8.3 STATISTICAL TIME DIVISION MULTIPLEXING

Characteristics

In a synchronous time division multiplexer, it is generally the case that many of the time slots in a frame are wasted. A typical application of a synchronous TDM involves linking a number of terminals to a shared computer port. Even if all terminals are actively in use, most of the time there is no data transfer at any particular terminal.

An alternative to synchronous TDM is statistical TDM. The statistical multiplexer exploits this common property of data transmission by dynamically allocating time slots on demand. As with a synchronous TDM, the statistical multiplexer has a number of I/O lines on one side and a higher speed multiplexed line on the other. Each I/O line has a buffer associated with it. In the case of the statistical multiplexer, there are n I/O lines, but only k, where $k < n$, time slots available on the TDM frame. For input, the function of the multiplexer is to scan the input buffers, collecting data until a frame is filled, and then send the frame. On output, the multiplexer receives a frame and distributes the slots of data to the appropriate output buffers.

Because statistical TDM takes advantage of the fact that the attached devices are not all transmitting all of the time, the data rate on the multiplexed line is less than the sum of the data rates of the attached devices. Thus, a statistical multiplexer can use a lower data rate to support as many devices as a synchronous multiplexer. Alternatively, if a statistical multiplexer and a synchronous multiplexer both use a link of the same data rate, the statistical multiplexer can support more devices.

Figure 8.12 contrasts statistical and synchronous TDM. The figure depicts four data sources and shows the data produced in four time epochs (t_0, t_1, t_2, t_3). In the case of the synchronous multiplexer, the multiplexer has an effective output rate of four times the data rate of any of the input devices. During each epoch, data are

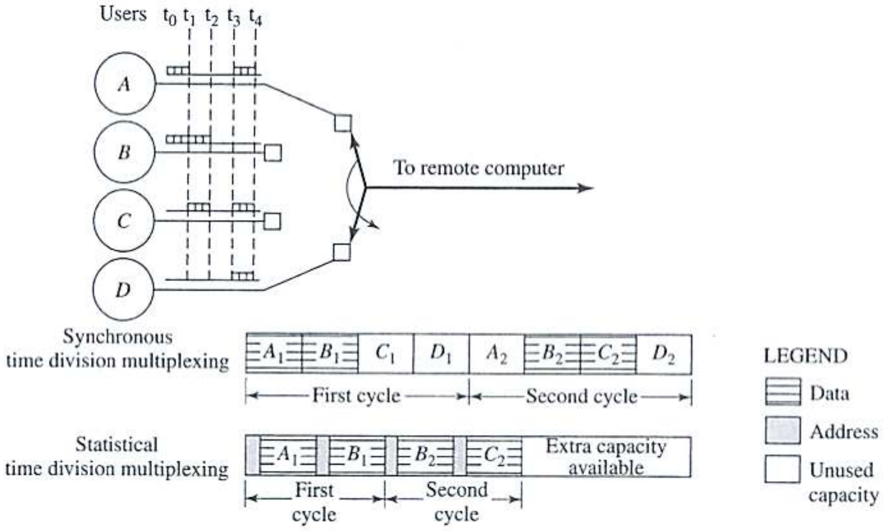

Users
t_0 t_1 t_2 t_3 t_4
A
B
C
D
To remote computer
Synchronous time division multiplexing
A_1 B_1 C_1 D_1 A_2 B_2 C_2 D_2
First cycle
Second cycle
Statistical time division multiplexing
A_1 B_1 B_2 C_2
Extra capacity available
First cycle
Second cycle
LEGEND
Data
Address
Unused capacity

Figure 8.12 Synchronous TDM Contrasted with Statistical TDM

that many of the
ıus TDM involves
all terminals are
rticular terminal.
statistical multi-
nically allocating
multiplexer has a
ine on the other.
tical multiplexer,
ble on the TDM
t buffers, collect-
:, the multiplexer
e output buffers.
attached devices
ıltiplexed line is
statistical multi-
s a synchronous
nous multiplexer
ın support more

