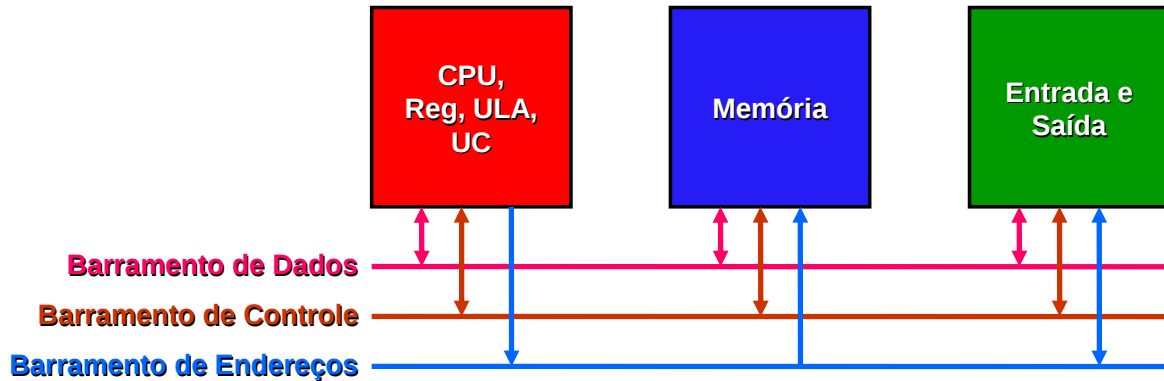


Aula 002

- **Sistema Computacional**

Sistema de Computador

Componentes Básicos do Sistema de Computador



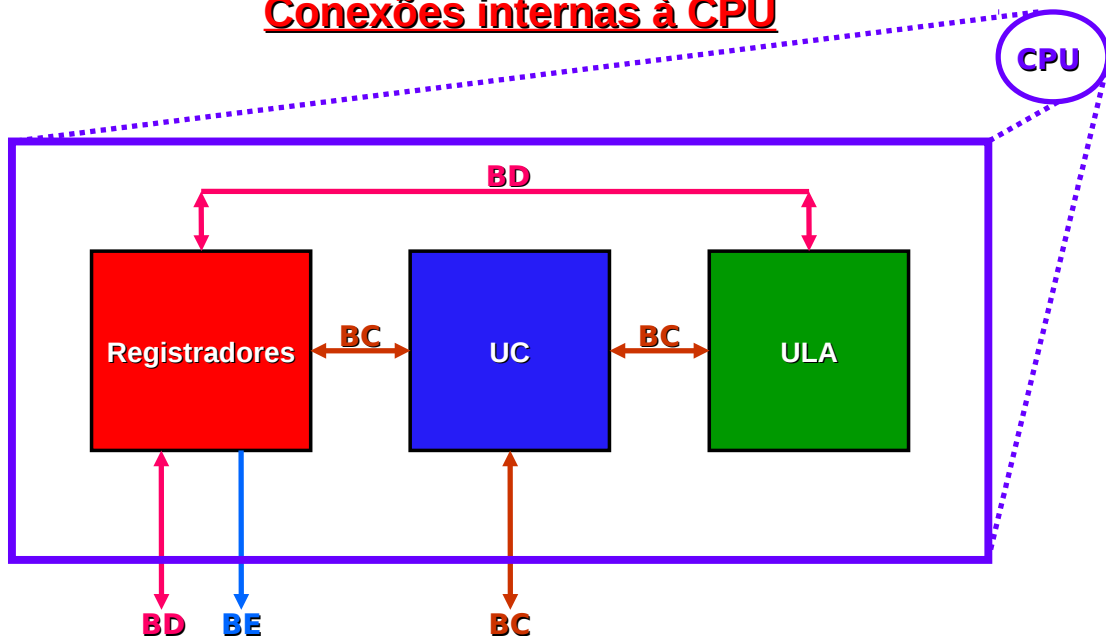
Barramentos:

- BD: Barramento de Dados transportam a informação útil (dados e instruções)
- BC: Barramento de Controle transportam informações de controle
- BE: Barramento de Endereços identificam o destino das informações

3

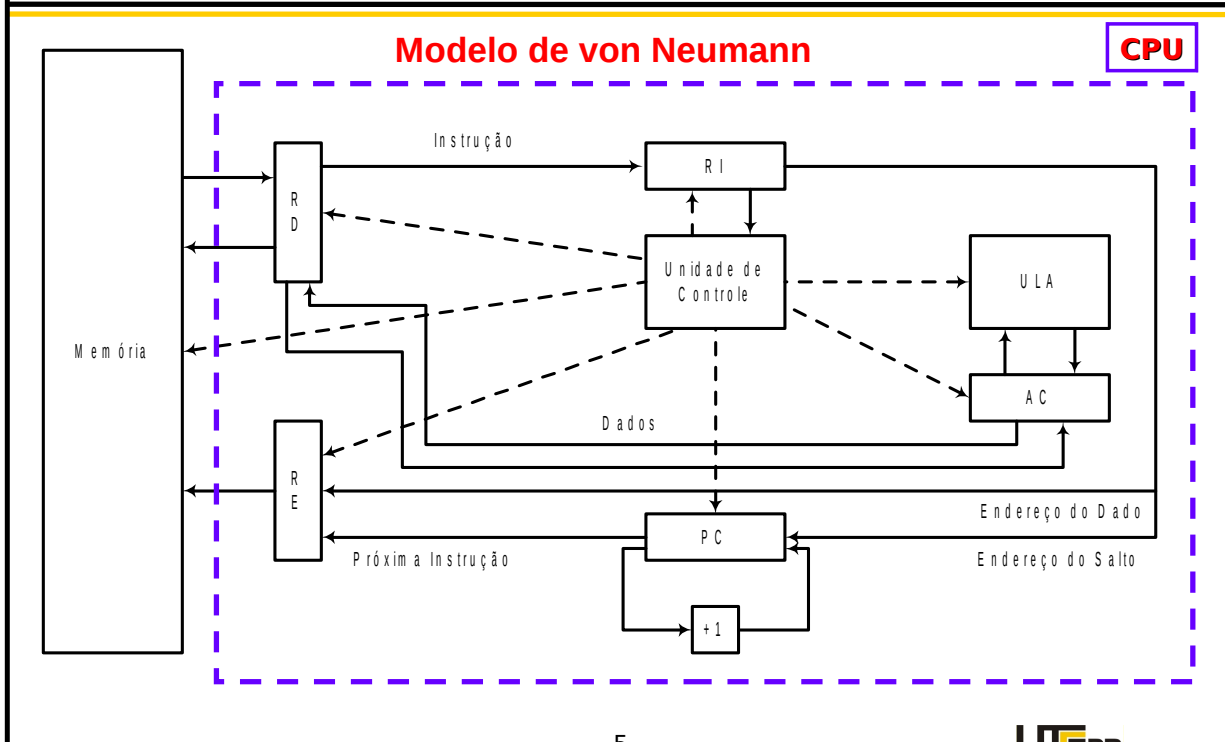
Zoom da CPU

Conexões internas à CPU

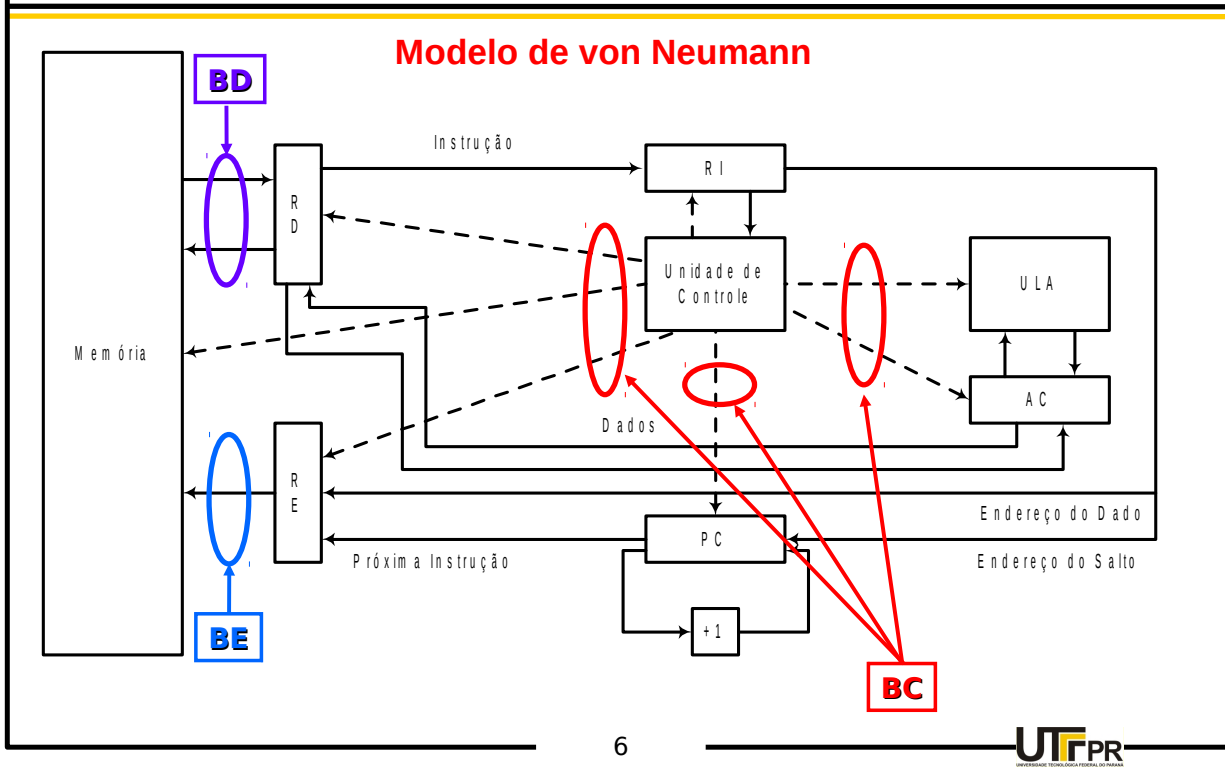


4

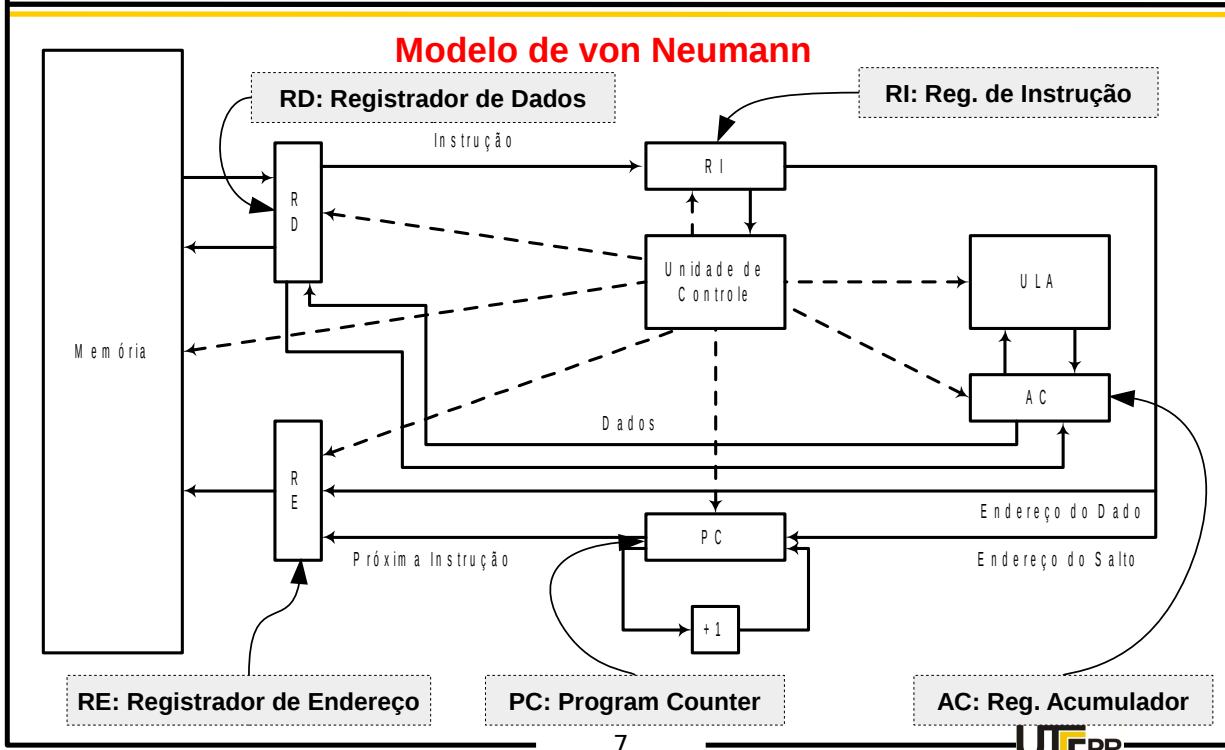
Arquitetura de von Neumann



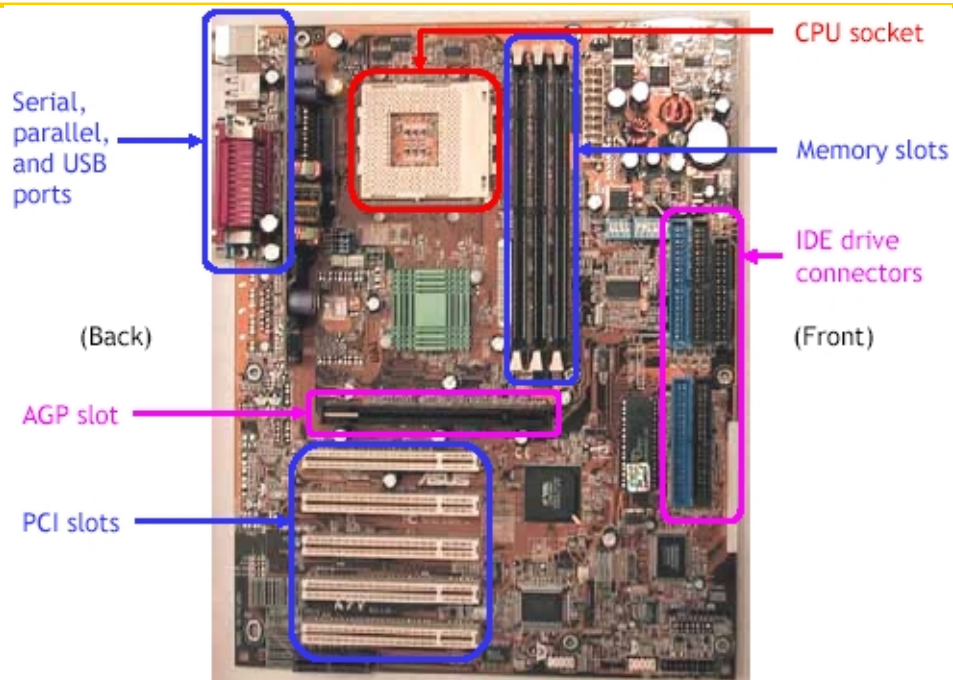
Arquitetura de von Neumann



Arquitetura de von Neumann



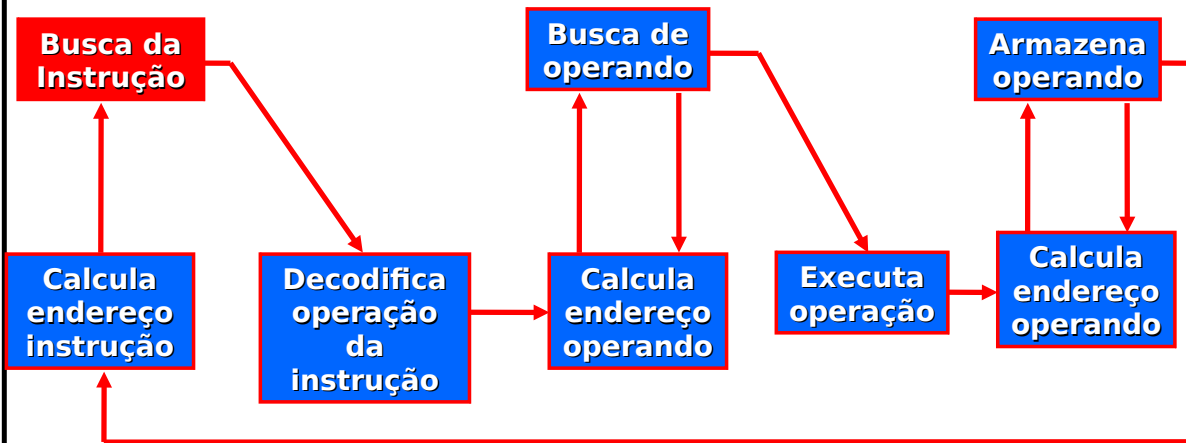
Exemplos de Barramentos



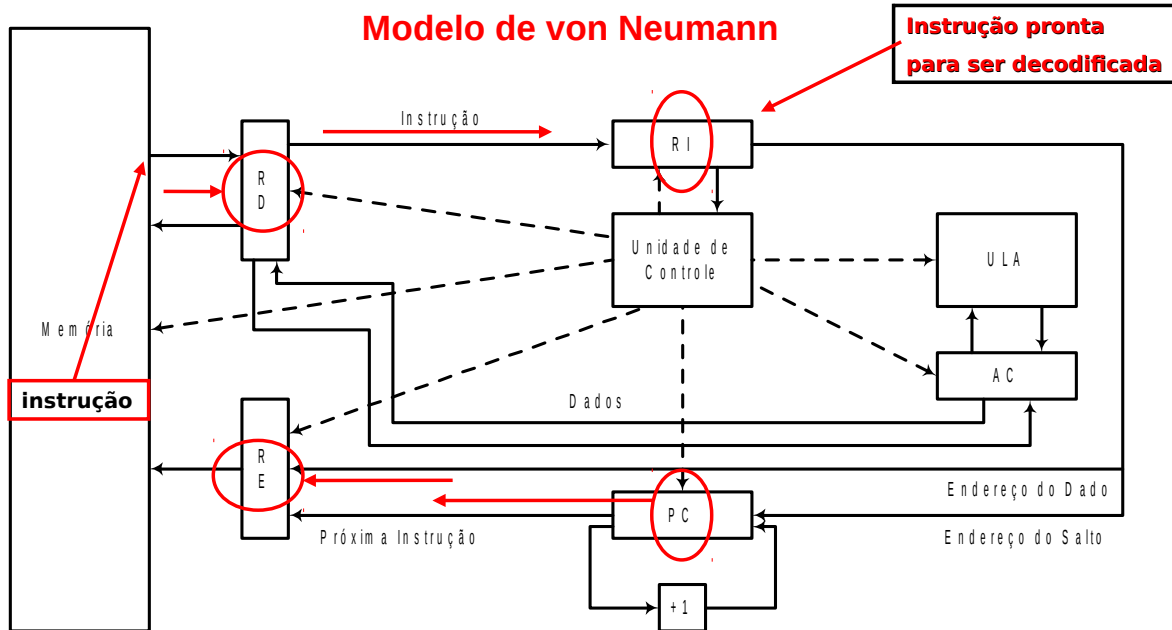
Ciclo de Instrução

Ciclo de Instrução

Mostra a interação entre os elementos da Arquitetura de von Neumann



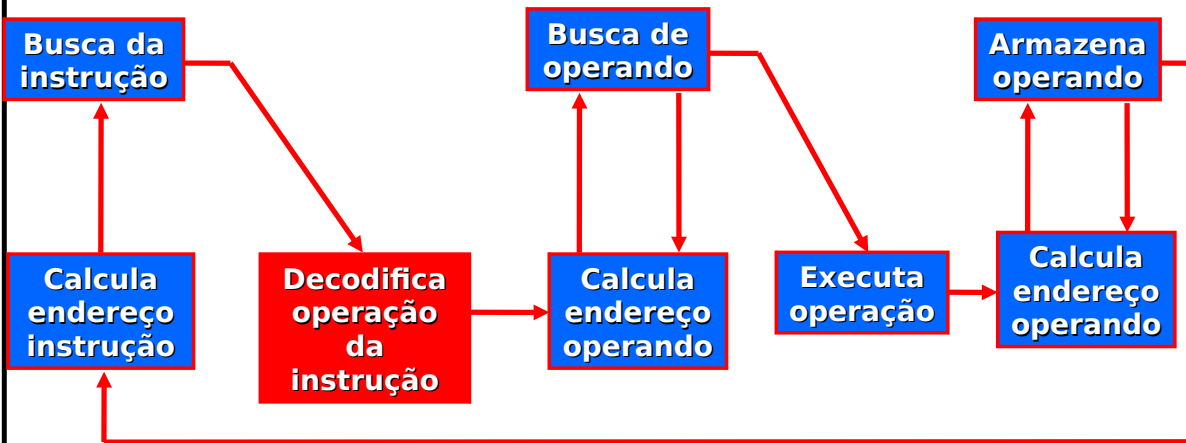
Busca de Instrução



Ciclo de Instrução

Ciclo de Instrução

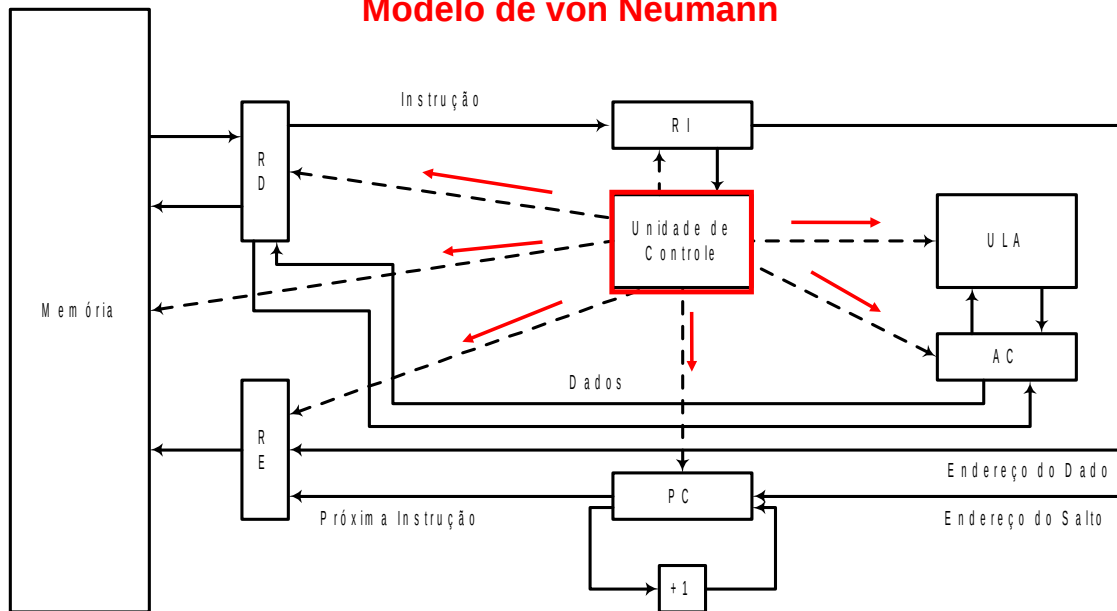
Mostra a interação entre os elementos da Arquitetura de von Neumann



