SDH Pointers

SDH provides payload pointers to permit differences in the phase and frequency of the Virtual Containers (VC-N) with respect to the STM-N frame. Lower-order pointers are also provided to permit phase differences between VC-1/VC-2 and the higher-order VC-3/VC-4.

On a frame-by-frame basis, the payload pointer indicates the offset between the VC payload and the STM-N frame by identifying the location of the first byte of the VC in the payload. In other words, the VC is allowed to "float" within the STM-1 frame capacity.

To make this possible, within each STM-N frame, there's a pointer, known as the VC Payload Pointer, that indicates where the actual payload container starts. For a VC-4 payload, this pointer is located in columns 1 and 4 of the fourth row of the Section Overhead. The bytes H1 and H2 (two 8-bit bytes) of the Overhead can be viewed as one value (see Figure 9).

The pointer value indicates the offset in bytes from the pointer to the first byte of the VC, which is the J1 byte. Because the Section Overhead bytes are not counted, and starting points are at 3-byte increments for a VC-4 payload, the possible range is:

Total STM-1 bytes – Section Overhead bytes = Pointer value range

For example:

(2430 – 81)/3 = 783 valid pointer positions

That is, the value of the pointer has a range of 0 to 782. For example, if the VC-4 Payload Pointer has a value of 0, then the VC-4 begins in the byte adjacent to the H3 byte of the Overhead; if the Payload Pointer has a value of 87, then the VC-4 begins in the byte adjacent to the K2 byte of the Overhead in the next row.

The pointer value, which is a binary number, is carried in bits 7 through 16 of the H1-H2 pointer word. The first four bits of the VC-4 payload pointer make provision for indicating a change in the VC, and thus an arbitrary change in the value of the pointer. These four bits, the N-bits, are known as the New Data Flag. The VC pointer value that accompanies the New Data Flag will indicate the new offset.

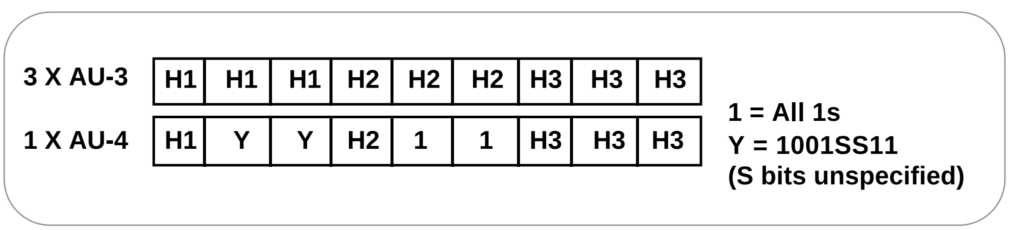
Payload Pointers

When there's a difference in phase or frequency, the pointer value is adjusted. To accomplish this, a process known as byte stuffing is used. In other words, the VC payload pointer indicates where in the container capacity a VC starts, and the byte stuffing process allows dynamic alignment of the VC in case it slips in time.

Positive Pointer Justification

When the data rate of the VC is too slow in relation to the rate of the STM-1 frame, bits 7, 9, 11, 13, and 15 of the pointer word are inverted in one frame, thus allowing 5-bit majority voting at the receiver (these bits are known as the I-bits or Increment bits). Periodically, when the VC is about one byte off, these bits are inverted, indicating that positive stuffing must occur.

An additional byte is stuffed in, allowing the alignment of the container to slip back in time. This is known as positive stuffing, and the stuff byte is made up of non-information bits. The actual positive stuff byte immediately follows the H3 byte (that is, the stuff byte is within the VC portion). The pointer is incremented by one in the next frame, and the subsequent pointers contain the new value.



▶ ***Figure 9.*** *Pointer 9-byte structure.*

Table 9. SDH Pointers

Byte	Description
H1 and H2	**Pointer bytes –** These two bytes, the VC payload pointer, specify the location of the VC frame. It's used to align the VC and STM-1 Section Overheads in an STM-N signal, to perform frequency justification, and to indicate STM-1 concatenation.
H3	**Pointer action byte –** This byte is used for frequency justification. Depending on the pointer value, the byte is used to adjust the fill input buffers. The byte only carries valid information in the event of negative justification, otherwise it's not defined.

