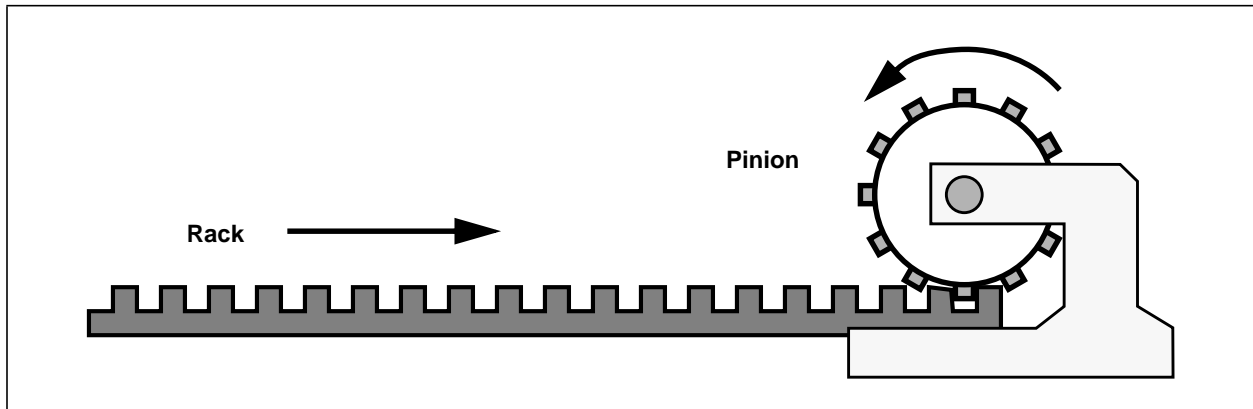


## 11.0 Too Many Zeroes on a T1 Span

Too many zeroes can cause a problem, since a T1 line is essentially inactive during a string of 0s. The problem is manifested by the receiver. To examine this more closely, a simple mechanical analogy of a digital transmission system is used.



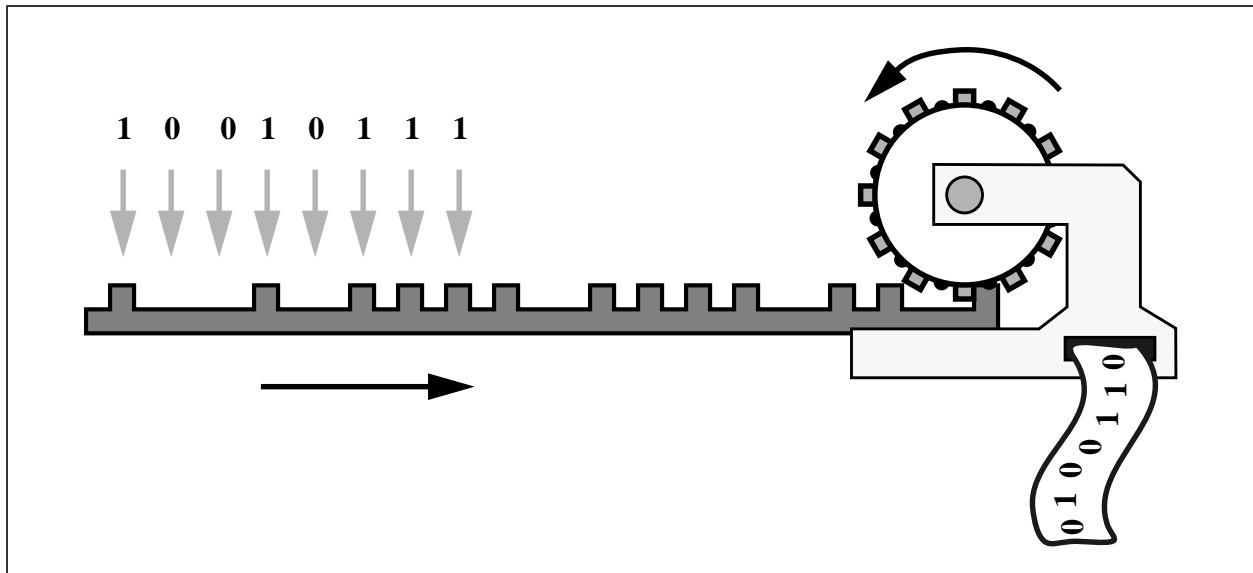
**Figure 40: Rack and Pinion mechanical assembly**

### 11.1 Too Many Zeroes: an Analogy

A rack and pinion (Figure 40) is a mechanical linkage that is sometimes used in cars to steer the front wheels. With some modification, we can design a simple digital communication system with this rack and pinion.

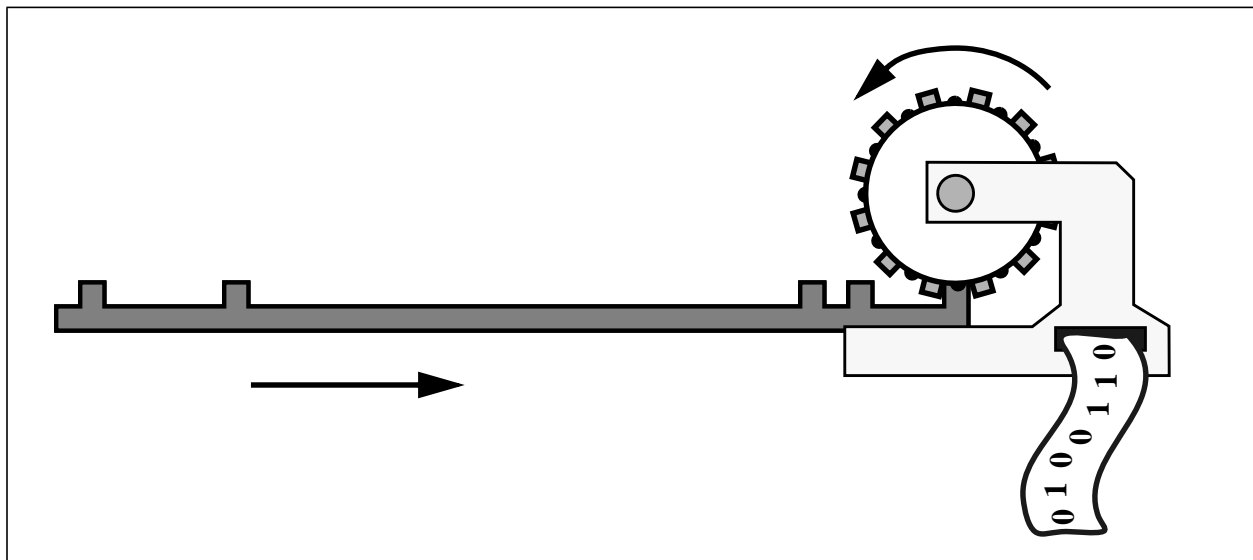
The timing problem caused by too many 0s on a T1 line is similarly manifested in this mechanical analogy.

In Figure 41, indicator switches are added between the cogs of the pinion gear. These sensor bumps are squashed when a tooth on the rack passes underneath the pinion gear. This indicates a 1 has been received from the information encoded into the rack. A tooth missing on the rack will not squash the sensor bump on the pinion, and therefore indicates a 0. By selectively eliminating teeth on the rack, a message can be hard-coded into the rack.



**Figure 41: “Rack and Pinion” Digital Transmission System**

To “transmit” the message, the rack is slid at a constant speed to the right. The pinion gear rotates on its axel and reads the 1s and 0s encoded into the rack’s teeth. The rotational inertial of the pinion keeps it spinning as 0s pass beneath it. The 1s keep the pinion spinning at the correct rate. With the pinion’s rotation synchronized to the rack’s travel, the system works well.

