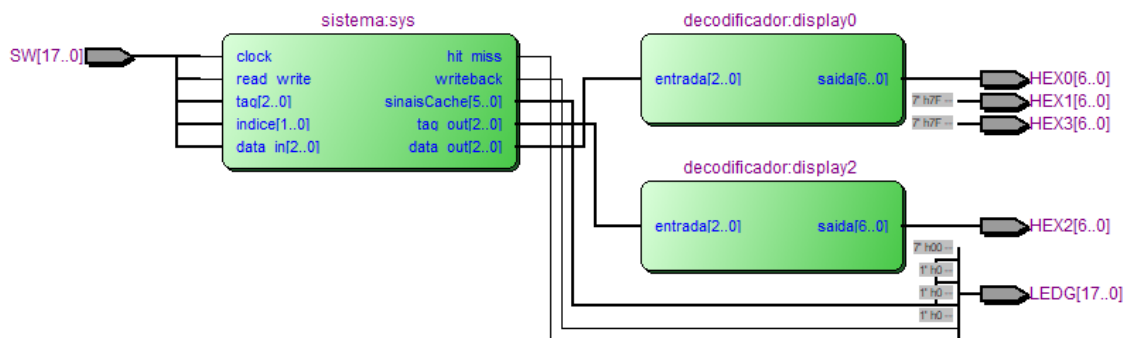


## Laboratório de Arquitetura e Organização de Computadores II

### Prática I – Parte III

Esta prática tem como objetivo a implementação de uma cache associativa por conjunto de 2 vias, com utilização do MIF e realização da leitura e escrita utilizando o display de 7 segmentos. Com isso, a primeira coisa a ser feita foi a integração dos sinais de entrada e saída da placa com a nossa prática em si, que passará a ser chamada de sistema. Nisso, foi criado módulos de decodificadores de sinais para as saídas a serem apresentadas nos display de 7 segmentos e o módulo chamado de sistema que como já referido vai lidar com os estímulos fornecidos pelas as entradas físicas da placa. A correlação correta entre pinos, leds e displays da placa com o sinal que cada um representa está na *tabela 01* no final deste documento.

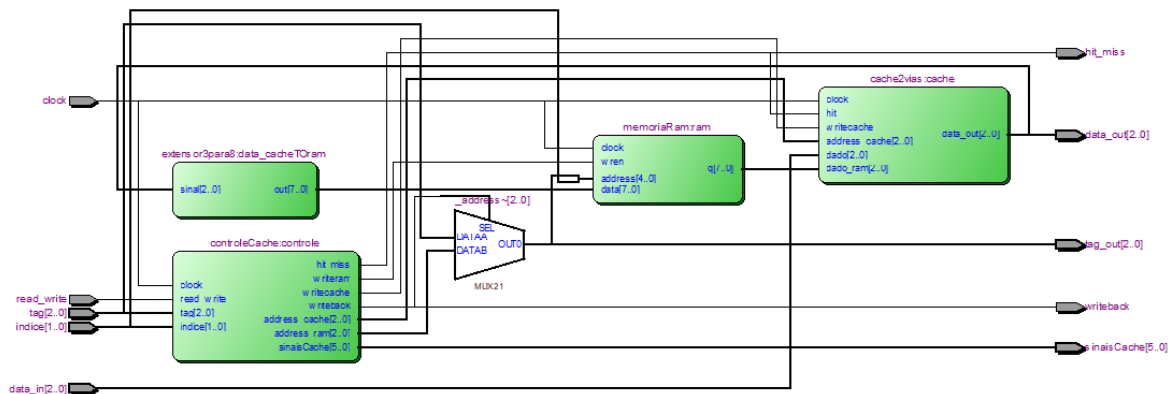
Figura 01 - Módulo que dialoga com a placa



Como visto na *figura 01* nosso sistema terá como entrada um clock, uma tag com 3 bits, um índice com dois bits, um dado com 3 bits e por fim um sinal para indicar leitura ou escrita na cache, sendo 1 escrita e 0 leitura. Já as saídas são hit ou miss, sendo hit igual a 1 e 0 igual a miss, writeback, a tag acessada, o dado presente na cache e sinais gerais da linha da cache com LRU, dirty e válido. Tendo deixado essa primeira parte clara, podemos então prosseguir para o funcionamento do nosso sistema, onde não iremos mais pensar no nosso mundo físico, ou seja, não vamos nos preocupar mais com os pinos da placa e com seus LEDs e displays de 7 segmentos.