ȝure depicts four
, t_1, t_2, t_3). In the
ve output rate of
epoch, data are

LEGEND

▤ Data

▨ Address

□ Unused capacity

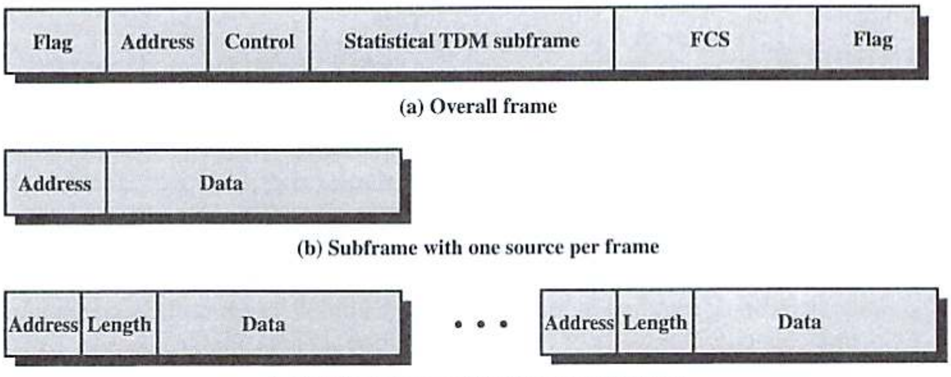
collected from all four sources and sent out. For example, in the first epoch, sources C and D produce no data. Thus, two of the four time slots transmitted by the multiplexer are empty.

In contrast, the statistical multiplexer does not send empty slots if there are data to send. Thus, during the first epoch, only slots for A and B are sent. However, the positional significance of the slots is lost in this scheme. It is not known ahead of time which source's data will be in any particular slot. Because data arrive from and are distributed to I/O lines unpredictably, address information is required to assure proper delivery. Thus, there is more overhead per slot for statistical TDM because each slot carries an address as well as data.

The frame structure used by a statistical multiplexer has an impact on performance. Clearly, it is desirable to minimize overhead bits to improve throughput. Generally, a statistical TDM system will use a synchronous protocol such as HDLC. Within the HDLC frame, the data frame must contain control bits for the multiplexing operation. Figure 8.13 shows two possible formats. In the first case, only one source of data is included per frame. That source is identified by an address. The length of the data field is variable, and its end is marked by the end of the overall frame. This scheme can work well under light load, but is quite inefficient under heavy load.

A way to improve efficiency is to allow multiple data sources to be packaged in a single frame. Now, however, some means is needed to specify the length of data for each source. Thus, the statistical TDM subframe consists of a sequence of data fields, each labeled with an address and a length. Several techniques can be used to make this approach even more efficient. The address field can be reduced by using relative addressing. That is, each address specifies the number of the current source relative to the previous source, modulo the total number of sources. So, for example, instead of an 8-bit address field, a 4-bit field might suffice.

Another refinement is to use a two-bit label with the length field. A value of 00, 01, or 10 corresponds to a data field of one, two, or three bytes; no length field is necessary. A value of 11 indicates that a length field is included.



(a) Overall frame

(b) Subframe with one source per frame

(c) Subframe with multiple sources per frame

Figure 8.13 Statistical TDM Frame Formats

Yet another approach is to multiplex one character from each data source that has a character to send in a single data frame. In this case the frame begins with a bit map that has a bit length equal to the number of sources. For each source that transmits a character during a given frame, the corresponding bit is set to one.

Performance

We have said that the data rate of the output of a statistical multiplexer is less than the sum of the data rates of the inputs. This is allowable because it is anticipated that the average amount of input is less than the capacity of the multiplexed line. The difficulty with this approach is that, while the average aggregate input may be less than the multiplexed line capacity, there may be peak periods when the input exceeds capacity.

The solution to this problem is to include a buffer in the multiplexer to hold temporary excess input. Table 8.6 gives an example of the behavior of such systems. We assume 10 sources, each capable of 1000 bps, and we assume that the average input per source is 50% of its maximum. Thus, on average, the input load is 5000 bps. Two cases are shown: multiplexers of output capacity 5000 bps and 7000 bps. The entries in the table show the number of bits input from the 10 devices each millisecond and the output from the multiplexer. When the input exceeds the output, backlog develops that must be buffered.

There is a tradeoff between the size of the buffer used and the data rate of the line. We would like to use the smallest possible buffer and the smallest possible data rate, but a reduction in one requires an increase in the other. Note that we are not so much concerned with the cost of the buffer—memory is cheap—as we are with the fact that the more buffering there is, the longer the delay. Thus, the tradeoff is really one between system response time and the speed of the multiplexed line. In this section, we present some approximate measures that examine this tradeoff. These are sufficient for most purposes.