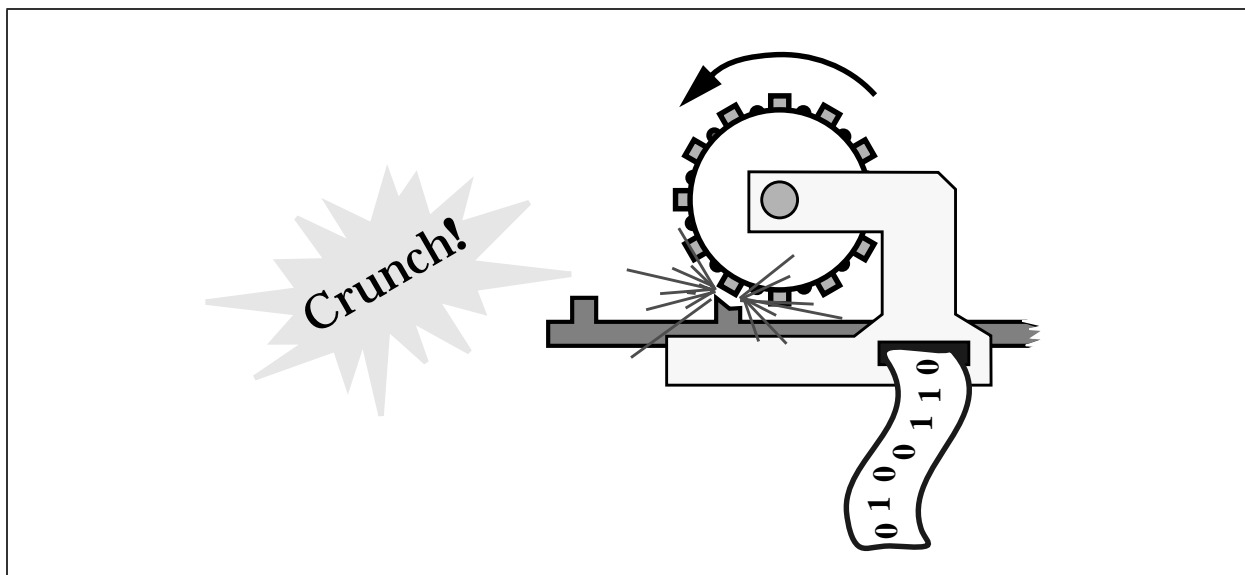


**Figure 42: Pinion loses Synchronization after a long string of Zeroes**

In Figure 42, the transmitter rack is encoded with a long string of 0s. That is, many teeth in a row are missing.

The rack is fed into the pinion at a constant speed. For a while, the pinion rotates and reads the 0s correctly. There is no problem, yet. But a problem is encountered when the pinion finally reaches a 1 at the end of the long string of 0s.

The pinion gradually slows its rate of rotation over the long string of 0s. There are no teeth on the rack to keep the pinion spinning at the proper speed. Pinion cogs gradually shift phase alignment relative to “bit positions” on the rack. As a result, when the pinion finally does encounter a tooth (Figure 43) it is no longer synchronized to the rack. Instead of the rack tooth slipping smoothly between the pinion’s cogs, the rack and cog teeth collide. Not only is the 1 encoded on the rack not sensed, but the system would probably break!

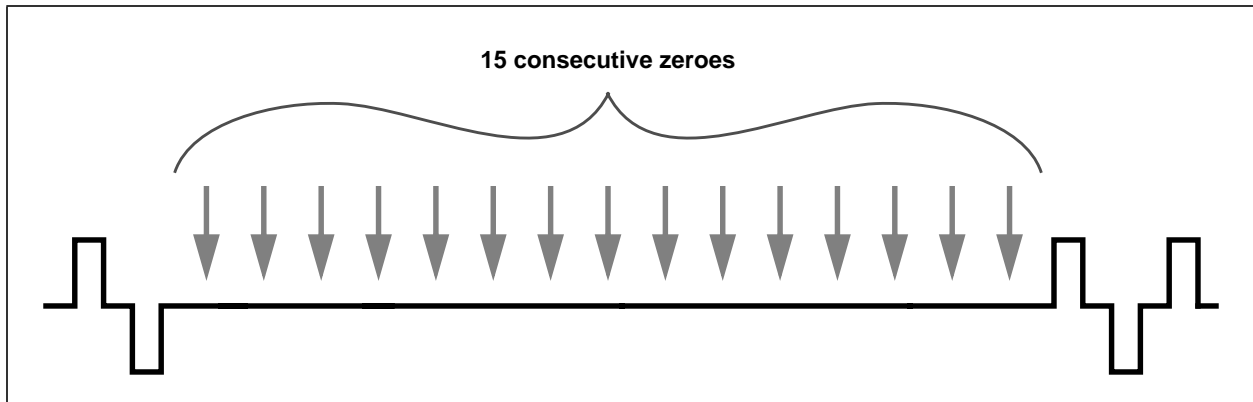


**Figure 43: So much for that Design!**

## 11.2 Zero String Suppression

The mechanical analogy’s problem is similar to what occurs in an actual T1 receiver. A T1 receiver relies upon the received 1s to keep its receiver synchronized to the incoming bit stream.

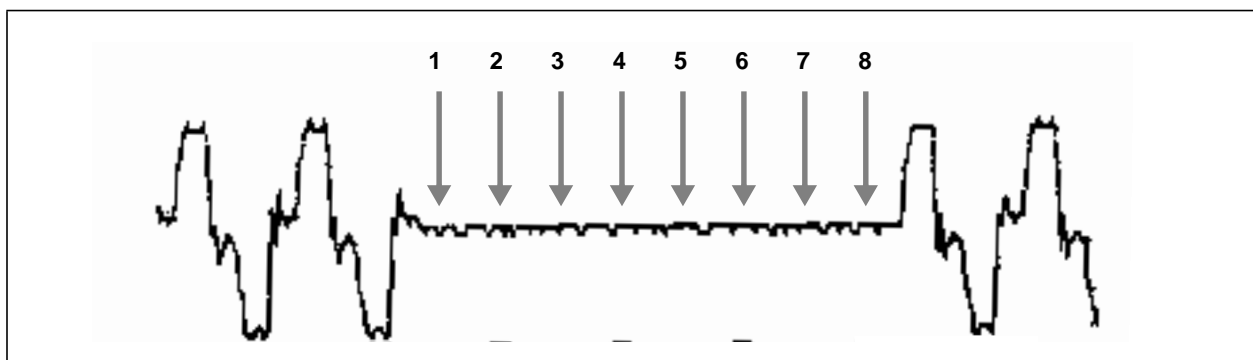
T1 receivers use an internal clock to determine when a bit position occurs on the T1 line. During a bit position the receiver tests the line for a pulse or absence of a pulse. In this way it can read a 1 or 0 transmitted from the far end. A long string of 0s can allow the receiver’s internal clock to get out of synchronization with the ideal bit positions, so when a 1 finally is encountered, the receiver may sample the line for a bit at the wrong moment and create an error. Older-design T1 receivers use ringing tank circuits to provide the internal clock. In these receivers the clock might die altogether if the 1s’ density on the T1 line is too low.



**Figure 44: Maximum Permissible String of Zeroes**

So, it is important to suppress long strings of 0s. T1 network regulations allow only fifteen consecutive 0s and an average 1s' density no lower than 12.5%. There are several zero string suppression techniques designed to force a T1 bit stream to comply with this requirement. These zero-suppression techniques look for eight consecutive 0s about to be transmitted onto the T1 span. If eight consecutive 0s are never allowed to occur on a T1 line, then both the maximum consecutive 0s string and the minimum 1s' density requirements are met.

