Power-On Reset and Power Sequencing for the Am79C02/03/031 DSLAC™ Devices

Application Note

The purpose of this application note is to consider the behavior of the Dual Subscriber Line Audio-Processing Circuit (DSLAC™) device on a linecard during power-up. It is necessary to understand the behavior of the DSLAC device during power-up to ensure a successful linecard design. This application note discusses the following topics:

- Power-on reset circuit operation
- Warm start
- Sequence for the application of signals and power rails

The DSLAC Power-On Reset Circuit

All DSLAC devices have an internal circuit for generating a power-on reset when power is first applied to the V_{CC} rail. The Am79C02 device also has an external hardware reset pin, but this is not present on the Am79C03 and the Am79C031 devices.

Power-On Reset Circuit Operation

A simplified circuit diagram for the power-on reset of the DSLAC devices is shown in Figure 1.

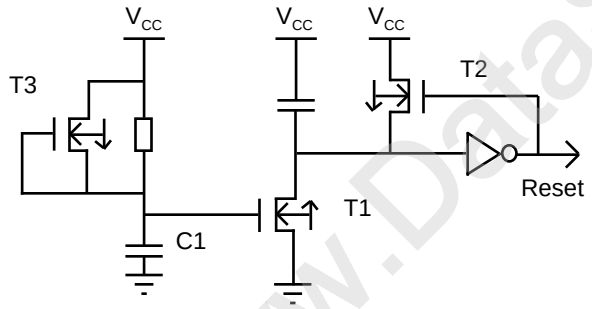


Figure 1. Simplified Power-On Reset Circuit

The reset signal is designed to be Low at initial power-up and become High shortly after V_{CC} reaches its normal operating voltage. Assume the V_{CC} rail has been at 0 V for some time so that capacitors C1 and C2 are completely uncharged and then V_{CC} is applied. It can be seen that initially the gate of the n-channel transistor T1 will still have 0 V applied to it (due to the uncharged state of C1) and will therefore be turned off. C2 is also uncharged, but since one plate of this capacitor is now at +5 V, the input to the inverter will also be at +5 V. The output of the inverter will therefore be Low. The p-channel transistor T2 will be turned on and therefore will tend to pull the input to the inverter High (T2 and the inverter in effect form a latch). As the capacitor C1 charges via the resistance, the transistor T1 starts to turn on, thus tending to pull the level at the input to the inverter Low. However, it is acting against T2, which tends to pull the inverter input High. C1 will continue to charge so that T1 becomes a lower impedance, eventually overpowering T2 so that the input to the inverter is finally pulled Low. The output of the inverter then switches High, turning off transistor T2, which latches the input of the inverter to the Low level and hence its output is latched High.

The power-on reset circuit has been designed to operate correctly when the V_{CC} rail is brought from 0 V to +5 V. Thorough investigation of the behavior of the power-on reset circuit shows that it behaves reliably, even with very slow V_{CC} ramps. However, one limitation that has been found is in situations where the V_{CC} rail does not reach 0 V before power is reapplied to the linecard. In these circumstances, the capacitor C1, which is normally discharged by the transistor T3, does not reach a low enough voltage to turn off transistor T1 so that it releases control of the input of the inverter. As a result, the circuit is not reset so that it is ready to generate another power-on reset to the DSLAC device on the next application of the V_{CC} rail.

Extensive characterization work indicates that the DSLAC device will operate correctly as long as the device does not have power reapplied when the voltage at the V_{CC} pin of the device is in the region 0.05 V to 0.25 V. If the V_{CC} rail does not drop any lower than 0.25 V, then the internal status of the serial microprocessor port registers is retained. In addition, a power-on reset is not required for normal operation of the serial port once power is reapplied, whereas the power-on reset will operate correctly as long as the V_{CC} rail voltage drops below 0.05 V before power is reapplied. As a result, in systems using the DSLAC device where it cannot be guaranteed that the V_{CC} rail to the DSLAC device will drop closer to 0 V than 0.05 V before it is reapplied, there is a small chance that the power-on reset circuit will not

Publication# **19756** Rev: **A** Amendment/**0**
Issue Date: **April 1996**

activate. In such circumstances, it is necessary to understand what action needs to be taken at the design stage to permit reliable, trouble-free operation in the field.