11

Decodificação de Instrução

Modelo de von Neumann

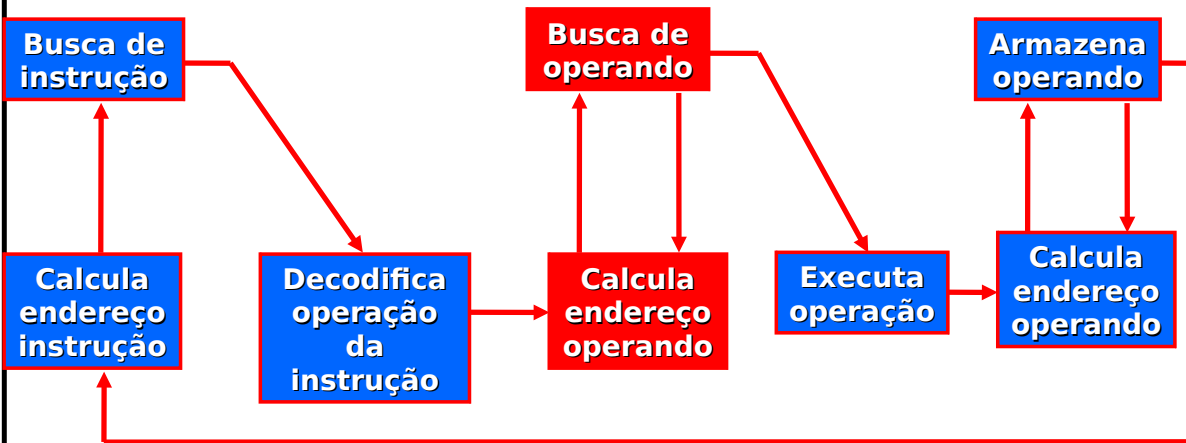


12

Ciclo de Instrução

Ciclo de Instrução

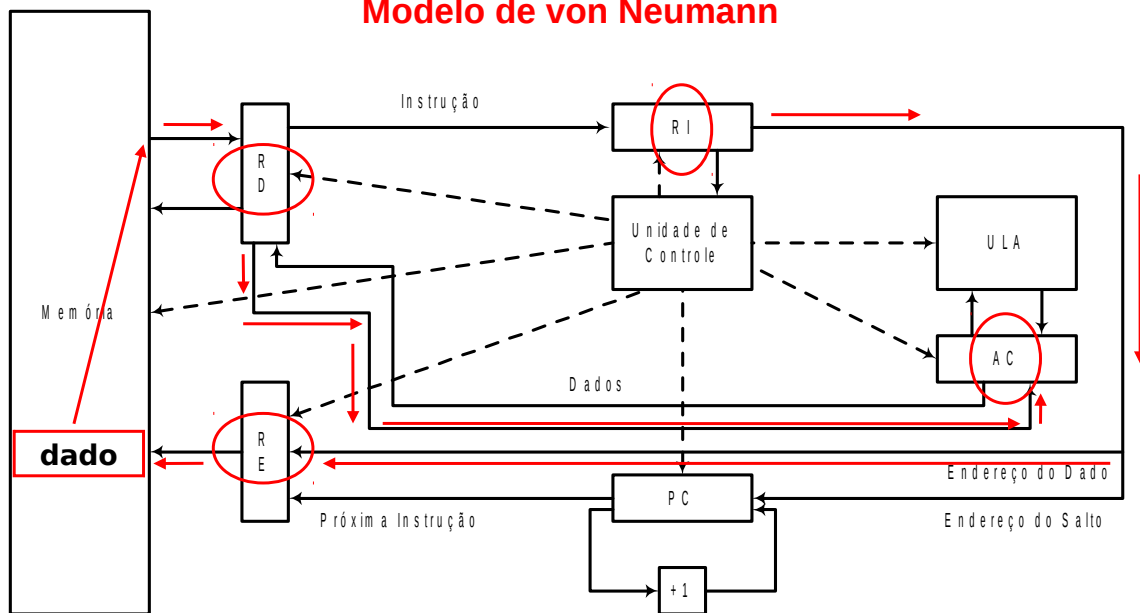
Mostra a interação entre os elementos da Arquitetura de von Neumann



13

Busca de Operando

Modelo de von Neumann

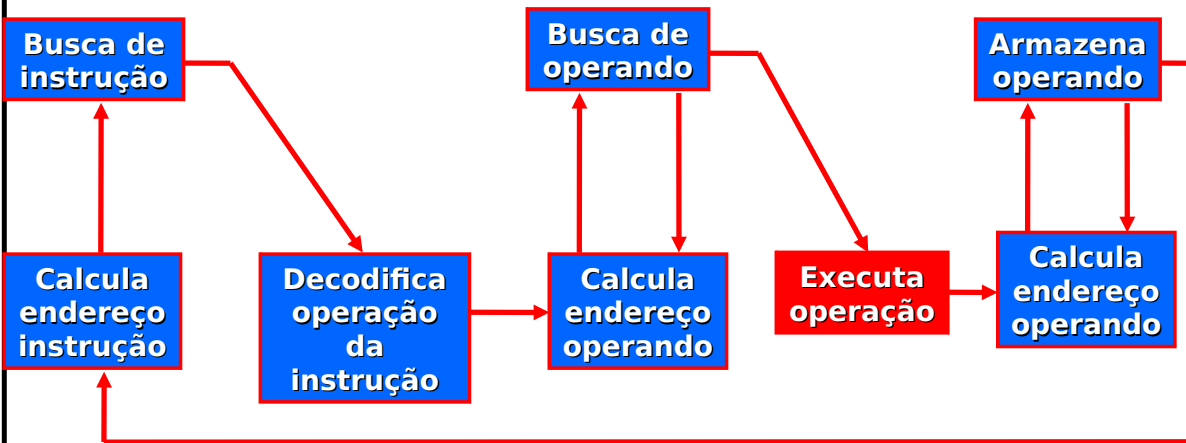


14

Ciclo de Instrução

Ciclo de Instrução

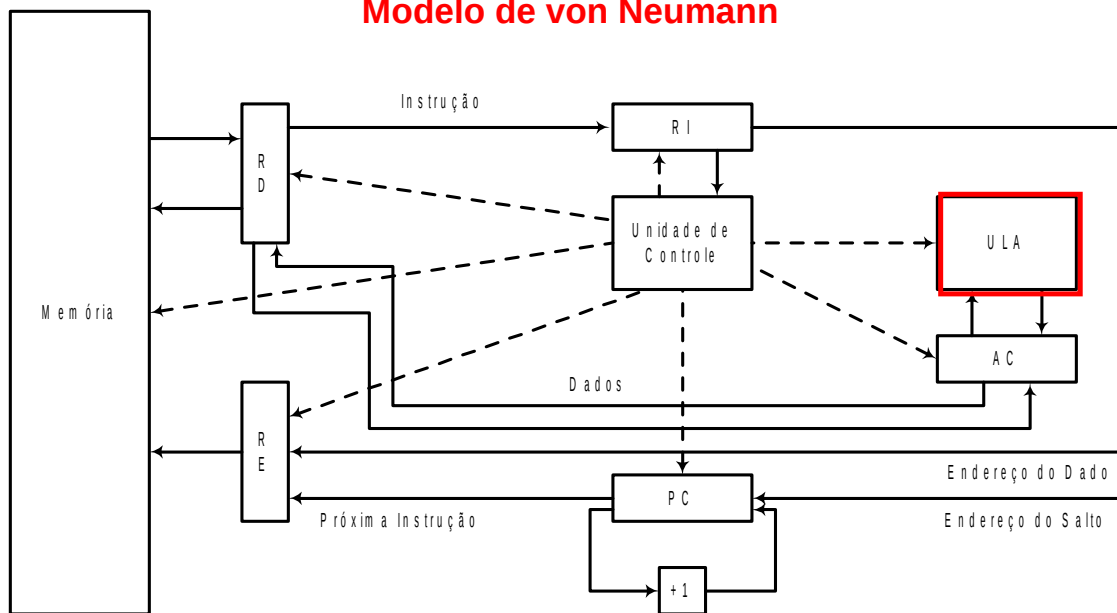
Mostra a interação entre os elementos da Arquitetura de von Neumann



15

Execução da Instrução

Modelo de von Neumann

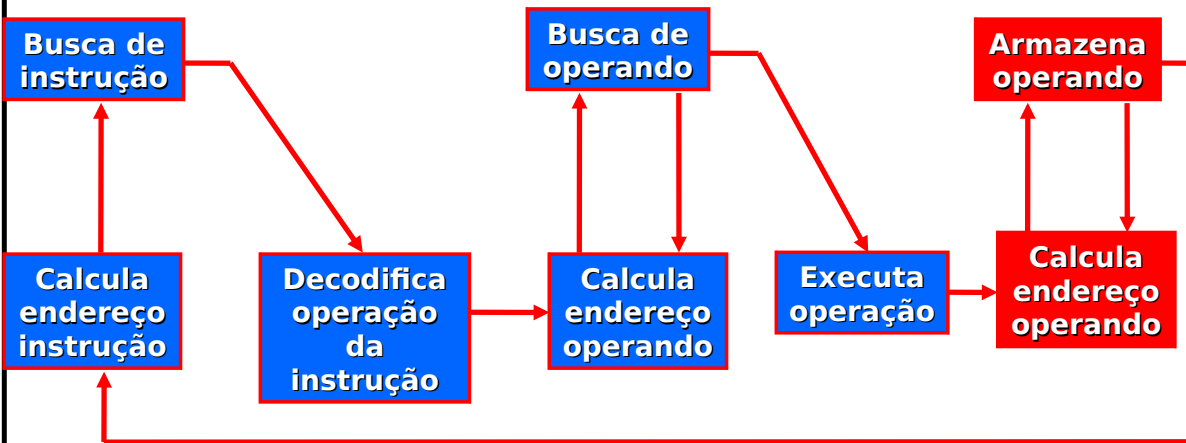


16

Ciclo de Instrução

Ciclo de Instrução

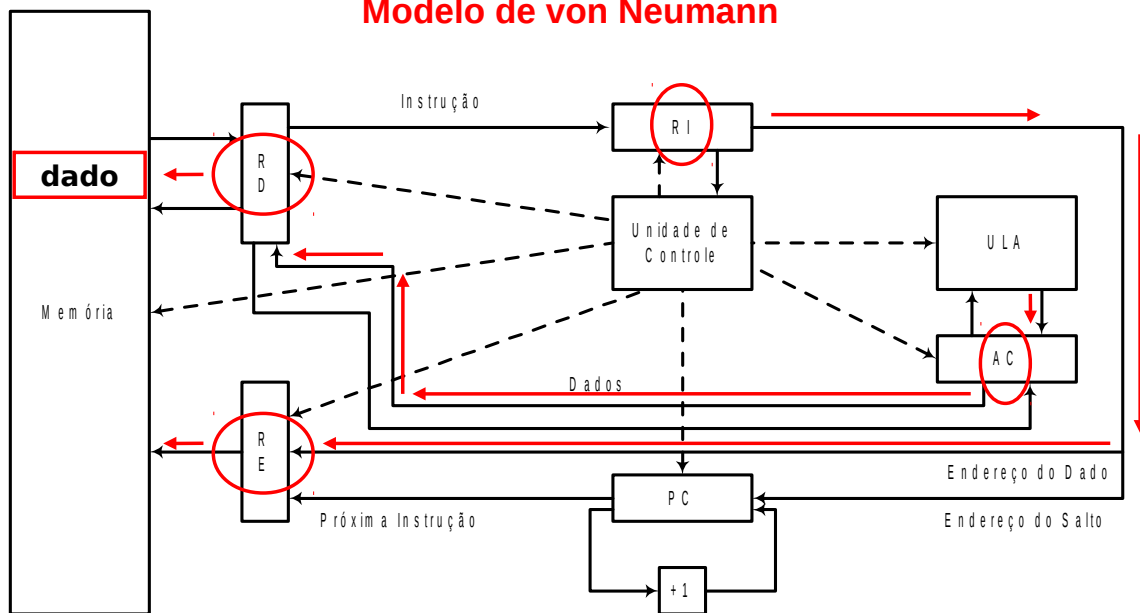
Mostra a interação entre os elementos da Arquitetura de von Neumann



17

Armazena Resultado

Modelo de von Neumann

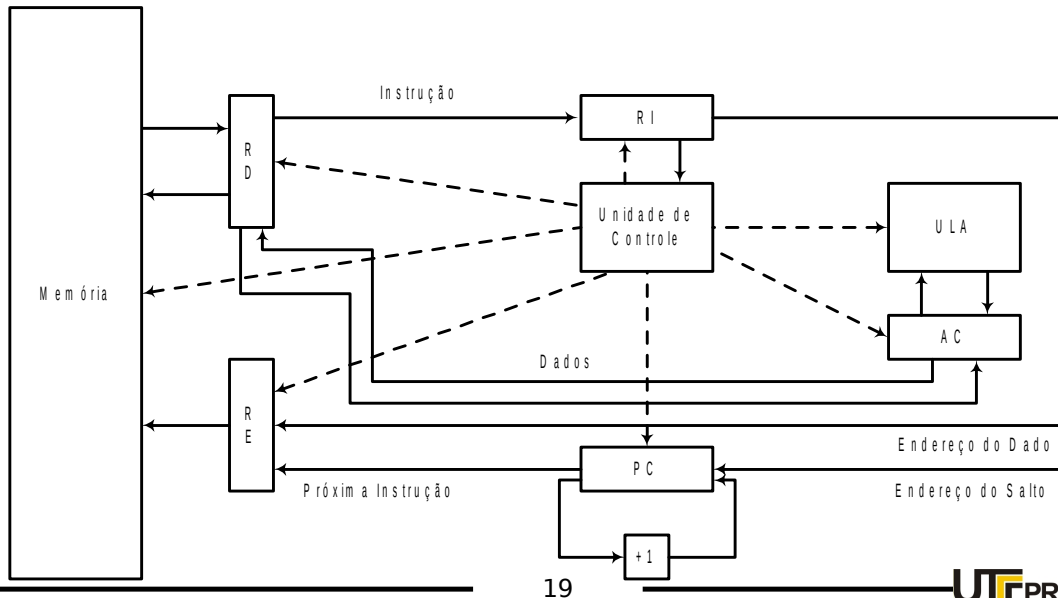


18

Exercício

Indique no diagrama da arquitetura de von Neumann:

1. O percurso da busca de instrução
2. O percurso da busca de operando

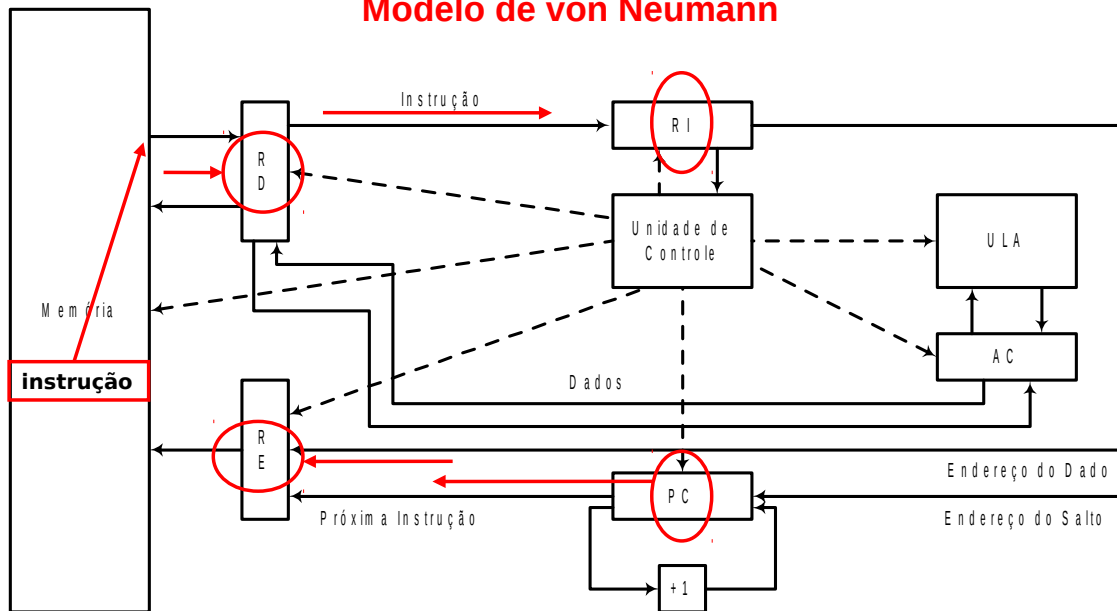


19

UTFPR

Solução: Busca de Instrução

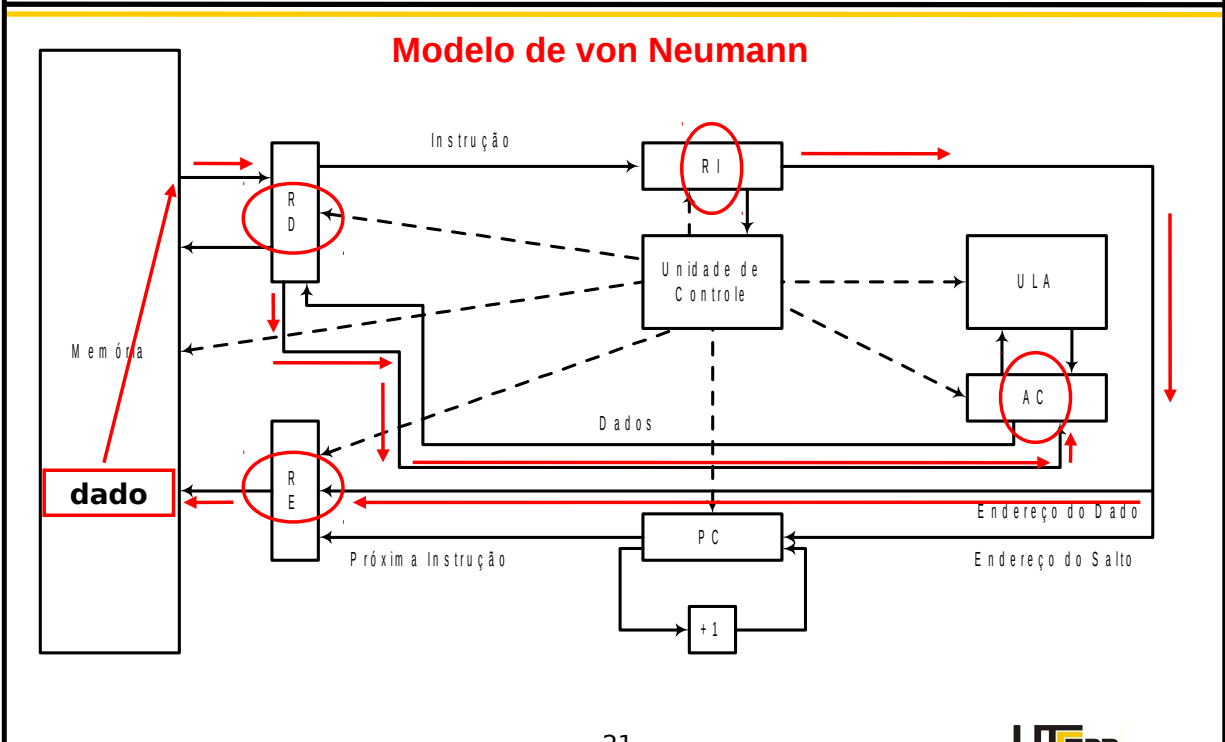
Modelo de von Neumann



20

UTFPR

Solução: Busca de Operando



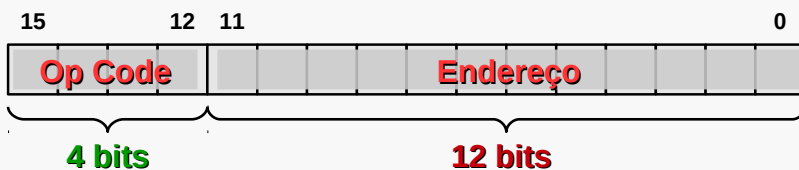
21

Exemplo de Ciclo de Instrução

Considere um processador com as seguintes características:

- Formato de instrução: 16 bits

- 4 bits para Código de Operação (Op Code)
- 12 bits para o campo de Endereços



Formato da Instrução

4 bits

12 bits

$2^4 = 16$ Instruções Diferentes

$2^{12} = 4096$ (4 K) Endereços Diferentes

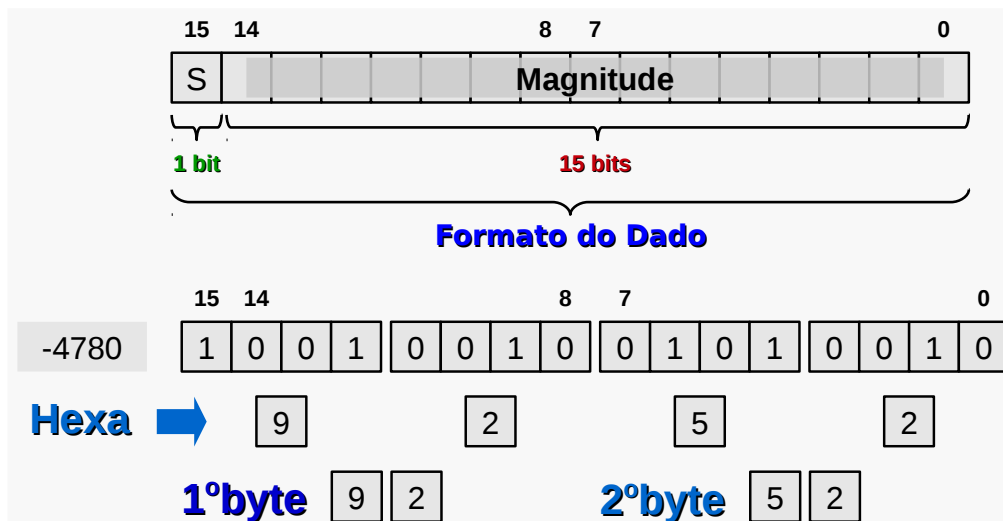
[1 K = $2^{10} = 1024$]

22

Exemplo de Ciclo de Instrução

Considere um processador com as seguintes características:

- Dado de 16 bits representado em Sinal-Magnitude



23

Conjunto de Instruções

- Carregar dados da Memória
 - LOAD → LDA
- Armazenar dados na Memória
 - STORE → STA
- Adição
 - ADD
- Subtração
 - SUB

24

Conjunto de Instruções

LDA

Instrução: *LDA* → Load AC from Memory

Op Code → **Binário:** 0001b

→ **Hexadecimal:** 1h

Operação: $AC \leftarrow [MEM]$

Descrição: Carrega AC a partir do conteúdo da posição de memória endereçada por [MEM]

Conjunto de Instruções

STA

Instrução: *STA* → Store AC to Memory

Op Code → **Binário:** 0010b

→ **Hexadecimal:** 2h

Operação: $[MEM] \leftarrow AC$

Descrição: Armazena o conteúdo do Acumulador na posição de memória endereçada por [MEM]

Conjunto de Instruções

ADD

Instrução: *ADD* → Add to AC from Memory

Op Code → **Binário:** 0011b

→ **Hexadecimal:** 3h

Operação: $AC \leftarrow AC + [MEM]$

Descrição: Soma o conteúdo do Acumulador ao conteúdo da posição de memória endereçada por [MEM]