The need to suppress 0s on a T1 span is why the 8-bit numbers assigned to the PCM words are mostly 1s for low amplitude values, and mostly 0s for high amplitude values (refer to Figure 23). Since human speech is mostly quieter tones with silence between words and sentences, assigning PCM words for low-amplitude values to be mostly 1s helps keep the T1 line overall 1s density high. Likewise, DS0 idle code is typically 0x7F (01111 1111) or 0xFF (1111 1111). Idle code is the PCM word that is repeatedly transmitted in a DS0 channel that is not actively carrying telephone call.



**Figure 45: String of Eight Zeroes**

Figure 45 shows an actual AMI T1 signal as recorded on an oscilloscope, with eight consecutive 0s. There is no zero string suppression technique used in this T1 transmission.

### 11.2.1 Bit 7 Stuffing

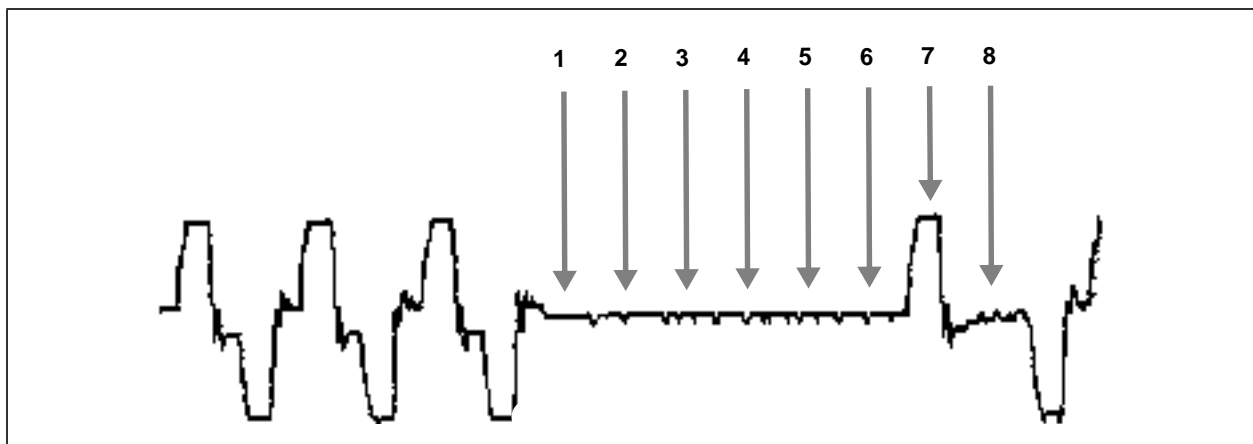
The simplest zero string suppression technique looks for any DS0 channel that contains an all-0s byte (PCM word), and changes bit 7 of the PCM word from a 0 to a 1. This breaks up the string of eight 0s. If this suppression is done for all the DS0 channels, then there is no way the 0s-string requirements can be violated.

Note that this zero string suppression technique alters the DS0 channel's contents. The value of all-0 byte was altered when bit 7 was changed from a 0 to a 1. If the DS0 contains PCM, then this corruption is usually not a problem. A listener cannot typically hear the consequences of bit-7 stuffing during a telephone call. However, if the DS0 channel is transferring some form of information where the exact value of the transmitted word is critical (e.g., the DS0 is transferring a data file, instead of speech from a telephone conversation), then bit-7 stuffing will cause an error in the information when the DS0 channel is recovered at the far end.

Bit 7 is chosen as the bit to be “stuffed” because it is the second least significant bit in the PCM sample. Altering the value of bit 7 does not corrupt the PCM sample as significantly as would altering a more significant bit.

*Why not alter the least significant bit of the PCM sample, bit 8? Altering bit 8 would corrupt the PCM sample even less than altering bit 7.*

This is true, but remember bit 8 is used for signaling in every sixth frame. Bit 7 was chosen for stuffing so it would not interfere with robbed-bit signaling. It is better to corrupt the speech than the signaling bits that control the state of the telephone call. Figure 46 shows bit-7 stuffing in action.



**Figure 46: Bit-7 Stuffing Zero Suppression on a T1 AMI signal**

### 11.2.2 Bipolar with Eight Zero Suppression (B8ZS)

**B8ZS** was developed to resolve the problem which bit 7 stuffing presents: data mutilation. Unlike bit-7 stuffing, B8ZS can be “undone” at the receiving end of the T1 span. B8ZS, like bit-7 stuffing, operates on a string of eight 0s. B8ZS replaces a string of eight 0s with a special code word on the T1 line. The special code word is recognized by the receiver at the far end, which removes the B8ZS word and replaces the original eight 0s. In this way, no DS0 data gets mutilated, even though T1 bits are altered.

The B8ZS word uses two BPVs to distinguish itself. The B8ZS word puts a 1 in the 4<sup>th</sup>, 5<sup>th</sup>, 7<sup>th</sup> and 8<sup>th</sup> bit positions where eight consecutive 0s would have been transmitted. The 4<sup>th</sup> and 7<sup>th</sup> positions create the telltale bipolar violations (Figure 47) pattern. That is, a 0000 0000 pattern is replaced by 000V A0VA, where V is a bipolar violation and A is a 1 that obeys AMI rules.

B8ZS does not limit itself to search for strings of eight 0s only within DS0 channel boundaries, as does bit-7 stuffing. A string of eight 0s anywhere in the DS1 stream can qualify for a B8ZS substitution.

Note, this suppression scheme requires a B8ZS-capable receiver, as well as a B8ZS transmitter, and B8ZS-capable repeaters (repeaters that do not correct BPVs). Bit-7 stuffing only requires a special transmitter. While B8ZS is a more useful zero suppression technique, it is also more expensive than bit-7 stuffing to implement.

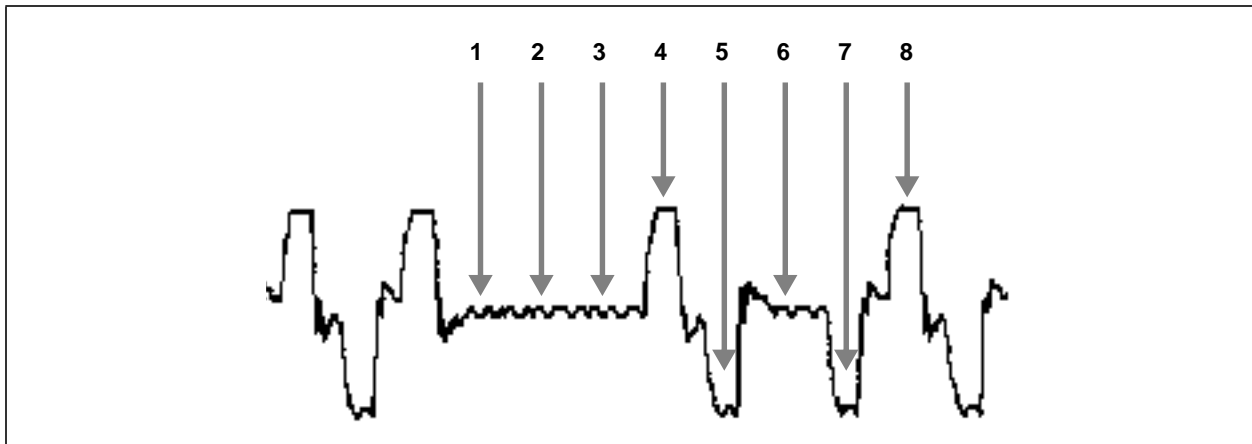


Figure 47: B8ZS

### 11.2.3 Zero-Byte Time Slot Interchange (ZBTSI)

**ZBTSI** is yet another 0-string suppression technique. ZBTSI became available in 1987 as a means to provide clear channel transmission without B8ZS line coding. The attraction for this ESF-based zero-suppression scheme is the clear-channel capability of B8ZS without requiring specialized B8ZS line equipment. Since ZBTSI uses standard AMI coding, T1 service providers need not replace all their AMI-only capable repeaters in a span in order to implement ZBTSI.