Typical Linecard V_{CC} Rail Power-Down Behavior

The power-down discharge of the decoupling capacitors is rapid at first (5 V to 1 V in less than a ms), but slows considerably once the voltage at the V_{CC} rail drops below 1 V, often taking several seconds to decay to near 0 V. This occurs because initially V_{CC} is near its normal operating range and the semiconductors on the linecard are still on and consuming power, which discharges the power supply quickly. However, once the V_{CC} rail drops to a low voltage (approximately 1 V), the semiconductor devices tend to turn off and therefore draw significantly less current. A typical V_{CC} discharge characteristic for a linecard is shown in Figure 2.

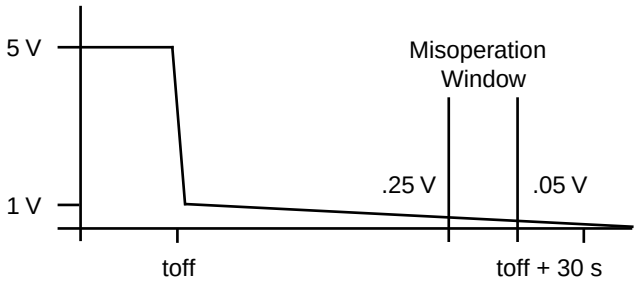


Figure 2. Typical V_{CC} Discharge

DSLAC Device Power-On Reset Behavior in Systems with Slow V_{CC} Ramp-Down

As stated previously, the power-on reset is designed to operate correctly when the V_{CC} rail rises from 0 V to +5 V. In systems with slow ramp-down of the V_{CC} rail, it is possible that the linecard may be powered up when in the region of misoperation, i.e., 0.05 V to 0.25 V. In such circumstances, the power-on reset of the DSLAC device may not operate. This most likely happens if power-up occurs after the decoupling capacitors have been allowed to discharge for several seconds during power-off, but not for so long that they have been allowed to discharge completely. This results in a window, called a misoperation window, often of several seconds if the last part of the ramp-down of V_{CC} is very slow. During this window, a power-up may not cause a power-on reset to be generated in the DSLAC device. This misoperation window is indicated in Figure 2.

Symptoms

If the DSLAC device fails to generate an internal power-on reset, the registers and counters inside the DSLAC device will power up in an undefined state. This will persist until either a genuine power-on reset occurs or a software reset is correctly accepted by the device. However, the serial microprocessor interface of the device also requires a hardware reset, and the device will be unable to accept a software reset until any problems with this interface have been cleared. The method used to clear the serial microprocessor is described below.

In normal operation, when the DSLAC device receives a single byte of data, it simply acts on this as a command. If the DSLAC device receives a command on the serial microprocessor port that is to be followed by multiple bytes of data, then it does nothing with this command other than generate a start address and a stop address related to the filter coefficient RAM (CRAM). The start address is read into a 7-bit counter and acts as a pointer into the CRAM where the data is to be loaded. This counter is incremented with each byte that is sent to the device to point to a new CRAM location until all the necessary data bytes have been written to or read from the CRAM. At this point, the contents of the counter become equal to the stop address (used to set a 7-bit comparator), and a new command may be sent to the device. The DSP engine simply takes its data from the appropriate area of the CRAM for each filter.

If the DSLAC device is powered down and then power is restored without generating an internal power-on reset, the internal registers associated with the serial microprocessor interface may be at two random 7-bit values. It will then be necessary to continue sending bytes to the device until the start address counter has reached the same value as the stop address comparator. In the worst case, the two values could be a full 7 bits apart so that it becomes necessary to send 128 bytes to the serial microprocessor port to clear the problem. The start address counter wraps around to the beginning once an overflow condition has occurred in this counter, so there is no danger of this locking up completely due to it starting with a value that is higher than the value contained in the stop address comparator.

Solutions

1. Using the Am79C02 device:

   The Am79C02 device has an external hardware reset pin, as well as an internal power-on reset circuit. This hardware reset can be operated to clear the DSLAC device's internal registers. This is the simplest solution and requires no further steps.

2. Modifying the V_{CC} rail discharge characteristic:

   The easiest method is to place a resistor between V_{CC} and digital ground. The resistor can usually be a low-power type of a fairly high value and still modify the discharge characteristic significantly, to the point where the window in which the problem may occur is very small and occurs very shortly after power-down. Using this method, it is possible to introduce a single resistor onto the linecard to reduce the discharge time constant of the decoupling capacitors to a few tens of milliseconds with no significant affect on the power dissipation of the linecard. Some line-

card designs may have a resistive device between V_{CC} and digital ground that will be sufficient to ensure rapid discharge of the V_{CC} rail.