Conjunto de Instruções

SUB

Instrução: *SUB* → Subtract to AC from Memory

Op Code → **Binário:** 0100b

→ **Hexadecimal:** 4h

Operação: $AC \leftarrow AC - [MEM]$

Descrição: Subtrai o conteúdo do Acumulador ao conteúdo da posição de memória endereçada por [MEM]

Programa Exemplo

Mnemônicos	Binário		Hexadecimal
	OpCode	Endereço	
LDA AC, [940]	0001	100101000000	1940h
ADD AC, [942]	0011	100101000010	3942h
SUB AC, [944]	0100	100101000100	4944h
STA [942], AC	0010	100101000010	2942h

Considerações Iniciais:

- PC contém inicialmente o valor 300H, o programa inicia nesse endereço
- Conteúdo da posição de memória [940H] é 0003H
- Conteúdo da posição de memória [942H] é 0002H
- Conteúdo da posição de memória [944H] é 0004H

29

Execução do Programa

Ciclo de Busca

	Memória
300	1940
302	3942
304	4944
306	2942
...	...
...	...
940	0003
942	0002
944	0004

Registradores	
300	PC
	RI
	AC

1940: AC ← [940]

30

Execução do Programa

Ciclo de Busca

	Memória
300	1940
302	3942
304	4944
306	2942
...	...
...	...
940	0003
942	0002
944	0004

Registradores	
300	PC
	RI
	AC

1940: AC ← [940]

31

Execução do Programa

Ciclo de Busca

	Memória
300	1940
302	3942
304	4944
306	2942
...	...
...	...
940	0003
942	0002
944	0004

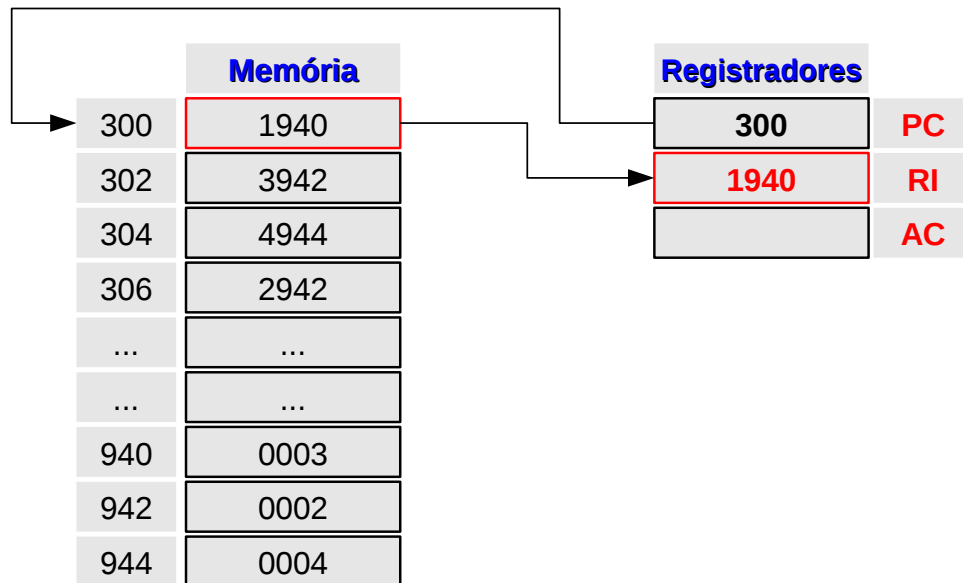
Registradores	
300	PC
	RI
	AC

1940: AC ← [940]

32

Execução do Programa

Ciclo de Busca

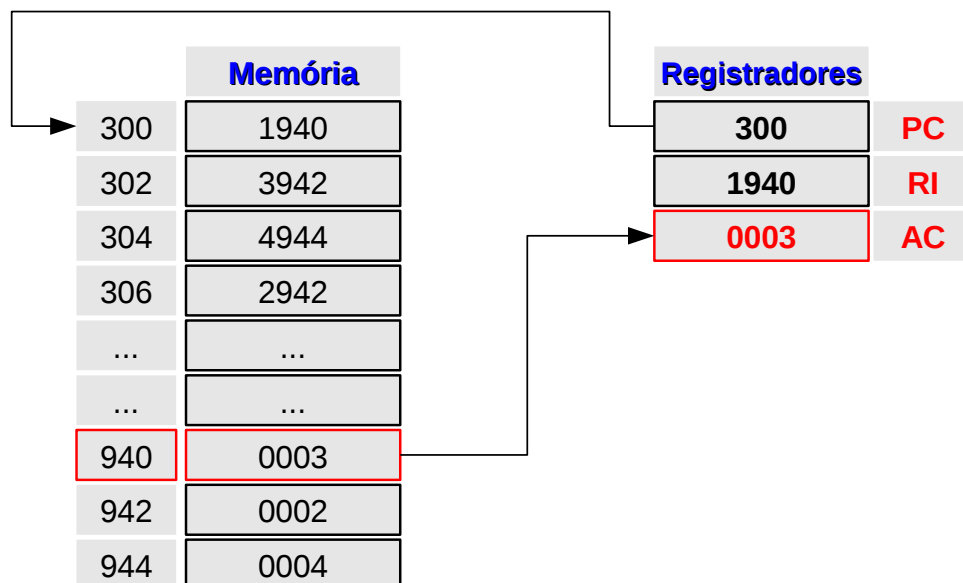


1940: AC ← [940]

33

Execução do Programa

Execução



1940: AC ← [940]

34

Execução do Programa

Ciclo de Busca

Memória		Registradores	
300	1940	302	PC
302	3942	1940	RI
304	4944	0003	AC
306	2942		
...	...		
...	...		
940	0003		
942	0002		
944	0004		

35

Execução do Programa

Ciclo de Busca

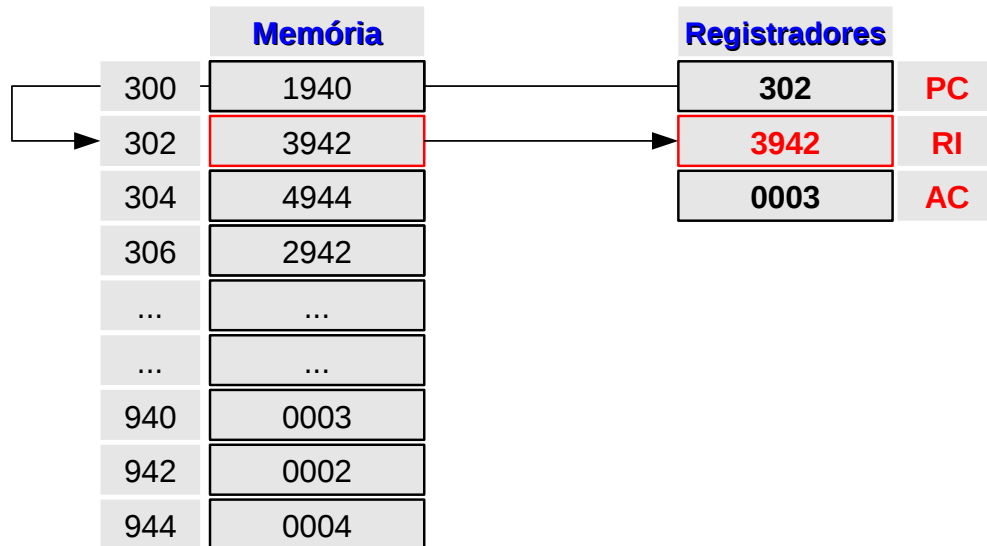
Memória		Registradores	
300	1940	302	PC
302	3942	1940	RI
304	4944	0003	AC
306	2942		
...	...		
...	...		
940	0003		
942	0002		
944	0004		

3942: AC ← AC + [942]

36

Execução do Programa

Ciclo de Busca

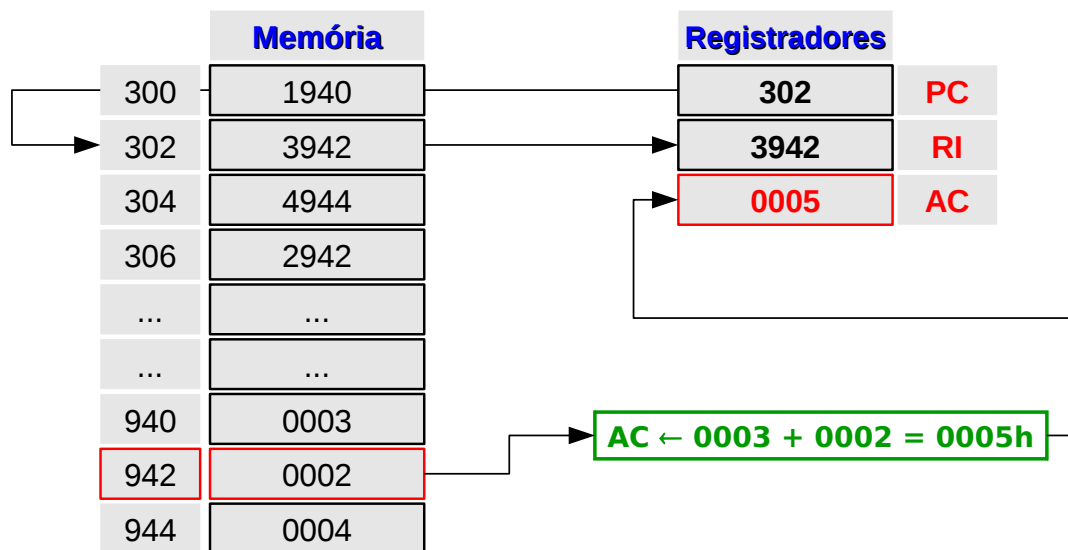


3942: AC ← AC + [942]

37

Execução do Programa

Execução



3942: AC ← AC + [942]

38

Execução do Programa

Ciclo de Busca

Memória		Registradores	
300	1940	304	PC
302	3942	3942	RI
304	4944	0005	AC
306	2942		
...	...		
...	...		
940	0003		
942	0002		
944	0004		

39

Execução do Programa

Ciclo de Busca

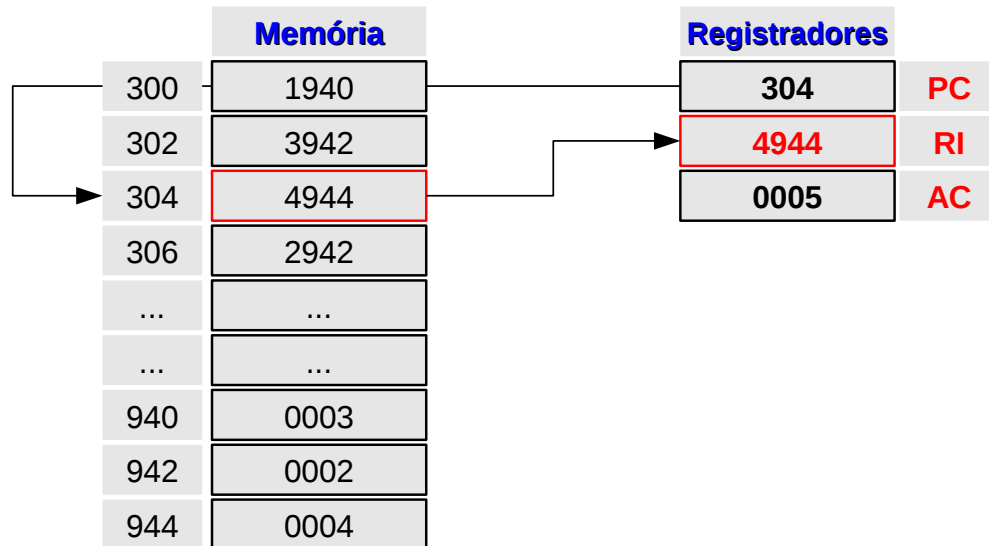
Memória		Registradores	
300	1940	304	PC
302	3942	3942	RI
304	4944	0005	AC
306	2942		
...	...		
...	...		
940	0003		
942	0002		
944	0004		

4944: AC ← AC - [944]

40

Execução do Programa

Ciclo de Busca

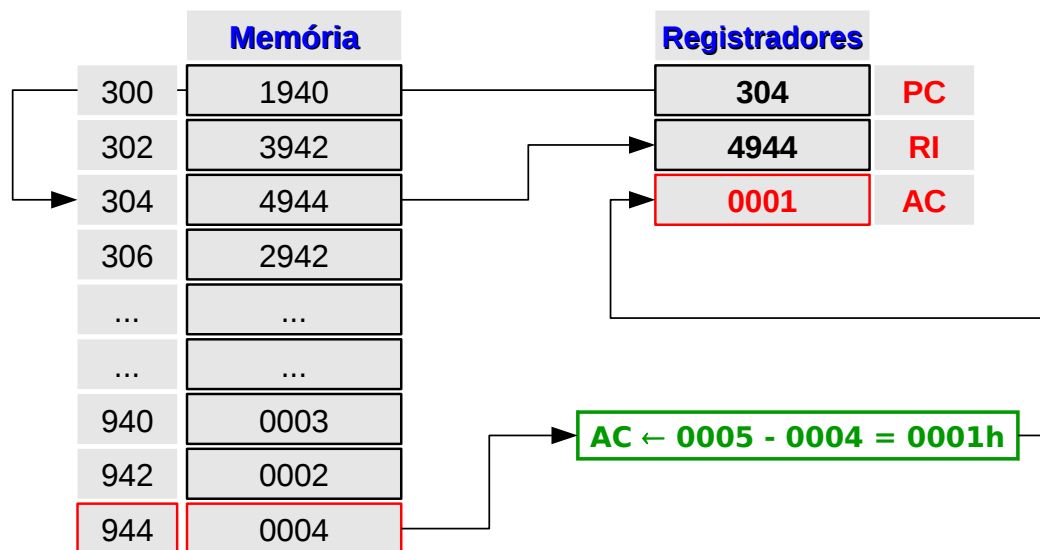


4944: AC ← AC - [944]

41

Execução do Programa

Execução



4944: AC ← AC - [944]

42

Execução do Programa

Ciclo de Busca

Memória		Registradores	
300	1940	306	PC
302	3942	4944	RI
304	4944	0001	AC
306	2942		
...	...		
...	...		
940	0003		
942	0002		
944	0004		

43

Execução do Programa

Ciclo de Busca

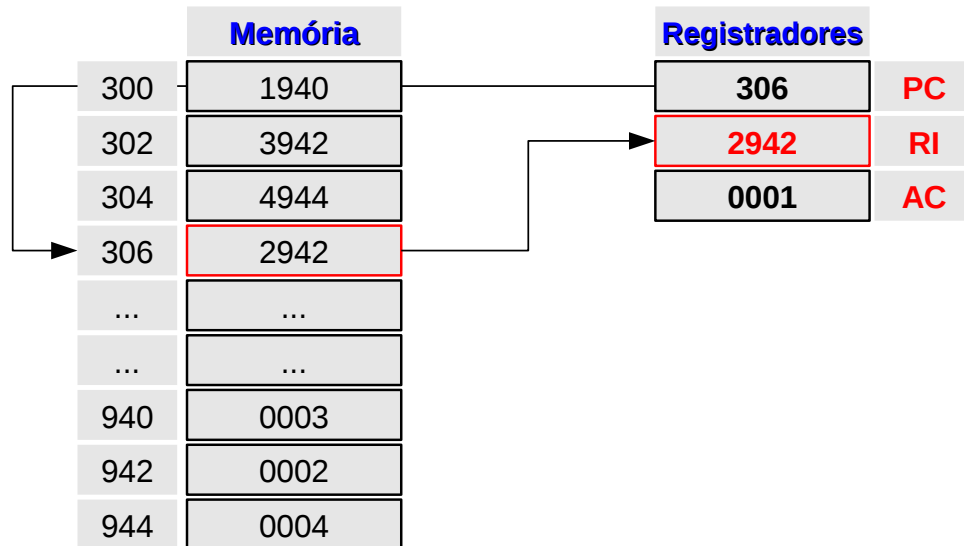
Memória		Registradores	
300	1940	306	PC
302	3942	4944	RI
304	4944	0001	AC
306	2942		
...	...		
...	...		
940	0003		
942	0002		
944	0004		

2942: [942] ← AC

44

Execução do Programa

Ciclo de Busca

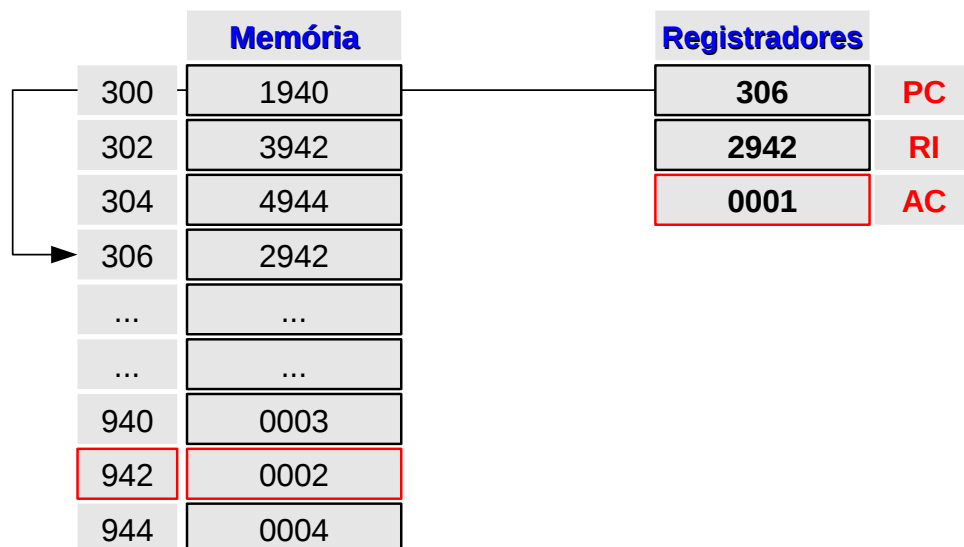


2942: [942] ← AC

45

Execução do Programa

Execução

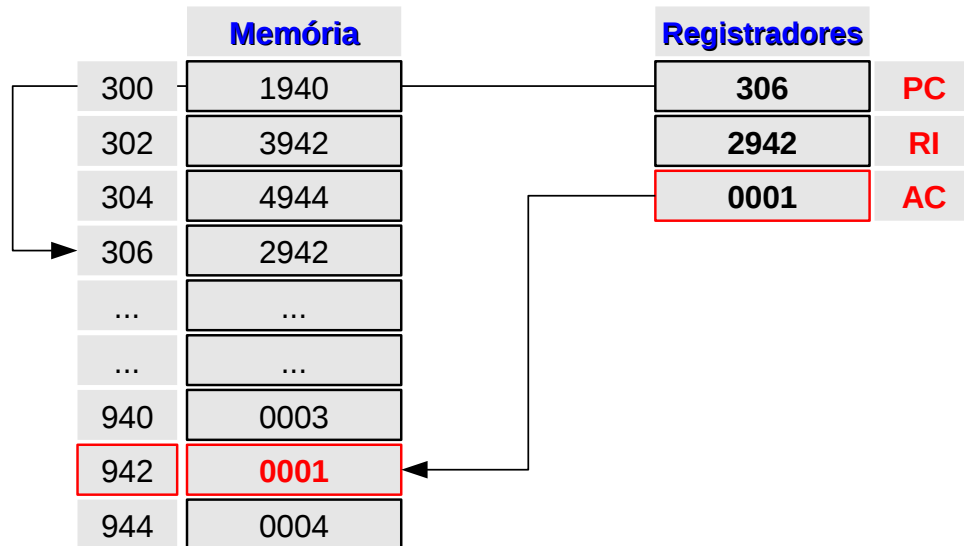


2942: [942] ← AC

46

Execução do Programa

Execução



2942: [942] ← AC

47

Exercício

Faça os diagramas da memória e dos registradores (PC, RI e AC) da CPU para mostrar a execução do seguinte programa:

Carregar o AC com o conteúdo da posição [MEM]:

$0001b \Rightarrow AC \leftarrow [MEM]$

AND do conteúdo da posição [MEM] com AC:
[MEM]

$0110b \Rightarrow AC \leftarrow AC \text{ and } [MEM]$

Armazenar o resultado na posição [MEM]:

$0010b \Rightarrow [MEM] \leftarrow AC$

Programa

LDA AC, [800h]
AND AC, [802h]
STA [804h], AC

Considere Inicialmente

[PC] = 500h
[800h] = F0E6
[802h] = 0FBF
[804h] = 0005

48

Solução

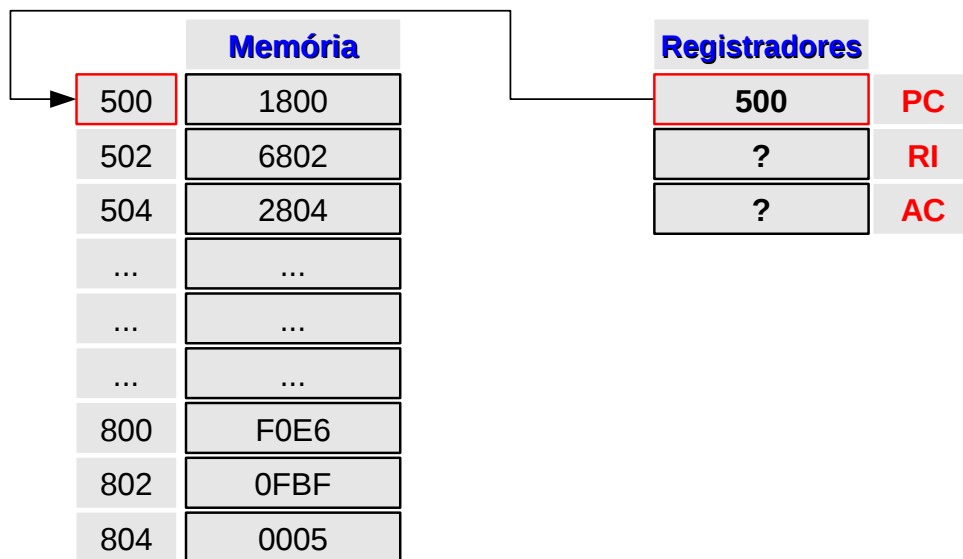
Ciclo de Busca

	Memória
500	1800
502	6802
504	2804
...	...
...	...
...	...
800	F0E6
802	0FBF
804	0005