ZBTSI is a complex scheme that uses half of the FDL bandwidth to indicate where strings of 0s exist. ZBTSI is a case of increased design complexity intended to enable simplified installation. Unfortunately, ZBTSI is a bit too complicated. It requires storing four extended superframes of PCM in order to generate a ZBTSI frame. This delays the transmission of voice signals. If voice travels over several T1 spans using ZBTSI, each span incurs additional delay due to the ZBTSI process. The accumulated delay can itself become a deterrent to telephone conversations.

The complexity of developing ZBTSI has prevented it from gaining widespread acceptance.

## 12.0 Jitter

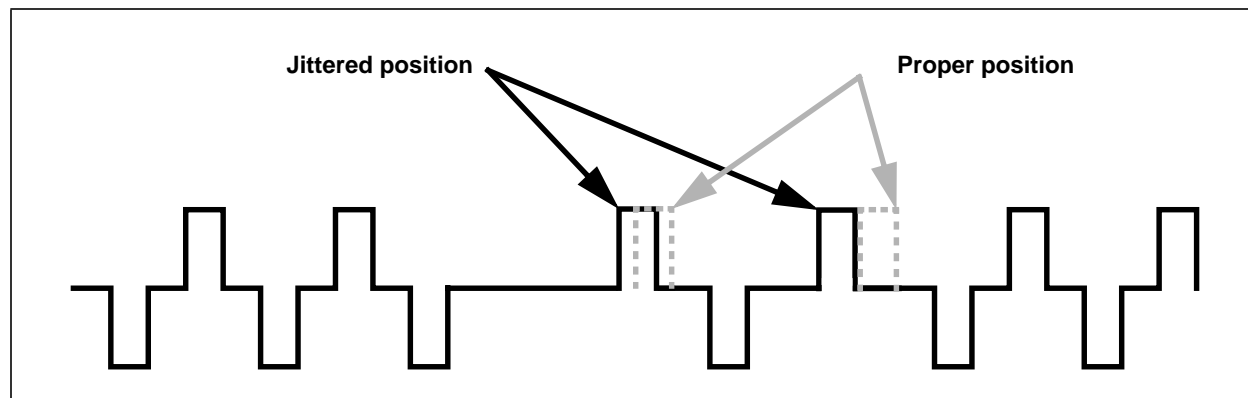
Jitter can be defined as a random shift of a pulse from its intended position.

Jitter is a complex phenomenon, but the simplest explanation is that it is a deviation of a pulse from its ideal position. Jitter, like long strings of zeroes, makes the receiver's job of recovering pulses difficult. If jitter becomes severe enough, the receiver can lose track of incoming pulses. When this happens, errors can occur in the recovered information.

Figure 48 shows two pulses that have deviated from their nominal positions (shown as dashed lines). These shifted pulses constitute jitter.

One pulse has moved completely outside its expected position. The receiver may well miss this pulse, causing an error of omission.

A prime source of jitter is repeaters. Jitter is cumulative, so it gets worse as a T1 signal passes through successive repeaters. Jitter is what limits the length of a T1 span. A T1 signal that gets re-timed, as does the signal passing through the slip buffer of a cross connect shown in Figure 51, gets any jitter removed.

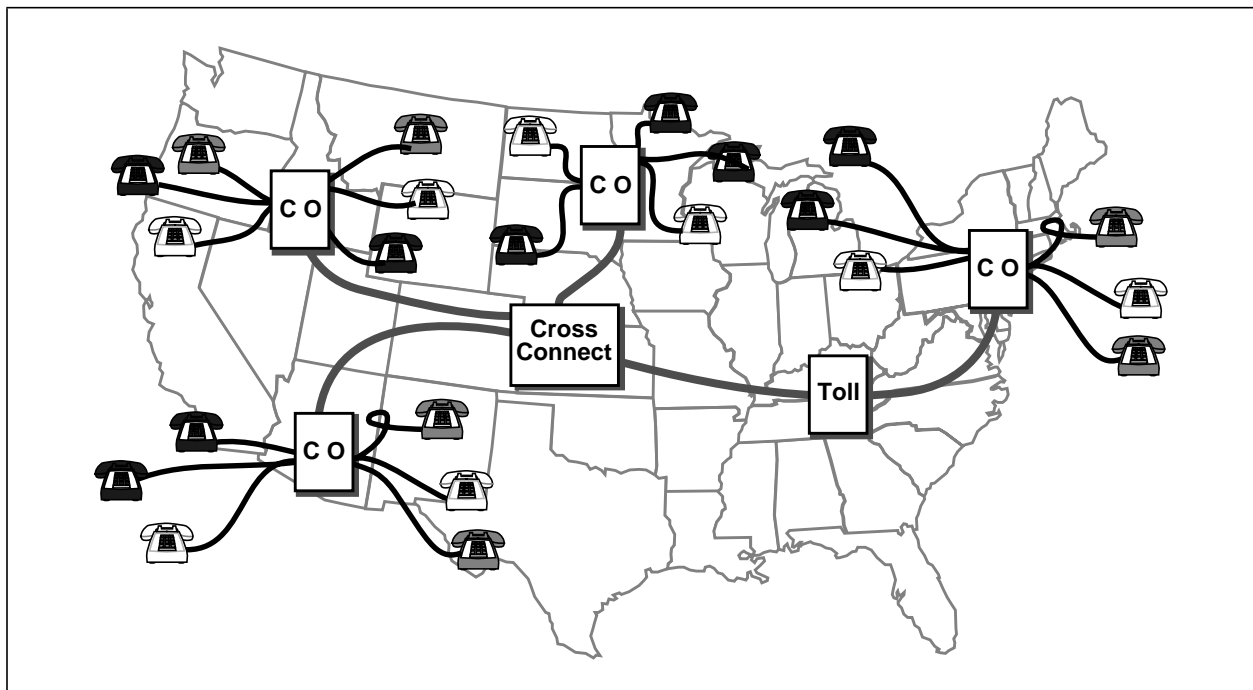


**Figure 48: Pulses out of position due to Jitter**

## 13.0 Slips

DS1 transmission requires that transmitting and receiving of DS1 bits occurs at the same rate (1.544 Mb/s). If transmit and receive rates differ, DS1 frames are either lost or processed twice, depending on whether the transmitter sends the DS1 bits faster or slower than the receiver is processing them.

Figure 51 shows a T1 network that contains a T1 terminating device, a cross connect. T1 spans terminate and originate at a cross connect, and they may arrive at the cross connect with slightly different average bit rates. Despite these varying T1 bit rates, the cross connect strives to keep the T1 network synchronous. It does this by reading the contents of each received T1 bit stream into a buffer (one buffer for each received T1 stream), and timing each T1 stream it transmits to a common 1.544 MHz clock. Timing differences between received T1 streams and the common transmit clock rate can result in controlled timing slips induced in the transmitted T1 streams.