3. Software initialization procedure:

   A software work-around has been developed to permit the serial microprocessor port on the Am79C03x device to be cleared and made ready to accept further commands, even if no power-on reset has been generated internally. This work-around requires serial bytes be sent to the Am79C03x device (which will be interpreted as data) until the start address counter is incremented to be equal to the stop address comparator, after which the Am79C03x device will accept the next byte as a command.

   Under worst-case conditions, it will be necessary to send up to 128 single-byte commands to the DSLAC device, followed by an MCLK select command and a software reset (these last two commands correspond to the initialization sequence given in the Power-up Sequence section on page 2-94 of the AMD Linecard Products for the Public Infrastructure Market Data Book 1995). The DSLAC device will then be in a position to correctly accept further programming commands, and will be in the inactive state with its C4–C1 pins set as inputs and the PCM output pins set to high impedance.

   Several possible scenarios should be considered:

   - **Optimized algorithm giving shortest recovery time.** If the Am79C03x device has not generated an internal power-on reset, then this can be detected by reading the revision code. If the revision code is not returned correctly, then assume that the device has not generated a power-on reset and take the corrective action as described above. If the device does return the correct revision code, then an MCLK select and software reset should be sent to the device and to one other register, written, and then read back to check that its value has been received correctly. If this is not successful, then again assume that the part has not generated a power-on reset and take corrective action. If the data can be read back correctly, then further programming of the device can continue.

     If corrective action is required, then it is possible to send a number of bytes to the Am79C03x device and check that it has recovered as described in the previous paragraph. This will reduce the time required to recover the device because in most cases it will not be necessary to send a full 128 bytes. The block diagram on the next page suggests a possible implementation of such an algorithm.

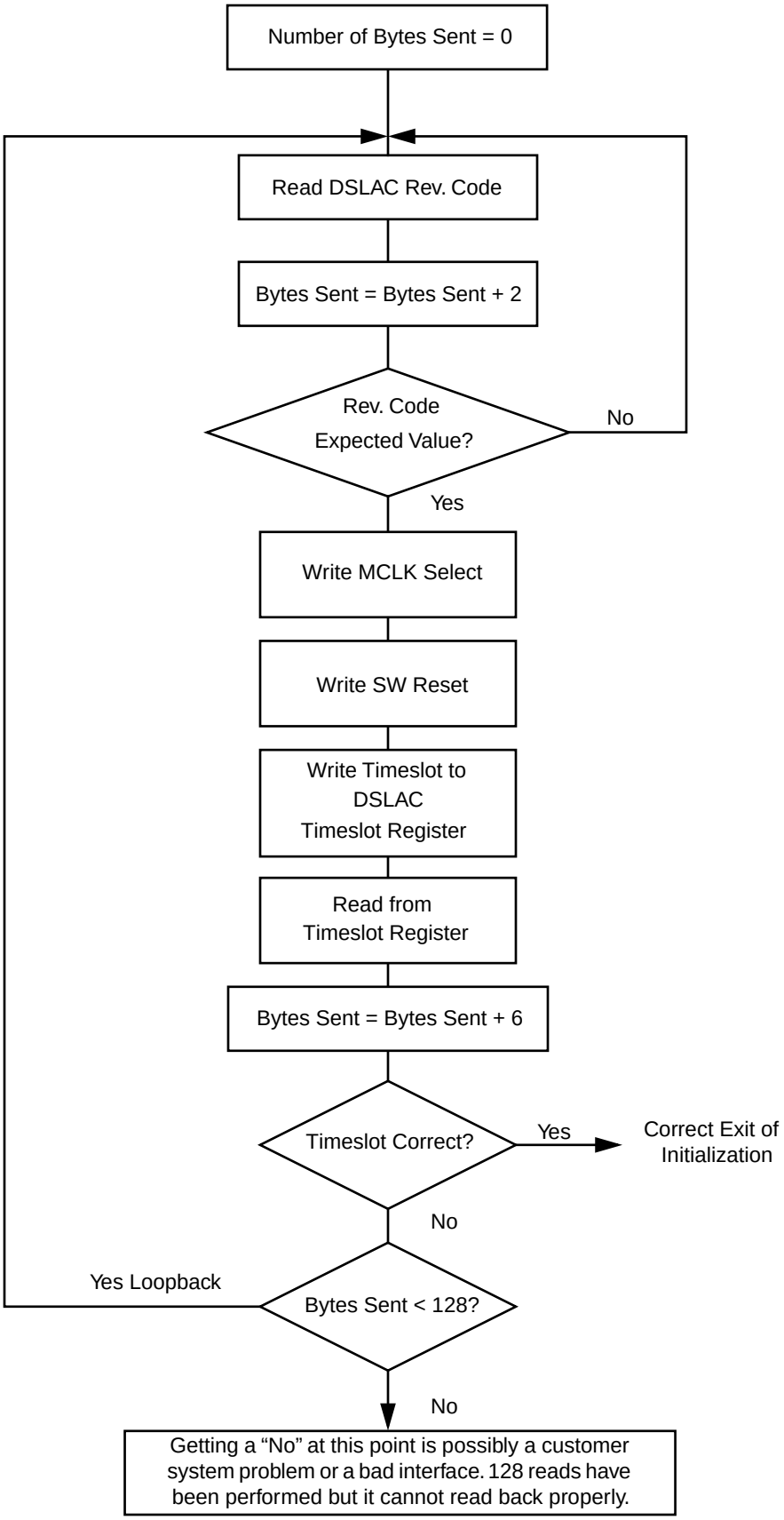
   - **Avoiding pin conflict in systems with read capability.** When writing serial data to the microprocessor interface port of the Am79C03x device, it is possible to cause a pin contention issue because the DSLAC device will be in an undefined state and may be attempting to write data to the pin. The DSLAC device is protected from damage in such a case, but the driver may not be. By reading data from the DSLAC device, the chip select will be toggled, and therefore the CRAM counter in the device will be incremented as required to clear the serial interface problem. However, there is no danger of an output conflict arising. Since it will not be possible to issue any commands to check for early recovery of the serial microprocessor port, it will be necessary to issue a full 128 read commands before issuing any commands.

