.2 template<typename T> MIPS::Queue< T >::~Queue ( ) [inline]

Destroi a fila e todos os seus elementos.

4.49.3 Member Function Documentation

4.49.3.1 template<typename T> T MIPS::Queue< T >::get (size_t *pos*) [inline]

Recupera o elemento que está na posição especificada da fila.

Parameters

pos	posição do elemento na fila.

Returns

elemento na posição especificada.

4.49.3.2 template<typename T> T MIPS::Queue< T >::pop ( ) [inline]

Retira um elemento da fila.

Returns

elemento no inicio da fila.

4.49.3.3 template<typename T> void MIPS::Queue< T >::push (T & *elem*) [inline]

Adiciona um elemento à fila.

Parameters

elem	elemento que será inserido na fila.

4.49.3.4 template<typename T> size_t MIPS::Queue< T >::size ( ) [inline]

Retorna o tamanho da fila.

Returns

tamanho atual da fila.

The documentation for this class was generated from the following file:

- include/mips/util/structure/queue.hpp

4.50 MIPS::Register Class Reference

`#include <register.hpp>`

Public Member Functions

- Register (const char *name, bool protect=false)
- ~Register ()
- void put (bit16_t value)
- bit16_t get ()
- const char * getName ()

4.50.1 Detailed Description

Classe que representa um registrador de 16 bits.

Author

Matheus Nogueira

4.50.2 Constructor & Destructor Documentation

4.50.2.1 MIPS::Register::Register (const char * *name*, bool *protect* = `false`)

Cria um novo registrador.

Parameters

name	nome do registrador. Cria um novo registrador.
name	nome do registrador.
protected	indica se o registrador é protegido para escrita.

4.50.2.2 MIPS::Register::~Register ( )

Destroi o registrador.

4.50.3 Member Function Documentation

4.50.3.1 bit16_t MIPS::Register::get ( )

Pega o valor 16 bits armazenado no registrador.

Returns

valor armazenado no registrador.

4.50.3.2 const char* MIPS::Register::getName ( )

Retorna o nome do registrador.

Returns

nome do registrador.

4.50.3.3 void MIPS::Register::put (bit16_t *value*)

Define um valor que o registrador irá guardar.

Parameters

value	valor 16 bits que será armazenado no registrador.

The documentation for this class was generated from the following file:

- include/mips/memory/register.hpp

4.51 MIPS::RegisterBank Class Reference

`#include <register_bank.hpp>`

Public Member Functions

- RegisterBank ()
- ~RegisterBank ()
- Register * getRegister (bit8_t id)
- Register * getPC ()
- void write (bit16_t result, bit8_t rd)

4.51.1 Detailed Description

Classe que representa o banco de registradores do MIPS. Esta é responsável por gerenciar os registradores do processador.

Author

Matheus Nogueira

4.51.2 Constructor & Destructor Documentation

4.51.2.1 MIPS::RegisterBank::RegisterBank ( )

Cria uma instância do banco de registradores.

4.51.2.2 MIPS::RegisterBank::~RegisterBank ( )

Destroi a instância do banco de registradores.

4.51.3 Member Function Documentation

4.51.3.1 Register* MIPS::RegisterBank::getPC ( )

Retorna o ponteiro para o registrador contador de programa.

Returns

ponteiro para o program counter

4.51.3.2 Register* MIPS::RegisterBank::getRegister (bit8_t *id*)

Retorna um ponteiro para o registrador identificado pelo código especificado.

Parameters

id	código do registrador.

Returns

ponteiro para o registrador.

4.51.3.3 void MIPS::RegisterBank::write (bit16_t *result*, bit8_t *rd*)

Escreve o valor de um registrador usando o seu indice para localizá-lo

Parameters

result	novo valor do registrador
rd	registrador de destino

The documentation for this class was generated from the following file:

- include/mips/memory/register_bank.hpp

4.52 MIPS::SignalExtender Class Reference

`#include <signal_extender.hpp>`

Static Public Member Functions

- static bit16_t extend (bit16_t num, bit8_t bits=8)

4.52.1 Detailed Description

Circuito responsável por extender um sinal para 16 bits.