Registradores	
500	PC
?	RI
?	AC

Solução

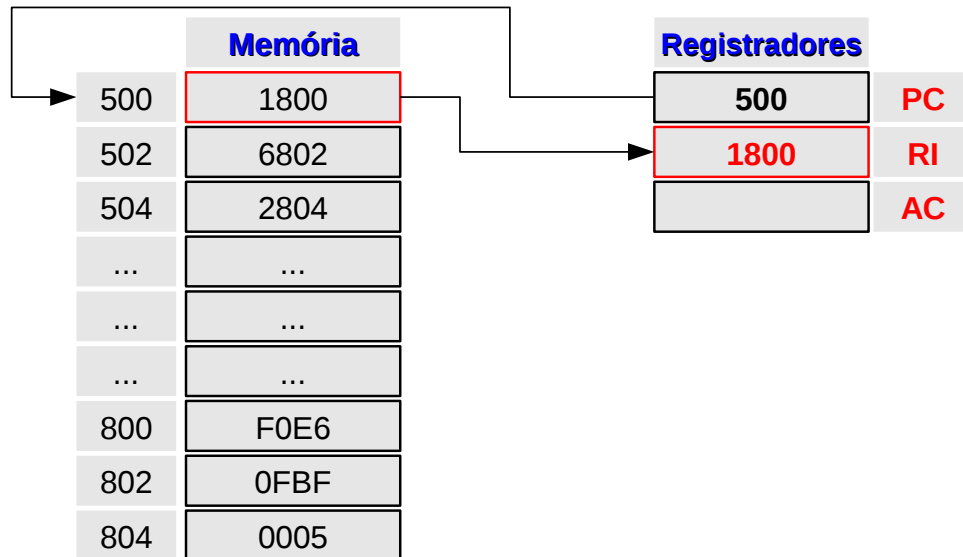
Ciclo de Busca



1800: AC ← [800]

Solução

Ciclo de Busca

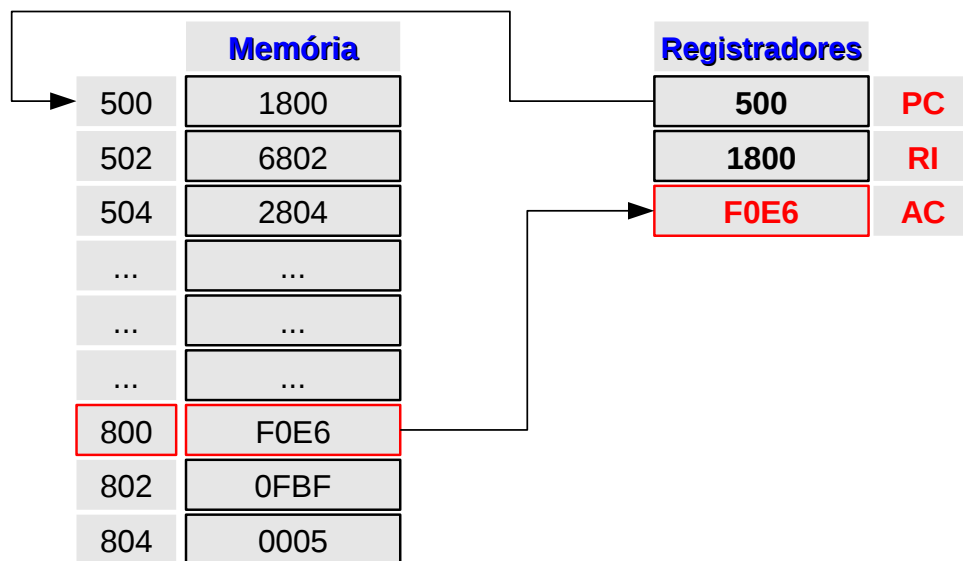


1800: AC ← [800]

51

Solução

Execução



1800: AC ← [800]

52

Solução

Ciclo de Busca

	Memória
500	1800
502	6802
504	2804
...	...
...	...
...	...
800	F0E6
802	0FBF
804	0005

Registradores	
502	PC
1800	RI
F0E6	AC

1800: AC ← [800]

53

Solução

Ciclo de Busca

	Memória
500	1800
502	6802
504	2804
...	...
...	...
...	...
800	F0E6
802	0FBF
804	0005

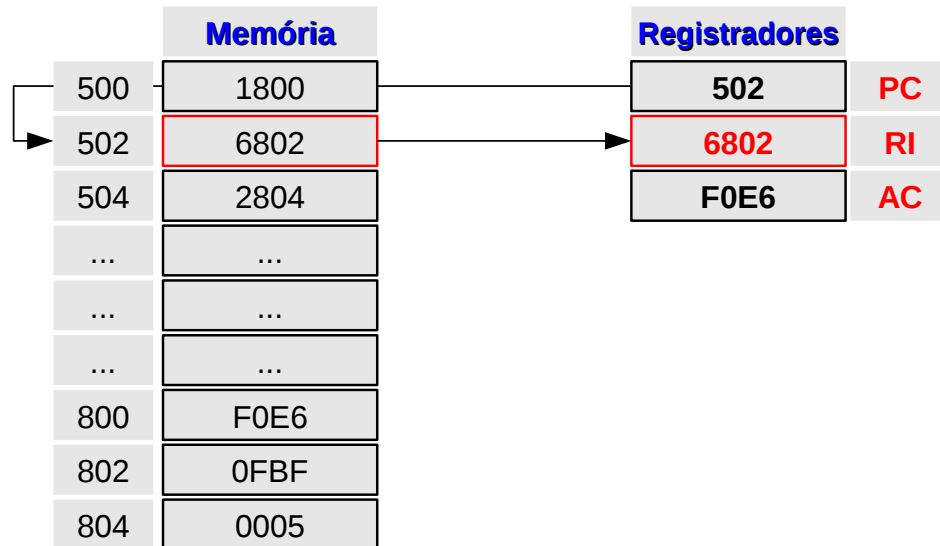
Registradores	
502	PC
1800	RI
F0E6	AC

6802: AC AND [802]

54

Solução

Ciclo de Busca

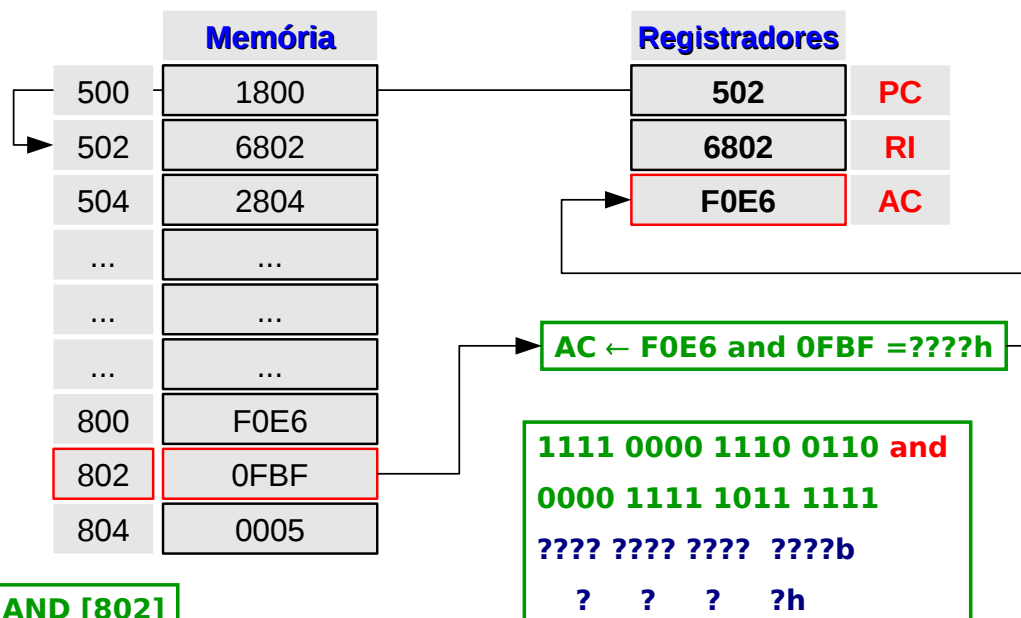


6802: AC AND [802]

55

Solução

Execução

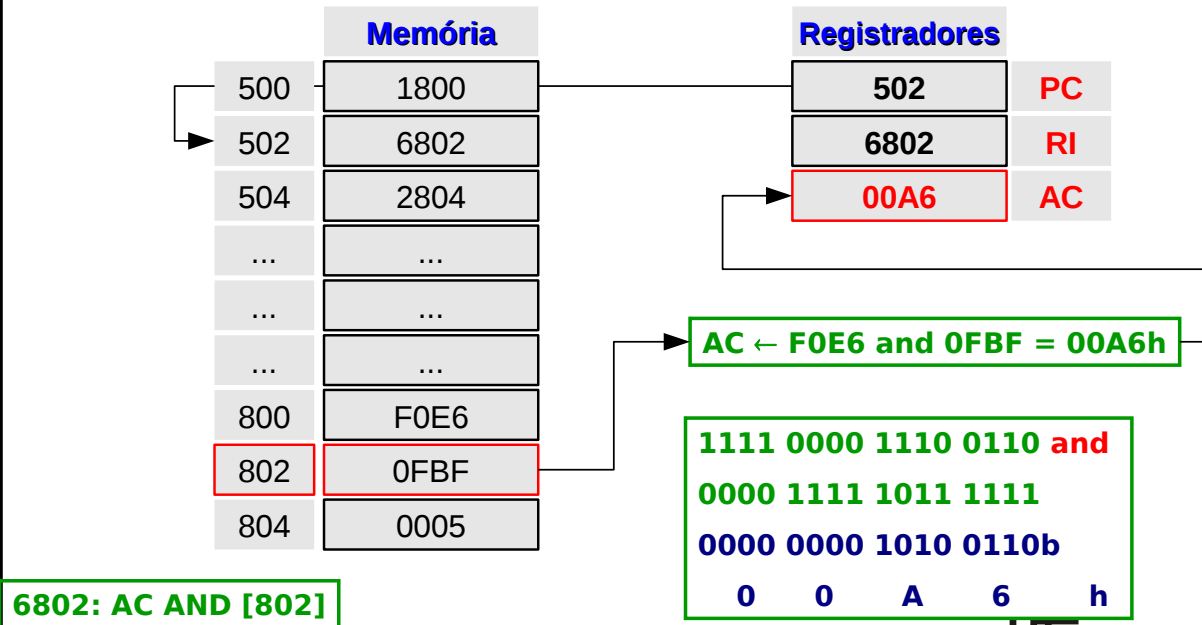


6802: AC AND [802]

56

Solução

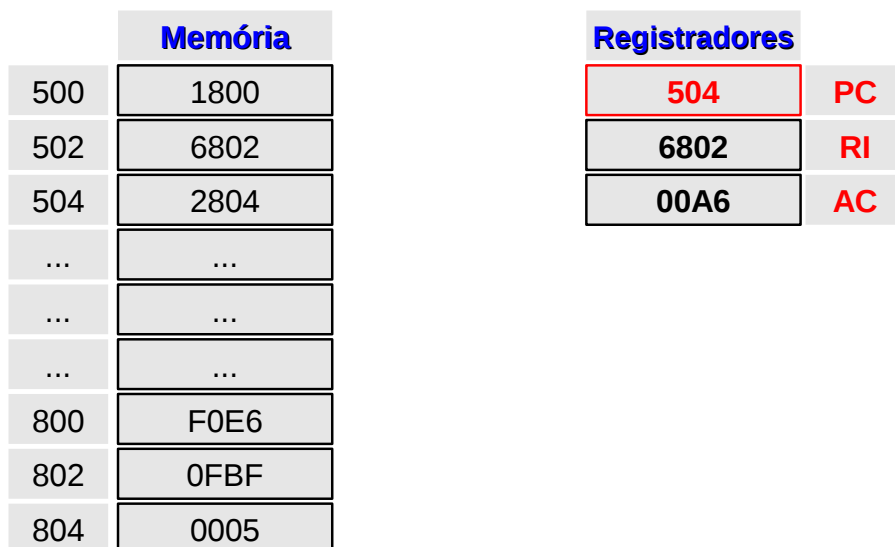
Execução



57

Solução

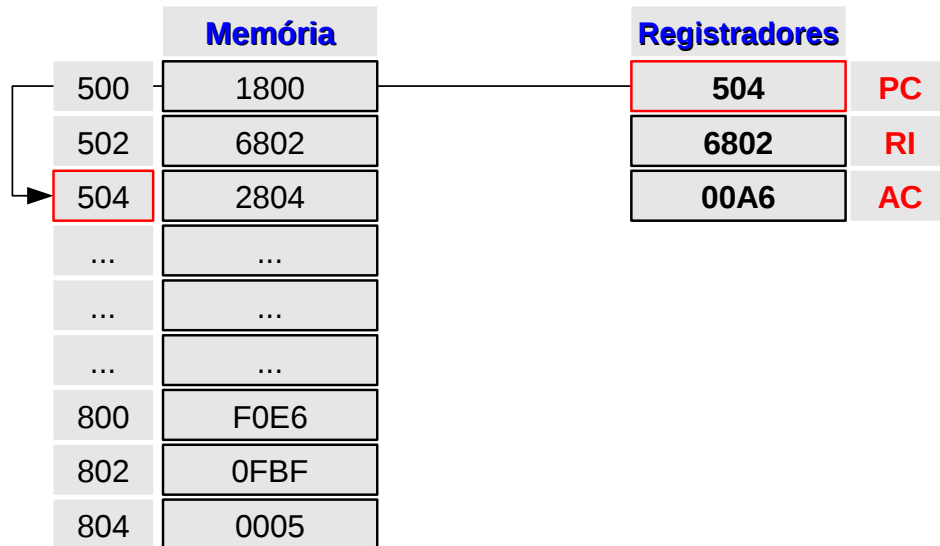
Ciclo de Busca



58

Solução

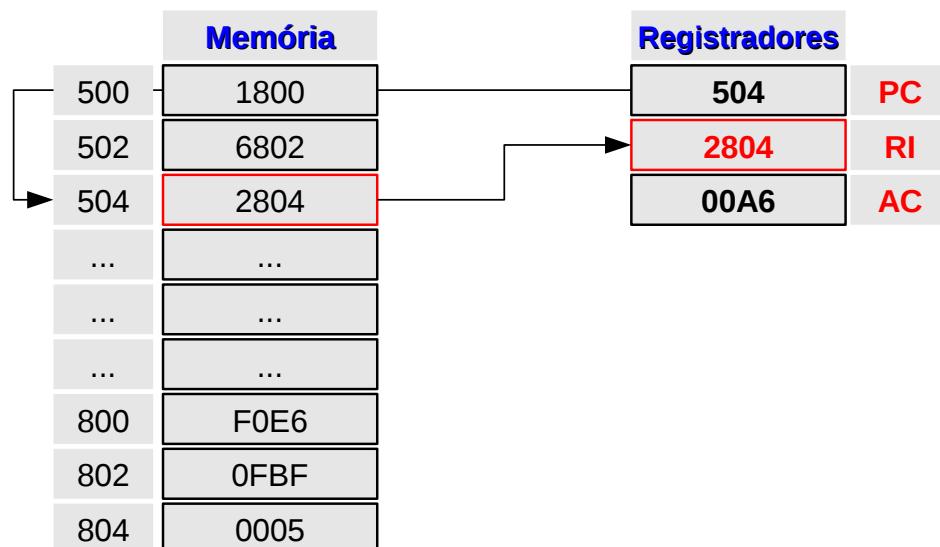
Ciclo de Busca



2804: [804] ← AC

Solução

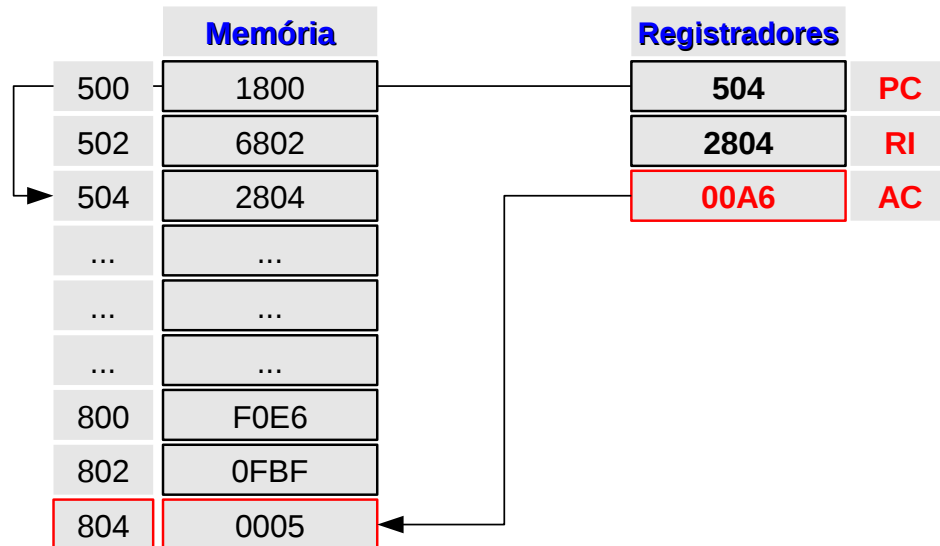
Ciclo de Busca



2804: [804] ← AC

Solução

Execução

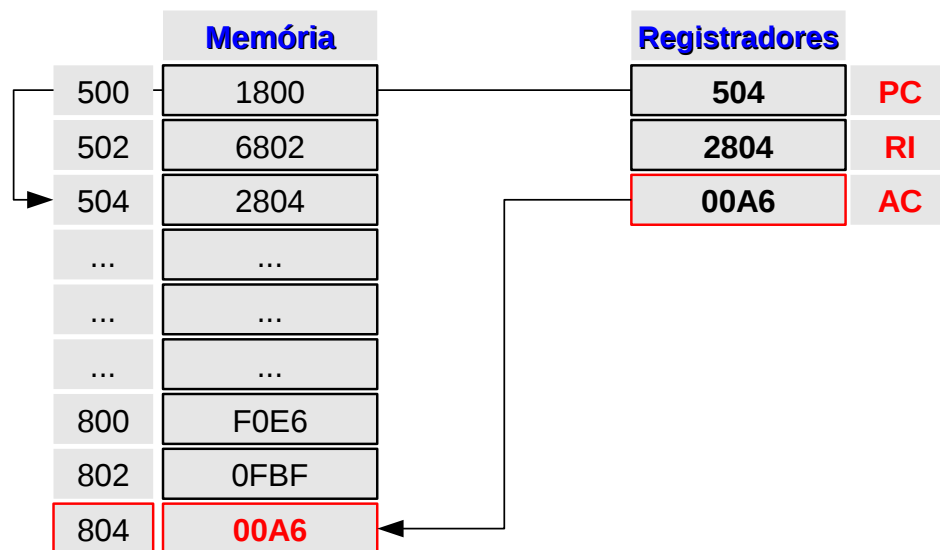


2804: [804] ← AC

61

Solução

Execução



2804: [804] ← AC

62

Referências

- Notas de aula do Prof. João Angelo Martini do DIN-UEM.



Universidade Tecnológica Federal do Paraná – UTFPR
Bacharelado em Ciência da Computação

BCC33B – Arquitetura e Organização de Computadores

Prof. Rogério A. Gonçalves

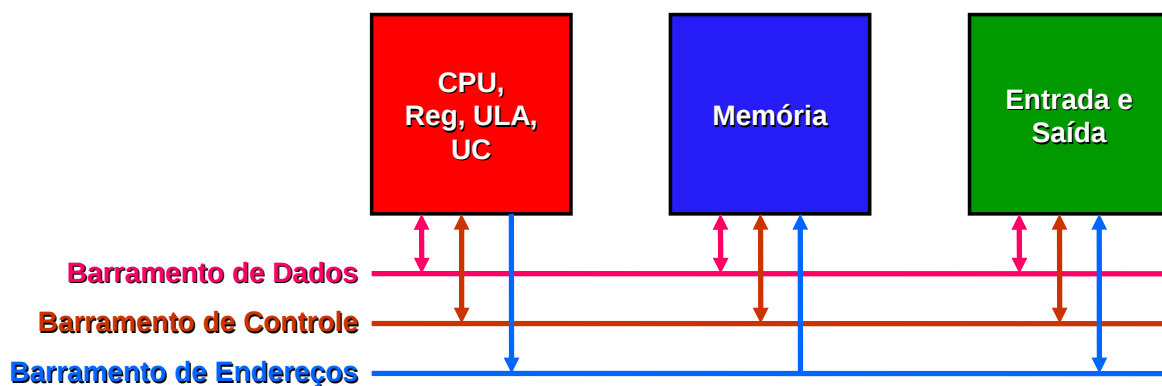
rogerioag@utfpr.edu.br

Aula 002

- Sistema Computacional

Sistema de Computador

Componentes Básicos do Sistema de Computador

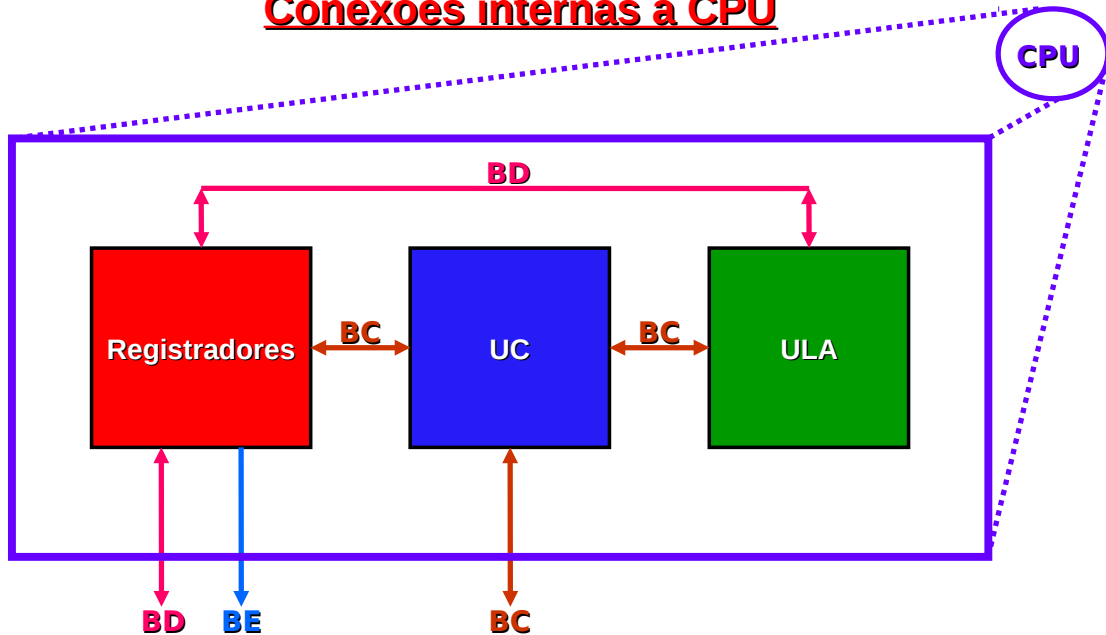


Barramentos:

- BD: Barramento de Dados transportam a informação útil (dados e instruções)
- BC: Barramento de Controle transportam informações de controle
- BE: Barramento de Endereços identificam o destino das informações