     Because the DSLAC device may recover at any time and expect to receive commands during the serial port recovery sequence, it will be necessary to ensure that there is a passive pull-up or pull-down on the DIO pin so that either FF_H (unused code) or 00_H (inactivate command) will be received by the device.

   - **Systems with DSLAC device write-only capability.** If the system has no ability to read the coefficient registers of the DSLAC device, then it will be necessary to assume a worst-case situation in which the Am79C03x device always requires a procedure to remove the problem with the serial interface port. In this case, it always requires a full 128 bytes to be sent to the device. In such cases, take into account that the DSLAC device may be attempting to output data onto its DIO pin during the time that bytes are being written to it, resulting in a contention issue. While the DSLAC device's DIO pin is protected from damage in such instances, the device that is driving the pin may not be. Note that with device drivers using passive pull-ups, it will be better to write FF_H to the Am79C03x device. This will present only a high value resistance to the DSLAC device in the event that the Am79C03x device is driving the DIO pin at the same time.

4. Advantages of being able to read data from the DSLAC device's registers:

   The impact of the failure of the DSLAC device's power-on reset can be greatly reduced if it is possible to read the registers in the DSLAC device. This enables the software initialization sequence to be enacted only when necessary and only for as long as required to get the serial microprocessor to communicate correctly. Being able to read the filter coefficients also has other benefits. For example, it is possible to perform some checks that the DSLAC device is responding correctly, and it is also possible to read the device revision code to ensure that the system software matches the device being used.


Number of Bytes Sent = 0
Read DSLAC Rev. Code
Bytes Sent = Bytes Sent + 2
Rev. Code Expected Value?
No
Yes
Write MCLK Select
Write SW Reset
Write Timeslot to DSLAC Timeslot Register
Read from Timeslot Register
Bytes Sent = Bytes Sent + 6
Timeslot Correct?
Yes
Correct Exit of Initialization
No
Yes Loopback
Bytes Sent < 128?
No
Getting a “No” at this point is possibly a customer system problem or a bad interface. 128 reads have been performed but it cannot read back properly.

Additional Information

DCLK Frequency

The maximum permissible DCLK frequency depends on the frequency of the internal clock that runs the DSP engine of the DSLAC device. This internal clock is derived from an internal PLL, which normally generates a clock frequency of 8.192 MHz from the MCLK signal. The maximum DCLK rate is half the rate of the PLL output. If the internal PLL does not receive a power-on or hardware reset but the MCLK signal is present, then the internal PLL may run not at 8.192 MHz but at the MCLK frequency. In this case, the DCLK frequency must not run at a rate faster than half the rate of the MCLK frequency or commands may not be received reliably. DCLK can be asynchronous to MCLK, and thus the phase relationship between these two signals is unimportant.