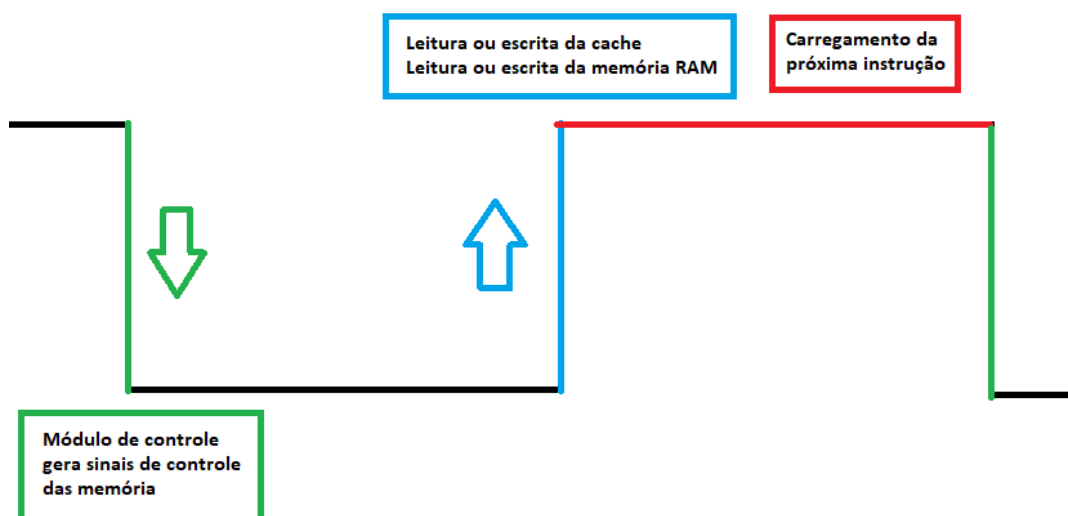
O nosso sistema é composto principalmente de três módulos, a memória cache, a memória RAM e o controle da cache, também há módulos auxiliares com um multiplexador para controle de endereço para a memória ram e um extensor de sinal de 3 bits para 8 bits para os dados a serem fornecidos para a memória ram. O diagrama do sistema com seus módulos e suas conexões pode ser visto na imagem abaixo.

Figura 02 - Sistema



Para entendermos o funcionamento do nosso sistema, iremos fazer o estudo por casos. Podemos encontrar 5 casos, que são write hit, write miss com write back, read hit, read miss e read miss com write back. Porém, antes, vale destacar que o módulo de controle, responsável por receber e interpretar os sinais de entrada para gerir a cache e a memória ram, é ativado com a borda de descida do nosso clock. Isso implica que a leitura e escrita na cache e na memória ram acontecem na borda de subida do clock.

Figura 03 - Clock do sistema



### **Write hit**

O caso de write hit é bem simples, se a tag fornecida pela entrada estiver em alguma das caches no índice indicado, basta escrever o valor de entrada na cache em questão. Para isso ocorrer o módulo de controle desabilita escrita na ram, ativa o sinal de hit, ativa sinal de escrita na cache, atualiza o sinal de dirty e a LRU da cache e por fim determina o endereço do registrador da cache que deverá ser escrito. Com isso, quando der a borda de subida do clock a cache receberá essas informações de dado e endereço e escreverá no registrador certo.

Aproveitarei esse caso, para destacar que as informações da cache com LRU, dirty, válido e a tag são armazenadas no próprio módulo de controle, para justamente facilitar o tratamento das informações, já os dados são armazenados no módulo da cache, que nada mais é que um banco de registradores.

### **Read hit**

O read hit acontece quando tanto a tag de entrada bateu com a tag em uma das caches tanto se o dado presente é válido pelo sinal de mesmo nome. Assim, o controle emite os sinais de leitura da cache, leitura da memória ram, determina o endereço do registrador da cache, atualiza lru, informa que deu hit e atualiza o sinal de válido. Com isso, quando a cache for acionada ela irá fornecer o valor presente no registrador no endereço indicado.

### **Read miss**

Até agora os processos se resumiram aos módulos de controle e do módulo de registradores da cache. Porém, em um processo de read miss sem write back, ou seja, quando o conteúdo presente na cache não é válido será necessário carregar um bloco da memória RAM para a nossa cache, para então poder ler seu conteúdo. Os sinais de controle são muito parecidos com o caso do read hit, porém o miss estará ativado o que levará ao dado fornecido pela ram ser escrito no registrador em questão.

