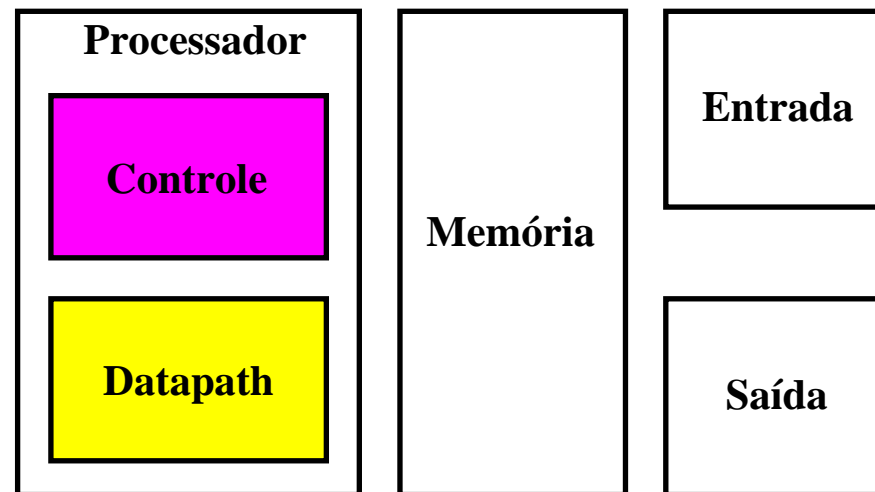


# Previsão dinâmica de desvios



# Previsão de desvios: idéia-base

- **Previsão:**
  - Uma hipótese (que se espera ser correta)
  - Antecipa busca de instruções na direção prevista
- **Objetivo:**
  - Não-degradação do desempenho
    - » Quando previsão correta
- **Garantia de correção:**
  - Código executado corretamente
    - » Quando previsão incorreta

# Previsão de desvios: mecanismo

- **Resultado da previsão**
  - **Previsão correta:**
    - » Instrução certa foi buscada, evitando a parada
  - **Previsão incorreta:**
    - » Instrução errada foi buscada: nova busca
    - » Penalidade
- **Impacto no desempenho**
  - **Depende da precisão**
    - » Frequência de previsões corretas
  - **Depende da penalidade de má previsão**
    - » Ciclos perdidos nas instruções mal previstas

# Mecanismo sob hipótese incorreta

- **Garantia de correção do código executado**
  - Sequência de instruções buscadas é anulada
    - » *Flushing*
  - Sequência apropriada de instruções é buscada
    - » *(Re)Fetching*
- **Melhoria de futuras previsões**
  - Como?

# Previsão de desvios: abordagens

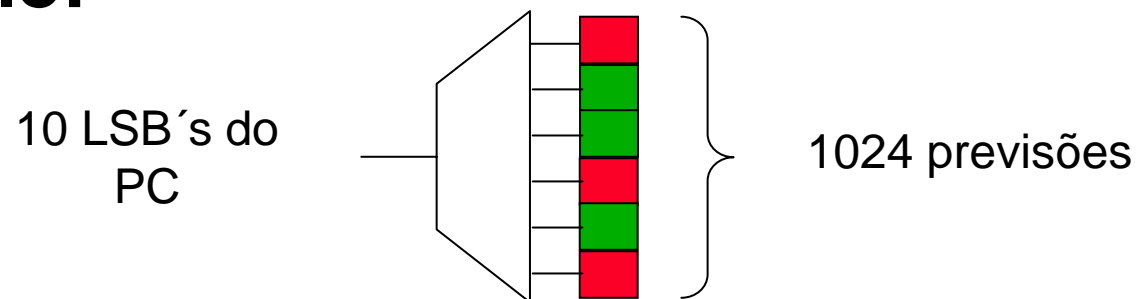
- **Previsão estática**
  - Assume-se direção preferencial para o desvio
    - » Por exemplo: desvio não-tomado (NT)
  - Direção fixa
    - » “Hardwired” no projeto do controlador
- **Previsão dinâmica**
  - Captura comportamento dinâmico dos desvios
    - » Em tempo de execução → HW
  - Direção da previsão é alterável
    - » De acordo com a **história recente** dos desvios