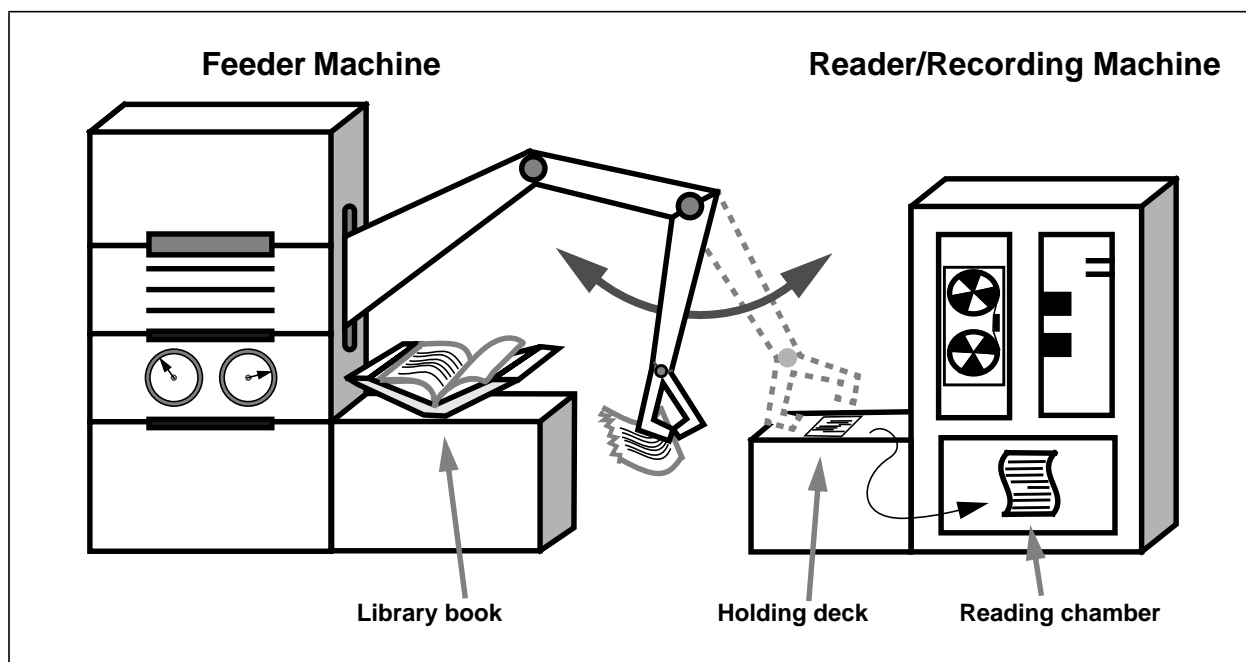


**Figure 49: A T1 Network that requires a Timing Hierarchy**

### 13.1 An Analogy for Timing Slips: The Book Reader/Recorder

To illustrate how slips cause problems in DS1 communications, image a machine that reads pages from a book and uses a computer-generated voice to record the contents of a book on cassette tape. After the machine records an entire book, the cassette tape is duplicated and sold to people who wish to listen to narrated books.

This machine can only read one page at a time. It holds one page on a holding deck in preparation while it is reading another page. The feeder portion of the machine rips pages out of a library book and drops them onto the holding deck. The reader/recorder pulls the page from the holding deck into the reading chamber, then reads and records the page. When the reader/recorder finishes recording a page, it must have a new page on deck, ready to feed into the reading chamber, or it will default to reading the page that is presently in the reading chamber and record it again. As the cassette tape runs continuously, the reader/recorder must always be reading something.



**Figure 50: The Book Reader/Recorder Machine**

If the page ripper arm runs too slow, it will not keep up with the demand of the reader/recorder. Eventually a point in time will be reached where the page reader requires a new page to read and the feeder will not yet have dropped that page onto the holding deck. This forces the reader/recorder to record the page in the chamber over again. This is called an **underflow** or **repeat slip**. The finished cassette tape is now flawed. It contains a page that is read twice.

If the page feeder is running too fast, it will be dropping pages onto the holding deck faster than the reader can process the pages in the reading chamber. Eventually the two machines will get far enough out of synchronization that the feeder will attempt to place a page onto the holding deck while the previous page is still on deck. The page ripper arm releases the new page, but it slides off the holding deck since the holding deck can only handle one sheet of paper. The page falls on the floor and never gets read by the reader. This is called an

**overflow** or **skip** slip. The holding deck has overflowed, and a page is skipped. The finished cassette tape is again flawed. It now has one page of the story omitted.

Clearly, slips are undesirable in this device, as they cause information to be repeated or lost in the recording of the book. Similarly, slips in a T1 network can cause PCM samples to be repeated or lost in a telephone conversation.

### 13.2 Slips in a T1 Network

Figure 51 shows a DS1 terminating device (the cross connect from Figure 51, for example) that requires synchronous timing to operate slip-free. DS1 bits are written into the frame buffer at the rate the incoming T1 line delivers them. Bits are clocked out from the frame buffer through the T1 transmitter at a rate dictated by the Transmit Clock. The ideal clocking arrangement for slip-free operation is when the Transmit Clock runs at exactly the same rate as the DS1 bits being written into the frame buffer. For every bit written into the buffer, one is read out. The write pointer lays down PCM information and then the read pointer comes along, reads the information from memory, and transmits it. No slips occur.

If the Transmit Clock runs slower than the average rate that the incoming bits are received, the DS1 terminal device will produce an overflow slip. A frame of DS1 information (possibly a PCM sample for each of 24 telephone conversations) will be lost.

If the Transmit Clock runs faster than the average rate at which incoming T1 bits are received, then the DS1 terminal will suffer a repeat, or underflow, skip. A DS1 frame of information will be repeated.

The frame buffer allows slips to be graceful by repeating or deleting an entire frame of information (thus allowing framing bits to be transmitted in their proper position to the next T1 terminating device downstream) instead of letting a slip repeat or skip only a portion of a frame (this would cause framing bits to be transmitted in a new relative position, and a downstream T1 terminating device would have to find the new position of the framing bits and re-synchronize itself to the framing pattern).

The frame buffer is a memory device that has sufficient capacity to store several frames of information. The buffer's operation is cyclical. When either the read or write pointer (an illustrative concept: the pointer "points" to the memory location where a read or write operation occurs) reaches the top of the memory, it jumps back down to the bottom of the memory and continues its cyclic operation. There are two independent cycles occurring within the memory; the write pointer cycle and the read pointer cycle.

If the incoming bit rate is equal to the Transmit Clock's rate, the two pointers chase each other around the memory stack, but one never catches the other. Every bit that is received gets transmitted. This is the desired mode of operation. The input and output T1 streams have synchronous bit rates and no slips occur.

A repeat slip occurs if the read pointer catches up with the write pointer (incoming bit rate is slower than the Transmit Clock's rate). The slip buffer does not allow the pointers to actually meet. To prevent this, the buffer jumps the read pointer's position backwards or forwards, depending on which pointer is about to catch the other. In this case, the read

pointer must jump back one frame to keep from overtaking the write pointer. The read pointer's jump consequently causes that frame to be re-transmitted (hence the repeat).

Likewise, a skip slip occurs if the write pointer catches up with the read pointer (incoming bit rate is faster than the Transmit Clock's rate). The write pointer can never jump, as it must smoothly write the incoming bits into memory in an uninterrupted, sequential fashion. So, the read pointer must jump ahead, skipping an entire frame of bits in the memory, in order to stay ahead of the write pointer. That frame that is jumped over never gets transmitted (hence the skip).

Pointers only jump at frame boundaries within the memory. In this way a whole frame can be neatly omitted or repeated in the transmit T1 stream. This is called a **controlled slip**.

Slips can be avoided by synchronizing the Transmit Clock to the incoming T1 line's bit rate.