Author

Matheus Nogueira

4.52.2 Member Function Documentation

4.52.2.1 static bit16_t MIPS::SignalExtender::extend (bit16_t *num*, bit8_t *bits* = `8`) `[static]`

Extende o sinal de um número.

Parameters

num	número que será extendido.
bits	número de bits do número de entrada. (Padrão: 8)

Returns

número de 16 bits

The documentation for this class was generated from the following file:

- include/mips/circuits/signal_extender.hpp

4.53 MIPS::SignalInversor Class Reference

`#include <signal_inversor.hpp>`

Public Member Functions

- SignalInversor ()
- bit16_t invert (bit16_t num)

4.53.1 Detailed Description

Classe responsável por realizar inversão de sinal de 16 bits.

Author

Matheus Nogueira

4.53.2 Constructor & Destructor Documentation

4.53.2.1 MIPS::SignalInversor::SignalInversor ( )

Cria um novo somador.

4.53.3 Member Function Documentation

4.53.3.1 bit16_t MIPS::SignalInversor::invert (bit16_t *num*)

Inverte o sinal de um número.

Parameters

num	número que terá seu sinal invertido.

Returns

número com o sinal invertido.

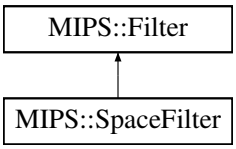
The documentation for this class was generated from the following file:

- include/mips/circuits/signal_inversor.hpp

4.54 MIPS::SpaceFilter Class Reference

`#include <space_filter.hpp>`

Inheritance diagram for MIPS::SpaceFilter:



Public Member Functions

- std::string filter (std::string &input)

4.54.1 Detailed Description

Classe responsável por remover espaços e tabulações de um texto.

Author

Matheus Matheus Nogueira

4.54.2 Member Function Documentation

4.54.2.1 std::string MIPS::SpaceFilter::filter (std::string & *input*) [virtual]

Remove todos os espaços e tabulações de uma string.

Parameters

input	string a ser filtrada.

Returns

string filtrada.

Implements MIPS::Filter.

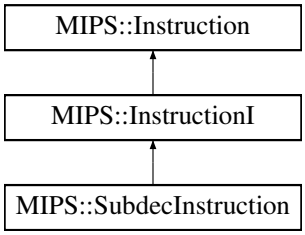
The documentation for this class was generated from the following file:

- include/mips/util/filter/space_filter.hpp

4.55 MIPS::SubdecInstruction Class Reference

`#include <subdec.hpp>`

Inheritance diagram for MIPS::SubdecInstruction:



Public Member Functions

- SubdecInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.55.1 Detailed Description

Classe que faz a operação de subdec no processador.

Author

Lucas Pereira

4.55.2 Constructor & Destructor Documentation

4.55.2.1 MIPS::SubdecInstruction::SubdecInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.55.3 Member Function Documentation

4.55.3.1 bit16_t MIPS::SubdecInstruction::execute ( ) [virtual]

Função que executa a operação de subtração.

Implements MIPS::InstructionI.

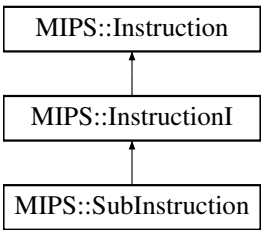
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/subdec.hpp

4.56 MIPS::SubInstruction Class Reference

`#include <sub.hpp>`

Inheritance diagram for MIPS::SubInstruction:



Public Member Functions

- SubInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.56.1 Detailed Description

Classe que faz a operação de ADD no processador.