### **Read ou Write miss com write back**

Esses dois casos estão sendo abordados em conjunto, pois muitos dos sinais atuam em um também atuam no outro, com a ressalva dos sinais que indicavam o próprio processo de escrita ou de leitura da cache. O processo de write back leva dois clocks para se completar o primeiro clock é idêntico para o write e para o read. Neste, o write back é habilitado assim como a escrita na ram, a leitura na cache e o sinal de miss. Além disso, o processo determina o endereço a ser acessado na cache, verifica qual o bloco mais antigo para ser levado para a ram e atualiza a lru.

Já no segundo clock o módulo de controle ignora as instruções de entrada e passa a agir com base nas informações antes determinadas, visto que o sinal de writeback está ativo. Assim, desabilita o write back para o próximo clock, mantém o sinal miss, descobre qual a cache mais antiga para determinar o endereço do registrador da cache, depois atualiza tag para a fornecida pela entrada, habilita a leitura da ram, atualiza dirty se for read e por último habilita ou desabilita escrita na cache.

Alguns últimos detalhes a serem mencionados são que o arquivo .mif foi inicializado com todos os valores iguais a zero. Com isso, quando se ver um dado igual a zero nos prints das simulações, entenda que esse dado foi trazido da memória RAM no caso de miss. Agora abaixo, estão as imagens das simulações que comprovam o funcionamento da cache associativa por conjunto de 2 vias.

*Figura 04 - Wave*

[illegible]

*Figura 05 - início da simulação*

```

=====
# ----- INICIO DA SIMULACAO -----
=====
#
# ----- CLOCK 01 -----
#
# ENTRADA
# WRITE 0 TAG 100 INDICE 0 DADO x
#
# SAIDA
# HIT x          WRITEBACK 0
# ADDRESS_CACHE x DADO-CACHE x
#
# RAM
# WREN x DADO-IN x ADDRESS 16 DADO-RAM 0
#
# CACHE 0
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      0      0      0  100  1
# 1      1      0      1  000  3
# 2      1      1      0  101  5
# 3      0      x      x  xxx  x
#
# CACHE 1
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      0      0      0  xxx  2
# 1      0      0      0  xxx  4
# 2      1      0      1  111  6
# 3      0      x      x  xxx  x

```

## 1 - Read miss TAG = 100 Índice= 00

Figura 06 - simulação parte 02

```
# ----- CLOCK 00 -----
#
# ENTRADA
# WRITE 0 TAG 100 INDICE 0 DADO x
#
# SAIDA
# HIT 0          WRITEBACK 0
# ADDRESS_CACHE 0 DADO-CACHE 1
#
# RAM
# WREN 0 DADO-IN 1 ADDRESS 16 DADO-RAM 0
#
# CACHE 0
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      1      0    1  100  1
# 1      1      0    1  000  3
# 2      1      1    0  101  5
# 3      0      x    x  xxx  x
#
# CACHE 1
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      0      0    0  xxx  2
# 1      0      0    0  xxx  4
# 2      1      0    1  111  6
# 3      0      x    x  xxx  x
```

Figura 07 - simulação parte 03

```
# ----- CLOCK 01 -----
#
# ENTRADA
# WRITE 0 TAG 100 INDICE 0 DADO x
#
# SAIDA
# HIT 0          WRITEBACK 0
# ADDRESS_CACHE 0 DADO-CACHE 0
#
# RAM
# WREN 0 DADO-IN 0 ADDRESS 16 DADO-RAM 0
#
# CACHE 0
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      1      0    1  100  0
# 1      1      0    1  000  3
# 2      1      1    0  101  5
# 3      0      x    x  xxx  x
#
# CACHE 1
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      0      0    0  xxx  2
# 1      0      0    0  xxx  4
# 2      1      0    1  111  6
# 3      0      x    x  xxx  x
```