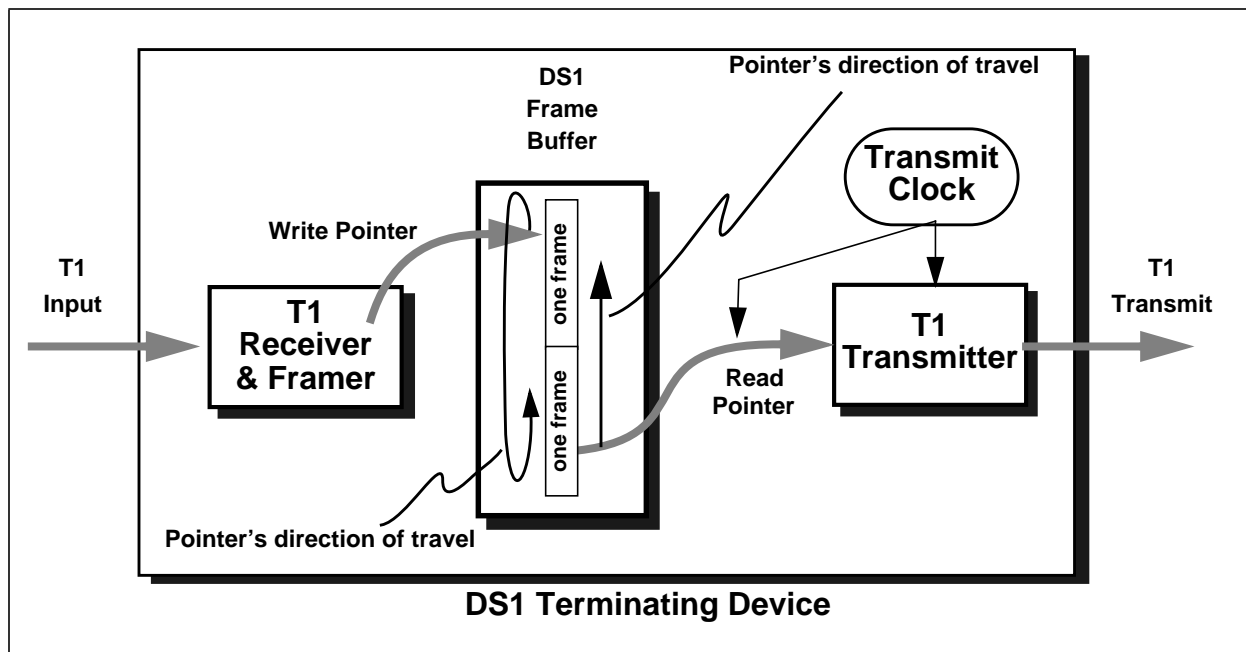


Figure 51: Controlled Slips in a DS1 Terminating Device

## 14.0 Alarms and Troubleshooting

T1 network elements use various alarms to indicate a problem with the network. If a T1 cable is cut, for example, the downstream T1 receiver will not receive the intended T1 signal. The receiver will declare an alarm to indicate the loss of the T1 signal.

These alarms are also designed to help locate the problem within the digital network. T1 alarms are of three basic types.

**Red alarm** indicates a loss of DS1 information at the T1 receiver. The receiver can lose DS1 framing because of a number of faults:

1. The T1 signal is lost
2. The 193<sup>rd</sup> bit position is lost and cannot be located
3. Corruption on the T1 line causes a **bit error rate (BER)** that is too high for an intelligible signal to be received
4. The **slip rate** is too high due to synchronization problems.
5. **Yellow alarm** is created in response to a red alarm. The yellow alarm is sent back towards the transmitter whose signal has been lost. This lets the transmitter know its transmitted information is not reaching the intended receiver.
6. In SF, a transmitter sending yellow alarm sets the second bit of each DS0 channel to 0. The receiver at the far end recognizes this as yellow alarm. The least significant bit of each DS0 is not used so as not to interfere with robbed bit signaling.
7. In ESF, yellow alarm is transmitted by a special pattern in the facility data link.
8. **Alarm Indication Signal (AIS)** is sent downstream from the T1 device that generates the red alarm. AIS indicates to downstream devices there is no DS1 information in the T1 signal. By receiving the AIS signal, a T1 device knows it is not receiving the intended DS1 and DS0 information. Further, it knows the problem does not lie in the span between itself and the upstream device that is sending the AIS signal.

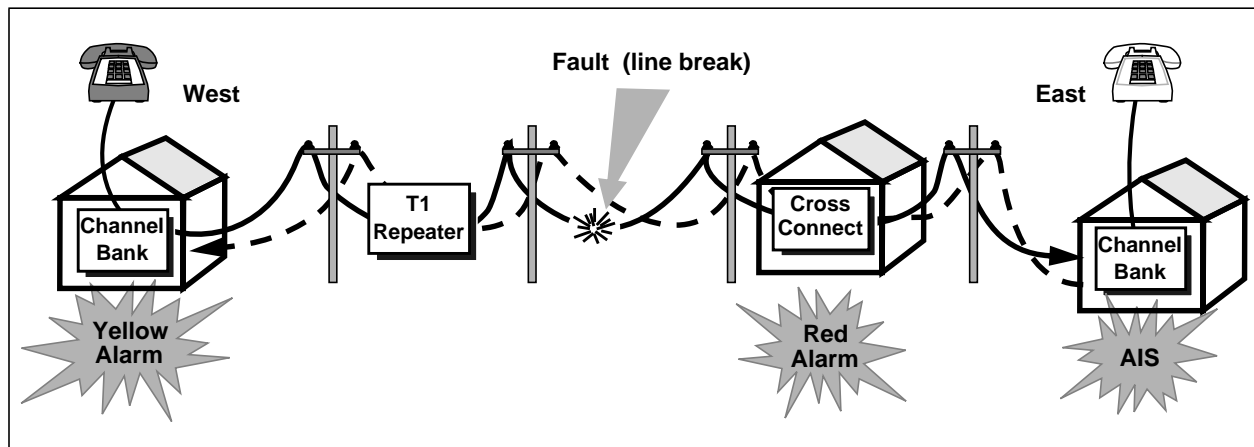


Figure 52: Typical T1 Alarm Scenario - T1 Signal Lost in one Direction (West to East)

