AN 10.12

Performance Analysis and Software Optimization on Systems Using the LAN91C111

1 Introduction

This application note describes one approach to analyzing the performance of a LAN91C111 implementation and fine-tuning the software to increase system throughput. Often, the user looks to the device driver for an answer to boost the overall Ethernet packet processing (receiving) speed in the system. While optimizations in the driver may well be sufficient to enhance the performance so that the system throughput requirements can be met, it by no means can be guaranteed. It is useful to understand all parameters that play a part in affecting the observed system performance. It is important to understand where the bottleneck is, and it is important to recognize the point where one has done all one could in software optimization.

1.1 Some Queuing Theory

The first step to establishing a framework for calculating the theoretical throughput is to adopt a mathematical model used commonly in network performance analysis. For this, the author went back to a textbook he used years ago. The reference to the textbook is given at the end of this application note. There are lots, and lots, of books on the subject of queuing theory. Only the results of the applicable model are presented here. If the reader is interested in the derivation, please enjoy yourself by referring to one of these books. The mathematical model that is most applicable to our case is known as an $M/M/1/K$ queuing system. This notation denotes a system as follows:

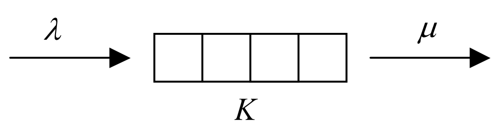

Figure 1.1 $M/M/1/K$ **Queuing System**

1. There is one input stream of packets, as there is only one PHY in our LAN91C111.
2. There is one process which retrieves the packets from the LAN91C111.
3. A certain number, K, of buffers form a queue, are responsible for receiving packets.

The packet traffic at the input of the queue is characterized by the average number of packets per second, λ, at which they arrive at the queue. The device driver takes the packets out of the queue at an average rate of μ packets per second. Both of these processes are modeled by a probabilistic distribution called the Poisson distribution. To cut to the chase, the following expressions give, in the steady state, the probability p_K , that a packet arrives at the input of the system and finds the queue full, at which the point the packet will be dropped by the LAN91C111.

$$p_K = \frac{(1-a)a^K}{1-a^{K+1}} \qquad \text{for} \quad \lambda \neq \mu$$

$$p_K = \frac{1}{K+1} \qquad \text{for} \quad \lambda = \mu$$

where $a = \lambda/\mu$.

Figure 1.2 Poisson Distribution

The throughput is related to the probability that a packet arrives at the queue and finds the queue *not* full, or 1 - p_K . This is proportional to the number of packets *not* dropped in a given period.

2 Performance as a Function of μ

It is apparent that the performance characteristics of the system depends on several variables: how fast the system removes the packet from the LAN91C111, how many buffers are allocated in the LAN91C111 for receiving packets, and, less intuitively, how fast the packets are arriving the LAN91C111. The following figure depicts the system throughput, expressed as the ratio between the rate of packets *not* dropped by the system and the rate of total packets arriving, for the given scenario. This we define as the bandwidth utilization, which is the same as 1 - p_K.