## 2 - Read miss TAG = 101 Índice = 00

Figura 08 - simulação parte 04

```
----- CLOCK 00 -----  
  
ENTRADA  
WRITE 0 TAG 101 INDICE 0 DADO x  
  
SAIDA  
HIT 0          WRITEBACK 0  
ADDRESS_CACHE 4 DADO-CACHE 2  
  
RAM  
WREN 0 DADO-IN 2 ADDRESS 20 DADO-RAM 0  
  
CACHE 0  
INDICE VALIDO DIRTY LRU TAG DADO  
0      1      0    0  100  0  
1      1      0    1  000  3  
2      1      1    0  101  5  
3      0      x    x  xxx  x  
  
CACHE 1  
INDICE VALIDO DIRTY LRU TAG DADO  
0      1      0    1  101  2  
1      0      0    0  xxx  4  
2      1      0    1  111  6  
3      0      x    x  xxx  x
```

Figura 09 - simulação parte 05

```
# ----- CLOCK 01 -----  
#  
# ENTRADA  
# WRITE 0 TAG 101 INDICE 0 DADO x  
#  
# SAIDA  
# HIT 0          WRITEBACK 0  
# ADDRESS_CACHE 4 DADO-CACHE 0  
#  
# RAM  
# WREN 0 DADO-IN 0 ADDRESS 20 DADO-RAM 0  
#  
# CACHE 0  
# INDICE VALIDO DIRTY LRU TAG DADO  
# 0      1      0    0  100  0  
# 1      1      0    1  000  3  
# 2      1      1    0  101  5  
# 3      0      x    x  xxx  x  
#  
# CACHE 1  
# INDICE VALIDO DIRTY LRU TAG DADO  
# 0      1      0    1  101  0  
# 1      0      0    0  xxx  4  
# 2      1      0    1  111  6  
# 3      0      x    x  xxx  x
```

### 3 - Read hit TAG = 100 Índice= 00

Figura 10 - simulação parte 06

```
# ----- CLOCK 00 -----  
#  
# ENTRADA  
# WRITE 0 TAG 100 INDICE 0 DADO x  
#  
# SAIDA  
# HIT 1          WRITEBACK 0  
# ADDRESS_CACHE 0 DADO-CACHE 0  
#  
# RAM  
# WREN 0 DADO-IN 0 ADDRESS 16 DADO-RAM 0  
#  
# CACHE 0  
# INDICE VALIDO DIRTY LRU TAG DADO  
# 0      1      0      1  100  0  
# 1      1      0      1  000  3  
# 2      1      1      0  101  5  
# 3      0      x      x  xxx  x  
#  
# CACHE 1  
# INDICE VALIDO DIRTY LRU TAG DADO  
# 0      1      0      0  101  0  
# 1      0      0      0  xxx  4  
# 2      1      0      1  111  6  
# 3      0      x      x  xxx  x
```

Figura 11 - simulação parte 07

```
----- CLOCK 01 -----  
  
ENTRADA  
WRITE 0 TAG 100 INDICE 0 DADO x  
  
SAIDA  
HIT 1          WRITEBACK 0  
ADDRESS_CACHE 0 DADO-CACHE 0  
  
RAM  
WREN 0 DADO-IN 0 ADDRESS 16 DADO-RAM 0  
  
CACHE 0  
INDICE VALIDO DIRTY LRU TAG DADO  
0      1      0      1  100  0  
1      1      0      1  000  3  
2      1      1      0  101  5  
3      0      x      x  xxx  x  
  
CACHE 1  
INDICE VALIDO DIRTY LRU TAG DADO  
0      1      0      0  101  0  
1      0      0      0  xxx  4  
2      1      0      1  111  6  
3      0      x      x  xxx  x
```



#### 4 - Write hit TAG = 000 Índice = 01 Dado = 111

Figura 12 - simulação parte 08

```
----- CLOCK 00 -----
:
: ENTRADA
: WRITE 1 TAG 000 INDICE 1 DADO 7
:
: SAIDA
: HIT 1          WRITEBACK 0
: ADDRESS_CACHE 1 DADO-CACHE 3
:
: RAM
: WREN 0 DADO-IN 3 ADDRESS 1 DADO-RAM 0
:
: CACHE 0
: INDICE VALIDO DIRTY LRU TAG DADO
: 0      1      0      1  100  0
: 1      1      1      1  000  3
: 2      1      1      0  101  5
: 3      0      x      x  xxx  x
:
: CACHE 1
: INDICE VALIDO DIRTY LRU TAG DADO
: 0      1      0      0  101  0
: 1      0      0      0  xxx  4
: 2      1      0      1  111  6
: 3      0      x      x  xxx  x
```