# Mecanismo sob hipótese incorreta

- **Garantia de correção do código executado**
  - Sequência de instruções buscadas é anulada
    - » *Flushing*
  - Sequência apropriada de instruções é buscada
    - » *(Re)Fetching*
- **Melhoria de futuras previsões**
  - **Hipótese de previsão é invertida**

# Tabela de Histórico de Desvios

- “Branch History Table” (BHT)
  - LSB’s do PC endereçam BHT
  - Cada elemento da tabela contém 1 bit
    - » Desvio foi tomado (T) ou não (NT) na última vez
  - Se previsão incorreta
    - » Inverte bit e re-armazena
- Exemplo:



# BHT: exemplo de limitação

- **Laço**

```
Loop:  ...  
      ...  
      ...  
      bne $s1, $s2, Loop
```

- **Suposição:**

- **Laço executa 10 iterações**
  - » **Desvio tomado 9 vezes seguidas**
  - » **Não tomado ao final da décima iteração**

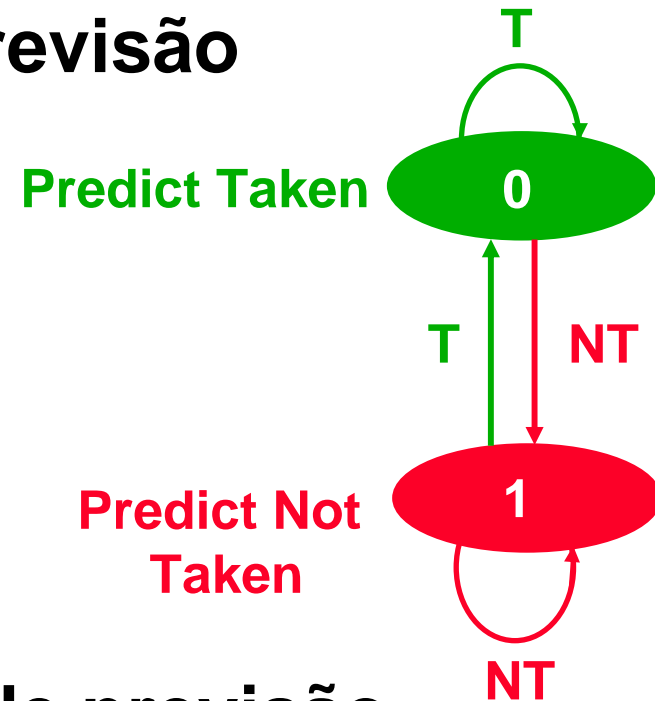


# **BHT: exemplo de limitação**

- **Previsões incorretas**
  - **Última iteração**
    - » **Fim do laço**
  - **Primeira iteração**
    - » **Na próxima execução do código do laço**
  - **Frequência do desvio: 90%**
  - **Precisão: 80%**
- **Taxa de previsões incorretas em laços**
  - **2 vezes (em média por execução)**
  - **Mesmo que desvio quase sempre tomado**