Author

Felipe Dias

4.56.2 Constructor & Destructor Documentation

4.56.2.1 MIPS::SubInstruction::SubInstruction (bit8_t *opcode*, Register * *rs*, Register * *rt*, bit8_t *shamt*, bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.56.3 Member Function Documentation

4.56.3.1 bit16_t MIPS::SubInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/sub.hpp

4.57 MIPS::Tokenizer Class Reference

`#include <tokenizer.hpp>`

Public Member Functions

- void tokenize (char *str, std::vector< char * > &vector)

4.57.1 Detailed Description

Classe responsável por quebrar uma string em diversos tokens.

Author

Matheus Nogueira

4.57.2 Member Function Documentation

4.57.2.1 void MIPS::Tokenizer::tokenize (char * *str*, std::vector< char * > & *vector*)

Retira os tokens de uma string e os armazena em um vector.

Parameters

str	string contendo os tokens
vector	vector que será usado para armazenar os tokens

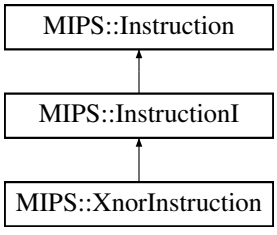
The documentation for this class was generated from the following file:

- include/mips/interpreter/parser/tokenizer.hpp

4.58 MIPS::XnorInstruction Class Reference

`#include <xnor.hpp>`

Inheritance diagram for MIPS::XnorInstruction:



Public Member Functions

- XnorInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.58.1 Detailed Description

Classe que faz a operação de XNOR no processador.

Author

Matheus Nogueira

4.58.2 Constructor & Destructor Documentation

4.58.2.1 MIPS::XnorInstruction::XnorInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.58.3 Member Function Documentation

4.58.3.1 bit16_t MIPS::XnorInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

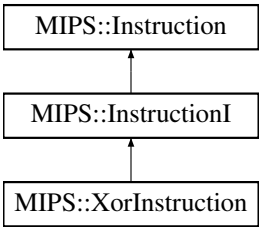
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/xnor.hpp

4.59 MIPS::XorInstruction Class Reference

`#include <xor.hpp>`

Inheritance diagram for MIPS::XorInstruction:



Public Member Functions

- XorInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.59.1 Detailed Description

Classe que faz a operação de XOR no processador.

Author

Felipe Dias

4.59.2 Constructor & Destructor Documentation

4.59.2.1 MIPS::XorInstruction::XorInstruction (bit8_t *opcode,* Register * *rs,* Register * *rt,* bit8_t *shamt,* bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.59.3 Member Function Documentation

4.59.3.1 bit16_t MIPS::XorInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

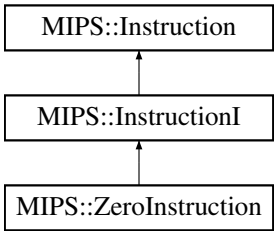
The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/xor.hpp

4.60 MIPS::ZeroInstruction Class Reference

`#include <zero.hpp>`

Inheritance diagram for MIPS::ZeroInstruction:



Public Member Functions

- ZeroInstruction (bit8_t opcode, Register *rs, Register *rt, bit8_t shamt, bit8_t funct)
- bit16_t execute ()

Additional Inherited Members

4.60.1 Detailed Description

Classe que faz a operação de ZERO no processador.

Author

Felipe Dias

4.60.2 Constructor & Destructor Documentation

4.60.2.1 MIPS::ZeroInstruction::ZeroInstruction (bit8_t *opcode*, Register * *rs*, Register * *rt*, bit8_t *shamt*, bit8_t *funct*) [inline]

Constroi uma nova instrução.

4.60.3 Member Function Documentation

4.60.3.1 bit16_t MIPS::ZeroInstruction::execute ( ) [virtual]

Função que executa a operação de soma.

Returns

resultado da operação

Implements MIPS::InstructionI.

