

Documentação Neander C

- [O Neander C](#)
- [O computador hipotético Neander](#)
- [Características](#)
- [Modos de endereçamento](#)
- [Conjunto de instruções](#)
- [Obtendo o Neander C](#)
- [Compilando o Neander C](#)
- [Executando o Neander C](#)
- [Programando o Neander C](#)
- [Considerações](#)

O Neander C

Neander C é um simulador do computador hipotético Neander escrito em linguagem C por Rúbens Antônio Rodrigues e Cleiton Sasaki. O Neander C possui todas as características do computador hipotético Neander

O computador hipotético Neander

O computador hipotético Neander foi criado para exercitar os conceitos da arquitetura de Von Neumann. Por ter uma arquitetura simples, facilita no aprendizado dos elementos envolvidos em uma arquitetura de computador. Facilita também no aprendizado da programação em linguagem de máquina (binário/assemble).

Esta pseudomáquina foi criada em homenagem ao homem de Neandertal, o antecessor do *homo sapiens*.

Características

O computador Neander, tem as seguintes características:

- largura de dados e endereços de 8 bits
- dados representados em complemento de dois
- um acumulador de 8 bits (AC)
- um apontador de programa de 8 bits (PC)
- um registrador de estados de com códigos de condição: negativo (N) ou zero (Z)

Modos de endereçamento

O Neander, só possui um modo de endereçamento: o modo de endereçamento direto (muitas vezes também chamado de absoluto). Neste modo de endereçamento, a palavra que segue o código da instrução contém, nas instruções de manipulação de dados, o endereço de memória do operando.

Conjunto de instruções

O conjunto de instruções do Neander compreende 11 instruções, codificadas por meio dos quatro bits mais significativos da palavra que contém o código da instrução.

Conjunto de instruções do Neander

Código	Instrução	Comentário
0000	NOP	Nenhuma operação
0001	STA end	armazena acumulador - (store)
0010	LDA end	carrega acumulador - (load)
0011	ADD end	soma
0100	OR end	"OU" lógico
0101	AND end	"E" lógico
0110	NOT	inverte (complementa) acumulador
1000	JMP end	desvio incondicional - (jump)
1001	JN end	desvio condicional - (jump on negative)
1010	JZ end	desvio condicional - (jump on zero)
1111	HLT	termino de execução - (halt)

Ações executadas

Instrução	Comentário
NOP	Nenhuma operação
STA end	MEM(end) <- AC
LDA end	AC <- MEM(end)
ADD end	AC <- MEM(end) + AC
OR end	AC <- MEM(end) OR AC
AND end	AC <- MEM(end) AND AC
NOT	AC <- NOT AC

```
JMP end PC <- end
```

```
JN end IF N=1 THEN PC <- end
```

```
JZ end IF Z=1 THEN PC <- end
```

Obtendo o Neander C

O Neander C é distribuído através de download no seu repositório oficial (<https://github.com/aquitemaluco/neanderc>). É possível obter o Neander C a partir dos arquivos fontes ou através de pacote .zip

Obtendo a partir dos fontes (é necessário ter o git instalado no computador):

Dentro de um diretório com permissões **rwX** execute o seguinte comando

```
git clone https://github.com/aquitemaluco/neanderc.git
```

Obtendo a partir de pacote .zip:

O pacote .zip pode ser obtido através de um navegador (Opera, Firefox, Chrome, etc.) e extraído por meio de uma ferramenta de extração (7-Zip, PeaZip, etc.) ou através dos comandos abaixo

```
wget https://github.com/aquitemaluco/neanderc/archive/master.zip  
unzip master.zip
```

Compilando o Neander C

O Neander C é desenvolvido para plataforma Linux, para sua compilação, é necessário ter instalado e devidamente configurado os pacotes de desenvolvimentos **gcc** e **make**

Para compilar, entre no diretório raiz dos arquivos fontes do Neander C e execute o seguinte comando:

```
make
```

Após a execução do comando será gerado dentro do diretório **bin/** o binário **neanderc**, é por meio dele que será iniciado o simulador Neander C

Executando o Neander C

A execução do Neander C é muito simples, não requer parâmetro ou configuração adicional, basta entrar do diretório **bin/** dentro do diretório dos fontes do Neander C e executar o binário

```
./neanderc
```

O menu do Neander C:

O menu principal do Neander C é composto por 4 opções

- *Opção 1 - Executar um arquivo.* Permite a execução de um arquivo de memória **.mem**.
- *Opção 2 - Modo interativo.* Permite escrever um programa diretamente na memória do simulador.
- *Opção 3 - Executar programa teste.* Permite executar um programa de teste pré-programado no simulador.
- *Opção 9 - Sair.* Encerra o simulador.

Executando um arquivo de memória .mem:

Após a seleção da opção de execução de um arquivo, aparecerá um prompt para fornecer o caminho (localização) do arquivo, basta fornecer o caminho e o simulador cuida do resto.

```
> ../examples/hlt.mem
```

Um arquivo de memória contem as instruções para o Neander C executar, o Neander C carrega o arquivo em sua memória, e após o carregamento, aguarda uma opção para a execução.

- *Opção 1 - Execucao normal.* Executa o programa contido na memoria e finaliza, mostrando o mapa da memória após a execução.
- *Opção 2 - Execucao por passo.* Executa o programa contido na memoria, com base na intervenção do usuário, e ao final mostra o mapa da memória após a execução.

Modo interativo:

O modo interativo permite programar diretamente dentro da memoria do simulador. Neste modo o simulador exibirá um prompt para leitura do endereço da memoria e, logo depois, outro para leitura da instrução que será armazenada no endereço especificado. A instrução pode ser um mnemônico das instruções existentes no Neander ou um dado (valor numérico) para a operação que será realizada. O modo interativo será encerrado automaticamente quando uma instrução **HLT** for programada em uma posição de memória

Programando o Neander C

A programação do Neander C é bem simples. Basta fornecer um endereço de memória e uma instrução/dado a ser armazenado no endereço fornecido.

Abaixo segue um exemplo de programa escrito em um arquivo **.mem**.

Programa hlt.mem

```
0 15
```

O programa hlt.mem, é composto por apenas uma linha, e seu conteúdo é: no endereço de memória zero tem a instrução **HLT** (1111b = 15) que significa termino da execução.

Programa sum.mem

```
128 30
129 67
0 2
1 128
2 3
3 129
4 1
5 130
6 15
```

O programa sum.mem, é composto por nove linhas linha, e seu conteúdo é: inserção direta do valor 30 no endereço 128 da memória (linha 1) e 67 no endereço 129 (linha 2), **LDA** (carrega) em **AC** o dado (valor = 30) que está no endereço 128 (linhas 3 e 4), **ADD** (soma) o dado que está no endereço 129 ao dado de **AC** e armazena o resultado em **AC** (linhas 5 e 6), **STA** (salva) o dado que está em **AC** no endereço 130 (linhas 7 e 8) e por fim, **HLT**, encerra o programa normalmente linha (9).

Considerações

O simulador Neander C é uma ótima ferramenta para auxiliar no aprendizado dos componente de um computador e também no aprendizado da programação através de linguagem de máquina.

É muito leve e não consome recursos excessivos do computador. Pode ser executado em qualquer computador com sistema operacional Linux.

Apesar de haver um certo grau de dificuldade na criação dos binários do Neander C por usuários que não são familiarizados com o Linux, a maior dificuldade não está realmente na criação dos binários e sim no uso do Linux.