Figura 13 - simulação parte 09

```
----- CLOCK 01 -----
:
: ENTRADA
: WRITE 1 TAG 000 INDICE 1 DADO 7
:
: SAIDA
: HIT 1          WRITEBACK 0
: ADDRESS_CACHE 1 DADO-CACHE 7
:
: RAM
: WREN 0 DADO-IN 7 ADDRESS 1 DADO-RAM 0
:
: CACHE 0
: INDICE VALIDO DIRTY LRU TAG DADO
: 0      1      0      1  100  0
: 1      1      1      1  000  7
: 2      1      1      0  101  5
: 3      0      x      x  xxx  x
:
: CACHE 1
: INDICE VALIDO DIRTY LRU TAG DADO
: 0      1      0      0  101  0
: 1      0      0      0  xxx  4
: 2      1      0      1  111  6
: 3      0      x      x  xxx  x
```

## 5 - Write hit TAG = 111 Índice= 10 Dado = 010

Figura 14 - simulação parte 10

```

# ----- CLOCK 00 -----
#
# ENTRADA
# WRITE 1 TAG 111 INDICE 2 DADO 2
#
# SAIDA
# HIT 1          WRITEBACK 0
# ADDRESS_CACHE 6 DADO-CACHE 6
#
# RAM
# WREN 0 DADO-IN 6 ADDRESS 30 DADO-RAM 0
#
# CACHE 0
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      1      0      1  100  0
# 1      1      1      1  000  7
# 2      1      1      0  101  5
# 3      0      x      x  xxx  x
#
# CACHE 1
# INDICE VALIDO DIRTY LRU TAG DADO
# 0      1      0      0  101  0
# 1      0      0      0  xxx  4
# 2      1      1      1  111  6
# 3      0      x      x  xxx  x

```

Figura 15 - simulação parte 11

```

----- CLOCK 01 -----

ENTRADA
WRITE 1 TAG 111 INDICE 2 DADO 2

SAIDA
HIT 1          WRITEBACK 0
ADDRESS_CACHE 6 DADO-CACHE 2

RAM
WREN 0 DADO-IN 2 ADDRESS 30 DADO-RAM 0

CACHE 0
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0      1  100  0
1      1      1      1  000  7
2      1      1      0  101  5
3      0      x      x  xxx  x

CACHE 1
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0      0  101  0
1      0      0      0  xxx  4
2      1      1      1  111  2
3      0      x      x  xxx  x

```

**6 - Write Miss com write back TAG = 110 Índice = 10 Dado = 011**  
**Primeiro clock**

*Figura 16 - simulação parte 12*

```

----- CLOCK 00 -----

ENTRADA
WRITE 1 TAG 110 INDICE 2 DADO 3

SAIDA
HIT 0          WRITEBACK 1
ADDRESS_CACHE 2 DADO-CACHE 5

RAM
WREN 1 DADO-IN 5 ADDRESS 22 DADO-RAM 0

CACHE 0
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    1  100  0
1      1      1    1  000  7
2      1      1    1  101  5
3      0      x    x  xxx  x

CACHE 1
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    0  101  0
1      0      0    0  xxx  4
2      1      1    0  111  2
3      0      x    x  xxx  x

```

*Figura 17 - simulação parte 13*

```

----- CLOCK 01 -----

ENTRADA
WRITE 1 TAG 110 INDICE 2 DADO 3

SAIDA
HIT 0          WRITEBACK 1
ADDRESS_CACHE 2 DADO-CACHE 0

RAM
WREN 1 DADO-IN 0 ADDRESS 22 DADO-RAM 5

CACHE 0
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    1  100  0
1      1      1    1  000  7
2      1      1    1  101  0
3      0      x    x  xxx  x

CACHE 1
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    0  101  0
1      0      0    0  xxx  4
2      1      1    0  111  2
3      0      x    x  xxx  x

```

**6 - Write Miss com write back TAG = 110 Índice = 10 Dado = 011**  
**Segundo clock**

*Figura 18 - simulação parte 14*

```

----- CLOCK 00 -----

ENTRADA
WRITE 1 TAG 110 INDICE 2 DADO 3

SAIDA
HIT 1          WRITEBACK 0
ADDRESS_CACHE 2 DADO-CACHE 0

RAM
WREN 0 DADO-IN 0 ADDRESS 26 DADO-RAM 5

CACHE 0
INDICE VALIDO DIRTY LRU TAG DADO
0       1       0    1  100  0
1       1       1    1  000  7
2       1       1    1  110  0
3       0       x    x  xxx  x

CACHE 1
INDICE VALIDO DIRTY LRU TAG DADO
0       1       0    0  101  0
1       0       0    0  xxx  4
2       1       1    0  111  2
3       0       x    x  xxx  x