The documentation for this class was generated from the following file:

- include/mips/instructions/format_I/zero.hpp

Chapter 5

File Documentation

5.1 include/mips/circuits/full_adder.hpp File Reference

```
#include <mips/core.hpp>
```

Classes

- class MIPS::FullAdder

5.1.1 Detailed Description

Somador de 16 bits. Referencia: http://isweb.redwoods.edu/instruct/calderwoodd/diglogic/full.↩
htm

5.2 include/mips/circuits/signal_extender.hpp File Reference

```
#include <mips/core.hpp>
```

Classes

- class MIPS::SignalExtender

5.2.1 Detailed Description

Estensor de sinal. Converte um número para 16 bits.

5.3 include/mips/circuits/signal_inversor.hpp File Reference

```
#include <mips/core.hpp>
```

Classes

- class MIPS::SignalInversor

5.3.1 Detailed Description

Inversor de sinal em números que utilizam a representação de complemento de 2.

5.4 include/mips/core.hpp File Reference

```
#include <mips/debug.hpp>
```

Typedefs

- typedef signed char MIPS::bit8_t
- typedef signed short MIPS::bit16_t
- typedef signed int MIPS::bit32_t
- typedef bit16_t MIPS::instruction_t

5.4.1 Detailed Description

Arquivo que contém tipos utilizados pelo emulaodr.

5.4.2 Typedef Documentation

5.4.2.1 typedef signed short MIPS::bit16_t

Tipo que representa um inteiro de 16 bits.

5.4.2.2 typedef signed int MIPS::bit32_t

Tipo que representa um inteiro de 16 bits.

5.4.2.3 typedef signed char MIPS::bit8_t

Tipo que representa um inteiro de 8 bits.

5.4.2.4 typedef bit16_t MIPS::instruction_t

Tipo que representa uma instrução de 16 bits.

5.5 include/mips/cpu.hpp File Reference

```
#include <mips/memory/register_bank.hpp>
#include <mips/memory/memory.hpp>
#include <mips/units/instruction_finder.hpp>
#include <mips/units/control.hpp>
#include <mips/decoder/instruction_decoder.hpp>
```

Classes

- class MIPS::CPU

5.5.1 Detailed Description

Arquivo contendo uma estrutura que representa a CPU m-RISC.

5.6 include/mips/debug.hpp File Reference

```
#include <iostream>
#include <cstdio>
#include <mips/core.hpp>
```

Macros

- #define **MESSAGE**(arg)
- #define **DEBUG**(arg)
- #define **FORMAT_DEBUG**(format, ...)
- #define **PRINT_BIN**(num)

5.6.1 Detailed Description

Arquivo que contém funções para auxiliar o processo de DEBUG da execução do emulador.

5.6.2 Macro Definition Documentation

5.6.2.1 #define DEBUG(*arg*)

Value:

```
{                                                   \
        std::cout << "[DEBUG] " << arg << std::endl;                        \
    }
```

5.6.2.2 #define FORMAT_DEBUG(*format, ...*)

Value:

```
{                                       \
        printf(format, __VA_ARGS__);                                        \
    }
```

5.6.2.3 #define MESSAGE(*arg*)

Value:

```
{                                               \
        std::cout << arg << std::endl;                                      \
    }
```

5.6.2.4 #define PRINT_BIN(*num*)

Value:

```
{                   \
        char x09878412bin = 0;                  \
        char i = 15;                            \
        for (; i >= 0; --i) {                   \
            x09878412bin = (num >> i) & 1;      \
            printf("%d", x09878412bin);         \
            if (i % 4 == 0)                         \
                printf(" ");                    \
        }                                       \
        printf("\n");                           \
    }
```

5.7 include/mips/instructions/format_I/add.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::AddInstruction

5.7.1 Detailed Description

Declaração da instrução de ADD.

Declaração da instrução de ASR.

Declaração da instrução de DECA.

Declaração da instrução de INCA.

Declaração da instrução de SUB.

5.8 include/mips/instructions/format_I/addinc.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::AddIncInstruction

5.8.1 Detailed Description

Declaração da instrução de ADDINC.