Zoom da CPU

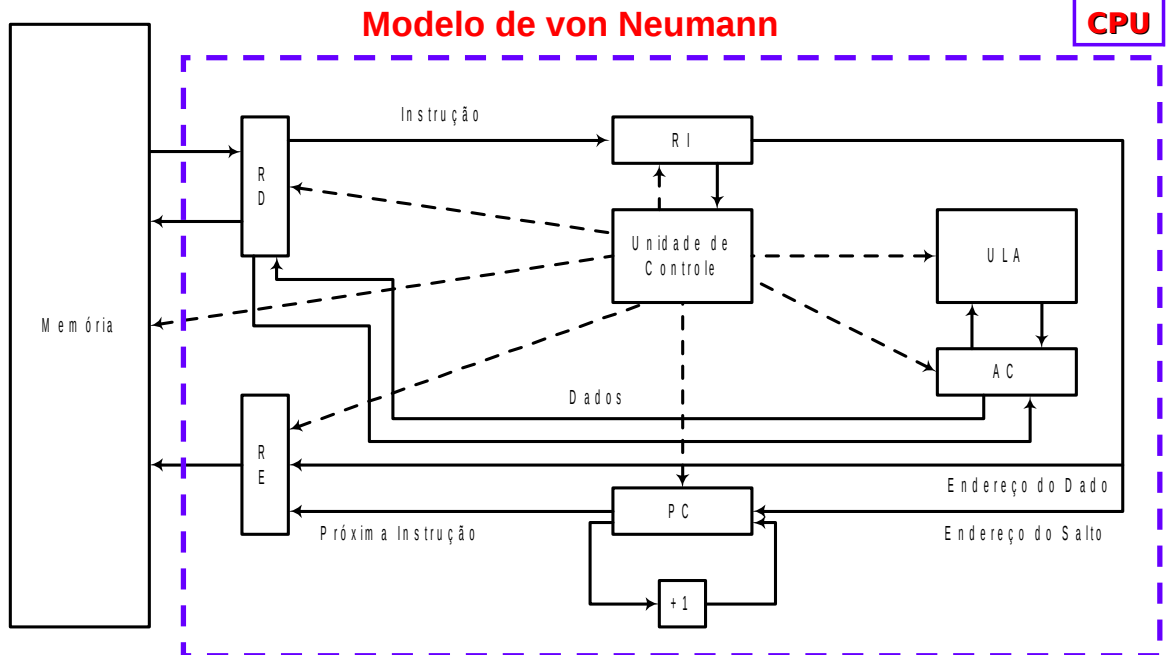
Conexões internas à CPU



4

Arquitetura de von Neumann

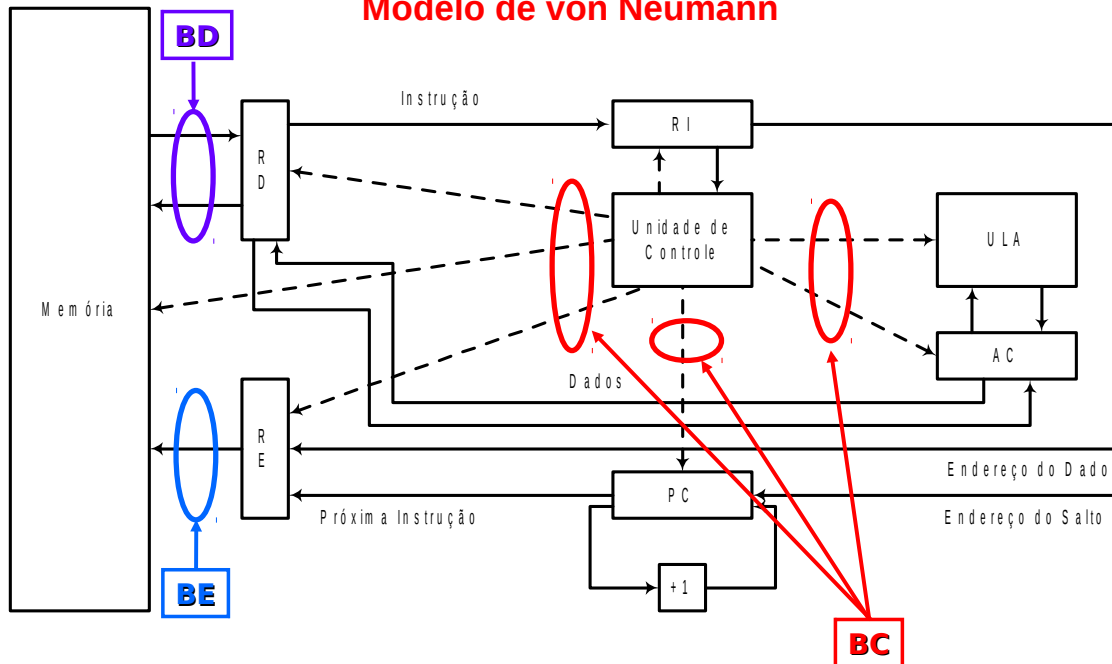
Modelo de von Neumann



5

Arquitetura de von Neumann

Modelo de von Neumann

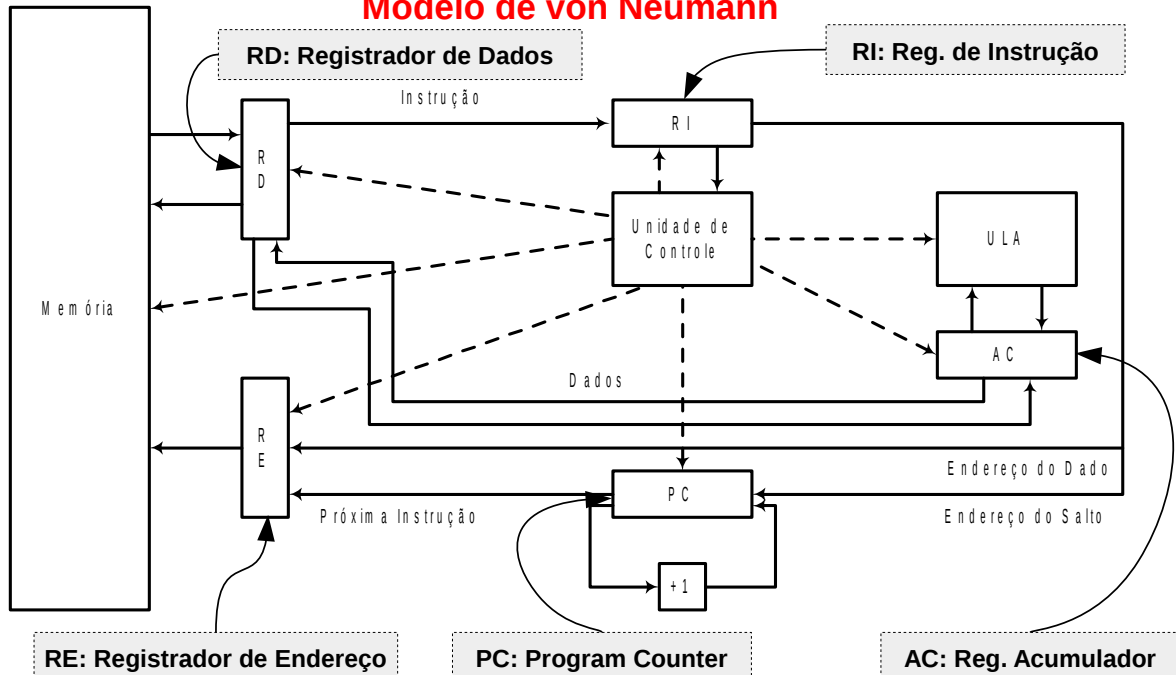


6

UTFPR

Arquitetura de von Neumann

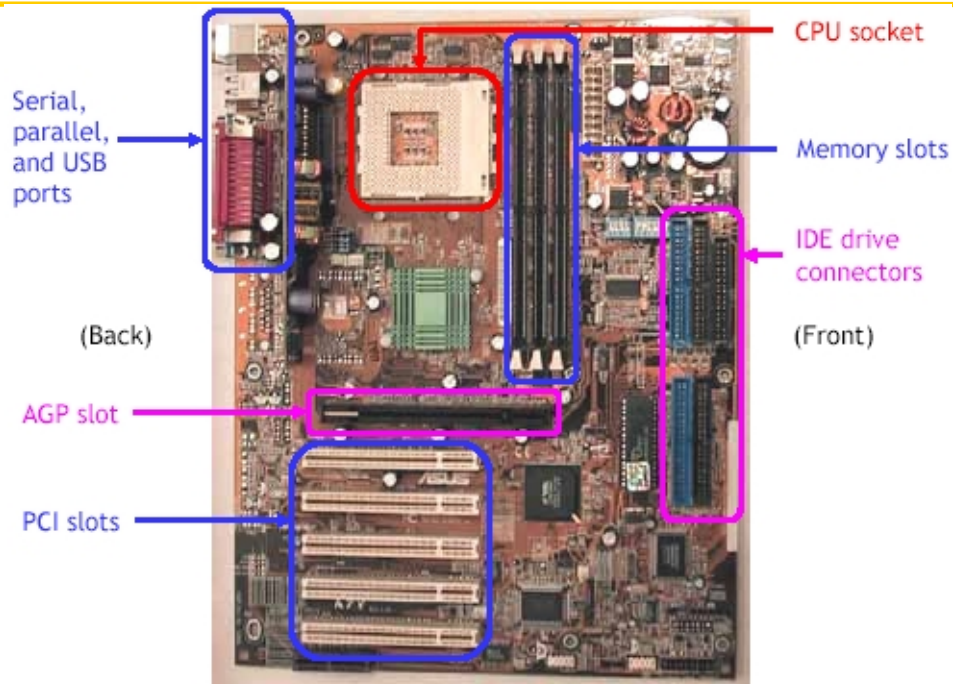
Modelo de von Neumann



7

UTFPR

Exemplos de Barramentos

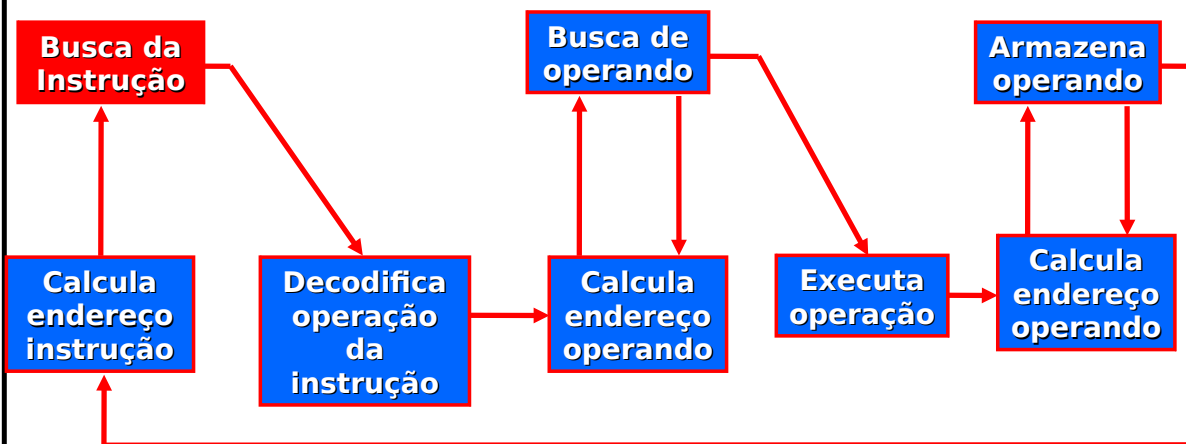


8

Ciclo de Instrução

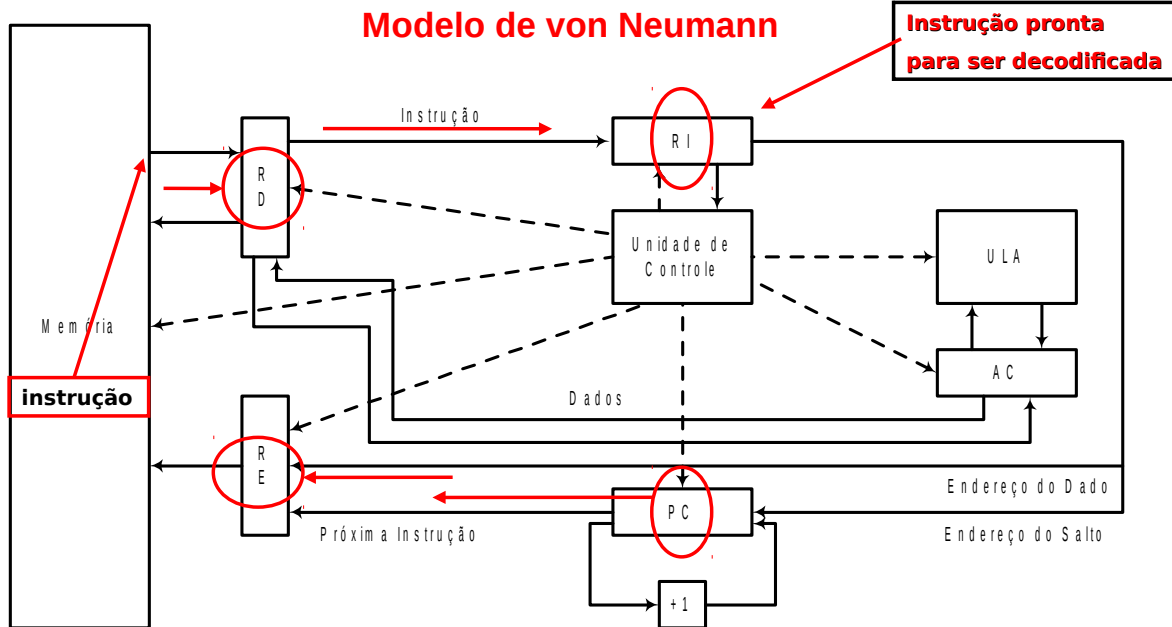
Ciclo de Instrução

Mostra a interação entre os elementos da Arquitetura de von Neumann



9

Busca de Instrução

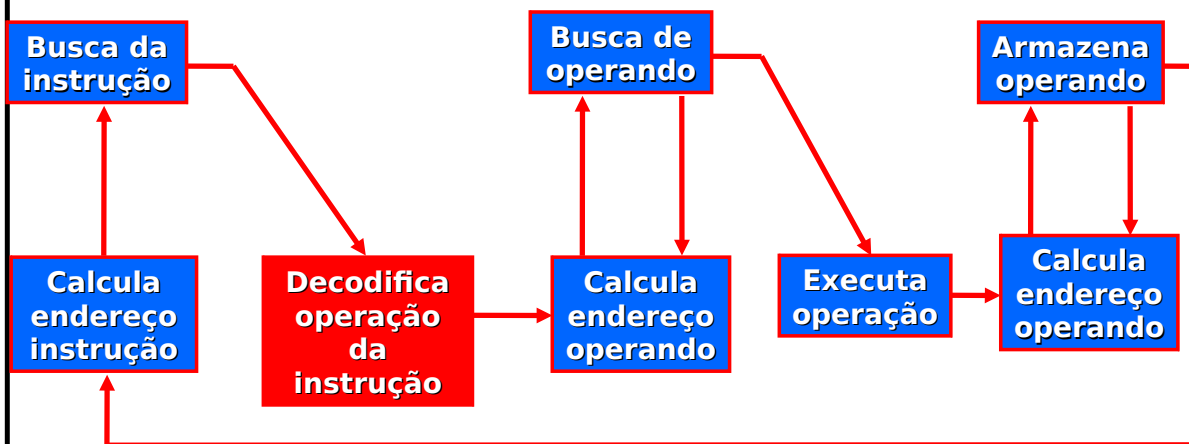


10

Ciclo de Instrução

Ciclo de Instrução

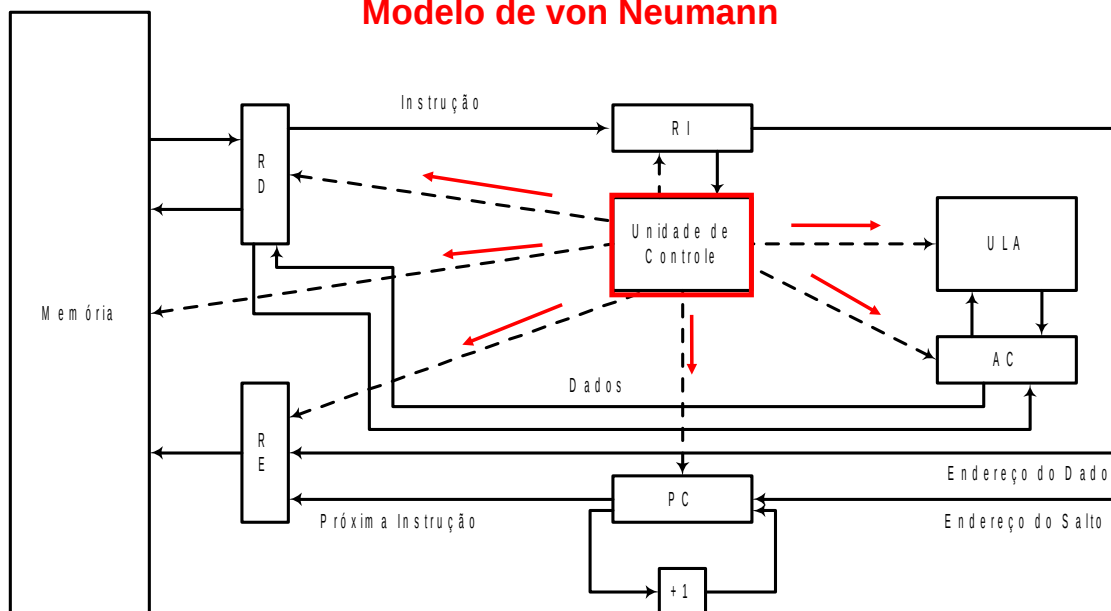
Mostra a interação entre os elementos da Arquitetura de von Neumann



11

Decodificação de Instrução

Modelo de von Neumann

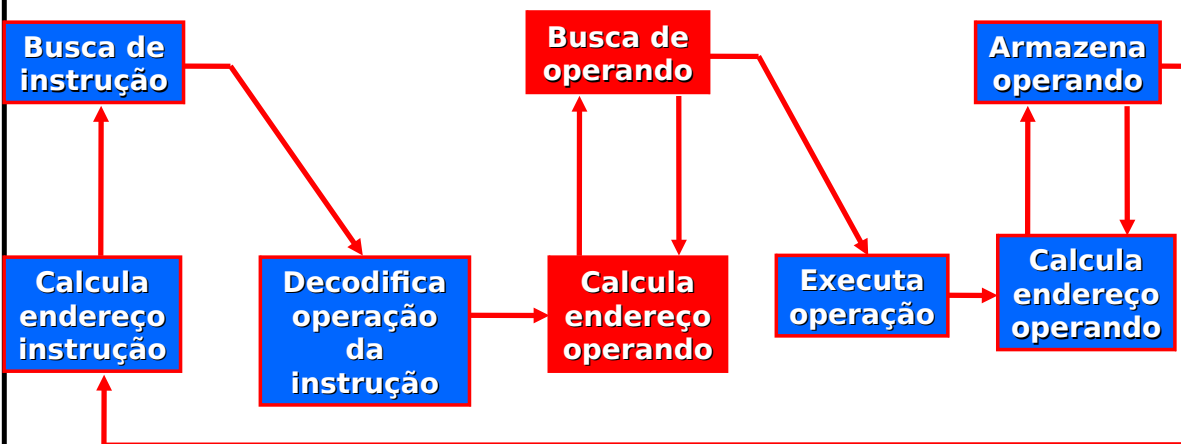


12

Ciclo de Instrução

Ciclo de Instrução

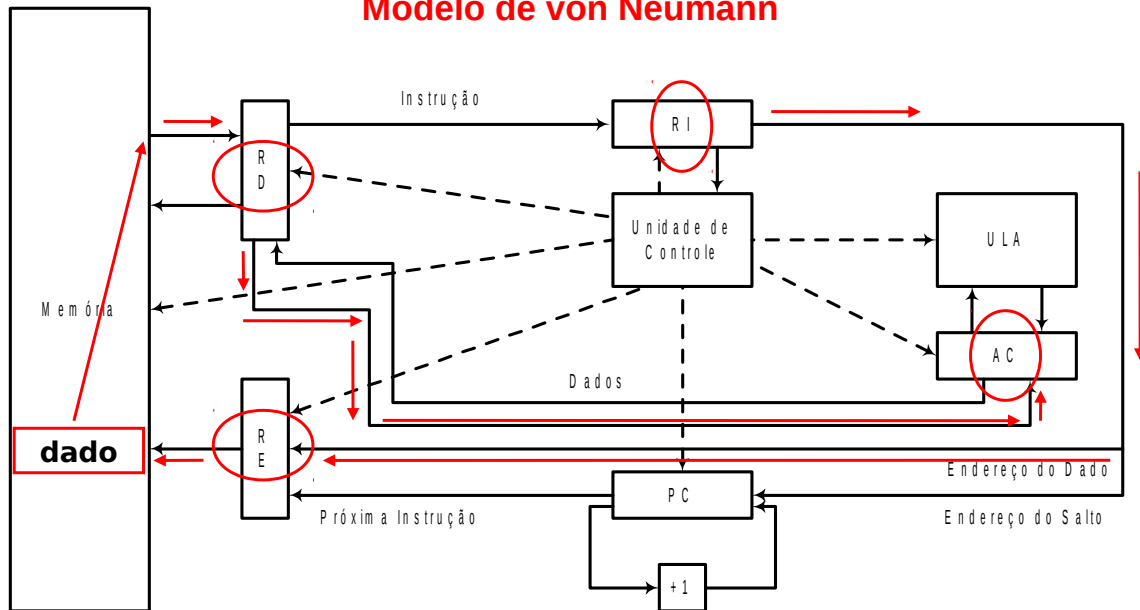
Mostra a interação entre os elementos da Arquitetura de von Neumann



13

Busca de Operando

Modelo de von Neumann

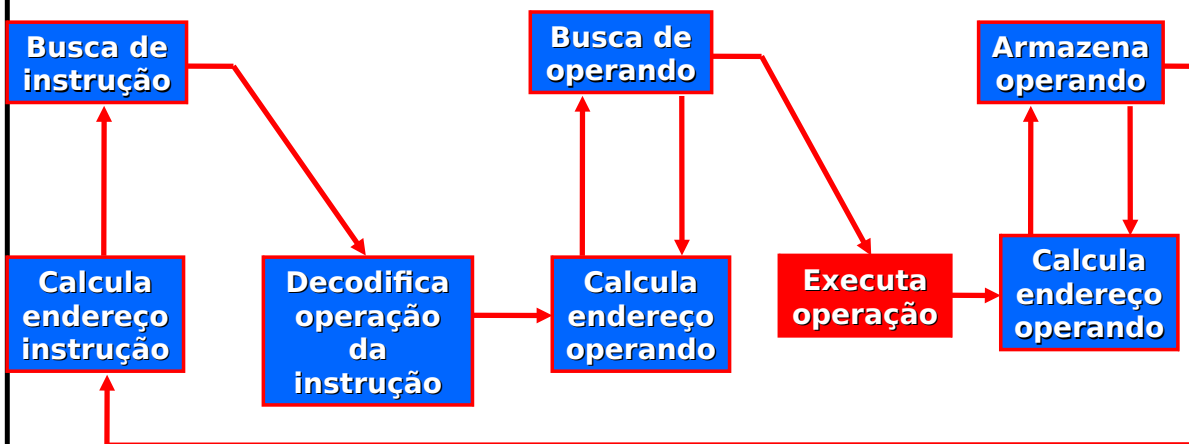


14

Ciclo de Instrução

Ciclo de Instrução

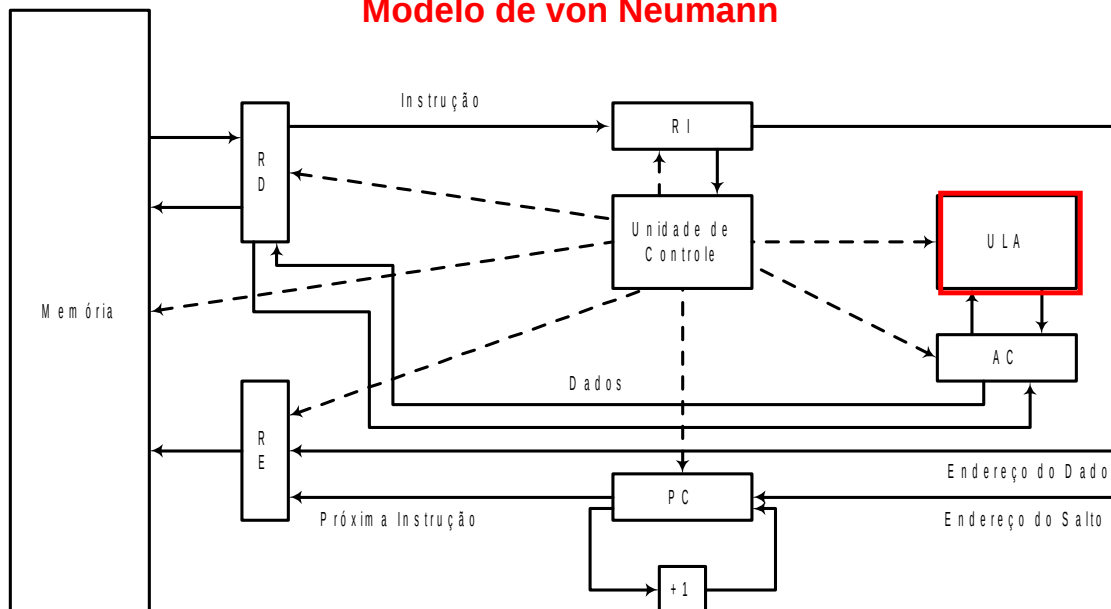
Mostra a interação entre os elementos da Arquitetura de von Neumann



