CS-534: Packet Switch Architecture
Spring 2006

Department of Computer Science

Exercise Set 8: Queueing Architectures SRAM Cost

Assigned: (normally: wk.8) 16/4/2008 (wk.9) - Due: (normally: wk.9) Wed. 7 May 2008 (wk.10)

8.1 Input/Output/Shared Queueing using SRAM chips

We wish to estimate the total buffer memory cost, in terms of number of SRAM chips, of a **12x12** switch with **10-Gigabit Ethernet** ports, in the following four cases: (a) output queueing, (b) block-crosspoint (block-shared) queueing, (c) input (virtual-output) queueing, (d) internal speedup with input and output queues.

To simplify our task, let us assume that the SRAM chips to be used: *(i)* have a shared DQ (read/write) data bus --so that we do not have to worry about read-to-write access ratio-- and *(ii)* use ZBT timing without any lost cycles when the bus direction changes between reads and writes (not a realistic assumption at such high clock rates as we have here). Further assume that each chip has a 32-bit (4-Byte) wide data bus, and uses double-clocking (DDR timing) with a burst-of-2. Hence, we can have **one access** (to an arbitrary address) **per clock cycle**, and each access concerns **8 Bytes** of data (read or written). The maximum allowable clock frequency for these SRAM chips is **200 MHz**.

The "10-Gigabit Ethernet" standard for high-speed links is 10 times faster than Gigabit Ethernet. Assume that the interframe gap and the preamble have the same size in both standards (I am not sure if this is true in reality); the ethernet header, CRC, and payload sizes should also be the same. Thus, assuming the same sizes as in exercise 3.3(e), the total **per-packet overhead is 38 Bytes**, and the **packet size is 46 to 1500 Bytes**, where by "packet" we mean the information that has to be stored in our buffer memories. In our switch queueing architectures, we will have two different kinds of buffer memories, each with a different packet size as its limiting factor:

i. Very wide memories, that provide the maximum possible throughput given today's SRAM technology and the packet size range. We choose the **segment size = 128 Bytes** (a power of 2, larger than twice the minimum packet size). For this segment size, the worst case segment rate of the link is **14.9 Msegments/second**, and it results for minimum-sized packets (packet rate = (10000 Mb/s) / ((38 B + 46 B) * 8 b/B) = 14.9 Mpck/s; segment rate = packet rate, since each packet fits in one segment). (We know that other "bad" segment rates to check are for packet sizes 132 B (just overflows from 1 to 2 segments), 260 B (just overflows from 2 to 3 segments), ..., and 1412 B (just overflows from 11 to 12 segments). The packet rate for 132-byte packets is = (10000 Mb/s) / ((38 B + 132 B) * 8 b/B) = 7.35 Mpck/s; the corresponding segment rate is twice that, i.e. 14.7 Mseg/s, since we need 2 segment accesses to store or read a packet of that size. The segment rate for 260-byte packets is 12.6 Mseg/s, etc).

ii. Narrow memories, that provide a throughput just twice as much as the link throughput, or a bit more than that (speed-up). These can be implemented using two of our SRAM chips in parallel (since each of our SRAM chips can provide a data throughput up to 200 MHz * 64 b = 12.8 Gb/s). To verify this, consider that each access to this 2-chip buffer reads or writes 16 Bytes (8 Bytes per chip). For each 46-byte minimum-sized packet, 3 memory accesses are needed, yielding a required access rate of: 2 (incoming + outgoing traffic) * 3 accesses/pck * (10000 Mb/s) / ((38 B/pck + 46 B/pck) * 8 b/B) = 2*3*14.9 Macc/s = 89.4 Macc/s; this is comfortably below our memory's capability for up to 200 Macc/s, but this is **not** the worst case for this narrow memory. The worst case access rate, for this memory, is for 93*16+4 = 1492-byte packets, which require 94 accesses to this memory, each (the 1500-byte maximum-sized packets also get accomodated with 94 accesses each, but they yield slightly lower packet rate). The required memory access rate for back-to-back 1492-byte packets is: 2 * 94 acc/pck * (10000 Mb/s) / ((38 B/pck + 1492 B/pck) * 8 b/B) = 2*94*0.817 = 153.6 Maccesses/second.

(a) Output Queueing:
Using this architecture, our 12x12 switch will need 12 output buffer memories. Each of these memories must provide a very high throughput, due to its fan-in of 12 links, hence it must follow organization *(i)* above. What is the peak segment rate that each of these memories must support? What SRAM clock frequency must we use to achieve that? How many SRAM chips do we need to use in parallel, in order to build each such memory? (*Hint:* just one memory access must suffice to read or write an entire 128-Byte segment). Given that our switch needs 12 output buffer memories, what is the total number of SRAM chips needed for the entire switch? If each SRAM chip consumes 2 Watts of power, how much power does the entire buffer memory consume? Fast SRAM chips are expensive; if each chip costs 20 Euro, what is the cost of buying all SRAM chips for one switch? Assume that the remaining components of the switch cost 3 times as much as the SRAM chips (a conservative estimate); if the average selling price (ASP) of the switch is 5 times its components cost (see Hennessy and Patterson, Computer Architecture, chapter 1), what would be the ASP of this switch?

