(66.20) Oganizacion de Computadoras: TP 1

Christian Angelone (93971)
christiangelone@gmail.com

Agustin Gaillard (94849)
agufiuba@gmail.com

1er cuatrimestre 2019

1 Introduccion

Este trabajo practico, trata de mostrar la forma en que trabaja una cache (N = 4)WSA en modo WT/WA con politi de replazo FIFOb

2 Disenio y Implementacion

Para el disenio de este programa que intenta emular una cache (N = 4)WSA, decidimos basarnos en primitivas que representen los conceptos vistos en clase. A continuacion presentamos los structs correspondientes:

```
typedef struct nways_cache {
  way (* ways)[NUM_OF_WAY
  queue (* fifo)[NUM_OF_BLOCKS_PER_Y];

} nways_cache;
```

```
typedef struct way {
   block (* blocks)[NUM_OF_BLOCKS_PER_WAY];
} way;

typedef struct block {
  bool has_data;
  unsigned int tag;
  unsigned char (* content)[BLOCKSIZE];
} block;

typedef struct main_memory {
  unsigned char (* content)[RAM_SIZE];
} main_memory;
```

3 Instrucciones de Compilacion

Ejecutar:

```
$ gcc artist_ant.c paint.s -o artist_ant
```

4 Instrucciones de compilacion y ejecucion

Ejecutar:

```
root@debmips:~/tp1$ make cache
root@debmips:~/tp1$ ./cache archivo.mem
```