Let us define the following parameters for a statistical time division multiplexer:

$$
\begin{aligned}
I &= \text{number of input sources} \\
R &= \text{data rate of each source, bps} \\
M &= \text{effective capacity of multiplexed line, bps} \\
\alpha &= \text{mean fraction of time each source is transmitting, } 0 < \alpha < 1 \\
K &= \frac{M}{IR} = \text{ratio of multiplexed line capacity to total maximum input}
\end{aligned}
$$

We have defined M taking into account the overhead bits introduced by the multiplexer. That is, M represents the maximum rate at which data bits can be transmitted.

The parameter K is a measure of the compression achieved by the multiplexer. For example, for a given data rate M, if $K = 0.25$, there are four times as many devices being handled as by a synchronous time division multiplexer using the same link capacity. The value of K can be bounded:

$$\alpha < K < 1$$

Table 8.6 Example of Statistical Multiplexer Performance

	Capacity = 5000 bps		Capacity = 7000 bps	
Input[a]	**Output**	**Backlog**	**Output**	**Backlog**
6	5	1	6	0
9	5	5	7	2
3	5	3	5	0
7	5	5	7	0
2	5	2	2	0
2	4	0	2	0
2	2	0	2	0
3	3	0	3	0
4	4	0	4	0
6	5	1	6	0
1	2	0	1	0
10	5	5	7	3
7	5	7	7	3
5	5	7	7	1
8	5	10	7	2
3	5	8	5	0
6	5	9	6	0
2	5	6	2	0
9	5	10	7	2
5	5	10	7	0

[a] Input = 10 sources, 1000 bps/source; average input rate = 50% of maximum.

A value of $K = 1$ corresponds to a synchronous time division multiplexer, because the system has the capacity to service all input devices at the same time. If $K < \alpha$, the input will exceed the multiplexer's capacity.

Some results can be obtained by viewing the multiplexer as a single-server queue. A queuing situation arises when a "customer" arrives at a service facility and, finding it busy, is forced to wait. The delay incurred by a customer is the time spent waiting in the queue plus the time for the service. The delay depends on the pattern of arriving traffic and the characteristics of the server. Table 8.7 summarizes results for the case of random (Poisson) arrivals and constant service time. This model is easily related to the statistical multiplexer:

$$\lambda = \alpha IR$$

Table 8.7 Single-Server Queues with Constant Service Times and Poisson (Random) Arrivals

Parameters

λ = mean number of arrivals per second

T_s = service time for each arrival

ρ = utilization; fraction of time server is busy

N = mean number of items in system (waiting and being served)

T_r = residence time; mean time an item spends in system (waiting and being served)

σ_r = standard deviation of T_r

Formulas

$$\rho = \lambda T_s$$

$$N = \frac{\rho^2}{2(1-\rho)} + \rho$$

$$T_r = \frac{T_s(2-\rho)}{2(1-\rho)}$$

$$\sigma_r = \frac{1}{1-\rho}\sqrt{\rho - \frac{3\rho^2}{2} + \frac{5\rho^3}{6} - \frac{\rho^4}{12}}$$

$$T_s = \frac{1}{M}$$

The average arrival rate λ, in bps, is the total potential input (IR) times the fraction of time α, that each source is transmitting. The service time T_s, in seconds, is the time it takes to transmit one bit, which is $1/M$. Note that

$$\rho = \lambda T_s = \frac{\alpha IR}{M} = \frac{\alpha}{K} = \frac{\lambda}{M}$$

The parameter ρ is the utilization or fraction of total link capacity being used. For example, if the capacity M is 50 kbps and $\rho = 0.5$, the load on the system is 25 kbps. The parameter N in Table 8.7 is a measure of the amount of buffer space being used in the multiplexer. Finally, T_r is a measure of the average delay encountered by an input source.

Figure 8.14 gives some insight into the nature of the tradeoff between system response time and the speed of the multiplexed line. It assumes that data are being transmitted in 1000-bit frames. Figure 8.14a shows the average number of frames that must be buffered as a function of the average utilization of the multiplexed line. The utilization is expressed as a percentage of the total line capacity. Thus, if the average input load is 5000 bps, the utilization is 100% for a line capacity of 5000 bps and about 71% for a line capacity of 7000 bps. Figure 8.14b shows the average delay experienced by a frame as a function of utilization and data rate. Note that as the utilization rises, so do the buffer requirements and the delay. A utilization above 80% is clearly undesirable.