Operating Power-On Reset in Current Am79C03x Designs

Problems relating to the failure of the power-on reset to operate for the Am79C03x device in some circumstances have only been found during the development stage. It is not expected that customers will experience problems with linecards using the Am79C03x device in the field if similar problems have not been discovered at an earlier stage in a laboratory environment.

In many customer systems, every register in the Am79C03x device will be completely reprogrammed a number of times at regular intervals once power has been applied to the system. In systems where this is not done, it is normal to read the coefficients back from the DSLAC device and resend them if an error is found. In these circumstances, any possible problem with the power-on reset of the device will automatically be cleared at an early stage once sufficient commands have been sent to the device by this procedure.

In the field, powering down the telecommunications equipment using the Am79C03x device is likely to be a rare event. In the unlikely event that a power-on reset does not occur correctly, it is most easily cured by powering down the equipment and trying again. The problem is only likely to occur in systems where the ramp-down of the V_{CC} rail is slow and in a small window where the system is powered down for no more and no less than a few seconds. Failure should only be expected in a minority of power-down/power-up sequences, even when the circumstances under which the problem may occur are present.

Power Interruption Bit (PIB)

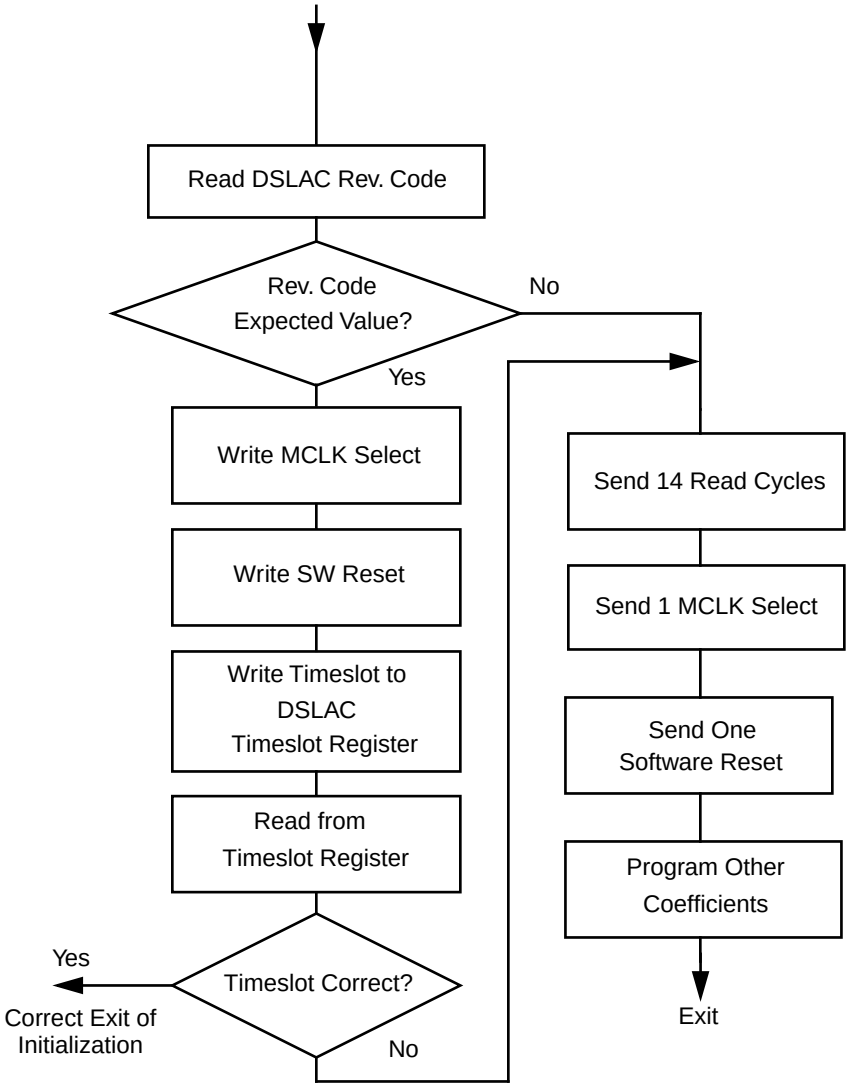
The DSLAC power interruption bit (PIB) can be read by issuing command 55_H. In normal circumstances, it gives an indication that a power interruption has occurred. This bit is set by the occurrence of a power-on reset. If such a reset does not occur when power is applied to the device (for the reasons described above), the PIB will not be set even though the device had been powered-down at some point. This is of little consequence since the inability to communicate with the DSLAC device will prevent reading of the PIB until a recovery procedure has been completed. If the recovery sequence is successful, it can be inferred that a short power interruption had occurred.