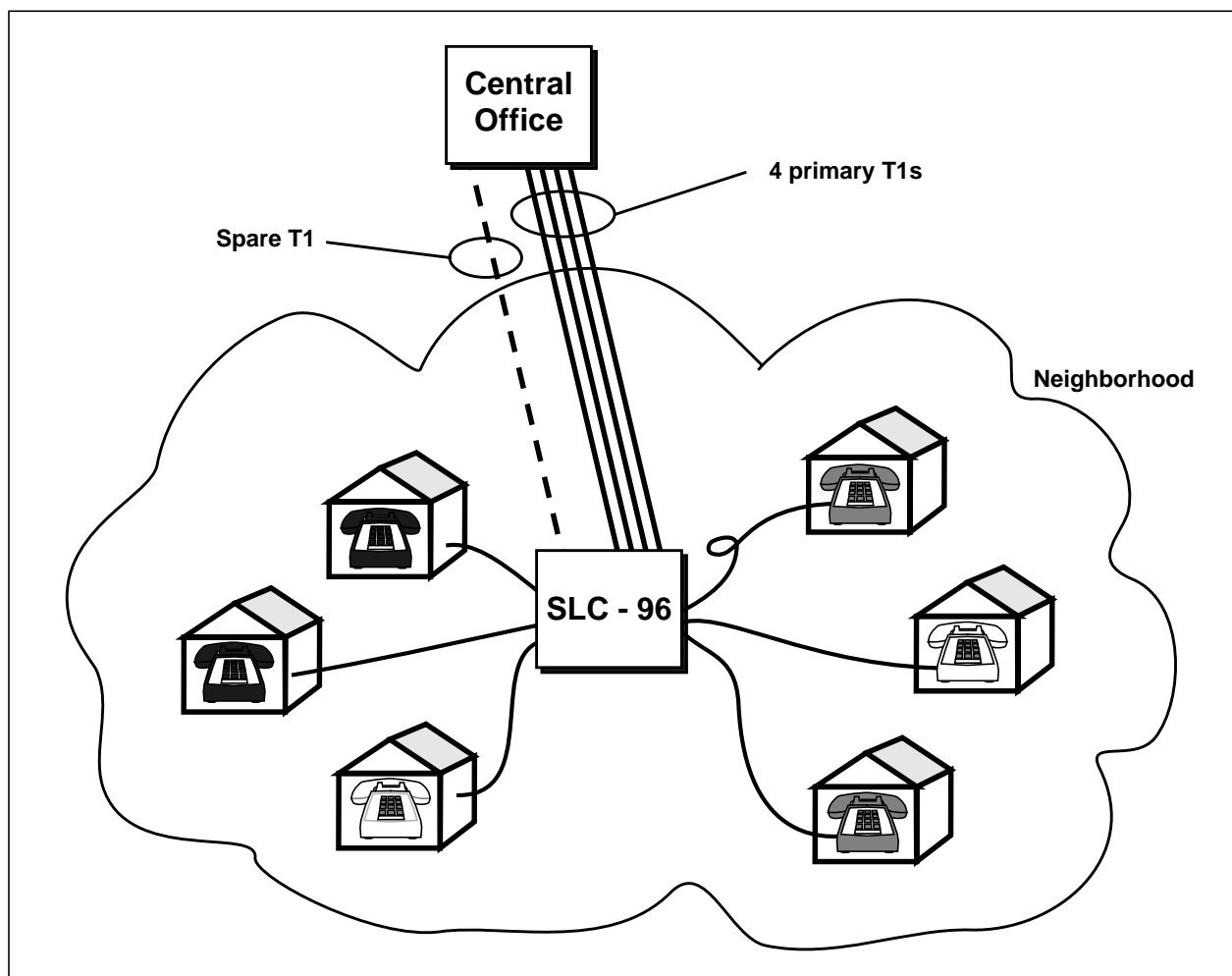
## 15.0 T1 Applications

T1 has applications beyond serving as an inter-office trunk facility. With its robust ability to encode multiple voice signals on a single transmission line, T1 applications have grown, as evidenced by these following examples.

### 15.1 Subscriber Loop Carrier (SLC-96)

The 24 channel multiplexing that T1 offers for trunk traffic can also be applied to subscriber loops (residential telephone lines). In the spring of 1979, D4 channel banks were used as the basis for a new local loop carrier system. The Subscriber Loop Carrier system (SLC-96) concentrates 24 residential telephone calls onto the 24 channels of a T1 span. Four T1 spans give the system its capacity of 96 simultaneous conversations.

SLC-96 has three modes. Mode 1, the most common SLC-96 application, uses four T1 spans as primary circuits. A fifth T1 span serves as a redundant span in case one of the four primary circuits has trouble.

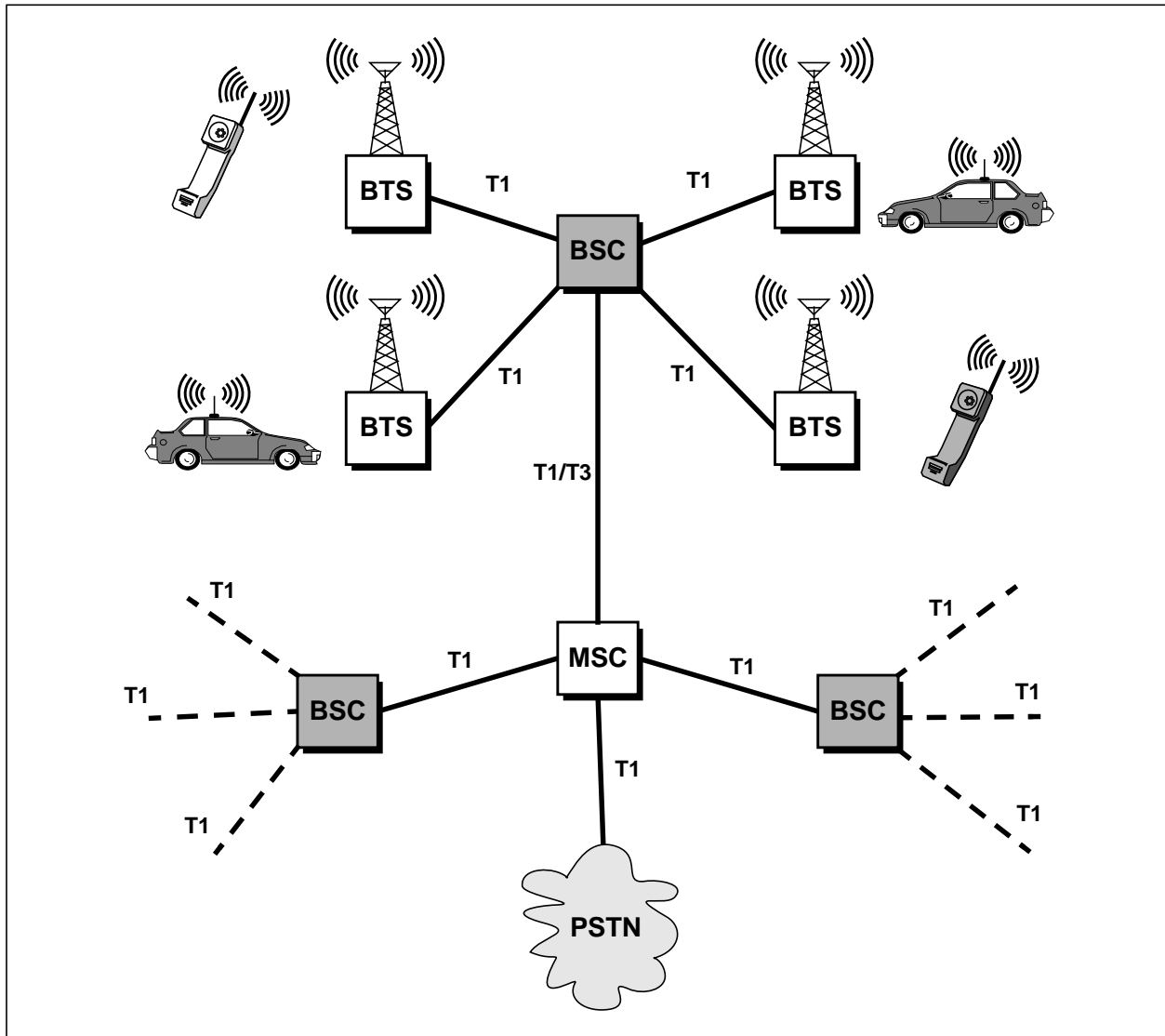


**Figure 53: SLC-96 (Mode 1)**

Today, SLC-96 is being supplanted by Next Generation Digital Loop Carrier (NG DLC).

## 15.2 Cellular Networks

Cellular telephone networks are a newer application for T1 spans. Voice and control information pass between a Base Switching Center (BSC) and its Base Transceiver Stations (BTSS) via T1 spans. T1 spans provide voice and control information links between the Mobile Switching Center (MSC) and its BSCs. T1 spans can also connect MSCs to the Public Switched Telephone Network (PSTN).



**Figure 54: T1 links a Cellular Telephone Network together**

### 15.3 T1 Deployment today: small business

T1 today is often deployed as trunking facilities to small business. T1 spans can connect a small business to a central office for telephone trunking purposes, as shown in Figure 55. Note also in this figure the banks connected to each other, but not directly to a central office. This can be for both voice trunking purposes, or for data transmission. In the example of a branch bank connected to a central bank, fractional T1 (DS0 by  $n$  to provide data bandwidth in multiples of 64 kb/s) provides a data link for data exchange the banks must do as part of their daily accounts updating.

