How Many Channels?

Part One: The Effects of FICON on Channel and Device Performance

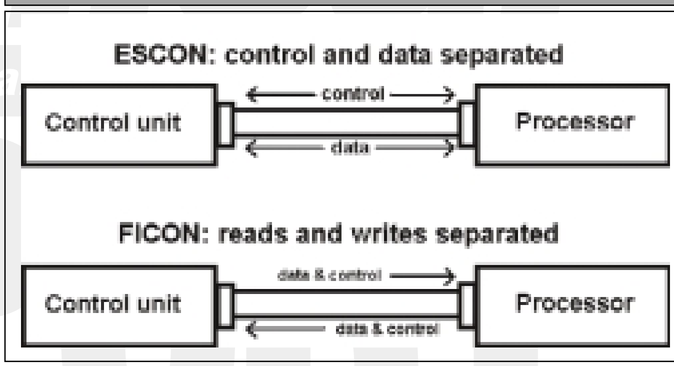
By Tom Aurand

The introduction of FICON into the mainframe arena promises significant relief of bandwidth constraints between processor and local peripherals, as well as boosting performance and distance. This article will compare the properties of FICON and ESCON and explore the general effects on I/O subsystem performance.

THE introduction of FICON into the mainframe arena promises significant relief of bandwidth constraints between processor and local peripherals, as well as boosting performance and distance. However, claims that channel path reductions of 8:1 to 12:1 are possible have been met with skepticism, and many capacity planners and performance managers have put into place ultra-conservative implementation plans of 2:1 reduction, and even straight 1:1 conversion.

FICON is different enough from ESCON that direct comparison is impossible using the usual measurements acquired from RMF. The difficulty in designing a maximum ROI implementation plan is a direct result of the relative complexity of predicting the performance of alternative FICON configurations. However, armed with sufficient knowledge and measurement data, it is possible to design an implementation plan with confidence without over-implementing, and help untie the Gordian's knot under the floor.

FIGURE 1: BI-DIRECTIONAL TRANSFER UNDER FICON SEPARATES READS FROM WRITES



A NEW PROTOCOL

Far from being simply an increase in data transfer rate, FICON differs from ESCON in three important ways.

1. **Increased data transfer rate**
   Yes, the data rate is higher. ESCON data transfer rate is nominally 20 MBPS (megabytes per second), while FICON is 100 MBPS (standard) or 200 MBPS (express).
2. **Bi-directional transfer**
   Although both ESCON and FICON employ two optical fibers in a pair, ESCON transfers reads and writes over the same fiber (the other used only for control information), while FICON transfers reads over one fiber, and writes over another. See FIGURE 1.
3. **Packet-switched protocol**
   ESCON channels connect in a dedicated mode, i.e., once connected to serve a data transfer operation, the channel is not available for any other purpose until the entire data transfer request is satisfied. FICON does not actually "connect," but rather transfers packets of data that can be interleaved from multiple I/O requests.

Each of these differences has important consequences to performance of I/O subsystems. Let us examine each of these in more detail.

1. Increased data rate

The nominal data rate is based on the bit transfer rate established by the cycle speed of the supporting hardware. However, this nominal rate is not the rate at which data is transferred. This is because, besides the data itself, control information must also be transferred to identify data packet boundaries, indicate the destination (port, device address) etc. The resulting *effective data rate* (the rate at which actual data can be moved over the channel path) is lower than the nominal rate. For ESCON, the effective data rate is approximately 17 MBPS.

The effective data rate for FICON *per fiber* is about 75 MBPS, or 150 MBPS for express. The consequences to performance are obvious;

you can move more data over FICON, 4.5 times as much (or nine times as much for express) on a single fiber. To keep things simple, we will refer only to the standard FICON data rates in the remainder of this article, unless explicitly stated as express. To get the numbers for express, simply multiply by two.

2. Bi-directional transfer

Because FICON uses both fibers to transfer data rather than only one, the maximum capacity of a FICON channel is actually twice that indicated by comparing just the data transfer rates. This gives a maximum effective data rate for FICON of 150 MBPS, or 300 MBPS for express! But, to complicate things, remember that one fiber is used strictly for reads, while the other is used to transfer writes. So, to achieve the maximum data rate, the read-to-write ratio would have to be 1:1. Since this is seldom exactly the case, the net effective data rate for the channel in practice will be somewhere between 75 and 150 MBPS (standard FICON). Because of the segregation of reads and writes, FICON channels have two path busy measurements; a read path busy, and a write path busy. Consequently, reads can experience different delays from writes, on the same channel path.

3. Packet-switched protocol

With ESCON, once the channel has begun data transfer for an I/O request ("connected"), it remains dedicated to this single request until all data for the I/O has been transferred. Since ESCON channels have a higher data rate than the physical storage device (17 MBPS as compared with roughly 4 to 5 MBPS for the device) the channel is "busy," but not always moving data if the data required is not immediately available in cache. This can cause a reduction in data rate for the channel, by the introduction of this "dead" time while the channel is connected but waiting for data to become available. This effect is more pronounced when the cache hit ratio for reads is low. Channel dead time exists under ESCON, although contemporary subsystems always transfer data from cache; i.e. the data is moved first to cache, then transferred to the channel.

