Jacob Branaugh, Brenn Kucey
November 10, 2013
CS 472 HW 4

(9.2) Why do computers use cache memory?

- Computers use cache memory because it transfers data quickly, and has a much larger capacity than the processor's registers. Cache memory is slightly slower than registers, but is significantly faster than RAM. There are only a handful of registers available, whereas cache can range from kB to MB. RAM, however, is usually available in the order of GB.

(9.3) What is the meaning of the following terms?

- Temporal locality

- Temporal locality describes the number of times a memory address is accessed within some amount of time.

- Spatial locality

- Spatial locality is a term to describe the physical closeness of related memory addresses.

(9.4) From first principles, derive an expression for the speedup ratio of a memory system with cache (assume the hit ratio is h and the ratio of the main storage access time to cache access time is k, where k < 1). Assume that the system is an ideal system and that you don't have to worry about the effect of clock cycle times.

- $S = \frac{1}{1-h(1-k)}$, $k = \frac{t_c}{t_m}$

(9.5) For the following ideal systems, calculate the speedup ratio S. In each case, t_c is the access time of the cache memory, t_m is the access time of the main store, and h is the hit ratio.

a) $t_m = 70ns, t_c = 7ns, h = .9$

$S = \frac{1}{1-.9(1-\frac{7}{70})} = \mathbf{5.263}$

b) $t_m = 60ns, t_c = 3ns, h = .9$

$S = \frac{1}{1-.9(1-\frac{3}{60})} = \mathbf{6.897}$

c) $t_m = 60ns, t_c = 3ns, h = .8$

$S = \frac{1}{1-.8(1-\frac{3}{60})} = \mathbf{4.167}$

d) $t_m = 60ns, t_c = 3ns, h = .97$

$S = \frac{1}{1-.97(1-\frac{3}{60})} = \mathbf{12.739}$

(9.6) For the following ideal systems, calculate the hit ratio h required to achieve the stated speedup ratio S.

a) $t_m = 60ns, t_c = 3ns, S = 1.1$

$h = \frac{1-\frac{1}{S}}{1-\frac{t_c}{t_m}} = \frac{1-\frac{1}{1.1}}{1-\frac{3}{30}} = \mathbf{0.096}$

b) $t_m = 60ns, t_c = 3ns, S = 2.0$

$h = \frac{1-\frac{1}{2}}{1-\frac{3}{60}} = \mathbf{0.526}$

c) $t_m = 60ns, t_c = 3ns, S = 5.0$

$h = \frac{1-\frac{1}{5}}{1-\frac{3}{60}} = \mathbf{0.842}$

d) $t_m = 60ns, t_c = 3ns, S = 15.0$

$h = \frac{1-\frac{1}{15}}{1-\frac{3}{60}} = \mathbf{0.982}$

(9.8) For the following systems that use a clocked microprocessor, calculate the maximum speedup ratio you could expect to see as h approaches 100%.

$S = \frac{1}{1-1(1-\frac{t_c}{t_m})} = \frac{1}{\frac{t_c}{t_m}} = \frac{t_m}{t_c}$, where t_m *and* t_c must be integer multiples of the time taken for one clock cycle.

a) $t_{cyc} = 20ns, t_m = 75ns, t_c = 15ns$

$t_m = 75ns \rightarrow t_m = 4\ cycles$
$t_c = 15ns \rightarrow t_c = 1\ cycle$
$S = \frac{4}{1} = \mathbf{4}$

b) $t_{cyc} = 20ns, t_m = 75ns, t_c = 25ns$

$t_m = 75ns \rightarrow t_m = 4\ cycles$
$t_c = 25ns \rightarrow t_c = 2\ cycle$
$S = \frac{4}{2} = \mathbf{2}$

c) $t_{cyc} = 10ns, t_m = 75ns, t_c = 15ns$

$t_m = 75ns \rightarrow t_m = 8\ cycles$
$t_c = 15ns \rightarrow t_c = 2\ cycle$
$S = \frac{8}{2} = \mathbf{4}$

(9.11) In a direct-mapped cache memory system, what is the meaning of the following terms.

- Word:
- Line: basic unit of storage in cache memory
- Set: a collection of lines equal to $\frac{size\ of\ main\ memory}{size\ of\ cache}$

(9.12) What is the binary encoding of the following instructions?

a) STRB r1, [r2]

- 1110 01 0 0 0 1 0 0 0010 0001 000000000000

b) LDR r3, [r4, r5]!

- 1110 01 1 1 1 0 1 1 0100 0011 00000 00 0 0101

c) LDR r3, [r4], r5

- 1110 01 1 0 1 0 1 1 0100 0011 00000 00 0 0101

d) LDR r3, [r4, #-6]!

- 1110 01 0 1 0 0 1 1 0100 0011 111111111010

(9.17) Write an ARM assembly language program that scans a string terminated by the null byte 0x00 and copies the string from a source location pointed at by r0 to a destination pointed at by r1.

```
START:                  ; start of loop
LDR   r2, [r0], #4      ; load byte into r2, increment r0 pointer
CMP   r2, #0x00         ; check if byte is null
BEQ   END               ; jump to end of loop if null byte
STRB  r2, [r1], #4      ; store read byte into r1, increment r1 pointer
B     START             ; jump to start of loop
END:                    ; end of loop, continue with program
```

(9.22) Write an ARM assembly language program to determine whether a string of characters with and odd length is a palindrome under the following constraints: The string of ASIC-encoded characters is stored in memory, At the start of the program, register r1 contains the address of the first character in the string, and r2 contains the address of the last character. On exit from the program, register r0 contains a 0 if the string is not a palindrome, and 1 if it is.

```
START:                  ; start of loop
CMP    r1, r2           ; check if registers are pointing to same location
MOVEQ  r0, #1           ; load immediate 1 into r0 if same location
BEQ    END              ; jump to end of loop
LDR    r3, [r1], #4     ; load data at location in r1 to r3, increment r1
LDR    r4, [r2], #-4    ; load data at location in r2 to r4, decrement r2
CMP    r3, r4           ; compare data at those locations
MOVNE  r0, #0           ; store immediate 0 to r0 if not same character
BNE    END              ; jump to end of loop
B      START            ; jump to start of loop
END:                    ; end of loop, continue with program
```

(9.23)

(9.26)

(9.28)

(9.35)

(9.41)

(9.42)

(9.43)

(9.45)

(9.46)

(9.57)