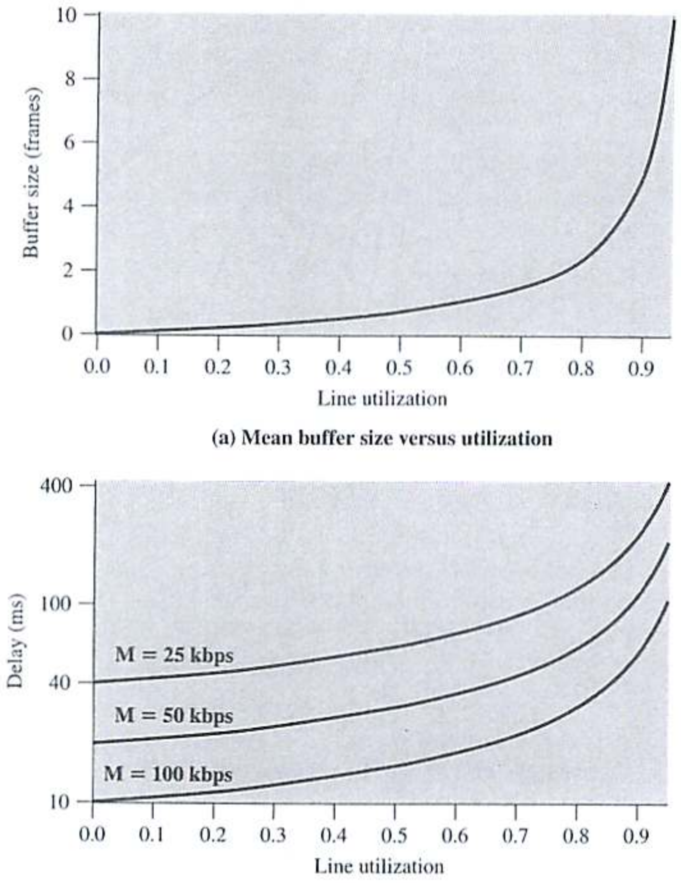
dom) Arrivals

rved)

the fraction
s, is the time

g used. For
1 is 25 kbps.
being used
tered by an

een system
a are being
r of frames
plexed line.
s, if the av-
of 5000 bps
erage delay
that as the
tion above



(a) Mean buffer size versus utilization

(b) Mean delay versus utilization

Figure 8.14 Buffer Size and Delay for a Statistical Multiplexer

Note that the average buffer size being used depends only on ρ, and not directly on M. For example, consider the following two cases:

Case I	Case II
$I = 10$	$I = 100$
$R = 100$ bps	$R = 100$ bps
$\alpha = 0.4$	$\alpha = 0.4$
$M = 500$ bps	$M = 5000$ bps

In both cases, the value of ρ is 0.8 and the mean buffer size is $N = 2.4$. Thus, proportionately, a smaller amount of buffer space per source is needed for multiplexers that handle a larger number of sources. Figure 8.14b also shows that the average delay will be smaller as the link capacity increases, for constant utilization.

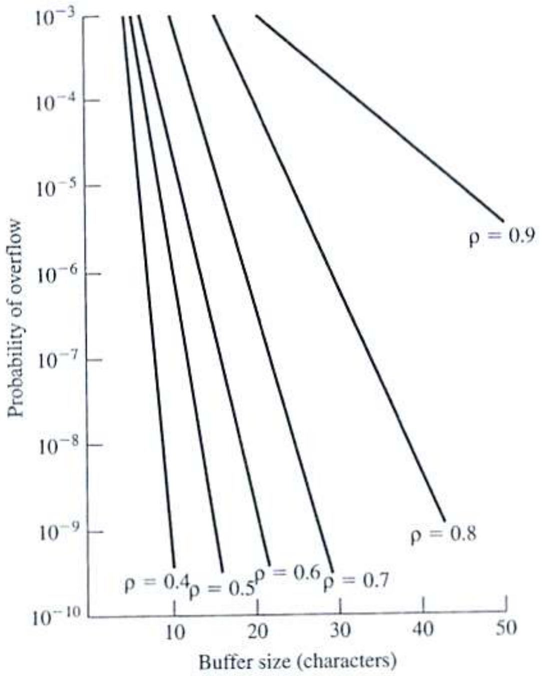


Figure 8.15 Probability of Overflow as a Function of Buffer Size

So far, we have been considering average queue length, and hence the average amount of buffer capacity needed. Of course, there will be some fixed upper bound on the buffer size available. The variance of the queue size grows with utilization. Thus, at a higher level of utilization, a larger buffer is needed to hold the backlog. Even so, there is always a finite probability that the buffer will overflow. Figure 8.15 shows the strong dependence of overflow probability on utilization. This figure and Figure 8.14 suggest that utilization above about 0.8 is undesirable.

