142.3 PCS receive data path

142.3.5 Receive data path state diagrams

142.3.5.1 Constants

EBD257
- Type: 257-bit block
- Description: The *EBD257* constant holds the value of the end-of-burst delimiter.
- Value: 0x00

EBD_TH
- Type: Integer
- Description: The *EBD_TH* constant holds the value of the Hamming threshold required to match the end-of-burst delimiter.
- Value: 36

FEC_CW_DELIM
- See 142.2.5.1.

FEC_CW_BIT_SZ
- Type: Integer
- Description: This constant represents the size of a full-length FEC codeword in bits.
- Value: 16962 (i.e., 257×66)

IBI_EQ
- See 143.3.3.3.

MATCH_TARGET
- Type: Integer
- Description: The number of parity delimiters required to match in order to declare the block alignment in the ONU (i.e., in the continuous reception mode).
- Value: 4

RATE_ADJ_EQ
- See 143.3.3.3.

RATE_ADJ_SIZE
- See 143.3.3.3.

SBD257
- Type: 257-bit block
- Description: The *SBD257* constant represents the start-of-burst delimiter, and its value is equal to either SP2 or SP3, depending on the most recently provisioned synchronization pattern (see 142.1.3.1). Once provisioned, this value does not change and is treated as constant by the state diagram.
- Value: See 142.1.3.1

SBD_TH

Type: Integer

Description: The *SBD_TH* constant holds the value of the Hamming threshold required to match the start-of-burst delimiter.

Value: 60

142.3.5.2 Variables

All variable definitions in this section assume an independent instance of the variable per each enabled receive channel.

Block257b

Type: 257-bit block

Description: The *Block257b* variable temporarily holds one 257-bit block removed from the head of *OutputFifo*.

Block66b

Type: 66-bit block

Description: The *Block66b* variable temporarily holds one descrambled 66-bit block.

Block72b

Type: 72-bit block

Description: The *Block72b* variable temporarily holds the value being passed to the xMII.

MatchCount

Type: Integer

Description: This counter tracks the number of consecutive successful detections of FEC codeword delimiters (*FEC_CW_DELIM*).

OutputFifo[]

Type: Array of 257-bit blocks

Description: The *OutputFifo* buffer holds one FEC codeword payload after it has been processed by the FEC Decoder. The *OutputFifo* supports operations *IsEmpty()* and *GetHead()* (see 142.1.1.5).

PayloadLeft

Type: Integer

Description: This variable holds the number of EQs remaining until the FEC codeword payload reaches the maximum allowed length.

PersistentFecFail

Type: Boolean

Description: This variable is set to true if the FEC decoder is unable to correct all errors in the three FEC codewords most recently received on a given channel. Otherwise, this variable is set to false. In the OLT, the *PersistentFecFail* value is reset when *SignalFail* becomes true, or the EBD is detected, i.e., the uncorrectable FEC codewords from the previous burst do not result in *PersistentFecFail* becoming true during the next burst.

RateAdjLeft

Type: Integer

Description: This variable holds the number of EQs remaining to be generated for the current FEC codeword to fill the gap left by the removal of FEC codeword parity data.

RxCwBuf[]

Type: An array of 257-bit blocks

Description: The *RxCwBuf* is a buffer capable of storing up to *FEC_CW_BLK_SZ* 257-bit blocks. The *RxCwBuf* supports operations *Append()*, *Clear()*, *IsEmpty()*, and *IsFull()* (see 142.1.1.5).

RxInput

Type: 257-bit block

Description: The *RxInput* is a buffer containing the 257 bits most recently received from the PMA sublayer on a given channel.

RxXcBuf[3:0]

Type: Array of four 66-bit blocks

Description: This buffer holds four 66-bit blocks resulting from the decoding of a 257-bit block.

SignalFail

Type: Boolean

Description: This Boolean variable is set based on the most recently received value of *PMA_SIGNAL.indication(SIGNAL_OK)* received on a given channel. It is true if the value of *SIGNAL_OK* was FAIL and false if the value was OK.

XcIndex

Type: Integer

Description: The *XcIndex* variable is an index to the *RxXcBuf[]* array and has a value ranging between 0 and 3, inclusively.

142.3.5.3 Functions

Decode257b(blk)

Description: This function accepts one 256B/257B encoded block *blk* and transcodes it into four 64B/66B encoded blocks. The result is returned as an array of four 66-bit blocks.

Decode66b(blk)

Description: This function accepts one 64B/66B encoded block *blk* and performs the decoding operation as described in 49.2.11 and Figure 49–17. The returned value is a 72-bit block.

Descramble(blk)

Description: This function accepts one 66-bit block *blk* and performs the descrambling operation on the 64-bit payload of the block, as described in 49.2.10. The returned value is a descrambled 66-bit block.

FecDecode(cw)

Description: The *FecDecode* function to passes one complete FEC codeword *cw* to the FEC Decoder. The FEC codeword may be full-length or shortened. The codeword length is intrinsic to the parameter *cw*.