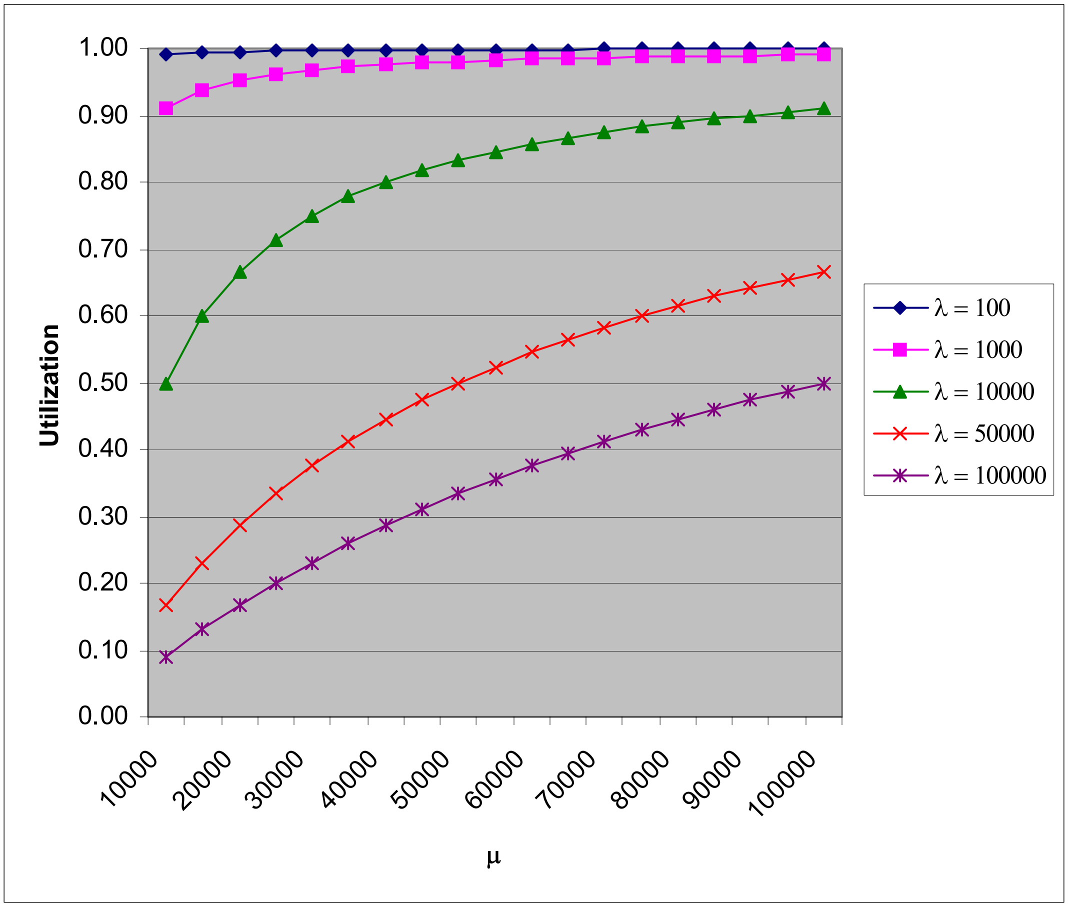

Figure 2.1 Bandwidth Utilization for K=1

It easy to see that the faster the CPU is able to take packets out of the LAN91C111 buffers, the better the system is able to receive all or most of the input packets. It seems obvious, but one can never expect to receive more than μ packets per second, on the average.

3 Performance as a function of *K*

The number of buffers allocated for receiving packets also affects the performance of the system. Figure 3.1 correlates the utilization to the number of receiving buffers for a particular packet arrival process characteristic.

