Upstream FEC Alignment

Detailed Proposal

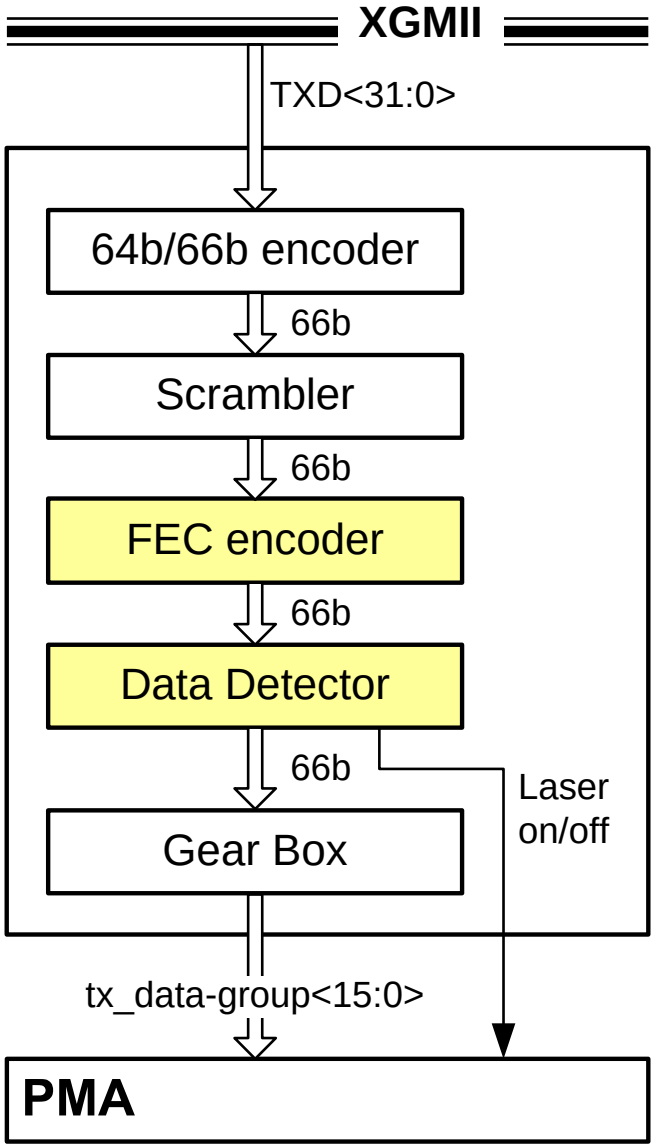
(See 3av_0701_effenberger_1.pdf
for general overview of this method)

Goal

- Develop a state machine to…
  - Turn laser on sufficiently ahead of data in order to send a synchronizing preamble
  - Generate synchronizing preamble and burst delimiter at the beginning of each burst
  - Align start of a FEC codeword with start of data in a burst (i.e., restart FEC)
  - Ensure that laser is not turned off until a complete FEC codeword is sent
  - Maintain constant PCS/PMA rate ratio by controlling the number of transmitted overhead blocks