Warm-Start Issues with DSLAC Devices

Systems using the DSLAC device can be considered as having distributed processor architectures. With these systems, it is necessary to ensure that the processors involved are correctly synchronized to each other when communicating data. In the case of the DSLAC device, this means it is necessary to ensure that the control processor is sending the data that the DSLAC device requires at the time that the DSLAC device is expecting to receive it, or vice versa. Problems can arise if this issue is not taken into consideration; they can occur during hot insertion of a linecard, during power-up (when the chip select lines to the DSLAC device may be changing state randomly), or when the control processor is reset (i.e., warm start). In such cases, the DSLAC device may be in a different state than is expected by the control processor, and it will be necessary to clear the problem before continuing.

Consider the worst case where the DSLAC device recently received a write Z-filter coefficients command, but immediately after this, the control processor received a local reset—perhaps due to a watchdog timeout. The control processor would then begin its sequence for initializing the DSLAC devices in the system, but the DSLAC device that was about to read Z-filter data would continue to do so, storing the initialization commands sent to the device as if they were filter data. The DSLAC device might continue behaving incorrectly by, for instance, treating subsequent data bytes sent by the control processor as if they were commands. The DSLAC device would only regain synchronism with the control processor at random, having missed much of the initialization sequence.

In the case of Am79C02 devices, it is possible to detect that the device is no longer responding to commands by performing a write/read cycle and checking the result. It is then possible to synchronize the Am79C02 device to the control processor by simply asserting the hardware reset. In the case of the Am79C03x device, there is no hardware reset and it is necessary to clear any potential problem by including a software module, an example of which is shown on the next page.



Note:

This flowchart represents a situation where the Am79C03x device is expecting to transfer a maximum 14 bytes of filter data. By sending 14 blank reads to the device, this ensures that even in this condition, the DSLAC device will complete its transfer of data and then be ready to accept further commands, at which point it may be reinitialized.

Note that there should be an external pull-up or pull-down resistance present on the DSLAC device's serial microprocessor port pin. This is required so that if it happens that the Am79C03x device is reading bytes into the serial port when synchronization was lost, it will read either FF_H (unused code) or 00_H (inactivate command), not a command to be followed by multiple bytes of data.

In systems where it is only possible to send data to the device, then a single byte command such as software reset should be used. It will, however, be necessary to ensure that the device driving the DSLAC device will not be damaged as a result of the DSLAC device driving this pin at the same time (the DSLAC serial microprocessor port is protected against such damage). It will also be necessary to assume that the DSLAC device needs to be resynchronized before sending any programming data to the serial port.

Note that the software that helps fix the problems experienced with the Am79C03x device's power-on reset will also cure problems related to processor synchronization.

Sequencing Applied Signals to the Am79C03x Devices

As has been noted previously, the DSLAC devices use a phase-locked loop to generate an internal 8.192-MHz clock from the MCLK signal. In the event that a hardware or power-on reset does not occur, the PLL will run at the applied MCLK frequency. If a hardware or power-on reset does occur and MCLK is not present, then the PLL will free run at 2 MHz. Since the maximum DCLK frequency is half the applied PLL frequency, then the maximum DCLK frequency with no MCLK present is 1 MHz—not the value of 4.096 MHz given in the data book. In order to avoid this problem and to permit the use of the DCLK signal at the highest possible speed, the MCLK signal should be present before the DSLAC device is given a hardware reset, either generated by the power-on reset or by an optional external reset in the case of the Am79C02 device.

Ideally, the power-up signal sequence should be as shown on page 2-94 of the AMD Linecard Products for the Public Infrastructure Market Data Book 1995.

It should also be noted that the signals fed to the DSLAC device may be rectified by input protection diodes and this rectified signal may be sufficient to keep the DSLAC device powered up. This has at times given rise to some confusion as a result of the DSLAC device coefficient data remaining present, even though power may have been removed from the V_{CC} rail. Also note that in such circumstances, the DSLAC device will not generate an internal power-on reset once power is restored, but that this is not a problem because each internal register retains its previous operating status and should behave normally when the power is reapplied to V_{CC}.