15

Execução da Instrução

Modelo de von Neumann

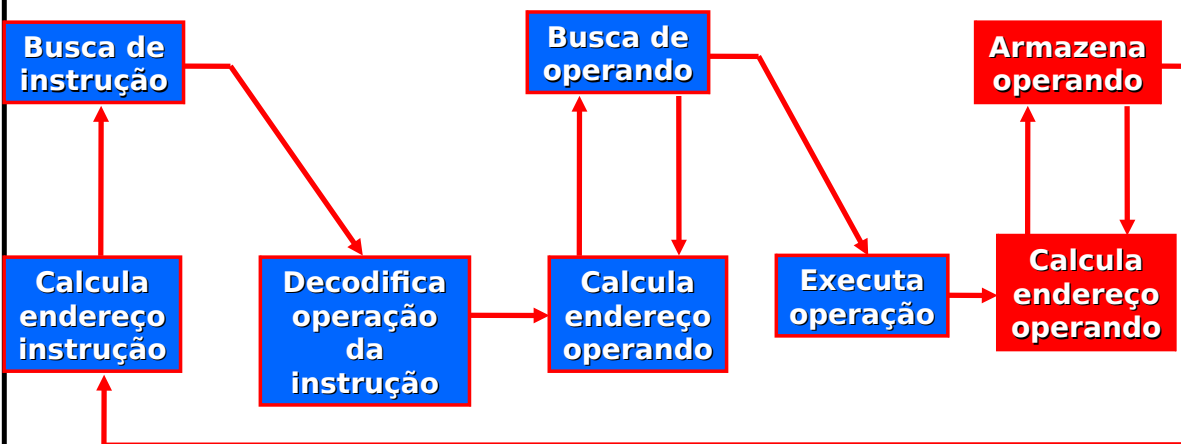


16

Ciclo de Instrução

Ciclo de Instrução

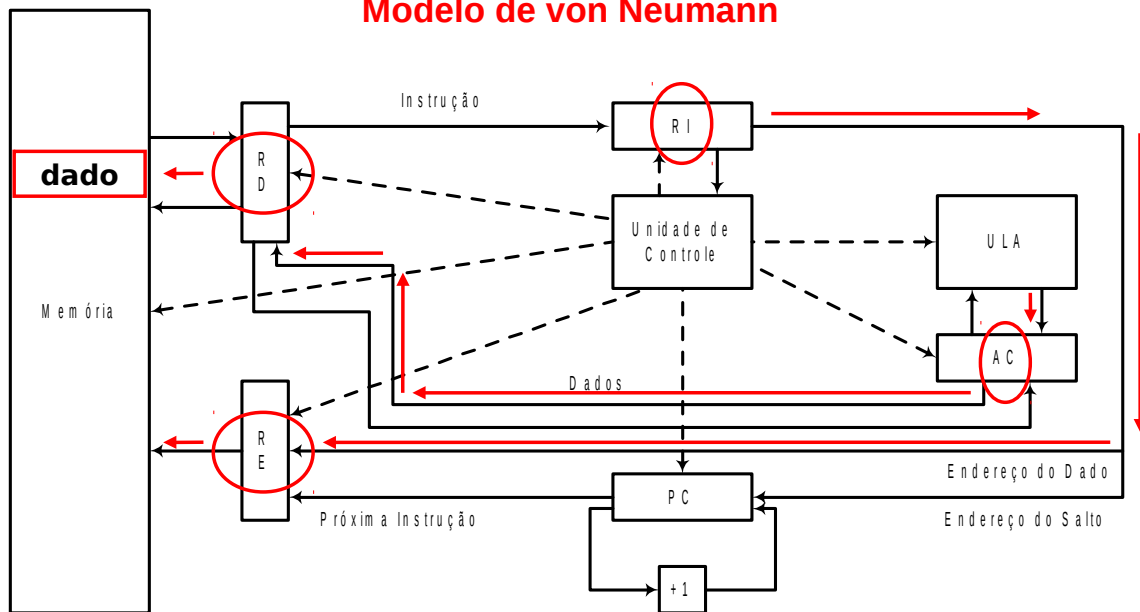
Mostra a interação entre os elementos da Arquitetura de von Neumann



17

Armazena Resultado

Modelo de von Neumann

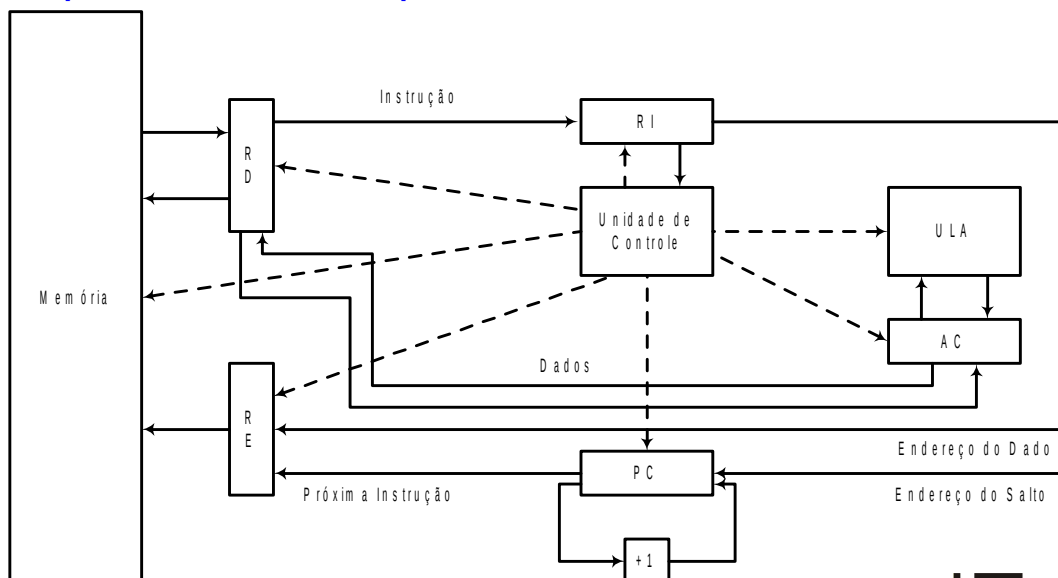


18

Exercício

Indique no diagrama da arquitetura de von Neumann:

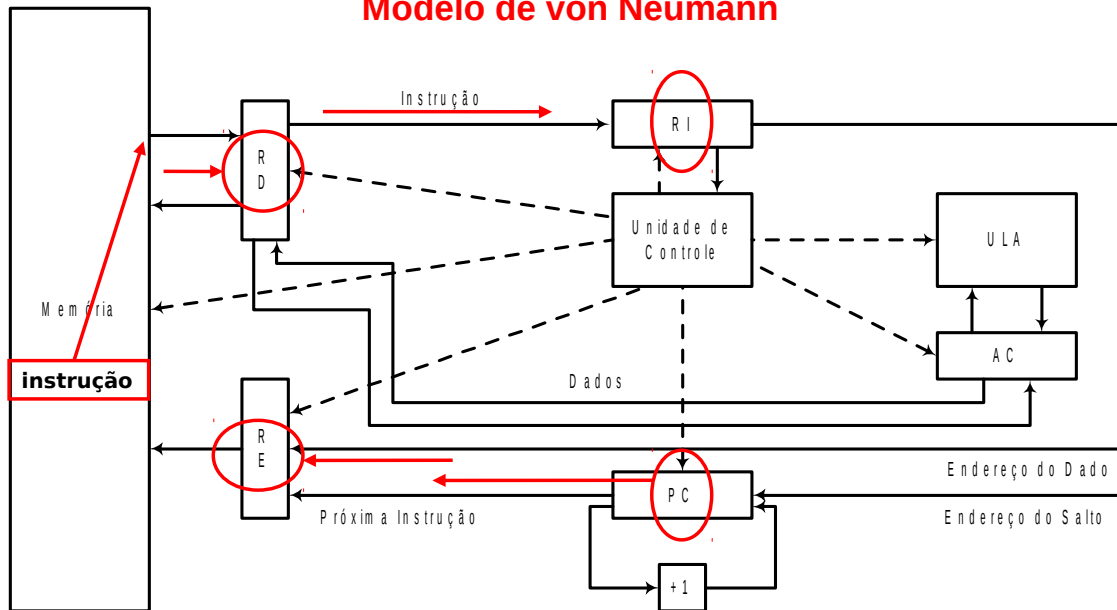
1. O percurso da busca de instrução
2. O percurso da busca de operando



19

Solução: Busca de Instrução

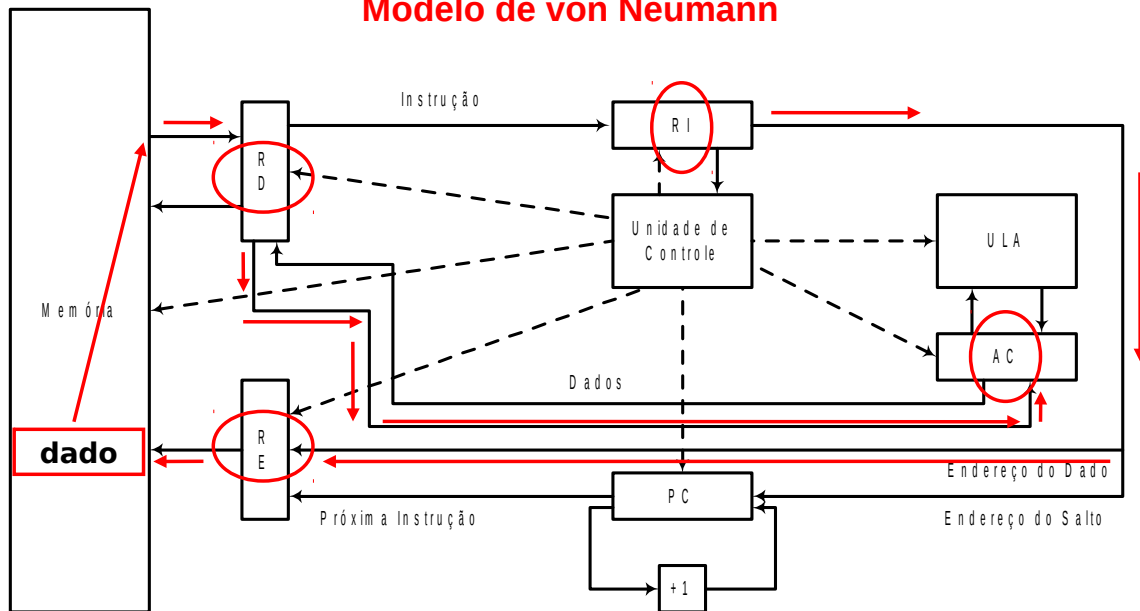
Modelo de von Neumann



20

Solução: Busca de Operando

Modelo de von Neumann

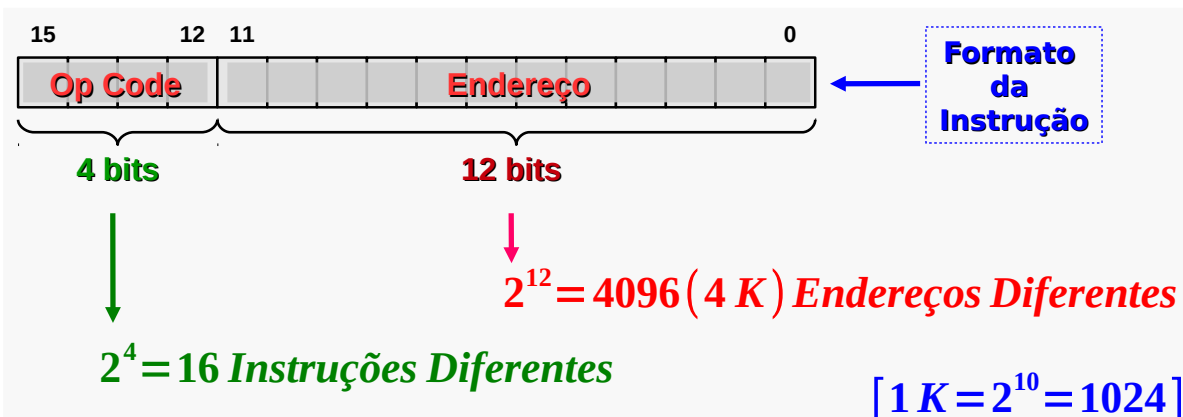


21

Exemplo de Ciclo de Instrução

Considere um processador com as seguintes características:

- Formato de instrução: 16 bits
 - 4 bits para Código de Operação (Op Code)
 - 12 bits para o campo de Endereços

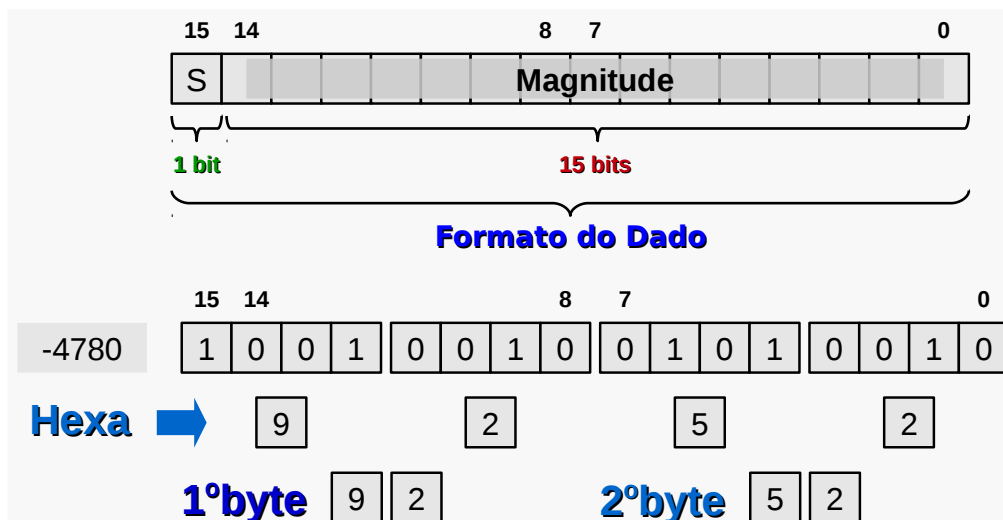


22

Exemplo de Ciclo de Instrução

Considere um processador com as seguintes características:

- Dado de 16 bits representado em Sinal-Magnitude



23

Conjunto de Instruções

- Carregar dados da Memória
 - LOAD → LDA
- Armazenar dados na Memória
 - STORE → STA
- Adição
 - ADD
- Subtração
 - SUB

Conjunto de Instruções

LDA

Instrução: *LDA* → Load AC from Memory

Op Code → **Binário:** 0001b
→ **Hexadecimal:** 1h

Operação: $AC \leftarrow [MEM]$

Descrição: Carrega AC a partir do conteúdo da posição de memória endereçada por [MEM]

Conjunto de Instruções

STA

Instrução: *STA* → Store AC to Memory

Op Code → **Binário:** 0010b

→ **Hexadecimal:** 2h

Operação: [MEM] ← AC

Descrição: Armazena o conteúdo do Acumulador na posição de memória endereçada por [MEM]

Conjunto de Instruções

ADD

Instrução: *ADD* → Add to AC from Memory

Op Code → **Binário:** 0011b

→ **Hexadecimal:** 3h

Operação: AC ← AC + [MEM]

Descrição: Soma o conteúdo do Acumulador ao conteúdo da posição de memória endereçada por [MEM]

Conjunto de Instruções

SUB

Instrução: *SUB* → Subtract to AC from Memory

Op Code → **Binário:** 0100b

→ **Hexadecimal:** 4h

Operação: $AC \leftarrow AC - [MEM]$

Descrição: Subtrai o conteúdo do Acumulador ao conteúdo da posição de memória endereçada por [MEM]

Programa Exemplo

Mnemônicos	Binário		Hexadecimal
	OpCode	Endereço	
LDA AC, [940]	0001	100101000000	1940h
ADD AC, [942]	0011	100101000010	3942h
SUB AC, [944]	0100	100101000100	4944h
STA [942], AC	0010	100101000010	2942h

Considerações Iniciais:

- PC contém inicialmente o valor 300H, o programa inicia nesse endereço
- Conteúdo da posição de memória [940H] é 0003H
- Conteúdo da posição de memória [942H] é 0002H
- Conteúdo da posição de memória [944H] é 0004H

Execução do Programa

Ciclo de Busca

	Memória
300	1940
302	3942
304	4944
306	2942
...	...
...	...
940	0003
942	0002
944	0004

Registradores	
300	PC
	RI
	AC

1940: AC ← [940]

30

Execução do Programa

Ciclo de Busca

	Memória
300	1940
302	3942
304	4944
306	2942
...	...
...	...
940	0003
942	0002
944	0004

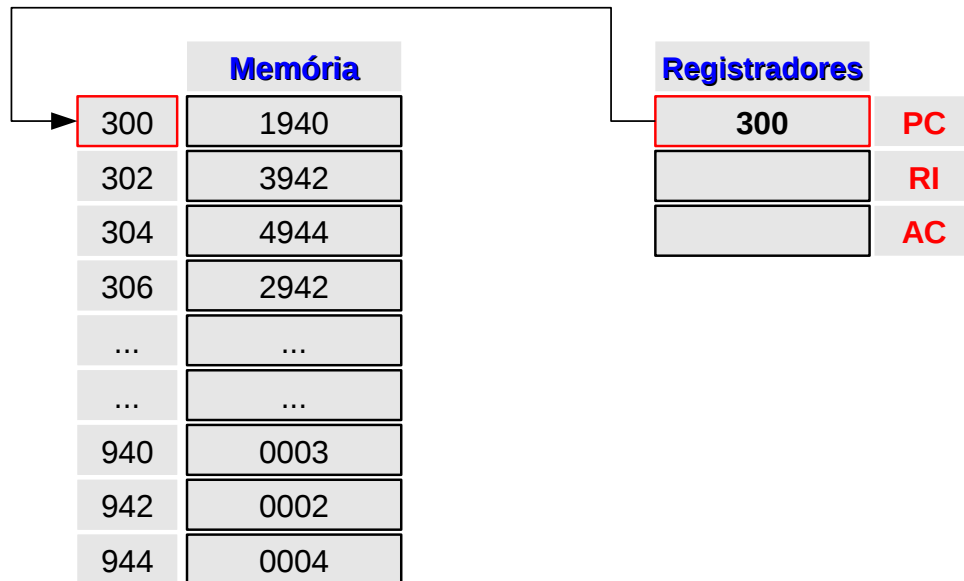
Registradores	
300	PC
	RI
	AC

1940: AC ← [940]

31

Execução do Programa

Ciclo de Busca

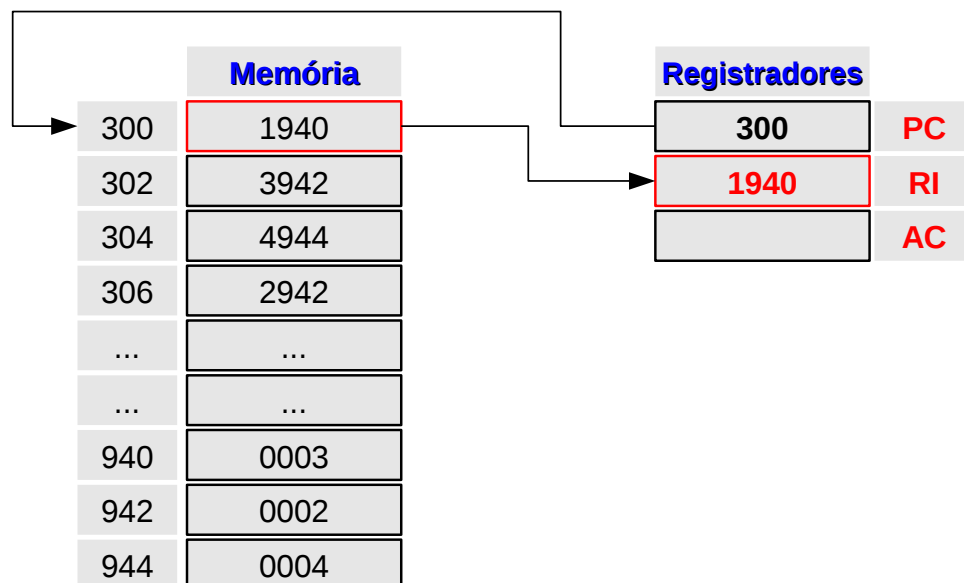


1940: AC ← [940]

32

Execução do Programa

Ciclo de Busca

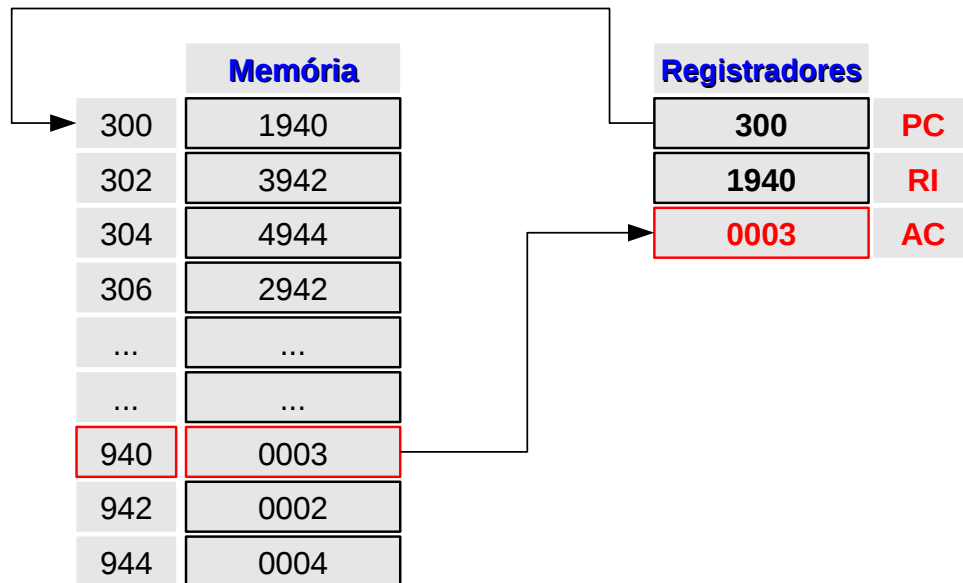


1940: AC ← [940]