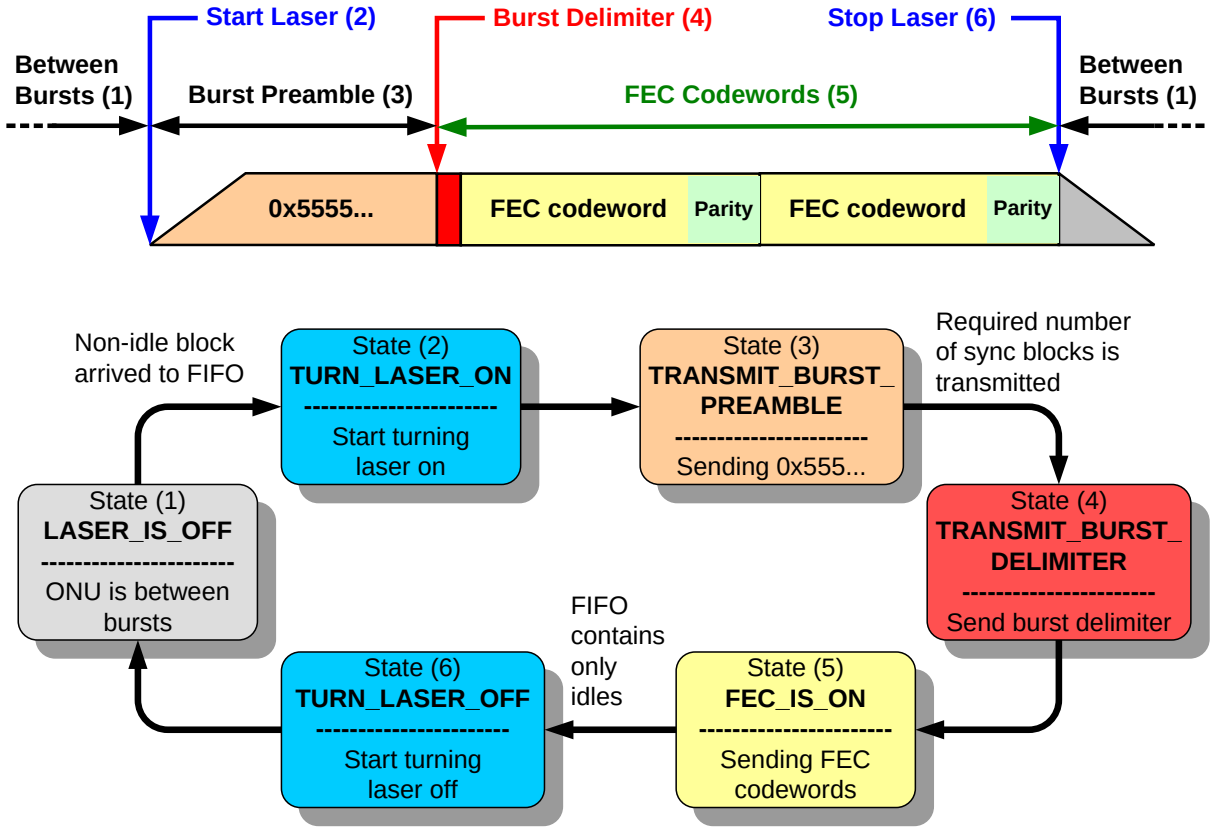
Proposed PCS Architecture

- PCS uses 64b/66b encoder, Scrambler, and GearBox as currently defined in 802.3
- New functions: FEC encoder and Data Detector
- FEC Encoder function includes FEC alignment mechanism.
- FEC Aligner and Data Detector can be combined in a single state machine as shown next