5.9 include/mips/instructions/format_I/and.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::AndInstruction

5.9.1 Detailed Description

Declaração da instrução de andnota.

5.10 include/mips/instructions/format_I/andnota.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::AndnotaInstruction

5.10.1 Detailed Description

Declaração da instrução de andnota.

5.11 include/mips/instructions/format_I/nand.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::NandInstruction

5.11.1 Detailed Description

Declaração da instrução de NAND.

5.12 include/mips/instructions/format_I/nor.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::NorInstruction

5.12.1 Detailed Description

Declaração da instrução de NOR.

5.13 include/mips/instructions/format_I/ones.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::OnesInstruction

5.13.1 Detailed Description

Declaração da instrução de ONES.

5.14 include/mips/instructions/format_I/or.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::OrInstruction

5.14.1 Detailed Description

Declaração da instrução de or.

5.15 include/mips/instructions/format_I/ornotb.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::OrnotbInstruction

5.15.1 Detailed Description

Declaração da instrução de ornotb.

5.16 include/mips/instructions/format_I/passa.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::PassaInstruction

5.16.1 Detailed Description

Declaração da instrução de PASSA.

5.17 include/mips/instructions/format_I/passnota.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::PassNotAInstruction

5.17.1 Detailed Description

Declaração da instrução de PASSNOTA.

5.18 include/mips/instructions/format_I/subdec.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::SubdecInstruction

5.18.1 Detailed Description

Declaração da instrução de SubDec.

5.19 include/mips/instructions/format_I/xnor.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::XnorInstruction

5.19.1 Detailed Description

Declaração da instrução de XNOR.

5.20 include/mips/instructions/format_I/xor.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::XorInstruction

5.20.1 Detailed Description

Declaração da instrução de XOR.

5.21 include/mips/instructions/format_I/zero.hpp File Reference

```
#include <mips/instructions/instruction_I.hpp>
```

Classes

- class MIPS::ZeroInstruction

5.21.1 Detailed Description

Declaração da instrução de ZERO.

5.22 include/mips/instructions/format_II/loadlit.hpp File Reference

```
#include <mips/instructions/instruction_II.hpp>
```

Classes

- class MIPS::LoadlitInstruction

5.22.1 Detailed Description

Instrução que carrega uma constante com sinal em um registrador.

5.23 include/mips/instructions/instruction.hpp File Reference

```
#include <mips/core.hpp>
```

Classes

- class MIPS::Instruction

5.23.1 Detailed Description

Arquivo contendo a estrutura abstrata que representa uma instrução qualquer em uma arquitetura de 16 bits.

5.24 include/mips/instructions/instruction_I.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/instructions/instruction.hpp>
#include <mips/memory/register.hpp>
```

Classes

- class MIPS::InstructionI

5.24.1 Detailed Description

Arquivo contendo uma classe que representa uma instrução do tipo (R)egister.

5.25 include/mips/instructions/instruction_II.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/instructions/instruction.hpp>
#include <mips/memory/register.hpp>
```

Classes

- class MIPS::InstructionII

5.25.1 Detailed Description

Arquivo descrevendo o formato de uma instrução do formato II.

5.26 include/mips/instructions/instruction_III.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/instructions/instruction.hpp>
#include <mips/memory/register.hpp>
```

Classes

- class MIPS::InstructionIII

5.26.1 Detailed Description

Arquivo descrevendo o formato de uma instrução do formato III.

5.27 include/mips/interpreter/encoder/encoder.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/interpreter/label.hpp>
#include <map>
#include <vector>
```

Classes

- class MIPS::Encoder

5.27.1 Detailed Description

Arquivo contendo um codificador genérico para instruções MIPS 16 bits.