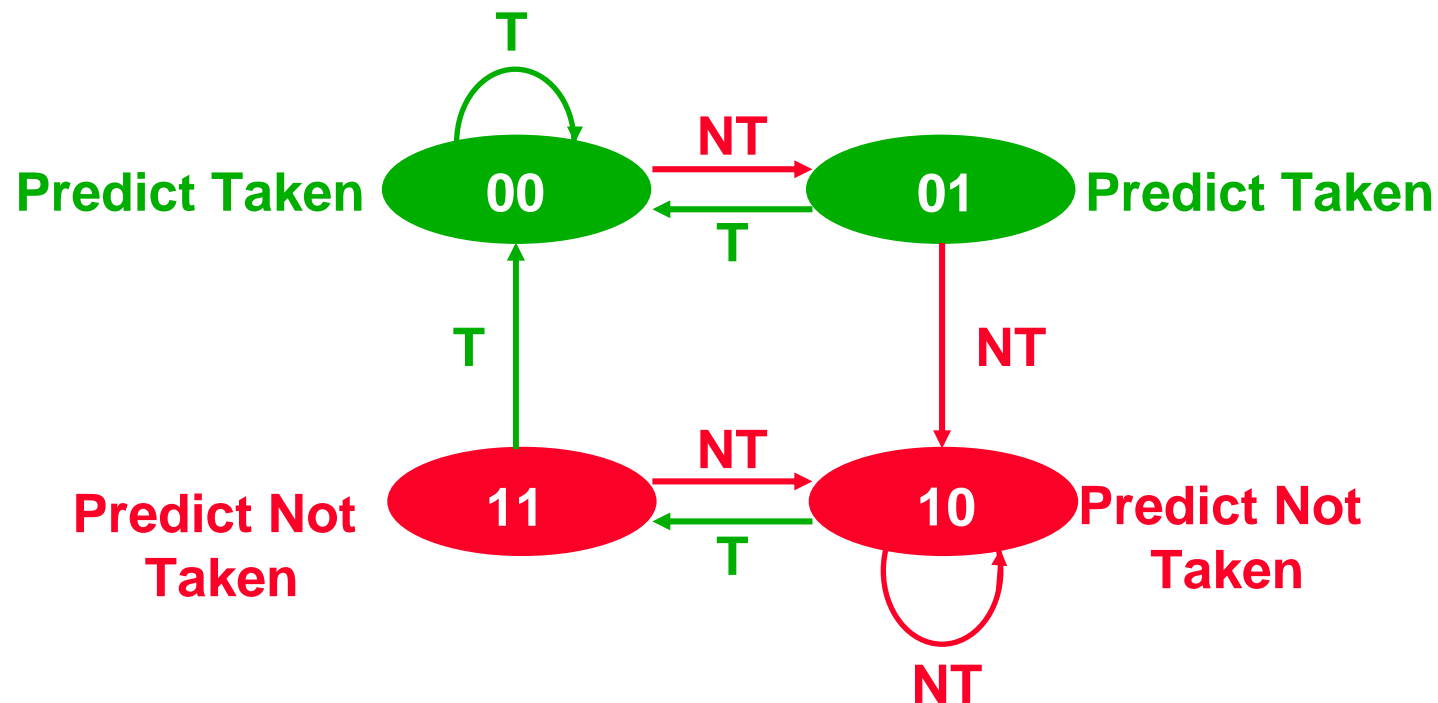
# BHT: relaxando a limitação

- Inversão da hipótese de previsão
  - Dois cenários de previsão
  - 1 bit por entrada da BHT
- Aumento para 4 cenários de previsão
  - Altere direção só depois de 2 previsões incorretas
  - 2 bits por entrada da BHT



# BHT com 2 bits

- Altere direção só depois de 2 previsões incorretas



# **BHT: implementação**

- **Alternativa 1: BHT em cache dedicada**
  - Indexada pelos LSBs do endereço do desvio
  - Acesso no estágio IF
- **Alternativa 2: BHT embutido na I-cache**
  - Par de bits extras nela armazenados

# **BHT: causas de previsão incorreta**

- **Direção errada para um dado desvio**
- **Obteve-se a história do desvio errado**
  - Quando se indexou a tabela
  - Pois índice contém só os LSBs do endereço

# BHT em pipelines curtos

- **Pequeno impacto**
  - BHT é eficiente quando:
    - » Endereço-alvo pré-calculado e ...
    - » Resultado do teste é **antecipado** pelo BHT
  - Mas eventos são (quase) simultâneos
    - » Endereço-alvo (ID é o estágio o mais cedo possível);
    - » Teste (ID ou EX)
      - Exemplos: beqz ou ble
- **Previsão estática suficiente para 5 estágios**
  - Previsão de desvio não-tomado
  - “Flushing” do pipeline se previsão incorreta

# BHT em pipelines longos

- **Crucial**
  - **Distância entre estágios envolvidos aumenta**
  - **Penalidade de má previsão aumenta**
    - » Mais instruções foram buscadas indevidamente
    - » Até o teste se resolver
  - **Várias instruções anuladas**
    - » Degradação do CPI

# Conclusão e perspectivas

- **Previsores dinâmicos**
  - Reduzem degradação do CPI
    - » Devida a “hazards” de controle
  - Prevêm o resultado do teste
    - » BHT
    - » **Previsores correlatados** (estudados em INE 5436)
- **Previsores de endereço-alvo**
  - **Branch target buffers** (estudados em INE 5436)
  - Armazenam o endereço-alvo dos últimos desvios previstos
    - » Para evitar seu cálculo e antecipar a busca