Cable Modem

To support data transfer to and from a cable modem, a cable TV provider dedicates two channels, one for transmission in each direction. Each channel is shared by a number of subscribers, and so some scheme is needed for allocating capacity on each channel for transmission. Typically, a form of statistical TDM is used, as illustrated in Figure 8.16. In the downstream direction, cable **headend** to subscriber, a cable scheduler delivers data in the form of small packets. Because the channel is shared by a number of subscribers, if more than one subscriber is active, each subscriber gets only a fraction of the downstream capacity. An individual cable modem subscriber may experience access speeds from 500 kbps to 1.5 Mbps or more, depending on the network architecture and traffic load. The downstream direction is also used to grant time slots to subscribers. When a subscriber has data to transmit, it must first request time slots on the shared upstream channel. Each subscriber is

: the average
upper bound
h utilization.
the backlog.
v. Figure 8.15
is figure and

ler dedicates
shared by a
capacity on
ised, as illus-
subscriber, a
ie channel is
ve, each sub-
able modem
or more, de-
1 direction is
to transmit,
subscriber is

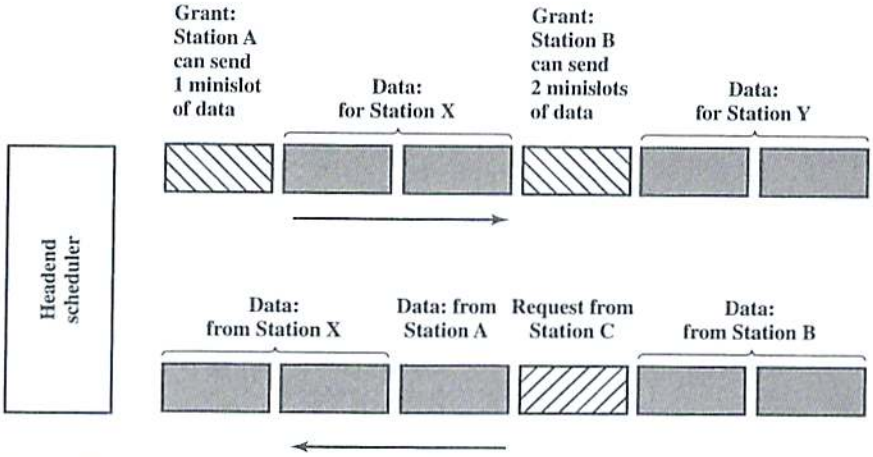


Figure 8.16 Cable Modem Scheme [DUTT99]

given dedicated time slots for this request purpose. The headend scheduler responds to a request packet by sending back an assignment of future time slots to be used by this subscriber. Thus, a number of subscribers can share the same upstream channel without conflict.

8.4 ASYMMETRIC DIGITAL SUBSCRIBER LINE

In the implementation and deployment of a high-speed wide area public digital network, the most challenging part is the link between subscriber and network: the digital subscriber line. With billions of potential endpoints worldwide, the prospect of installing new cable for each new customer is daunting. Instead, network designers have sought ways of exploiting the installed base of twisted-pair wire that links virtually all residential and business customers to telephone networks. These links were installed to carry voice-grade signals in a bandwidth from zero to 4 kHz. However, the wires are capable of transmitting signals over a far broader spectrum—1 MHz or more.

ADSL is the most widely publicized of a family of new modem technologies designed to provide high-speed digital data transmission over ordinary telephone wire. ADSL is now being offered by a number of carriers and is defined in an ANSI standard. In this section, we first look at the overall design of ADSL and then examine the key underlying technology, known as DMT.

ADSL Design

The term *asymmetric* refers to the fact that ADSL provides more capacity downstream (from the carrier's central office to the customer's site) than upstream (from customer to carrier). ADSL was originally targeted at the expected need for video on demand and related services. This application has not materialized. However, since the introduction of ADSL technology, the demand for high-speed access to the