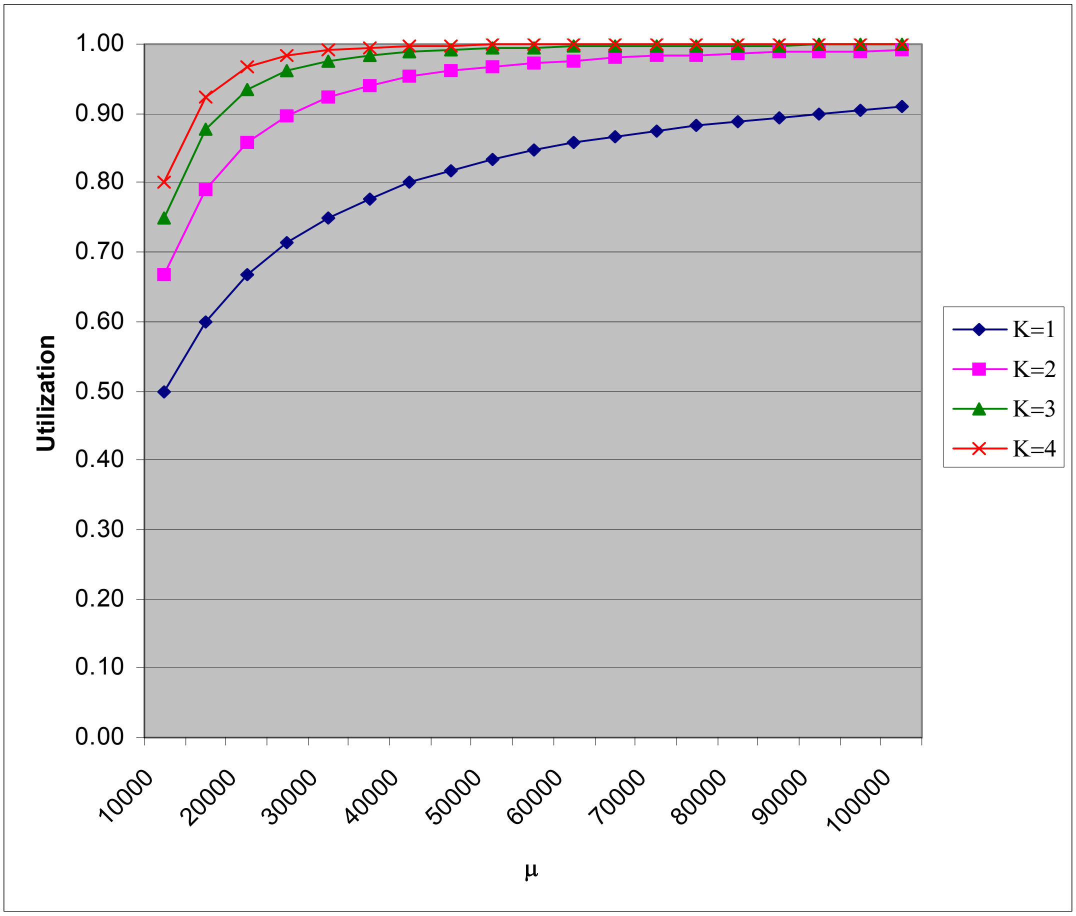

Figure 3.1 Bandwidth Utilization for λ = 10000

It can be seen that, given our mathematical model, increasing the number of buffers in the queue beyond 2 does not greatly affect the system throughput when the system's processing speed is high compared to the packet arrival rate.

4 System Performance Analysis Procedure

Follow the steps below to analyze the system before attempting to optimize. Knowing where the bottleneck is gives you "the most bang for the buck" in terms of using your efforts wisely.

1. Are packets currently being discarded by the driver (or anywhere else in software) because the TCP/IP stack input buffer is full? If so, the application software or the operating system is too slow. Optimizing the driver will not help to increase the system performance.
2. Determine the number of buffers allocated for receiving in the LAN91C111. This is your K.
3. Attempt to measure how much time, in seconds, it takes to process one packet in the Interrupt Service Routine (ISR). This includes the following times:

a. Interrupt latency

b. Time required to copy the packet data from the LAN91C111 to main memory

The reciprocal of this number is your μ. You will need to instrument your software to measure this over a large number of packets. μ is the average processing rate.

4. Plot the equation on the first page for 1 - p_K , as a function of λ, with the K and μ values found from the two steps above. It should look something like one of the following curves.

