Haowen Xue

ECE 437 HW8

Mg257

5.11.1

Given the address stream shown, and the initial TLB and page table states provided above, show the final state of the system. Also list for each reference if it is a hit in the TLB, a hit in the page table, or a page fault.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Addr | Addr in Binary | Page number | Page number(base10) | Final status |
| 4669 | 0001001000111101 | 0001 | 1 | Page Fault |
| 2227 | 0000100010110011 | 0000 | 0 | Page Hit |
| 13916 | 0011011001011100 | 0011 | 3 | TLB Hit |
| 34587 | 1000011100011011 | 1000 | 8 | Page Fault |
| 48870 | 1011111011100110 | 1011 | 11 | Page Hit |
| 12608 | 0011000101000000 | 0011 | 3 | TLB Hit |
| 49225 | 1100000001001001 | 1100 | 12 | Page Hit |

|  |  |
| --- | --- |
| Valid | Physical Page |
| 1 | 5 |
| 1 | 13 |
| 0 | Disk |
| 1 | 6 |
| 1 | 9 |
| 1 | 11 |
| 0 | Disk |
| 1 | 4 |
| 1 | 14 |
| 0 | Disk |
| 1 | 3 |
| 1 | 12 |
| 1 | 15 |

Final TLB and Page Table:

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 12 | 15 |
| 1 | 8 | 14 |
| 1 | 3 | 6 |
| 1 | 11 | 12 |

5.11.2

Repeat 5.11.1, but this time use 16 KiB pages instead of 4 KiB pages. What would be some of the advantages of having a larger page size? What are some of the disadvantages?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Addr | Addr in Binary | Page number | Page number(base10) | Final status |
| 4669 | 0001001000111101 | 0000 | 0 | Page Fault |
| 2227 | 0000100010110011 | 0000 | 0 | TLB Hit |
| 13916 | 0011011001011100 | 0000 | 0 | TLB Hit |
| 34587 | 1000011100011011 | 0010 | 2 | Page Fault |
| 48870 | 1011111011100110 | 0010 | 2 | TLB Hit |
| 12608 | 0011000101000000 | 0000 | 0 | TLB Hit |
| 49225 | 1100000001001001 | 0011 | 3 | TLB Hit |

|  |  |
| --- | --- |
| Valid | Physical Page |
| 1 | 5 |
| 0 | Disk |
| 0 | Disk |
| 1 | 6 |
| 1 | 9 |
| 1 | 11 |
| 0 | Disk |
| 1 | 4 |
| 0 | Disk |
| 0 | Disk |
| 1 | 3 |
| 1 | 12 |
| 1 | 13 |

Final TLB and Page table:

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 2 | 13 |
| 1 | 7 | 4 |
| 1 | 3 | 6 |
| 1 | 0 | 5 |

The advantage of a larger page size could bring the cache associative mapping be increased and be more efficient. And the virtual address size is decreased. Therefore, the mapping of the block offset is also reduced.

However, the disk searching latencies could be increased to be high, which could slow down the process.

5.11.3

Show the final contents of the TLB if it is 2-way set associative. Also show the contents of the TLB if it is direct mapped. Discuss the importance of having a TLB to high performance. How would virtual memory accesses be handled if there were no TLB.

Final TLB for 4669:

Final TLB for 2227:

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 11 | 12 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 0 | 4 | 9 |

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 11 | 12 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 0 | 4 | 9 |

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 0 | 5 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 0 | 4 | 9 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Addr | Addr in Binary | Page number | Page number(base10) | Final status |
| 4669 | 0001001000111101 | 0001 | 1 | Page Fault |
| 2227 | 0000100010110011 | 0000 | 0 | Page Hit |
| 13916 | 0011011001011100 | 0011 | 3 | TLB Hit |
| 34587 | 1000011100011011 | 1000 | 8 | Page Fault |
| 48870 | 1011111011100110 | 1011 | 11 | Page Hit |
| 12608 | 0011000101000000 | 0011 | 3 | TLB Hit |
| 49225 | 1100000001001001 | 1100 | 12 | Page Hit |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Addr | Addr in Binary | Page number | Page number(base10) | Final status |
| 4669 | 0001001000111101 | 0001 | 1 | Page Fault |
| 2227 | 0000100010110011 | 0000 | 0 | Page Hit |
| 13916 | 0011011001011100 | 0011 | 3 | TLB Hit |
| 34587 | 1000011100011011 | 1000 | 8 | Page Fault |
| 48870 | 1011111011100110 | 1011 | 11 | Page Hit |
| 12608 | 0011000101000000 | 0011 | 3 | TLB Hit |
| 49225 | 1100000001001001 | 1100 | 12 | Page Hit |

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 0 | 5 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 1 | 0 | 6 |

Final TLB for 13916:

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 2 | 14 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 1 | 0 | 6 |

Final TLB for 34587:

Final TLB for 48870:

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 2 | 14 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 1 | 2 | 12 |

Final TLB for 12608:

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 2 | 14 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 1 | 0 | 6 |

Final TLB for 12608:

|  |  |  |
| --- | --- | --- |
| Valid Bit | Tag | Physical page number |
| 1 | 3 | 15 |
| 1 | 0 | 13 |
| 1 | 3 | 6 |
| 1 | 0 | 6 |

The high performance for TLB could reduce the time for physical memory lookup mapping. If there is no TLB, the virtual memory would have to lookup and map each memory location once a time, which would lead to a inefficient performance.

5.11.4

Given the parameters shown above, calculate the total page table size for a system running 5 applications that utilize half of the memory available.

Number of page table entries = (2^32 ) / (2^10 \* 2^3 ) = 2^19

Page Table Size = 2^19 \* 2^2 = 2MiB

Sine it only utilize half of the memory available:

Utilized size = 1 MiB

Therefore, Total page size used = 1MiB \* 5 = 5MiB.