5.28 include/mips/interpreter/encoder/encoder_factory.hpp File Reference

```
#include <mips/interpreter/encoder/encoder.hpp>
#include <mips/interpreter/encoder/format_I_encoder.hpp>
#include <mips/interpreter/encoder/format_II_encoder.hpp>
#include <mips/interpreter/encoder/format_III_encoder.hpp>
#include <mips/interpreter/encoder/format_IV_encoder.hpp>
#include <mips/interpreter/encoder/format_V_encoder.hpp>
#include <mips/interpreter/encoder/format_VI_encoder.hpp>
#include <mips/interpreter/encoder/format_VII_encoder.hpp>
```

Classes

- class MIPS::EncoderFactory

5.28.1 Detailed Description

Arquivo contendo uma fábrica de codificadores de instruções.

5.29 include/mips/interpreter/encoder/format_I_encoder.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/interpreter/encoder/encoder.hpp>
#include <vector>
```

Classes

- class MIPS::FormatIEncoder

5.29.1 Detailed Description

Arquivo contendo o codificador de instruções do formato I.

5.30 include/mips/interpreter/encoder/format_II_encoder.hpp File Reference

```
#include <mips/interpreter/encoder/encoder.hpp>
#include <vector>
```

Classes

- class MIPS::FormatIIEncoder

5.30.1 Detailed Description

Arquivo contendo o codificador de instruções do formato II.

5.31 include/mips/interpreter/encoder/format_III_encoder.hpp File Reference

```
#include <mips/interpreter/encoder/encoder.hpp>
#include <vector>
```

Classes

- class MIPS::FormatIIIEncoder

5.31.1 Detailed Description

Arquivo contendo o codificador de instruções do formato III.

5.32 include/mips/interpreter/encoder/format_IV_encoder.hpp File Reference

```
#include <mips/interpreter/encoder/encoder.hpp>
#include <vector>
```

Classes

- class MIPS::FormatIVEncoder

5.32.1 Detailed Description

Arquivo contendo o codificador de instruções do formato IV.

5.33 include/mips/interpreter/encoder/format_V_encoder.hpp File Reference

```
#include <mips/interpreter/encoder/encoder.hpp>
#include <vector>
```

Classes

- class MIPS::FormatVEncoder

5.33.1 Detailed Description

Arquivo contendo o codificador de instruções do formato V.

5.34 include/mips/interpreter/encoder/format_VI_encoder.hpp File Reference

```
#include <mips/interpreter/encoder/encoder.hpp>
#include <vector>
```

Classes

- class MIPS::FormatVIEncoder

5.34.1 Detailed Description

Arquivo contendo o codificador de instruções do formato VI.

5.35 include/mips/interpreter/encoder/format_VII_encoder.hpp File Reference

```
#include <mips/interpreter/encoder/encoder.hpp>
#include <vector>
```

Classes

- class MIPS::FormatVIIEncoder

5.35.1 Detailed Description

Arquivo contendo o codificador de instruções do formato VII.

5.36 include/mips/interpreter/exception/interpreter_exception.hpp File Reference

```
#include <exception>
#include <stdexcept>
#include <mips/core.hpp>
```

Classes

- class MIPS::InterpreterException

5.36.1 Detailed Description

Arquivo contendo uma exceção que é lançada pelo interpretador de assembly.

5.37 include/mips/interpreter/interpreter.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/interpreter/label.hpp>
#include <mips/util/file_reader.hpp>
#include <iostream>
```

Classes

- class MIPS::Interpreter

5.37.1 Detailed Description

Arquivo contendo a declaração do interpretador de instruções MIPS 32.

5.38 include/mips/interpreter/label.hpp File Reference

Classes

- struct MIPS::Label

5.38.1 Detailed Description

Arquivo contendo uma estrutura que armazena um label do codigo assembly.

5.39 include/mips/interpreter/parser/tokenizer.hpp File Reference

```
#include <vector>
```

Classes

- class MIPS::Tokenizer

5.39.1 Detailed Description

Arquivo contendo a classe responsável por separar uma string em tokens.

