[add variables to 93.2.2.3:]

payloadBalance

TYPE: 16 bit unsigned

Counts the number of transmitted bytes for which parity-compensating idle time has not yet been inserted.

beginTime

TYPE: 32 bit unsigned

Records the local time (in TQs) at which the current transmitted stream began.

In the OLT, this variable is set to value that the LocalTime variable has at system initialization.

In the ONU, this variable is assigned in the GATE Processing ONU Activation state diagram.

[Replace definition of FEC_Overhead_min() (and associated table 93-1) in 93.2.2.4 with the following:]

BytesSinceTQ()

This function refers to a timer that operates in units of .8ns and returns the number of byte times that have passed since the last tick of the local MPCP clock.

FEC_Overhead_delay(length)

This function calculates the amount of time (in octet times) that the MPCP control multiplexer must wait following transmission of a frame of size *length* so as to allow the insertion of parity data into the frame by the PHY layer.

As described in Clause @@92.2.3@@, FEC encoder adds 32 parity octets for each block of 216 data or control octets.

Parameter *length* represents the size of an entire frame including preamble, SFD, DA, SA, Length/Type, and FCS (but not IPG).

The following formula is used to calculate the overhead:

```
FEC_Overhead_delay(length)
{
   payloadBalance = (((localTime - beginTime) * 20 ) % 248) + BytesSinceTQ() +
                   length + IPGLen

   return  32 * round_down(payloadBalance / 216 )
}
```

FEC_Overhead_tx(length)

This function calculates the size of additional overhead, to be added by the FEC encoder, while encoding a frame of size length, using worst-case assumptions about FEC parity requirements for the frame. The function is used to check if the packet fits the grant.

The returned value is in TQ.

Parameter length represents the size of an entire frame including preamble, SFD, DA, SA, Length/Type, and FCS (but not IPG)

The following formula is used to calculate the overhead:

For the ONU the initial conditions are set at beginning of a grant:
beginTime = start_of grant_time + laser_on + sync_time

```
FEC_Overhead_tx(length)
{
   payloadBalance = (((localTime - beginTime) * 20 ) % 248) + BytesSinceTQ() +
                    length + IPGLen

   return  round_up((248 * round_up( payloadBalance / 216)) - payloadBalance) / 20)

}
```

[Make the following changes to the OLT control multiplexer state diagram (93-12):]

- Change the first line in the Start_Packet_Initiate_timer state to this:

```
Packet_initiate_delay =  sizeof(data_tx) + tailGuard +
                 FEC_Overhead_Delay(sizeof(data_tx) + tailGuard)
```

[Make the following changes to the ONU control multiplexer state diagram (93-13):]

- In state Check Size, change “Fec_overhead_max()” to “Fec_overhead_tx()”

- Change the first line in the Start_Packet_Initiate_timer state to this:

```
Packet_initiate_delay =  sizeof(data_tx) + tailGuard +
                 FEC_Overhead_Delay(sizeof(data_tx) + tailGuard)
```

[Make the following changes to diagram 93-29:]

- In Random_Wait state, change FEC_overhead_max() to FEC_Overhead_Tx()

- Add the following line to the beginning of the START_TX state

```
BeginTime <= LocalTime + laser_on + sync_time
```