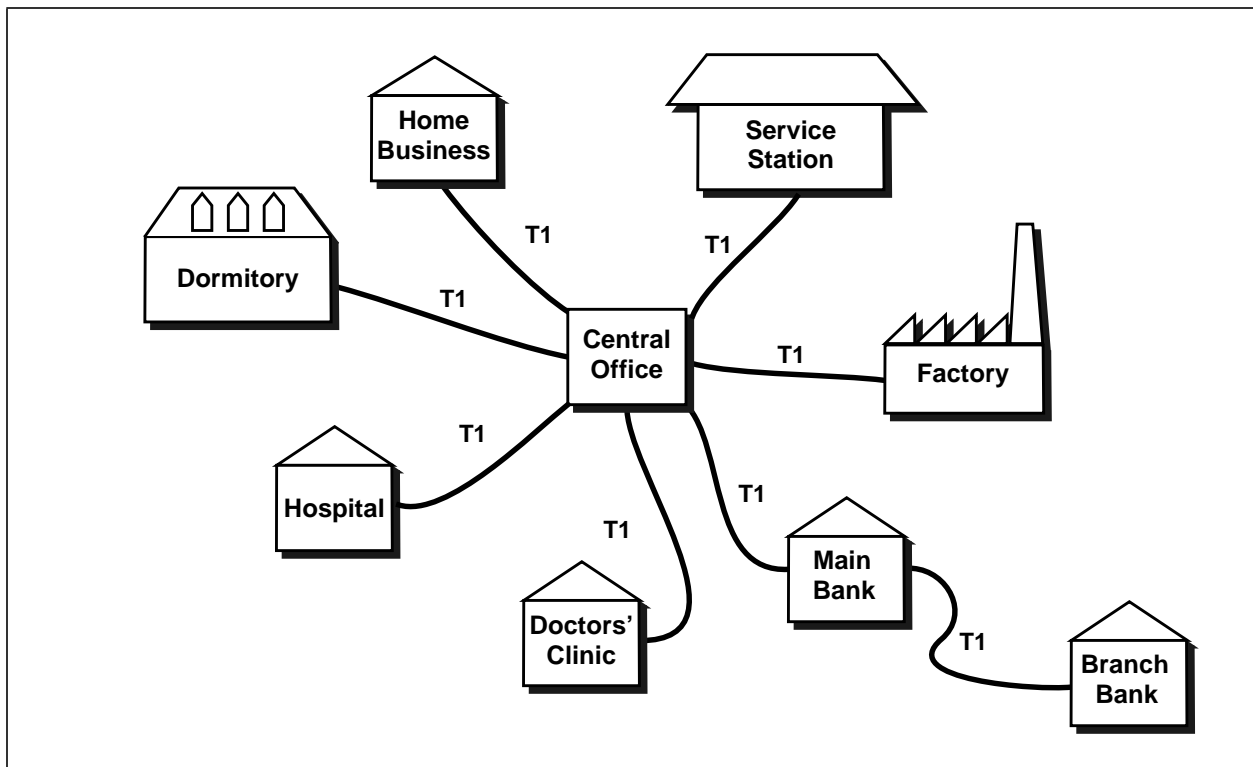


Figure 55: Small business as a user of T1

## 16.0 History of the D-series Channel Banks

Table 4 gives some comparative data on aspects of Bell Laboratories' D-series channel banks.

Designation	When Introduced	Density	Remarks
D1	1962	72 channels per bay	Voice encoding used 7 bits per TS. Only 100 out of 127 PCM quantization levels used. Later referred to as D1A. Channel order (for TS 1 through 24): 1, 13, 2, 14, 3, 15, 4, 16, 5, 17 ... 12, 24.
D1D/D2	1969	96 channels per bay	Voice encoding used $7\frac{5}{6}$ bits per TS on average, introducing toll quality voice transmission. D2 was the new channel bank that introduced SF framing. D1D was a retrofit to D1 channel banks to make them compatible with the new framing format. Note D4 framing, as SF is often called, actually started with the D2 channel bank. Channel order: 12, 13, 1, 17, 5, 21, 9, 15, 3, 19, 7, 23, 11, 14, 2, 18, 2, 22, 10, 16, 4, 20, 8, 24.
D3			Channel order: 1, 2, 3, 4, 5, ... 23, 24.
D4	late 1976	144 channels per bay	First system offering 48 channels for two DS1/T1s in one channel bank. Interchangeable modules allow 2 T1s, or a single T1C. Two channel banks can be ganged together to create a T2. Most widely used version today.
DCT			Digital Carrier Trunk.
D5		96 channels per channel bank	Microprocessor-based controller administers up to 20 D5 systems. Manual adjustments to channel bank modules are eliminated and replaced by administration system.

**Table 4: D-series channel bank evolution**

**16.1 T1 or T-1?**

Both formats are used and acceptable for designating “tee one”. I have chosen to use T1 in this document, as this is the format used in articles from Bell Laboratories’ technical publication Bell Laboratories Record. They should know. They invented it.

**17.0 Acronyms and Contractions**

AMI.....Alternate Mark Inversion

ANSI .....	American National Standards Institute
AT&T.....	American Telephone & Telegraph
BPV.....	Bipolar Violation
BSC.....	Base Switching Center
BTS .....	Base Transceiver Station
CH.....	Channel - designation for a communication link transmitted on a TS
CO.....	Central Office
CODEC.....	Coder-Decoder
Companding.....	Compressing - expanding
CRC .....	Cyclic Redundancy Check
D/A.....	Digital to Analog
DC.....	Direct Current
DCT.....	Digital Carrier Trunk
DSn .....	Digital Signal, level n
ESF.....	Extended Superframe
FDL.....	Facility Data Link
FDM.....	Frequency Domain Multiplexing
kb/s.....	kilobits per second
LSB .....	Least-significant bit
Mb/s .....	Megabits per second
MSC .....	Mobile Switching Center
NETS.....	Network Enhancing Technologies Solutions
NG DLC.....	Next Generation Digital Loop Carrier
NIU .....	Network Interface Unit
PAM.....	Pulse Amplitude Modulation
PCM .....	Pulse Code Modulation
PSTN.....	Public Switched Telephone Network
RZ .....	Return to Zero
SF .....	Superframe
SLC-96.....	Subscriber Loop Carrier - 96 channels
TS .....	timeslot - an 8-bit position with a frame which holds a channel
TDM.....	Time Domain Multiplexing
ZBTSI .....	Zero Byte Time Slot Interchange

## 18.0 Where to learn more

1. All You Wanted to Know About T1 But Were Afraid to Ask, by Bob Wachtel, an on-line document at [www.dcbnet.com/notes/9611t1.html](http://www.dcbnet.com/notes/9611t1.html). A best read.

2. T1 Tutorial, an on-line document by Larus Corporation at [www.laruscorp.com/home.htm](http://www.laruscorp.com/home.htm).
3. ANSI T1.107-1988, Digital Hierarchy - Formats Specifications
4. AT&T, Technical Reference 62411, Accunet T1.5 Service, December 1990
5. Bell System Technical Reference Pub 43801, Digital Channel Bank Requirements and Objectives, November 1982
6. "The D4 Digital Channel Bank Family", J.J. Lange, Guest Editor, The Bell System Technical Journal, Volume 61, Number 9, Part 3, November 1982.
7. AT&T, Technical Reference 54016, Requirements for Interfacing Digital Equipment to Services employing the Extended Superframe Format, September 1989
8. HP 37741A Tester User's Manual, Hewlett Packard
9. "Understanding Telephone Electronics", John L. Fike, Ph.D., P.E., George E. Friend, and Texas Instruments Learning Center. Published by Texas Instruments Learning Center. Available through Radio Shack.
10. "The Guide to T-1 Networking", by William A. Flanagan. Published by Telecom Library Inc. ISBN 0-936648-26-0.

#### 18.1 Comments, corrections, questions?

Feel free to e- mail comments or questions to me at [ken@tellabs.com](mailto:ken@tellabs.com).

END

