An Industry Standard Benchmark Consortium

Networking Version 2.0

Benchmark Name: IP Packet Check

Highlights

- **Simulates a network router with four data sets to compare memory bus effects**
- **Focuses on checksum calculations and logical compare operations**

Application

The IP Packet Check benchmark performs a subset (essentially the IP Header Validation) of the network layer forwarding function of the Internet protocol suite as specified in RFC1812, "Requirements for IP Version 4 Routers" which can be found at http://www.faqs.org/rfcs/rfc1812.html. The benchmark provides an indication of the potential performance of a microprocessor in an IP router system.

A TCP/IP router normally examines the IP protocol header as part of the switching process. It generally removes the Link Layer header from a received message, modifies the IP header, and replaces the Link Layer header for retransmission. In this benchmark, the Link Layer header has already been removed and will not be replaced, i.e. all processing is done at Layer 3, on the assumption that lower level functions are handled by hardware or an interrupt service routine.

Benchmark Description

The benchmark simulates a router with four network interfaces. It initializes a buffer of programmable size (512 KB, 1 MB, 2 MB and 4 MB for reporting purposes) with IP datagrams. The header is always the minimum 20 bytes and is made up random characters except in the byte positions to be checked (IP version, checksum, and length). A checksum for the IP header is calculated and stored in each datagram. Errors are introduced in certain headers and an error count is logged. Datagrams are allowed to be aligned on the best natural boundary of the microprocessor and padding is added between them.

As a benchmark, the IP packet size is chosen randomly to be either 46 bits (small packets) or 1500 bits (large packets) in size. Packet receipt is simulated by creating a dummy store queue of 512 KB (approximately 370 packets), 1 MB (~720 packets), 2 MB (~1400 packets) or 4 MB (~2800 packets) outside of the timing loop. One timed iteration of the benchmark consists of processing each packet header pointed to by the receive queue and moving the descriptor to a holding queue. Results are reported in iterations per second for each of the buffer sizes but can be equated to packets per second by the conversion of ~660, ~1320, ~2640 or ~5280 packet headers checked per iteration, respectively.

Two descriptor queues are created with a pointer to the next descriptor and a pointer to the datagram header. One queue is called the receive queue (`rx_queue` in the code) and the other queue is the holding queue (`hold_que` in the code). IP datagrams are often stored like this in actual systems using descriptors that are separate from the datagram. A descriptor has a next member that allows it to be put in a linked list and a pointer to a datagram.

Benchmark Description (continued)

As each datagram is processed by the benchmark algorithm it is removed from the receive queue and placed in the holding queue. Processing consists of:

1. Checking that the packet length is large enough to hold the minimum length legal IP datagram (>=20 bytes).
2. Checking that the IP checksum is correct (a bad packet counter is incremented if the checksum is not correct)
3. Checking that the IP version number is 4
4. Checking that the IP header length field is large enough to hold the minimum length legal IP datagram (20 bytes = 5 words)
5. Checking that the IP total length field is large enough to hold the IP datagram header, whose length is specified in the IP header length field

Cache route lookup, the routing decision, and test for local delivery, which would normally be a part of packet routing are not implemented in Version 1.0 of this benchmark. Cache route lookup, however, is implemented in the EEMBC Route Lookup benchmark and those results can be combined with IP Packet Check to get a better indication of microprocessor performance in an IP router system.

A single iteration of the benchmark is complete when the receive queue of packet descriptors is empty. At the end of one iteration, the receive queue and the holding queue are switched allowing the next iteration to execute with a full receive queue.

Analysis of Computing Resources

The IP Packet Check benchmark performs integer math on 16 bit unsigned quantities (the checksum calculation) and shift and logical compare operations (the IP version number and length checks). These operations and accessing the data from memory are primarily what is tested by this benchmark. Though the buffer sizes in memory are large, the checksum and verification process is only over the IP headers, which tend to take up residence in cache; therefore even at the largest buffer sizes, this benchmark has a high cache hit rate for microprocessors with 32KB of Data Cache. (The headers and packet descriptors for a 1MB buffer come very close to fitting in 32 KB of L1.) The code size is trivial and easily fits in even a small L1 Instruction Cache.

Special Notes

Do not directly compare the results of IP Packet Check benchmark to EEMBC Networking Version 1 Packet Flow benchmark. Even though the two benchmarks test the same function, the algorithm was changed in IP Packet Check to allow a user specified alignment without impacting the number packets processed.