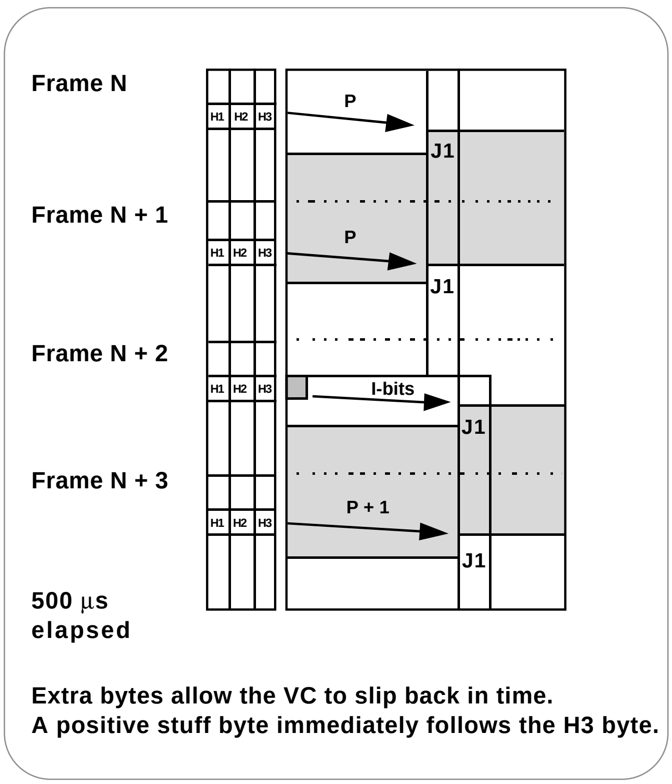
Simply put, if the VC is running more slowly than the STM-1 frame, every now and then “stuffing” an extra byte in the flow gives the VC a one-byte delay (see Figure 10).

Negative Pointer Justification

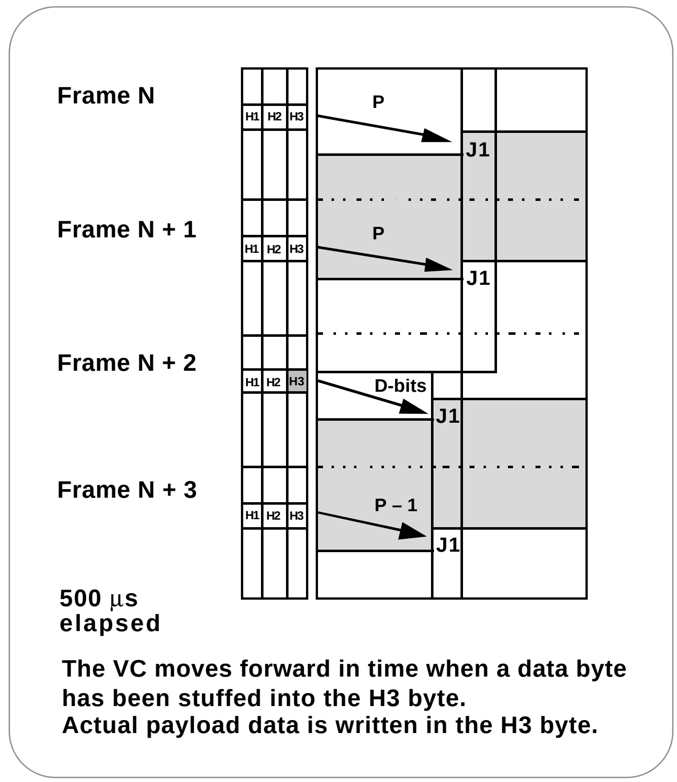
Conversely, when the data rate of the VC is too fast in relation to the rate of the STM-1 frame, bits 8, 10, 12, 14, and 16 of the pointer word are inverted, thus allowing 5-bit majority voting at the receiver (these bits are known as the D-bits, or Decrement bits). Periodically, when the VC is about one byte off, these bits are inverted, indicating that negative stuffing must occur.

Because the alignment of the container advances in time, the payload capacity must be moved forward. Thus, actual data is written in the H3 byte, the negative stuff opportunity within the Overhead; this is known as negative stuffing. The pointer is decremented by one in the next frame, and the subsequent pointers contain the new value. Simply put, if the VC is running more quickly than the STM-1 frame, every now and then pulling an extra byte from the flow and stuffing it into the Overhead capacity (the H3 byte) gives the VC a one-byte advance (see Figure 11).

In both positive or negative cases, there must be at least three frames in which the pointer remains constant before another stuffing operation (and, therefore a pointer value change) can occur.



▶ ***Figure 10.*** *Payload pointer – positive justification.*



▶ ***Figure 11.*** *Payload pointer – positive justification.*