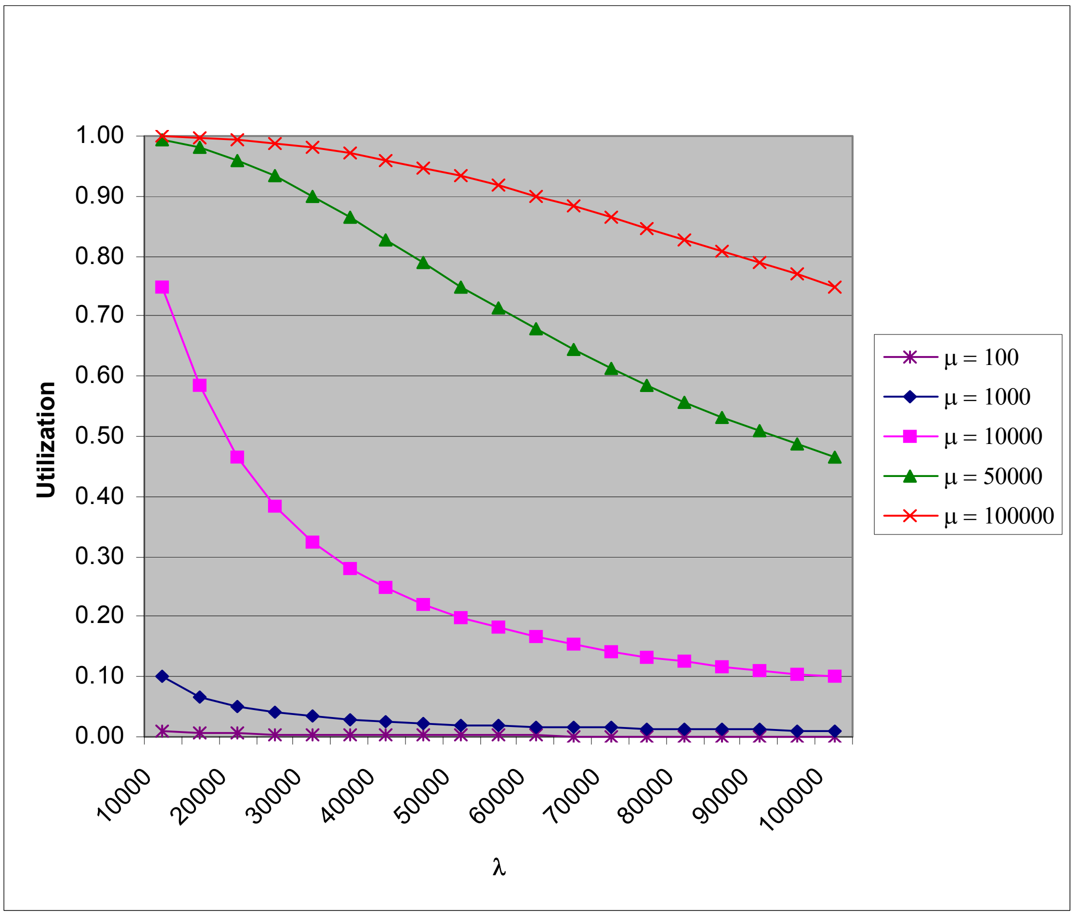

Figure 4.1 Bandwidth Utilization for K=3

5. Measure how quickly the packets arrive at your system. Over a period of time, calculate the average number of packets arriving per second. This is your actual λ. Since your system is dropping packets (or else, you might not care to do this analysis in the first place), you should not use your software to do the measurement. You will want to do this measurement by using a network sniffer or such similar equipment.

6. Measure the number of packets received by the software. Correlate λ to the chart from **Step 3**. Does it make sense in terms of how many packets are currently being dropped?

5 System Performance Optimization Recommendations

Try the following techniques in software to increase the system throughput performance. Do them one at a time. Each time after modifying the software, reassess the new performance by repeating **Step 6** described in the System Performance Analysis Procedure.

1. Allocate the right number of buffers, K, in the LAN91C111 for receiving. Larger is better, but take care to not negatively affect packet transmission performance to the point of not meeting system requirements. The effectiveness of varying this parameter depends on the "burstiness" of the packets arriving at the input of the system, which in turn is dependent on the network application and the instantaneous network condition.
2. Rewrite the ISR to make copying packet data from the LAN91C111 buffers to main memory more efficient. Consider rewriting it in assembly language if necessary. Copy all packets present in the buffers for each ISR invocation. There is no downside to taking this optimization action (except that assembly code is less readable, harder to maintain, and more difficult to port).
3. Increase the interrupt priority for packet receives to lower the interrupt latency. Shortening the interrupt latency lowers the buffer depth, K, requirement. Take care to prevent starving other (non-networking) processes of CPU cycles.

If you still need more performance gain after taking the actions above, there are usually things that can be tried to modify the hardware design in order to increase μ.. However, it is difficult to make specific recommendations without seeing the particular designs. We heard stories that involve the LAN91C111 on a 16-bit bus with 20 wait states – there is no software in the world that is going to fix that! As our rule of thumb, the raw bus bandwidth should be about twice the maximum data bandwidth through an I/O device. For the LAN91C111, that means 50 MBps, or 80 nS cycle time on a 32-bit bus if you want to do 100 Mbps full-duplex.

6 References

Schwartz, Mischa, "Computer-Communication Network Design and Analysis," Prentice-Hall, Inc., Englewood Cliffs, N.J., 1977.

80 Arkay Drive
Hauppauge, NY 11788
(631) 435-6000
FAX (631) 273-3123