33

Execução do Programa

Execução

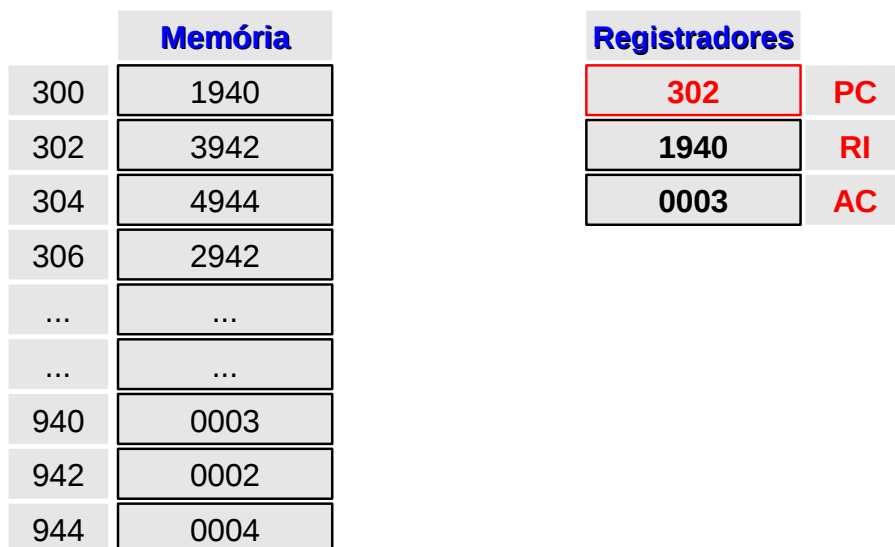


1940: AC ← [940]

34

Execução do Programa

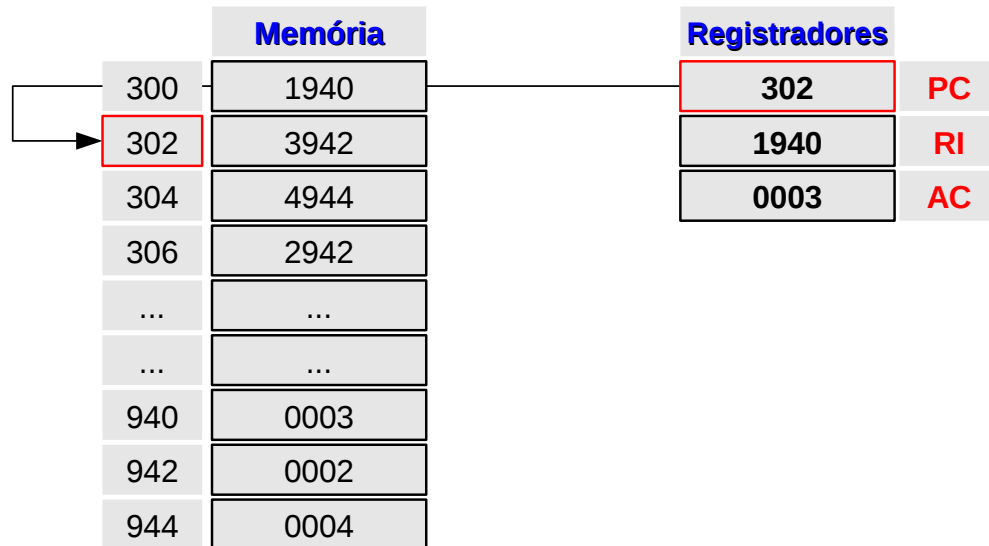
Ciclo de Busca



35

Execução do Programa

Ciclo de Busca

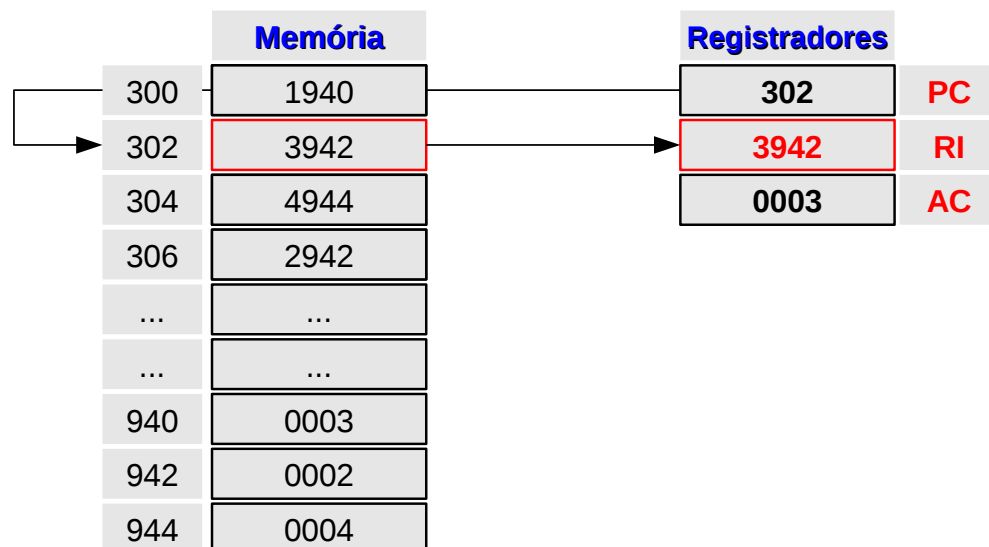


3942: AC ← AC + [942]

36

Execução do Programa

Ciclo de Busca

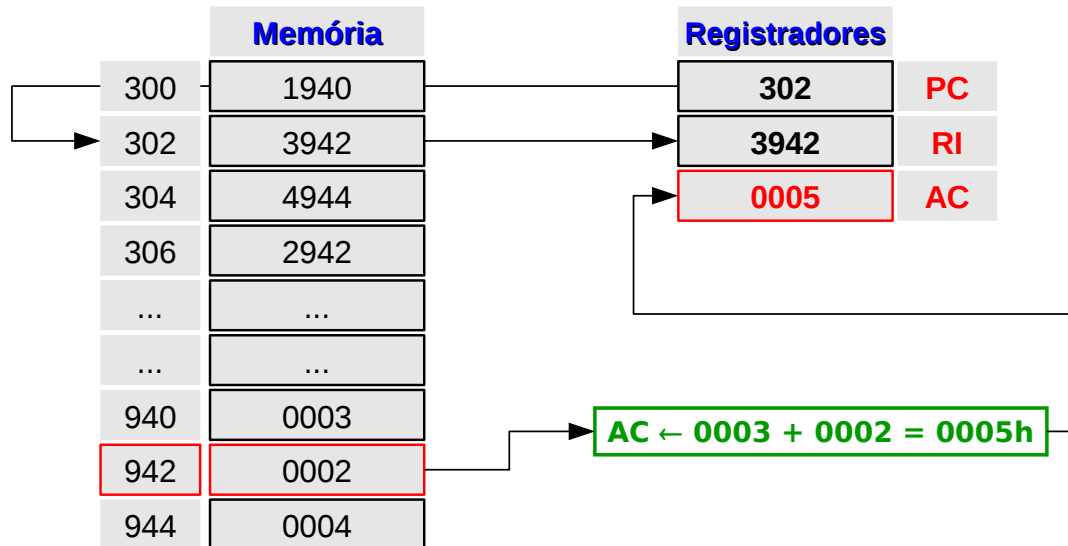


3942: AC ← AC + [942]

37

Execução do Programa

Execução

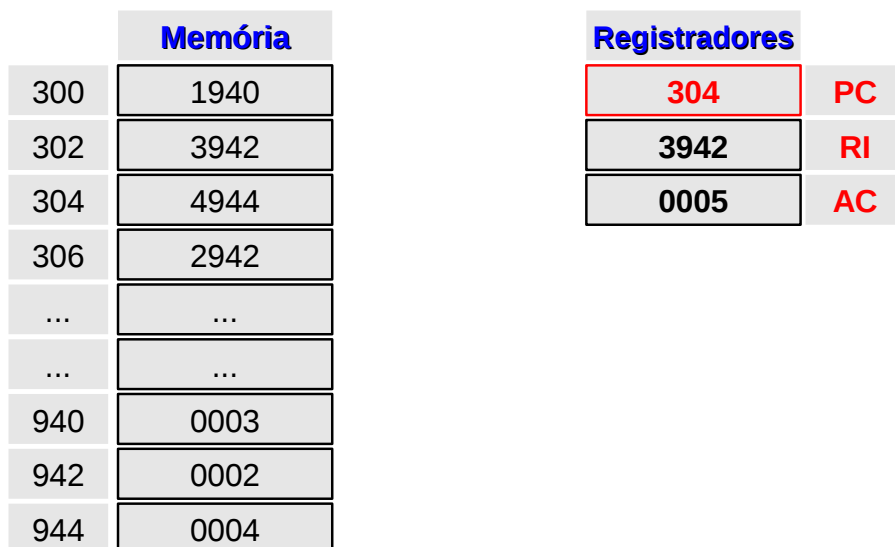


3942: $AC \leftarrow AC + [942]$

38

Execução do Programa

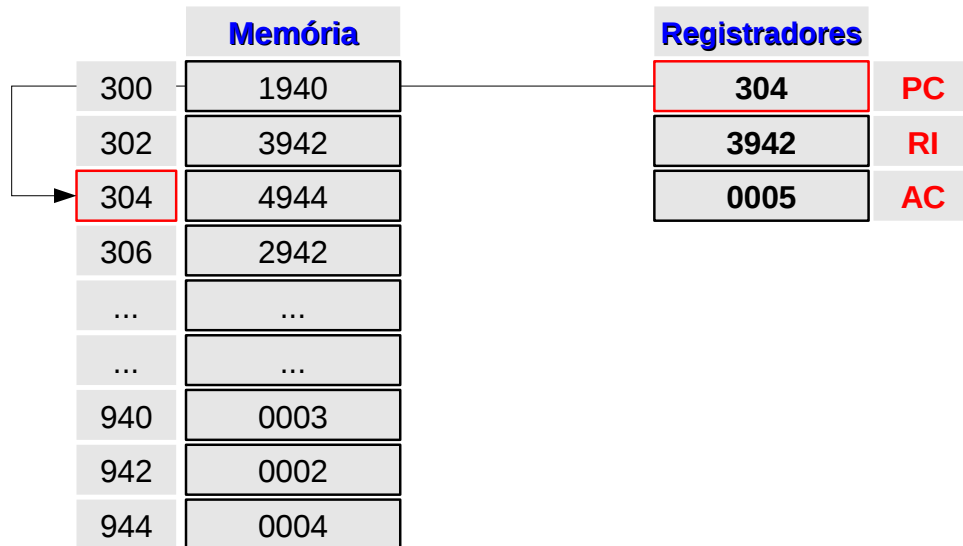
Ciclo de Busca



39

Execução do Programa

Ciclo de Busca

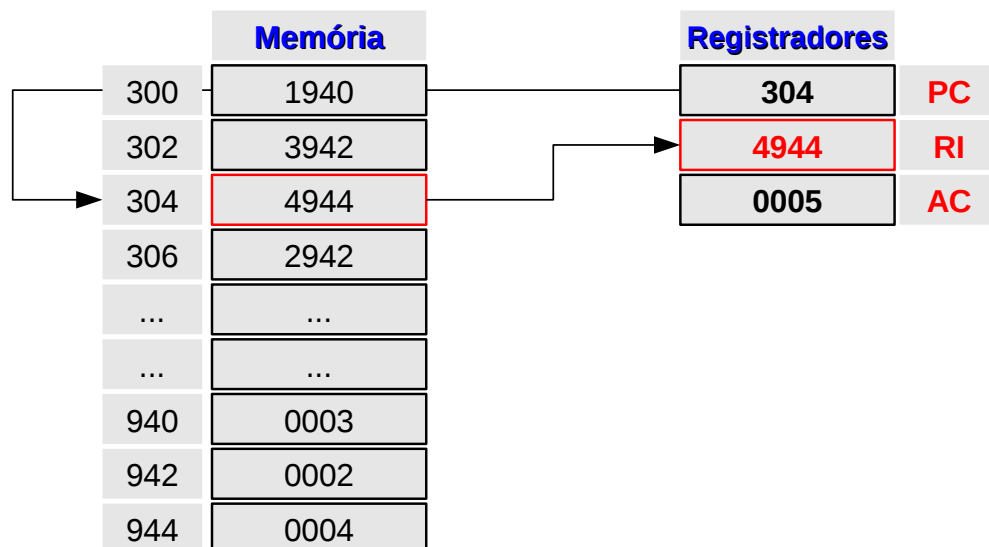


4944: AC ← AC - [944]

40

Execução do Programa

Ciclo de Busca

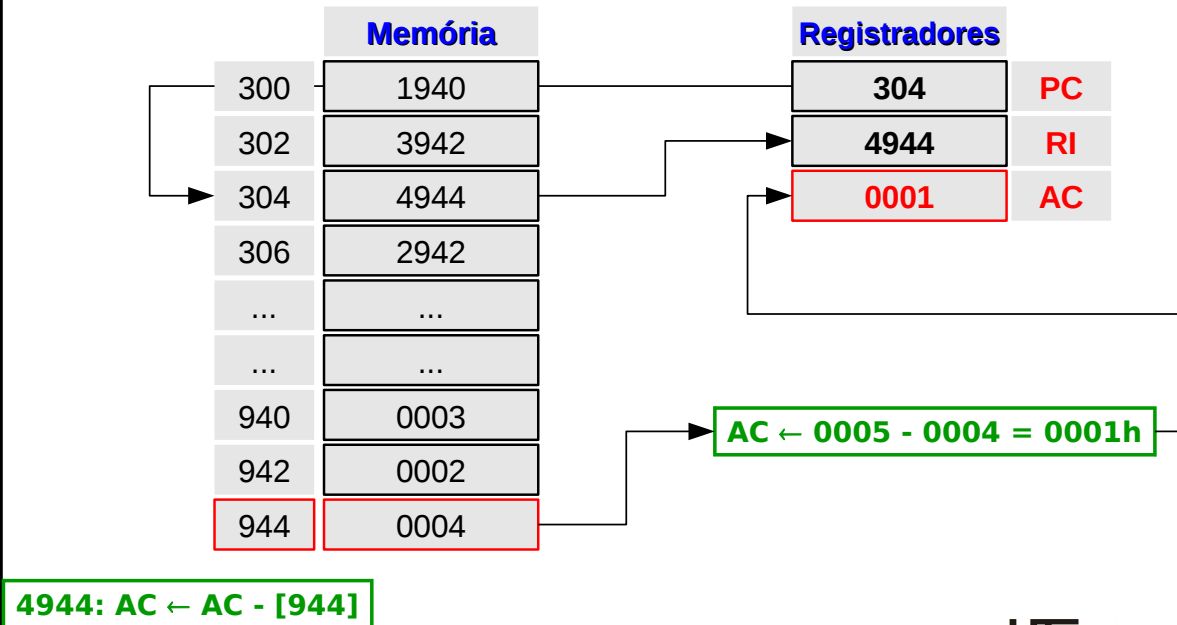


4944: AC ← AC - [944]

41

Execução do Programa

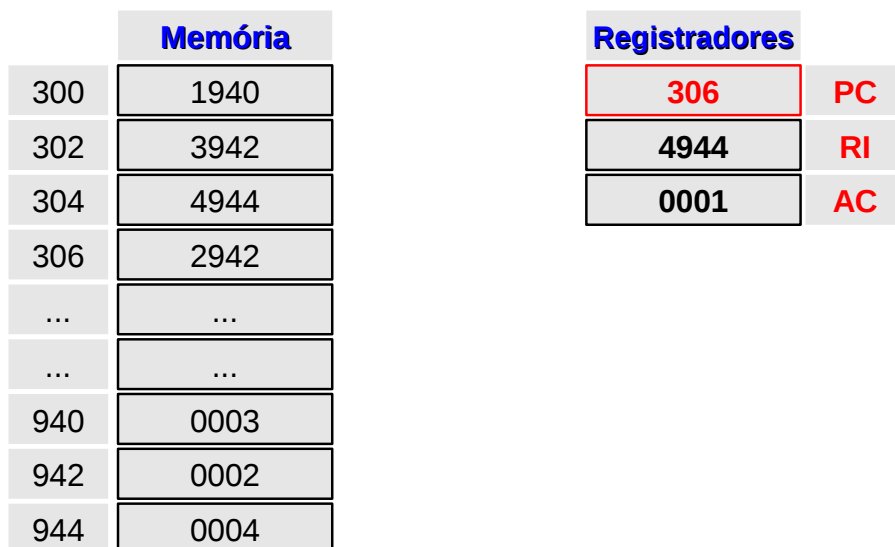
Execução



42

Execução do Programa

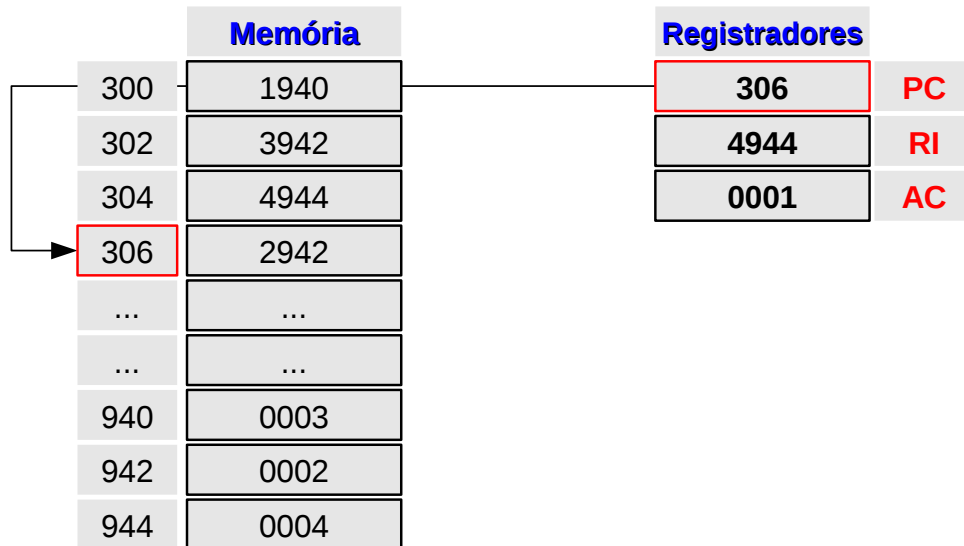
Ciclo de Busca



43

Execução do Programa

Ciclo de Busca

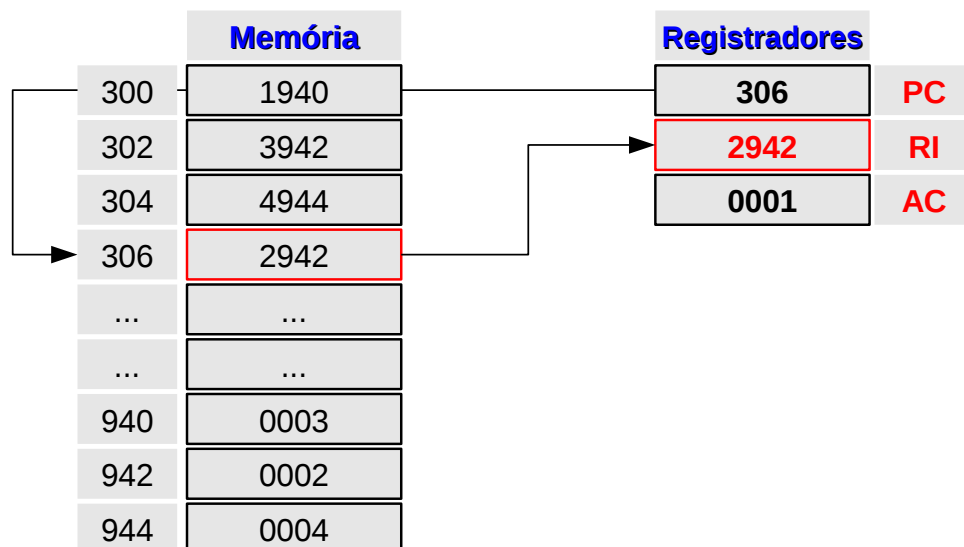


2942: [942] ← AC

44

Execução do Programa

Ciclo de Busca

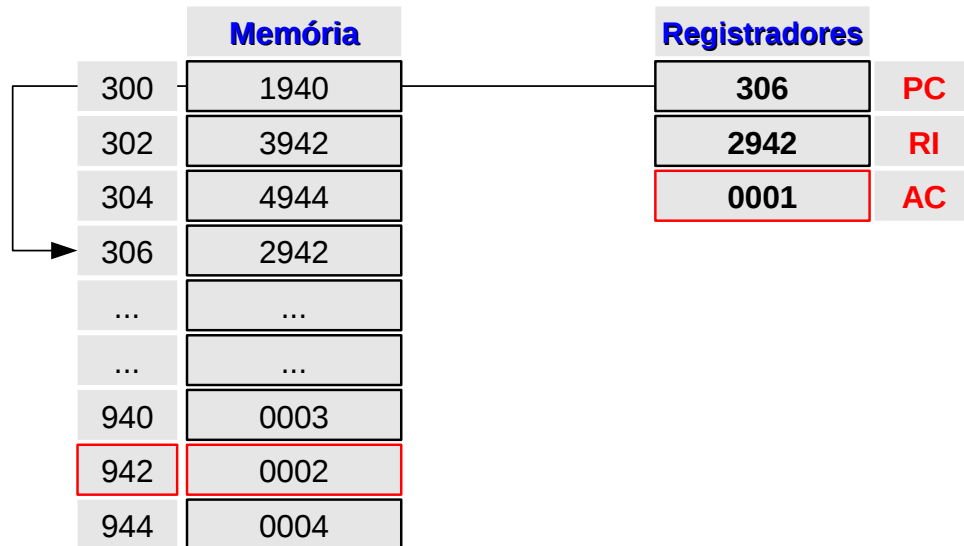


2942: [942] ← AC

45

Execução do Programa

Execução

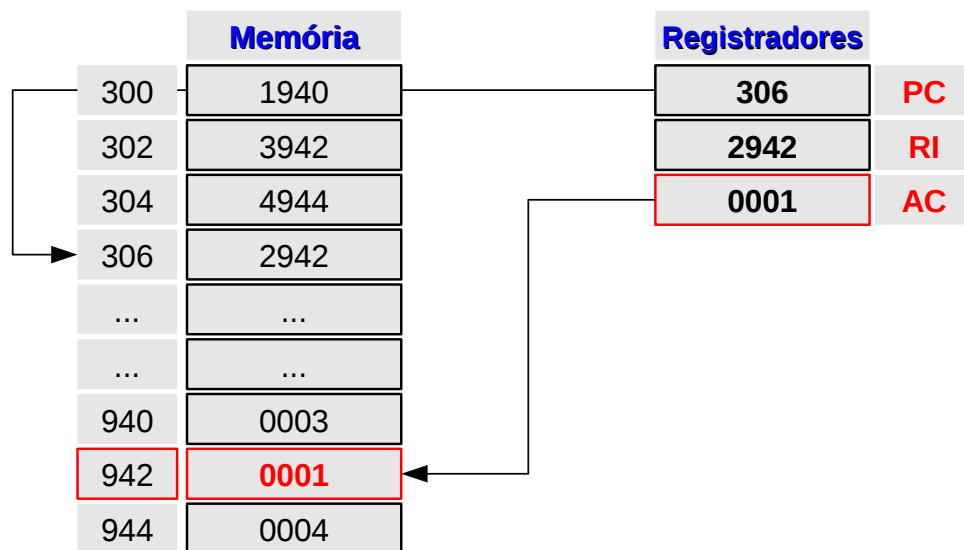


2942: [942] ← AC

46

Execução do Programa

Execução



2942: [942] ← AC

47

Exercício

Faça os diagramas da memória e dos registradores (PC, RI e AC) da CPU para mostrar a execução do seguinte programa:

