Basic Synchronous Connections

September 15, 1995

The continuing demand for greater throughput is an ever-present theme in data communication. This demand is fueled by more powerful computing systems--in the hands of a greater number of people--trying to push more data around in the same amount of time. One consequence of the demand for greater throughput is a migration from asynchronous (async) to synchronous(sync) communications: typically over one of the many digital telco services (DDS, ISDN, etc.).

This article will attempt to remove some of the mystery from sync communications. First it will show how a basic sync connection is made using a pair of CSU/DSUs and a DDS link. Then it will add a sync tailcircuit to the basic layout using short range modems. The basic layout is very simple: A sync host computer at the local site is connected to a Patton Model 2500 CSU/DSU, which is then plugged into a 56Kbps DDS telco circuit. At the remote site, a sync workstation is connected to a Patton Model 2500 CSU/DSU, and then to a 56Kbps DDS telco circuit.

Timing is Everything

In the async realm, this simple connection would be a no-brainer. Just use straight through cabling from DTE to DCE on either end, make sure the telco lines are pinned correctly, then plug in and run your diagnostics. The only signals you need to think about are transmit data (TD) and receive data (RD).

Why is async so easy?

Mostly because the timing, which coordinates the communication process, is short term. The timing begins at the front of each start bit, and typically lasts only 10 bits beforere cycling. In async, the host doesn't know (or care) about the timing process in the workstation or in either CSU/DSU. As long as each piece of hardware is set up the same (start & stop bits, word length, data rate)communication flows smoothly from end-to-end.

Moving from the async realm to the sync realm, timing goes from short term to long term. Instead of the timing being reset with every data byte, the timing must now be synchronized between all the hardware components through the use of a master clock (the master clock may reside in the telco network, or in one of the individual components). Along with synchronization comes the demand for greater timing accuracy. Accuracy of 96-98% is allowable in async communication. In sync communication, timing accuracy must be 99.99% or better. However, the benefit of synchronization-which often justifies the effort and expense-is increased data throughput.

Each sync hardware component must be acutely "aware" of the timing of the others. To achieve this awareness, two signals are added in sync communication that are not present in async. These are transmit clock (TC)and receive clock (RC). Proper implementation of these two signals insures that all the hardware components are marching to the same drummer.

How the Clock Ticks

Let's focus on the local host (DTE) and local CSU/DSU (DCE) to see how synctiming works. As you might imagine, TD is controlled by the TC signal. The TC signal originates in the master clock of the telco's DDS network.(All other hardware clocks set themselves by the master clock.) The CSU/DSU receives TC and passes it to the host. At the moment the host sees TC go from low to high, it sends the next data bit to the CSU/DSU on TD. The CSU/DSU then sends that signal down the DDS line. On the remote end, the CSU/DSU receives the signal from the DDS line. From this signal, it extracts the data and recovers the timing. Then the remote CSU/DSU delivers both to the remote workstation. The received data goes to the computer on RD in coordination with the RC signal. At the moment the RC signal goes from high to low, the computer samples (reads) the data bit.

Of course, the actual picture is far more complicated, because both the local host and the remote workstation are transmitting and receiving simultaneously. But the main point to remember in sync communication are that all hardware components pass timing signals along with the data-timing signals that are synchronized by a single master clock. This end-to-end synchronization enhances data throughput. In the section following, we'll add asynchronous tail circuit to the basic layout and see how the system copes with it.

Above we showed how to connect alocal sync host to a sync workstation at a remote site. The communicationlink was made with two CSU/DSUs (Patton Model 2500s) on a 56kbps DDS line.In this section the user at the remote site needs to move the sync workstation toanother office about 500 feet away on a different floor. Since the remoteCSU/DSU cannot be moved out of the phone closet-and 500 feet exceeds the distance specification of RS-232--we need to extend the connection between the CSU/DSU and the workstation. This requires the construction of asynchronous "tail circuit".

What is a Tail Circuit?
A synchronous tail circuit is an extension on a preexisting sync communications link that must synchronize its clocks (transmit and receive)with the preexisting link. Our tail circuit will employ a pair of Patton Model 1080A short range modems, which support a sync data rate of 56 kbps and allow multiple clock sources. Short RS-232 cables will connect the Model 1080As to the Model 2500 and to the workstation. A four-wire unshielded twisted pair (UTP) cable will connect the Model 1080As to each other, completing the tail circuit. The construction of our tail circuit involves three links: The CSU/DSU to 1080A link, the 1080A to 1080A link,and the remote 1080A to sync workstation link. The 1080A to 1080A link isa standard 4-wire UTP (unshielded twisted pair) connection. The remainder of the article will focus on connecting up the outer two links correctly.

The Timing of the Tail
The simpler of the two outer links is between the remote Model 1080A (DCE) and the sync workstation (DTE). This uses a straight through RS-232 cable. The first section we learned that one critical element which separates a DDS sync communication from async is the need for the entire synchronous link to follow a single clock source. The original point-to-point connection used the DDS network clock as its timing source. Adding the sync tail circuit is merely an extension of the overall link, so the tail circuit must follow the same clock.

How do we take the master clock from the DDS circuit and supply it to the tail?
After the CSU/DSU recovers the clock from the DDS line, we apply it directly to

the External Transmit Clock (XTC) on the local 1080A. The XTC signal is used by the 1080A to synchronize data transmission to the other 1080A (remote)over the UTP. Now look at the remote 1080A connected to the workstation. By setting this 1080A to Receive Recover Clock, the clock is recovered from the RCX signal on the UTP and is used to control the transmission of data from the workstation back toward the CSU/DSU and DDS circuit. This keeps all data on the tail circuit, both received and transmitted, synchronized to one clock--the DDS system clock.

Making a Sync Null modem

The CSU/DSU and the local 1080A are both DCE's, and we know that two DCE's cannot be connected directly together. So we must make a null modem to interconnect them properly-a sync null modem. The rules for wiring the RS-232 signals together apply to async and sync null modems:

1)Never connect two outputs together.
2) An input must be connected to an output.

First, we connect the data signals (TD, RD) and control signals(RTS,CD,DSR, & DTR) together properly. All that remains are the timing signals (TC, RC, & XTC, if necessary). From our earlier discussion, we know to take the RC signal of the CSU/DSU and connect it to XTC on the 1080A. (You must configure the clock source on the local 1080A for External Transmit Clock.) Consequently the Transmitted Data on the 1080A is synchronized to the XTC signal, and ultimately to the DDS system clock.

Now with the information presented in these 2 sections of "Basic Synchronous Connections" you have the tools to connect two synchronous devices over a dedicated digital link, and to extend one (or both) connections using asynchronous tail circuit. If this two-part series has been helpful to you, please let the Friday Facts Editor hear about it. Or tell him what other technical topics you would like to see covered. The Editor is listening!

PATTON Electronics Co.

7622 Rickenbacker Drive, Gaithersburg, MD 20879 USA
Phone: +301-975-1000 or Fax: +301-869-9293