(b) Block-Crosspoint (Block-Shared) Queueing:
According to this architecture, our 12x12 switch is made of a 2x2 array of buffer memories, where each buffer memory serves 6 inputs and 6 outputs, i.e. each buffer memory forms a small 6x6 shared-buffer switch. Each of these memories must again provide a very high throughput, so it must follow again organization *(i)* above. What is the peak segment rate that each of these memories must support? What SRAM clock frequency must we use to achieve that? How many SRAM chips do we need to use in parallel, in order to build each such memory? What is the total number of SRAM chips needed for the entire switch, in this case? How much power does the entire buffer memory consume, now? What is the cost of buying all these SRAM chips, and what would be the ASP of this switch?

(c) Input (Virtual-Output) Queueing:
For this architecture, we need 12 input buffer memories for our 12x12 switch. (Within each buffer memory, multiple logical queues (per-output, per-priority, etc) should be implemented; this does not affect our throughput calculation, here, provided of course that the (separate!) queue-pointer memories can keep up with the required operation rate). Each of these memories must now provide a throughput just twice as much as the link throughput, hence it will follow organization *(ii)* above. What SRAM clock frequency must we use to achieve that, for the worst-case (for this buffer) packet size? How many SRAM chips do we need for the entire switch? How much power do all the buffer memories consume? What is the cost of buying all these SRAM chips, and what would be the ASP of this switch?

(d) Internal Speedup with Input and Output Queues:
By running each input buffer of question (c) with a faster clock, its throughput is increased. Using the fastest allowable clock (200 MHz), the aggregate memory access rate can reach 200 Maccesses/second. How much of that access rate can be consumed by the incoming link, for worst-case (for this buffer) packet sizes? How many accesses per second remain available for the crossbar to use? **What speedup factor** does that represent? Besides this worst-case analysis, also make an average-case calculation for the speedup factor, assuming 320 Byte packets (320 = 2.5 * 128), which require 3 segment accesses each. This switch also includes output buffer memories: how many are these, what aggregate access rate must each provide, and what clock frequency and how many SRAM chips do they need? How many SRAM chips do we need for the entire switch, how much power do they consume, what is the cost of buying them, and what would be the ASP of this switch?

8.2 Single-Chip Shared-Buffer Switch

We wish to estimate how many **10-Gigabit Ethernet** ports a single-chip shared-buffer switch can have, and what would be the capacity, area, and power consumption of the corresponding SRAM blocks on the chip. In other words, for an NxN shared-buffer switch, implemented on a single chip, with 10-Gigabit Ethernet ports, we wish to estimate how large N can be. The segment size will be **128 Bytes**, as in the above exercise 8.1.

Assume that the chip will be implemented in the 0.13-micron CMOS technology that we saw in class (section 2.2), and that the shared buffer will be built out of 1K x32 two-port SRAM blocks, operating at 330 MHz (their worst-case maximum rate, as we saw).

(a) Number of Ports:
What is the maximum access rate that a shared buffer can offer, when implemented using the above SRAM blocks? Given the worst-case segment rate analysis for 10-Gigabit Ethernet that was presented in the above exercise 8.1 (14.9 Msegments/second/port), and given that our shared-buffer architecture uses the "very wide memory" organization above, how many ports, N, can this buffer support?

(b) Number of Blocks, SRAM Capacity, Area, and Power Consumption:
How many SRAM blocks are needed in parallel to yield the performance required by (a), i.e. in order for an entire segment to be accessible in just one clock period? What is the capacity of the resulting shared buffer in Kbits, KBytes, and Ksegments? How much silicon area will these SRAM blocks occupy on the chip? How much power will they consume at the clock frequency at which they need to operate?

--- *Optional Part:* ---
(c) Evaluation - Adaptation:

- Is the power consumption (b) realistic or excessive? Say that an upper bound for the power consumption of a large and sophisticated chip, like this one, is 25 Watts, and that 18 out of those 25 Watts can be dedicated to packet I/O (pin) drivers/receivers and to the shared buffer SRAM, thus leaving 7 W for all remaining logic. Assume, as in section 2.2, that I/O pin power consumption is 20 mW/Gbaud; thus, each 10-Gigabit Ethernet port consumes approx. 0.5 Watt (10 Gbps = 12.5 GBaud input rate plus 12.5 Gbaud output rate --given 8B/10B encoding-- i.e. 25 GBaud x 20 mW/Gbaud = 500 mW). How many ports would that allow us to have? --assume, for simplicity, that power consumption scales linearly with the number of ports.
- Is the silicon area cost (b) realistic, excessive, or small? Say that a reasonable area for a large and sophisticated chip, like this one, is 240 square mm, and that one third of that (80 mm^2) can be dedicated to the shared buffer SRAM, leaving a second third to chip I/O, and another third to all remaining logic. How much buffer space would that allow us to have? --assume, for simplicity, that buffer space scales linearly with silicon area. Given the final number of ports (minimum of (a) and (c) above) and the final buffer space based on area as calculated here, what would be the per-port buffer space allocation if buffer space were allocated equally to each output port (bits, Bytes, segments)?

Up to the Home Page of CS-534

Last updated: 16 Apr. 2008, by M. Katevenis.