0-10v:

The first remote lighting consoles had a lot of wires. An individual wire was run from each channel control to each dimmer. Adjusting the fader on the control desk changed the voltage of the signal being sent to the dimmer.

This system was known as **0-10v Analog** control. A 0v signal correlated to 0% dimmer output. 5v and 10v corresponded to 50% and 100%, respectively. The problem with this system is obvious and has already been stated: It took a lot of connections to control the dimmers.

Multiplexing was the solution to this problem. Multiplexing allows numerous signals to be sent at the same time, over a single wire. Instead of each dimmer "listening" to its own control wire, it listens to a stream of control signals which are sent to all of the dimmers.

By knowing its own numeric designation, a dimmer is able to pick its own control value out of the stream.

AMX:

AMX, or **Analog Multiplexing,** is a system whereby control voltages are sent down a cable one at a time. For example:

10v 4v 8.5v 9v 3.2v

A dimmer set to listen to the fourth value in the stream would be set to 90% output.

The above example is a five value **Frame** of data. AMX can handle up to 192 values per frame.

A **Clock Pulse** is also sent – along a different wire – at the same time as the AMX value. Each pulse is like an instruction saying "Read the value now!"

In order for a dimmer to identify the correct value to listen to, it must count the number of clock pulses after a **Reset Pulse**. (A very long pulse on the clock line.)

The above example with a clock pulse; AMX actually uses a 0-5 volt signal. Each voltage is present on the line for 8 microseconds.

Clock:	xxxx	x	x	x	x	x
Data:		5v	2v	4.25v	4.5v	1.6v

The **Starting Channel** of a dimmer rack tells the rack how to assign the dimmer values. A control interpreter within the rack counts the clock pulses, and sends the correct values along to the dimmers it controls. For example, a 2 dimmer rack set to Start Channel 3 would listen to the 3rd and 4th control values.

Problems with AMX:

- Over long runs, the resistance of the control cable affects the signal and reduces the dimmer output.
- 192 channel limitation

DMX

Along came **DMX**, or **Digital Multiplexing**. Instead of analog values being sent down the wire in sequence, digital values (composed of 0s and 1s) are sent. Each value is sent as a single byte of data – that is, a value ranging from 0 to 255.

Unlike AMX, DMX has no clock signal. Instead of counting clock pulses, DMX systems count the number of byes after a start code. A 2 dimmer rack set to start channel 5 would listen to the 5th and 6th bytes of data.

The DMX data stream is more robust. It does not suffer much from signal degradation over long runs. Also, up to **512 bytes** of data can be sent per DMX frame.

Control Cable for DMX is technically supposed to be a 5 pin XLR connector. However, in 95% of applications, only the first three pins are used. This has lead to many manufacturers using a 3 pin XLR connector, commonly found in microphone cables. (The pin correlation is 1 to 1.. that is, pins 1, 2, and 3 in the 5 pin connector correlate to 1, 2, and 3 of the 3 pin.)

Only electronic **Splitter/Optoisolator/Amplifier** modules should be used to split or redistribute DMX signal. These devices read the values in the DMX frame, store it, and re-broadcast it to other devices.

Also, the last device in a DMX system should have a **Terminator,** a 120 ohm resistor connecting pins 2 and 3.

Questions:

1) What is Multiplexing?
2) Why is a clock pulse necessary in AMX? Why is it not necessary in DMX?
3) How might you connect newer DMX systems to older AMX systems?