5 Ejecucion de casos de prueba y resultados

```
  # Prueba1
  root@debmips:~/tp1$ ./cache prueba1.mem

  [MISS] write_byte: Block bringed to cache and wrote (value: 'E') in cache
  Wrote (value: 'E') in memory (address: 0x00000000)

[MISS] write_byte: Block bringed to cache and wrote (value: 'D') in cache (t
  Wrote (value: 'D') in memory (address: 0x00000400)

[MISS] write_byte: Block bringed to cache and wrote (value: 'C') in cache (t
  Wrote (value: 'C') in memory (address: 0x00000800)

[MISS] write_byte: Block bringed to cache and wrote (value: 'B') in cache (t
  Wrote (value: 'B') in memory (address: 0x00001000)
[MISS] write_byte: Block bringed to cache and wrote (value: 'A') in cache (t
  Wrote (value: 'A') in memory (address: 0x00002000)

[MISS] read_byte: Read from memory (address: 0x00000000, value: 'E') and blo

[HIT] read_byte: Read from cache (value: 'D', tag: 2, set: 0, offset: 0)

[HIT] read_byte: Read from cache (value: 'C', tag: 4, set: 0, offset: 0)

[MISS] read_byte: Read from memory (address: 0x00002000, value: 'A') and blo

[MR: 0.78]

CACHE ==========================================================================
         SET0:         SET1:         SET2:         SET3:
SET4:         SET5:         SET6:         SET7:

FIFO:   <W2,W2,W3,]         <W0,W1,W2,W3,]         <W0,W1,W2,W3,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]         <W0,W1,W2,W3,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]

       old: W2      old: W0      old: W0      old: W0
old: W0      old: W0      old: W0      old: W0

  W0:  [B0]: E&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W1:  [B0]: A&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W2:  [B0]: C&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W3:  [B0]: B&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

CACHE ==========================================================================

 # Prueba2
  root@debmips:~/tp1$ ./cache prueba2.mem

  [MISS] read_byte: Read from memory (address: 0x00000000, value: '&') and b

[HIT] read_byte: Read from cache (value: '&', tag: 0, set: 0, offset: 31)

[MISS] write_byte: Block bringed to cache and wrote (value: '
') in cache (tag: 0, set: 1, offset: 0)
  Wrote (value: '
') in memory (address: 0x00000040)
```

```
[HIT] read_byte: Read from cache (value: '
', tag: 0, set: 1, offset: 0)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 1, offset: 0)
  Wrote (value: '') in memory (address: 0x00000040)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 1, offset: 0)

[MR: 0.33]

CACHE ==========================================================================
         SET0:         SET1:         SET2:         SET3:
SET4:         SET5:         SET6:         SET7:

FIFO:   <W1,W2,W3,]         <W1,W2,W3,]         <W0,W1,W2,W3,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]         <W0,W1,W2,W3,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]

       old: W1      old: W1      old: W0      old: W0
old: W0      old: W0      old: W0      old: W0

  W0:  [B0]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W1:  [B0]: (null) [B1]: (null) [B2]: (null) [B3]: (null) [B4]: (null) [B5]

  W2:  [B0]: (null) [B1]: (null) [B2]: (null) [B3]: (null) [B4]: (null) [B5]

  W3:  [B0]: (null) [B1]: (null) [B2]: (null) [B3]: (null) [B4]: (null) [B5]

CACHE ==========================================================================

 # Prueba3
  root@debmips:~/tp1$ ./cache prueba3.mem

[MISS] write_byte: Block bringed to cache and wrote (value: '') in cache (ta
  Wrote (value: '') in memory (address: 0x00000080)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 2, offset: 1)
  Wrote (value: '') in memory (address: 0x00000081)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 2, offset: 2)
  Wrote (value: '') in memory (address: 0x00000082)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 2, offset: 3)
  Wrote (value: '') in memory (address: 0x00000083)
```

```
[MISS] read_byte: Read from memory (address: 0x00000480, value: '&') and blo

[MISS] read_byte: Read from memory (address: 0x00000880, value: '&') and blo

[MISS] read_byte: Read from memory (address: 0x00000c80, value: '&') and blo

[MISS] read_byte: Read from memory (address: 0x00001080, value: '&') and blo

[MISS] read_byte: Read from memory (address: 0x00000080, value: '') and block

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 2, offset: 1)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 2, offset: 2)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 2, offset: 3)

[MR: 0.50]

CACHE ====================================================================
         SET0:         SET1:         SET2:         SET3:
SET4:         SET5:         SET6:         SET7:

FIFO:   <W0,W1,W2,W3,]         <W0,W1,W2,W3,]         <W0,W0,W0,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]         <W0,W1,W2,W3,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]

       old: W0       old: W0       old: W0       old: W0
old: W0       old: W0       old: W0       old: W0

  W0:  [B0]: (null) [B1]: (null) [B2]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W1:  [B0]: (null) [B1]: (null) [B2]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W2:  [B0]: (null) [B1]: (null) [B2]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W3:  [B0]: (null) [B1]: (null) [B2]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

CACHE ====================================================================

 # Prueba4
  root@debmips:~/tp1$ ./cache prueba4.mem
[MISS] write_byte: Block bringed to cache and wrote (value: '') in cache (ta
  Wrote (value: '') in memory (address: 0x00000000)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 0, offset: 1)
```

```
  Wrote (value: '') in memory (address: 0x00000001)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 0, offset: 2)
  Wrote (value: '') in memory (address: 0x00000002)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 0, offset: 3)
  Wrote (value: '') in memory (address: 0x00000003)

[HIT] write_byte: Wrote (value: '') in cache (tag: 0, set: 0, offset: 4)
  Wrote (value: '') in memory (address: 0x00000004)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 0)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 1)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 2)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 3)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 4)

[MISS] read_byte: Read from memory (address: 0x00001000, value: '&') and blo

[MISS] read_byte: Read from memory (address: 0x00002000, value: '&') and blo

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 0)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 1)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 2)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 3)

[HIT] read_byte: Read from cache (value: '', tag: 0, set: 0, offset: 4)

[MR: 0.18]

CACHE ================================================================================
         SET0:         SET1:         SET2:         SET3:
SET4:         SET5:         SET6:         SET7:

FIFO:   <W3,]         <W0,W1,W2,W3,]         <W0,W1,W2,W3,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]         <W0,W1,W2,W3,]
<W0,W1,W2,W3,]         <W0,W1,W2,W3,]

       old: W3       old: W0       old: W0       old: W0
```

```
old: W0      old: W0      old: W0      old: W0

  W0:  [B0]:  [B1]: (null) [B2]: (null) [B3]: (null) [B4]: (null) [B5]: (nul

  W1:  [B0]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W2:  [B0]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W3:  [B0]: (null) [B1]: (null) [B2]: (null) [B3]: (null) [B4]: (null) [B5]

CACHE ==========================================================================

 # Prueba5
  root@debmips:~/tp1$ ./cache prueba5.mem

[ERROR] Invalid line (R) in file: invalid address.
[MISS] read_byte: Read from memory (address: 0x00001000, value: '&') and blo

[MISS] read_byte: Read from memory (address: 0x00002000, value: '&') and blo

[HIT] read_byte: Read from cache (value: '&', tag: 8, set: 0, offset: 0)

[MISS] read_byte: Read from memory (address: 0x00000000, value: '&') and blo

[HIT] read_byte: Read from cache (value: '&', tag: 8, set: 0, offset: 0)

[MR: 0.60]

CACHE ==========================================================================
         SET0:        SET1:        SET2:        SET3:
SET4:        SET5:        SET6:        SET7:

FIFO:   <W3,]        <W0,W1,W2,W3,]        <W0,W1,W2,W3,]
<W0,W1,W2,W3,]        <W0,W1,W2,W3,]        <W0,W1,W2,W3,]
<W0,W1,W2,W3,]        <W0,W1,W2,W3,]

       old: W3      old: W0      old: W0      old: W0
old: W0      old: W0      old: W0      old: W0

  W0:  [B0]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W1:  [B0]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W2:  [B0]: &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

  W3:  [B0]: (null) [B1]: (null) [B2]: (null) [B3]: (null) [B4]: (null) [B5]
```

```
CACHE ===========================================================
```

6 Conclusiones

- La prueba 4 es la que mejor rendimiento tuvo, con un missrate de 0.18
- La prueba 1 es la de peor rendimiento tuvo, con un missrate de 0.78
- En pruemedio el missrate de la cache segun la pruebas corridas es de 0.48, lo que nos indica que aproximadamente la mitad de los accesos a cache terminan en hit, lo que a nuestro criterio es un muy buen resultado.