MatchFound(value1, value2, threshold)

Description: This function compares bit by bit its arguments *value1* and *value2* and returns a Boolean true if the number of bits that are different is less or equal to the *threshold*, otherwise the function returns false.

OutputBlock(eq)

Description: This function accepts one 72-bit block *eq* and outputs two 36-bit blocks over the xMII. This is a blocking function and the control is not returned to the calling state until after the second 36-bit block is sent.

ResetScrambler()

Description: This function resets the scrambler/descrambler function to the value *IBI_EQ* (see 143.3.3.3).

Shift(buffer, n)

Description: This function receives 257-bit blocks from the PMA via the *PMA_UNITDATA.indication(rx_code_group<256:0>)* primitive and inserts *n* new bits at the end of the FIFO *buffer*, while removing the same number of old bits at the head of the *buffer*. The *Shift()* function is blocking and its execution takes exactly *n* bit times at the given receiving line rate.

142.3.5.4 OLT Synchronizer Process state diagram

The OLT Synchronizer Process is responsible for receiving unaligned 257-bit blocks from the PMA sublayer and aligning these blocks to the correct 257-bit block boundary. This process hunts for *SBD257* and *EBD257* values, allowing for a certain Hamming distance (see *SBD_TH* and *EBD_TH*). The 257-bit blocks that are received between the SBD and the EBD are accumulated in the *RxCwBuf* buffer. When a complete full-length or shortened FEC codeword is stored in the *RxCwBuf*, the buffer content is passed to the FEC Decoder function (see 142.x.x.x)

The OLT Synchronizer Process shall implement an instance of state diagram as depicted in Figure 142-17 for every enabled receive channel.

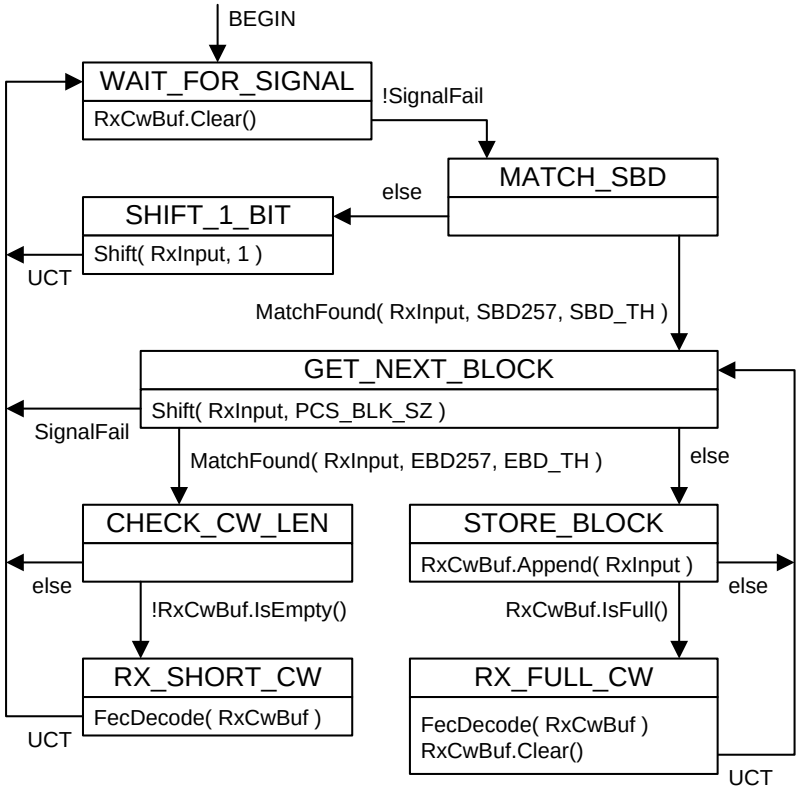

Figure 142-17 – OLT Synchronizer Process state diagram

142.3.5.5 ONU Synchronizer Process state diagram

The ONU Synchronizer Process is responsible for receiving unaligned 257-bit blocks from the PMA sublayer and aligning these blocks to the correct 257-bit block boundary. This process hunts for 10-bit FEC codeword delimiters *FEC_CW_DELIM*. The delimiter is expected to have the exact match and the block alignment is declared when the *FEC_CW_DELIM* values are matched *MATCH_TARGET* times and are exactly one FEC code-word size apart.

The received blocks are accumulated in the *RxCwBuf* buffer. When a complete full-length FEC codeword is stored in the *RxCwBuf*, the buffer content is passed to the FEC Decoder function (see 142.x.x.x). In the ONU, shortened FEC codewords are disallowed.

The ONU Synchronizer Process shall implement an instance of state diagram as depicted in Figure 142-18 for every enabled receive channel.