Carregar o AC com o conteúdo da posição [MEM]:

0001b \Rightarrow AC \leftarrow [MEM]

AND do conteúdo da posição [MEM] com AC:
[MEM]

0110b \Rightarrow AC \leftarrow AC and

Armazenar o resultado na posição [MEM]:

0010b \Rightarrow [MEM] \leftarrow AC

Programa

LDA AC, [800h]

AND AC, [802h]

STA [804h], AC

Considere Inicialmente

[PC] = 500h

[800h] = F0E6

[802h] = 0FBF

[804h] = 0005

Solução

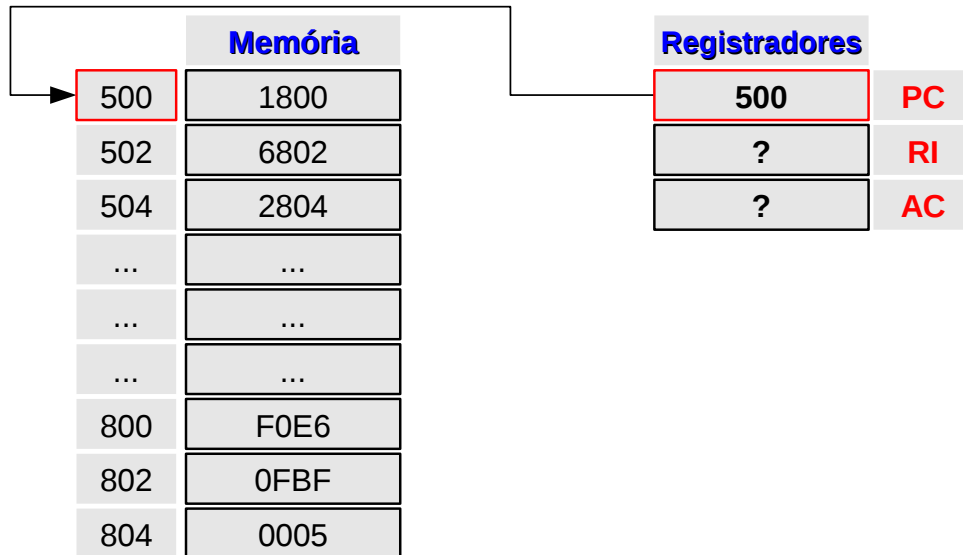
Ciclo de Busca

	Memória
500	1800
502	6802
504	2804
...	...
...	...
...	...
800	F0E6
802	0FBF
804	0005

Registradores	
500	PC
?	RI
?	AC

Solução

Ciclo de Busca

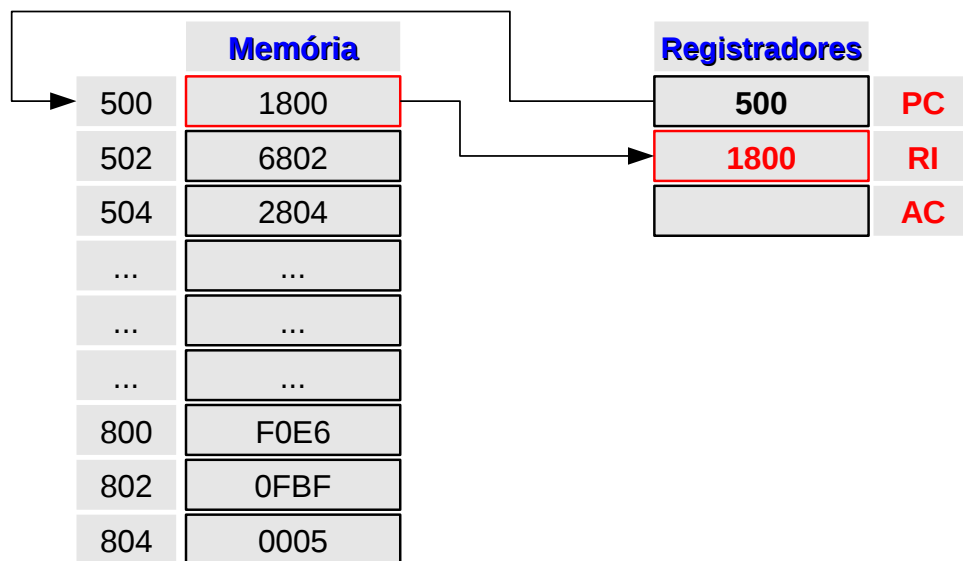


1800: AC ← [800]

50

Solução

Ciclo de Busca

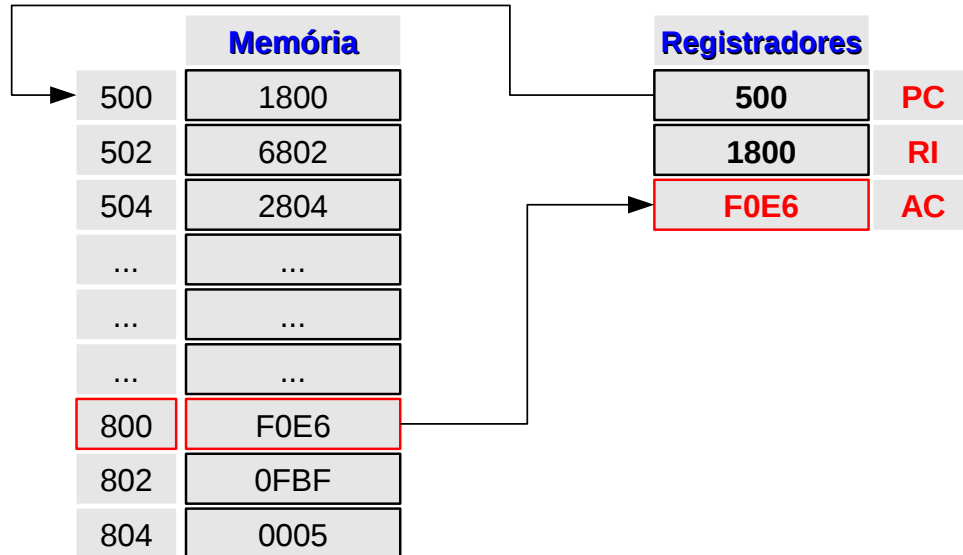


1800: AC ← [800]

51

Solução

Execução

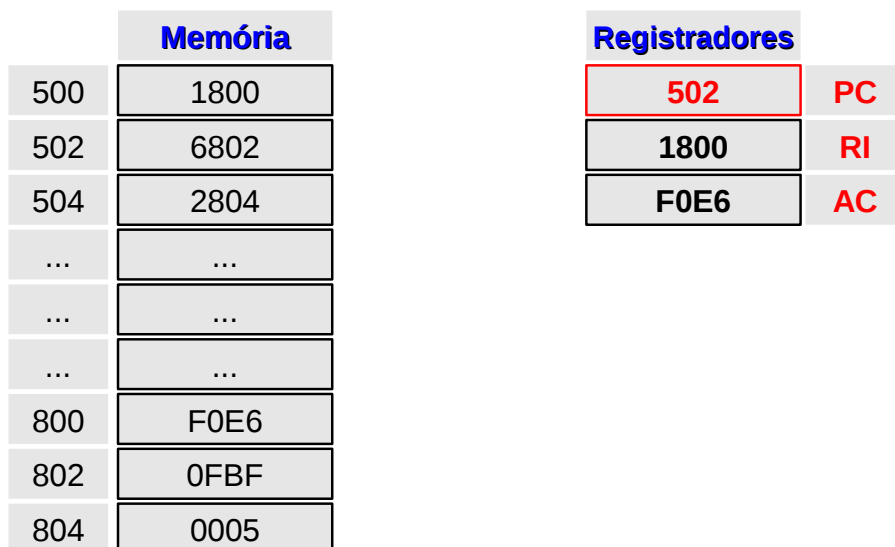


1800: AC ← [800]

52

Solução

Ciclo de Busca

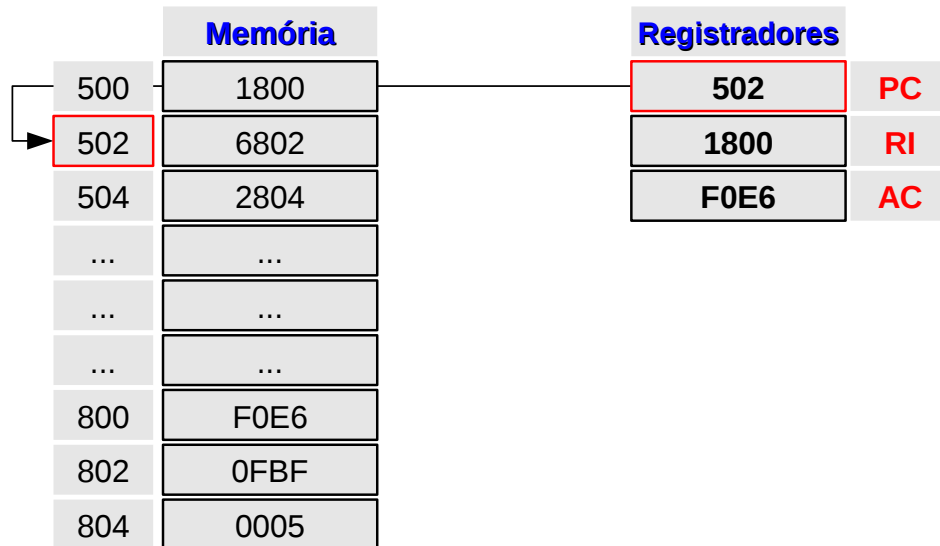


1800: AC ← [800]

53

Solução

Ciclo de Busca

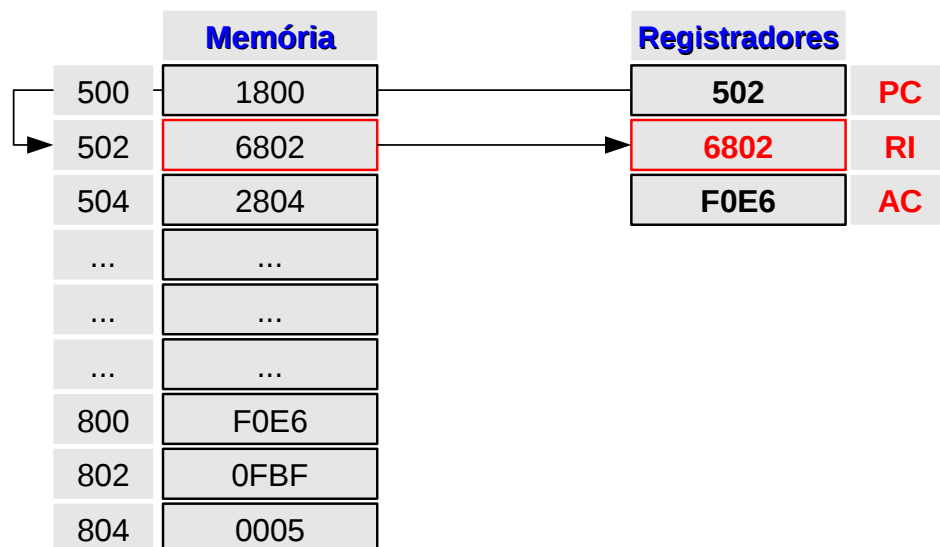


6802: AC AND [802]

54

Solução

Ciclo de Busca

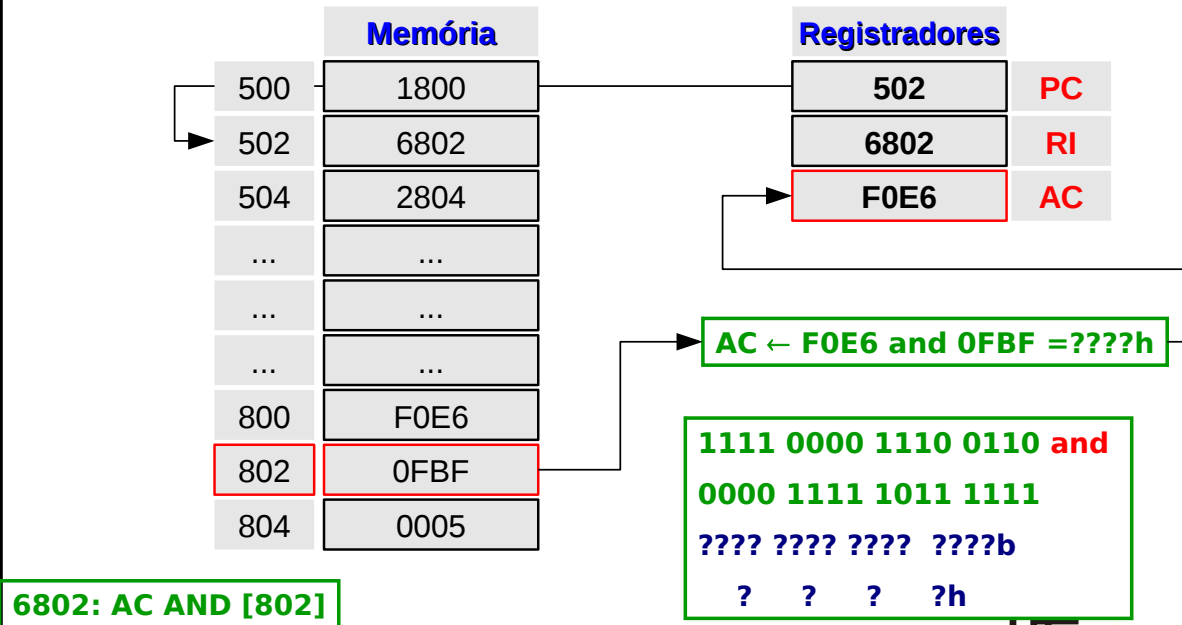


6802: AC AND [802]

55

Solução

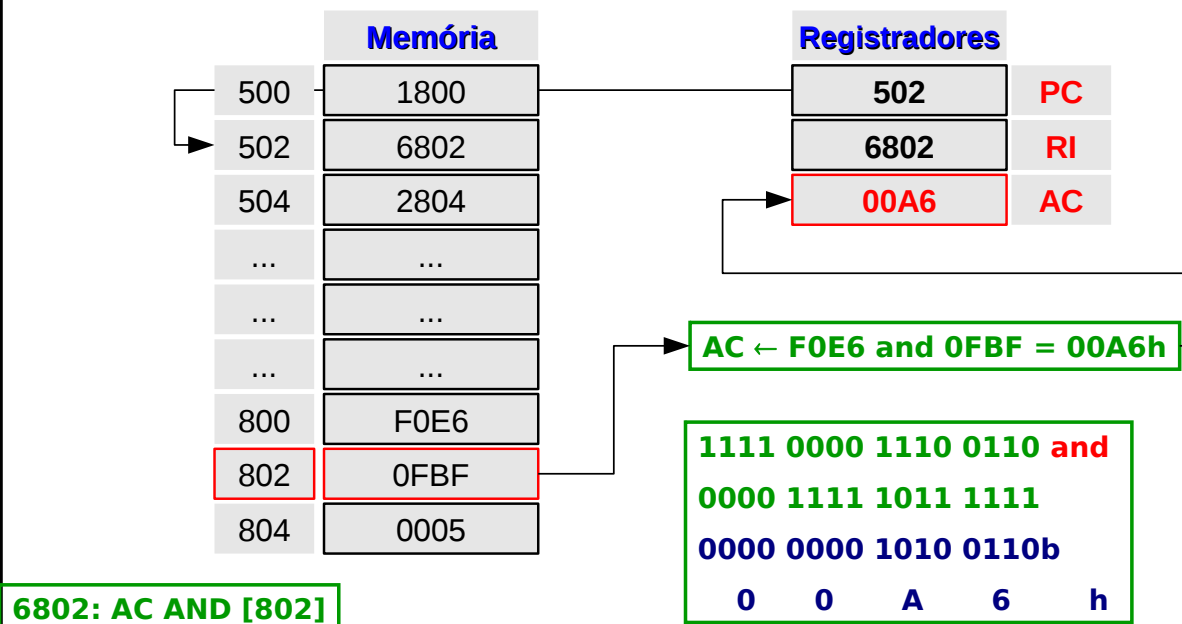
Execução



56

Solução

Execução



57

Solução

Ciclo de Busca

Memória		Registradores	
500	1800	504	PC
502	6802	6802	RI
504	2804	00A6	AC
...	...		
...	...		
...	...		
800	F0E6		
802	0FBF		
804	0005		

58

Solução

Ciclo de Busca

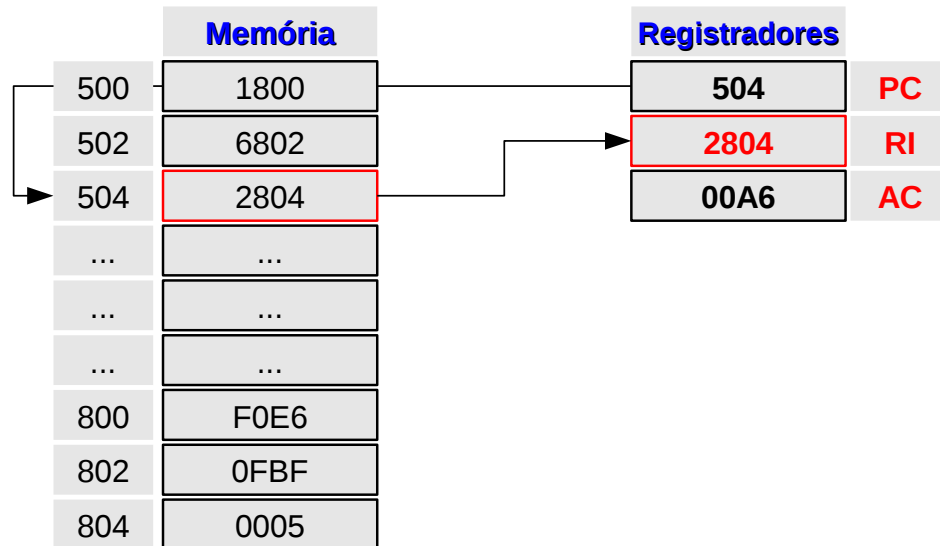
Memória		Registradores	
500	1800	504	PC
502	6802	6802	RI
504	2804	00A6	AC
...	...		
...	...		
...	...		
800	F0E6		
802	0FBF		
804	0005		

2804: [804] ← AC

59

Solução

Ciclo de Busca

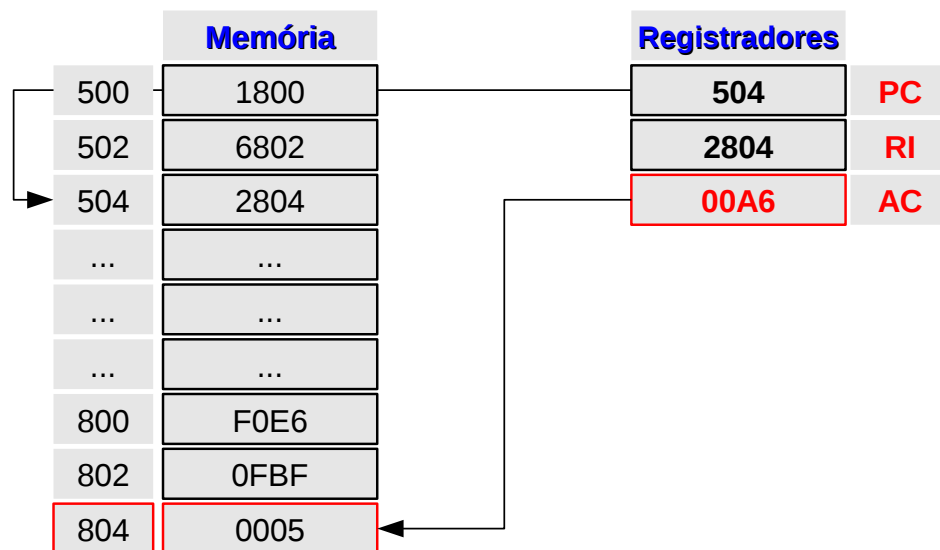


2804: [804] ← AC

60

Solução

Execução

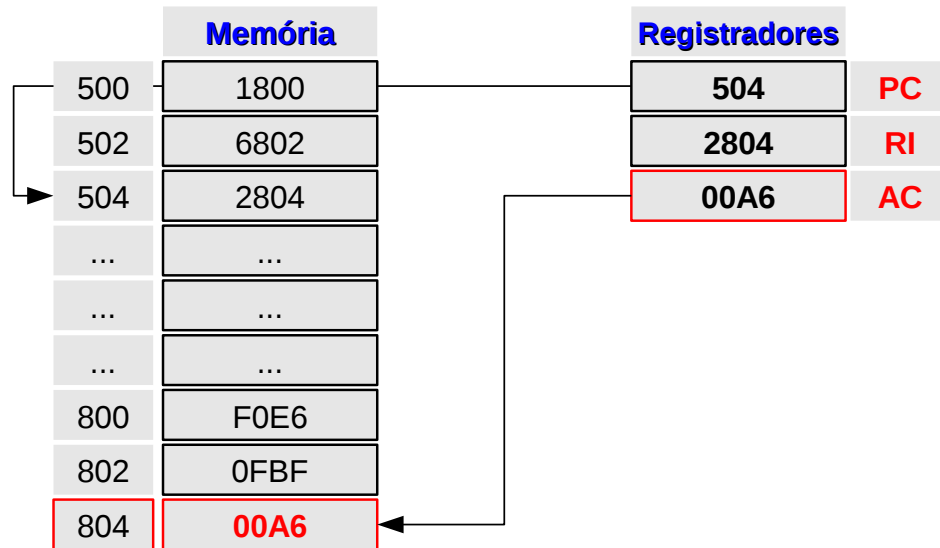


2804: [804] ← AC

61

Solução

Execução



2804: [804] ← AC

62

Referências

- Notas de aula do Prof. João Angelo Martini do DIN-UEM.

63