```

*Figura 19 - simulação parte 15*

```

----- CLOCK 01 -----

ENTRADA
WRITE 1 TAG 110 INDICE 2 DADO 3

SAIDA
HIT 1          WRITEBACK 0
ADDRESS_CACHE 2 DADO-CACHE 3

RAM
WREN 0 DADO-IN 3 ADDRESS 26 DADO-RAM 0

CACHE 0
INDICE VALIDO DIRTY LRU TAG DADO
0       1       0    1  100  0
1       1       1    1  000  7
2       1       1    1  110  3
3       0       x    x  xxx  x

CACHE 1
INDICE VALIDO DIRTY LRU TAG DADO
0       1       0    0  101  0
1       0       0    0  xxx  4
2       1       1    0  111  2
3       0       x    x  xxx  x

```

**7 - Read Miss com write back TAG = 001 Índice = 10**  
**Primeiro clock**

Figura 20 - simulação parte 16

```

----- CLOCK 00 -----

ENTRADA
WRITE 0 TAG 001 INDICE 2 DADO x

SAIDA
HIT 0          WRITEBACK 1
ADDRESS_CACHE 6 DADO-CACHE 2

RAM
WREN 1 DADO-IN 2 ADDRESS 30 DADO-RAM 0

CACHE 0
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    1  100  0
1      1      1    1  000  7
2      1      1    0  110  3
3      0      x    x  xxx  x

CACHE 1
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    0  101  0
1      0      0    0  xxx  4
2      1      1    1  111  2
3      0      x    x  xxx  x

```

Figura 21 - simulação parte 17

```

----- CLOCK 01 -----

ENTRADA
WRITE 0 TAG 001 INDICE 2 DADO x

SAIDA
HIT 0          WRITEBACK 1
ADDRESS_CACHE 6 DADO-CACHE 0

RAM
WREN 1 DADO-IN 0 ADDRESS 30 DADO-RAM 2

CACHE 0
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    1  100  0
1      1      1    1  000  7
2      1      1    0  110  3
3      0      x    x  xxx  x

CACHE 1
INDICE VALIDO DIRTY LRU TAG DADO
0      1      0    0  101  0
1      0      0    0  xxx  4
2      1      1    1  111  0
3      0      x    x  xxx  x

```

**7 - Read Miss com write back TAG = 001 Índice = 10**  
**Segundo clock**

*Figura 22 - final da simulação*

```
----- CLOCK 00 -----  
  
ENTRADA  
WRITE 0 TAG 001 INDICE 2 DADO x  
  
SAIDA  
HIT 1          WRITEBACK 0  
ADDRESS_CACHE 6 DADO-CACHE 0  
  
RAM  
WREN 0 DADO-IN 0 ADDRESS 6 DADO-RAM 2  
  
CACHE 0  
INDICE VALIDO DIRTY LRU TAG DADO  
  0      1      0    1  100  0  
  1      1      1    1  000  7  
  2      1      1    0  110  3  
  3      0      x    x  xxx  x  
  
CACHE 1  
INDICE VALIDO DIRTY LRU TAG DADO  
  0      1      0    0  101  0  
  1      0      0    0  xxx  4  
  2      1      0    1  001  0  
  3      0      x    x  xxx  x
```

*Tabela 01 - Pinagem da placa*

<b>Entradas</b>	
SW 0 a 2	Dado
SW 3 a 4	Índice
SW 5 a 7	Tag
SW 8	Read ou Write
SW 17	Clock
<b>Saídas</b>	
LEDG 0	Hit ou Miss
LEDG 1	Write back
LEDG 2	LRU da cache 0 do índice fornecido
LEDG 3	LRU da cache 1 do índice fornecido
LEDG 4	Válido da cache 0 do índice fornecido
LEDG 5	Válido da cache 1 do índice fornecido
LEDG 6	Dirty da cache 0 do índice fornecido
LEDG 7	Dirty da cache 1 do índice fornecido
HEX 0	Dado de saída da cache
HEX 2	Tag acessada