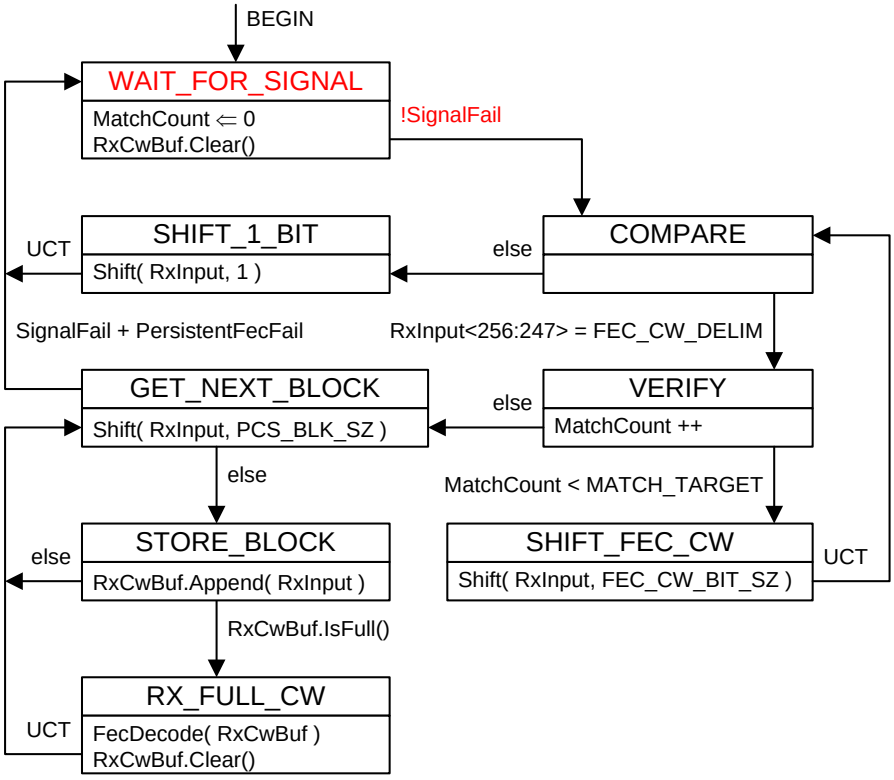

Figure 142-18 – ONU Synchronizer Process state diagram

142.3.5.6 PCS BER monitor Process

142.3.5.7 PCS Output Process

The PCS Output Process receives corrected information bits from the FEC Decoder. The FEC Decoder outputs an entire payload of a FEC codeword into the *OutputFifo* buffer. The FEC codeword payload consists of 56 257-bit blocks, however, in the OLT, the payload of a last codeword in a burst may contain fewer than 56 blocks.

The PCS Output Process converts the 257-bit blocks into EQs by first transcoding each 257-bit block into four 66-bit blocks, then descrambling each block, and finally, decoding each 66-bit block into a 72-bit block. The 72-bit blocks are passed to xMII for transfer to the MCRS.

The PCS Output Process shall implement an instance of state diagram as depicted in Figure 142-19 for every enabled receive channel.

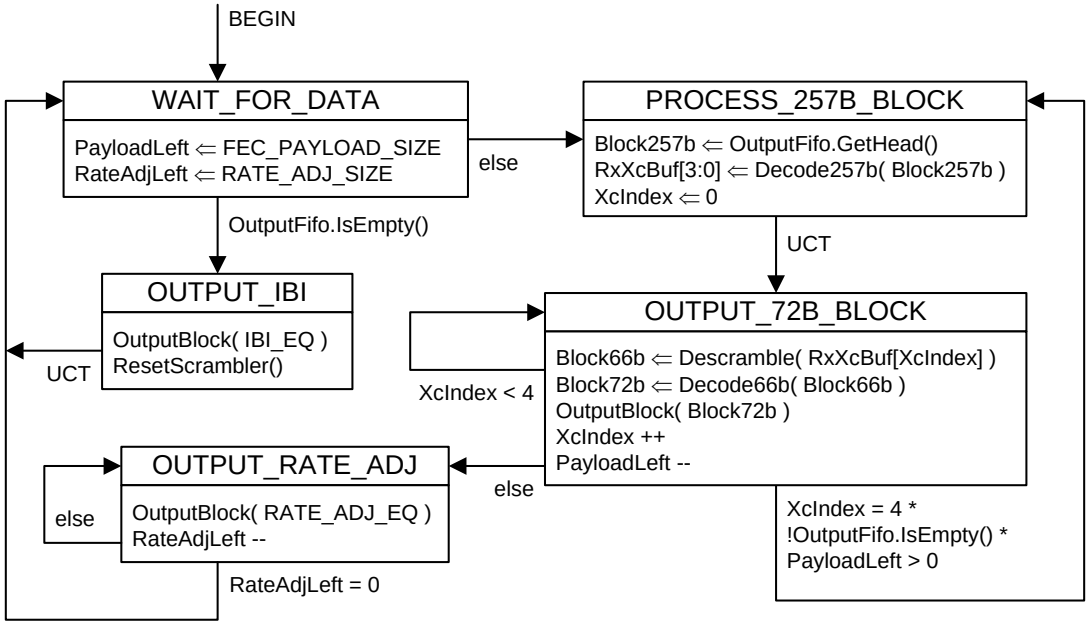

Figure 142-19 – PCS Output Process state diagram