FICON eliminates channel dead time through the packet-switched protocol. Under FICON, data packets for different I/O requests can be interleaved together (but reads are not interleaved with writes because they utilize different fibers). The channel is no longer dedicated to a single request for the duration of connect time. So, under FICON, if you add up the connect time for all devices, you are likely to find a larger number than the total of all channel busy time. This is because the connect time as measured by RMF includes all time *from the start of transfer of the first data block to the end of the last data block*, so connect time for a particular request can encompass data transfer time associated with interleaved packets from other I/O requests.

FICON and Device Response Time

What does all this mean? At first thought, it may seem very clear what effect FICON has on device response time; it just reduces connect time by the difference in data rate, right? Unfortunately for capacity planners and performance managers, this is not even close to the actual effect. To understand how FICON influences the measurements of response time components, it is necessary to examine each component carefully in light of the new FICON protocol.

FICON and connect time

As we have seen, RMF measures connect time under FICON the same way it does under ESCON; connect time starts with the start of the first data packet transfer and ends with the end of the transfer of the last data packet. Under ESCON, this time includes the actual transfer time (bytes transferred divided by data rate) and "dead" time, while waiting for data from the device.

Under FICON, RMF connect time includes transfer time, but the dead time is eliminated. However, transfer time for interleaved packets, also referred to as *frames*, is introduced and is called *frame-pacing delay*. The frame pacing delay increases as path busy increases, just like wait for CPU increases as CPU busy increases. So, it is actually possible to have *higher* connect time under FICON than ESCON to transfer the same number of bytes, if the paths are busy enough. In fact, the connect time under FICON cannot be less than the rate-adjusted ESCON connect time for the same I/O request under the same conditions, but it can be (and usually is) more, perhaps much more, if the paths are very busy.

FICON and pend time

Pend time under both ESCON and FICON includes the time waiting for the initial channel connect to transfer the first CCW information to the control unit. Also included is the actual time to transfer the CCW, which under FICON is subject to frame pacing delay. Although there is a small gain under FICON associated with the reduction in transfer time of the CCW information, the primary difference in pend time is associated with the change in path busy (and the separation of read/write) under FICON. So, if you reduce the number of FICON paths to the point that the FICON write path busy is higher than the ESCON path busy was, it is possible to experience higher pend times under FICON as a result.

FICON and disconnect time

Disconnect time under FICON is influenced by the same factors as pend time. Of course, there is a component of disconnect time associated with device busy, device positioning and transfer of data to cache that is completely device dependent, and will be no different under FICON. This device-dependent component is usually the portion of disconnect of most concern, since it can be quite large and a pesky performance problem under certain conditions. However, there is another component of disconnect time associated with channel reconnect that is subject to delays due to path busy under FICON. So, once again, it is possible to experience increased disconnect time under FICON when the number of paths is reduced.

FICON and IOS queue time

IOS queue time is a result of waiting for another request to complete against a particular device. When a device is already active processing a request, other I/O requests to the same device from the same system are queued on the processor until I/O that arrived previously have been processed. This means that, the longer the service time, the more IOS queue time, and shorter IOS queue time for shorter service time. Therefore, changes to IOS queue time under FICON are directly related to the net change in pend, connect, and disconnect time for the device. As is the case with these other response time components, it is again possible for IOS queue time to increase with FICON, if the channel paths are reduced enough.

NOT JUST PATH BUSY

Armed with the general understanding of how FICON influences each of the components of device response time, one might conclude that a straight 1:1 swap-out of ESCON to FICON will improve performance. This is true. A conversion without reduction of numbers of paths will result in dramatic reduction in channel path utilization, with corresponding reduction in all components of response time.

One might also conclude that reducing the number of paths will not hurt response times, as long as the channel path utilization does not exceed the original ESCON percent busy. This is the origin of claims of reduction of channel paths of up to 12:1, since the transfer rate is five times faster (5:1 reduction?) and there are two fibers (10:1?), and the ESCON channel dead time is eliminated. But this is a gross oversimplification. The reason is that elongation of response time is not due only to path busy, but also depends on the number of paths available. The problem is complicated also by the fact that under FICON we have *two* channel path busy numbers; one for reads, one for writes.

CONCLUSION

In part two of this article in the next issue, we will present a method for predicting the performance of a FICON configuration based on actual performance statistics measured under ESCON. We will show how to calculate expected FICON path busy, and borrow from Queuing Theory (but avoid the complex math) to build an optimum FICON configuration.

NaSPA member Mr. Aurand is founder and President of Advanced Technological Research, Inc., a software and services company marketing product line Lexonix Technologies, specializing in batch systems tuning and mainframe performance. Mr. Aurand has over 25 years of experience in all aspects of mainframe application and systems management, from application development and programming, to operating system support, capacity planning, and hardware management. Mr. Aurand has an extensive mathematical and statistical background and has developed several modeling systems for statistical analysis of performance of mainframe CPU and peripheral devices.