5.40 include/mips/memory/memory.hpp File Reference

```
#include <mips/core.hpp>
```

Classes

- class MIPS::Memory

5.40.1 Detailed Description

Arquivo contendo a estrutura responsável por encapsular a memória de nstruções e a memória de dados utilizada pelo processador.

5.41 include/mips/memory/memory_exception.hpp File Reference

```
#include <exception>
#include <stdexcept>
#include <mips/core.hpp>
```

Classes

- class MIPS::MemoryException

5.41.1 Detailed Description

Arquivo contendo uma exceção que é lançada pelo interpretador de assembly.

5.42 include/mips/memory/register.hpp File Reference

```
#include <mips/core.hpp>
```

Classes

- class MIPS::Register

5.42.1 Detailed Description

Arquivo contendo a definição de um registrador.

5.43 include/mips/memory/register_bank.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/memory/register.hpp>
```

Classes

- class MIPS::RegisterBank

5.43.1 Detailed Description

Arquivo contendo o banco de registradores contido no processador MIPS.

5.44 include/mips/units/control.hpp File Reference

Classes

- class MIPS::ControlUnit

5.44.1 Detailed Description

Arquivo contendo a unidade de controle do processador.

5.45 include/mips/units/instruction_finder.hpp File Reference

```
#include <mips/core.hpp>
#include <mips/memory/memory.hpp>
#include <mips/memory/register_bank.hpp>
```

Classes

- class MIPS::InstructionFinder

5.45.1 Detailed Description

Arquivo contendo o buscador de instruções do processador.

5.46 include/mips/util/event/event.hpp File Reference

Classes

- class MIPS::Event

Enumerations

- enum MIPS::EventType { MIPS::REGISTER_UPDATE, **TEST** }

5.46.1 Detailed Description

Arquivo contendo um evento base que é despachavel pelo despachante de eventos do emulador de MIPS.

5.46.2 Enumeration Type Documentation

5.46.2.1 enum MIPS::EventType

Enum utilizado para listar todos os tipos de eventos suportados pelo emulador MIPS.

Enumerator

REGISTER_UPDATE Indica que um registrador foi alterado.

5.47 include/mips/util/event/event_dispatcher.hpp File Reference

```
#include <mips/util/structure/queue.hpp>
#include <mips/util/event/event_listener.hpp>
#include <mips/util/event/event.hpp>
#include <vector>
```

Classes

- class MIPS::EventDispatcher
- struct MIPS::EventDispatcher::ListenerMap

5.47.1 Detailed Description

Arquivo contendo o despachante de eventos utilizado por diversos componentes do núcleo do emulador MIPS. Esse despachante é responsável por criar uma interface de despache de eventos para outros componentes, dessa forma, um componente pode comunicar com outro componente usando troca de mensagens, onde estas são transmitidas via eventos.

5.48 include/mips/util/event/event_listener.hpp File Reference

```
#include <mips/util/event/event.hpp>
```

Classes

- class MIPS::EventListener

5.48.1 Detailed Description

Arquivo contendo a classe abstrata a qual deve ser pai de todos os objetos que devem ouvir um evento de outro objeto.

5.49 include/mips/util/file_reader.hpp File Reference

```
#include <mips/util/filter/filter.hpp>
#include <fstream>
#include <vector>
```

Classes

- class MIPS::FileReader

5.49.1 Detailed Description

Arquivo contendo uma classe responsável por ler e limpar o conteúdo de um arquivo do disco.

5.50 include/mips/util/filter/filter.hpp File Reference

```
#include <string>
```

Classes

- class MIPS::Filter

5.50.1 Detailed Description

Arquivo contendo um filtro de string abstrato.

5.51 include/mips/util/filter/space_filter.hpp File Reference

```
#include <mips/util/filter/filter.hpp>
```

Classes

- class MIPS::SpaceFilter

5.51.1 Detailed Description

Arquivo contendo um filtro que retira tabulações e espaços em branco desnecessários de um texto.