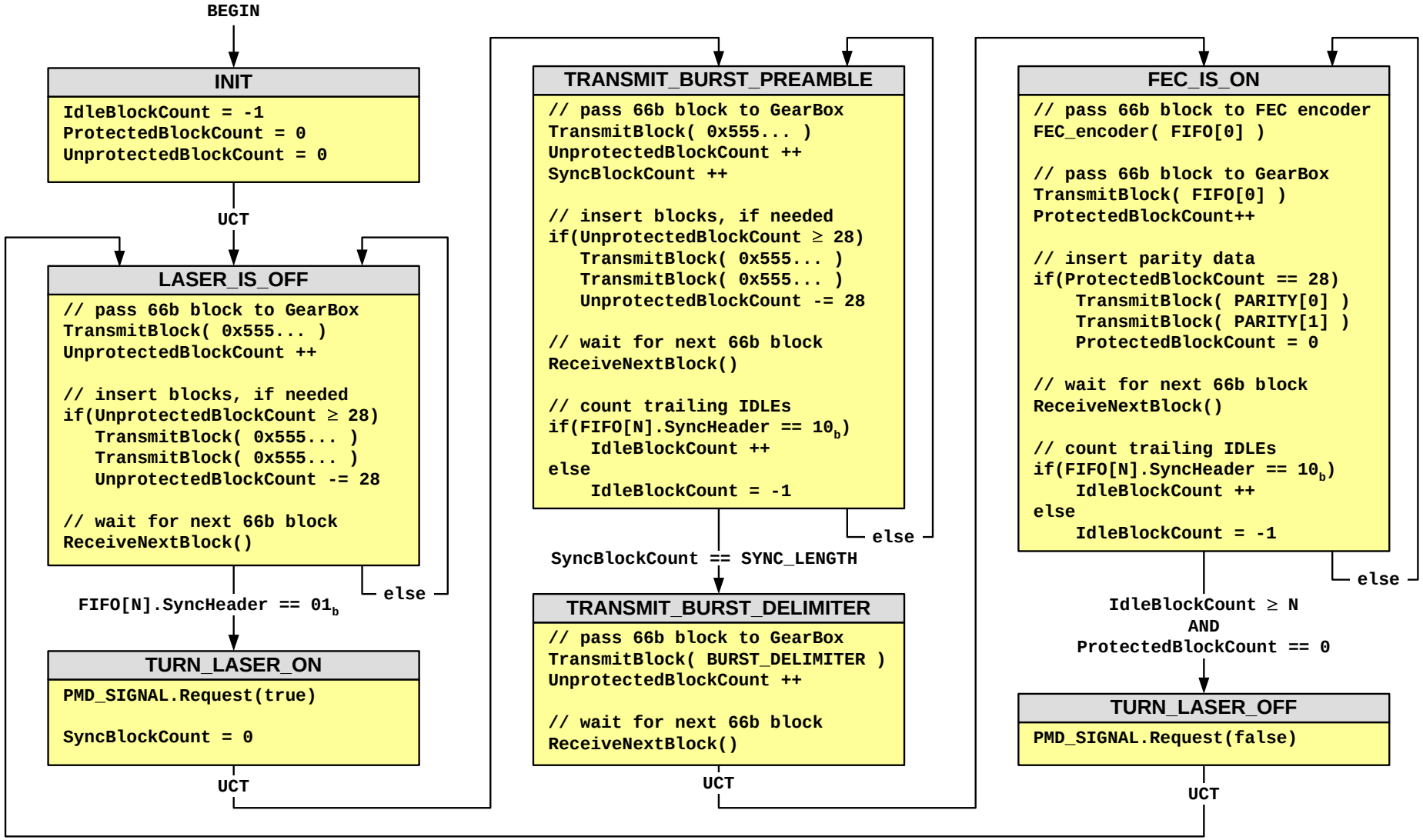


Conceptual Diagram

- There are 6 stages in burst's "lifetime"
  1. Between bursts
  2. Turning laser on
  3. Sending preamble
  4. Sending delimiter
  5. Sending FEC-protected data
  6. Turning laser off
- The stages always follow in the same order



FEC Alignment and Data Detection



The state machines assumes 2 parity blocks per every 28 payload blocks. It is trivial to change it to any other ratio.

State Machine Description (1)

- **Constants**

  BURST_DELIMITER – a special 66-bit value used to find the beginning of a FEC codeword (see 3av_0701_effenberger_1.pdf).

  SYNC_LENGTH – required number of sync blocks per burst. The value of this constant is derived from *SyncTime* parameter passed from the OLT to ONUs.

- **Variables**

  IdleBlockCount – number of consecutive non-data blocks ending with the most recently received block. The non-data blocks are represented by sync header 10.

  ProtectedBlockCount – number of blocks added to a payload of a current FEC codeword. After reaching the full payload size (28), this variable is reset to 0.

  UnprotectedBlockCount – similar to ProtectedBlockCount, but counts payload size outside of grant. This variable is used to determine a number of overhead blocks to be added to maintain proper PHY rate outside the FEC-protected grant area.

  SyncBlockCount – number of synchronization blocks transmitted in current burst.

State Machine Description (2)

- **Functions**

TransmitBlock(tx_block<65:0>)

This function passes its argument to the GearBox for further transmission to the PMA.

ReceiveNextBlock()

This function is used to receive the next 66-bit block from the scrambler. *RecevieNextBlock()* is a blocking function – it does not return until a next block becomes available and is stored in the tail position in the FIFO, as represented by the code *FIFO[N] = tx_data<65:0>.*

```
ReceiveNextBlock()
{
    // shift FIFO forward
    FIFO[0] = FIFO[1]
    FIFO[1] = FIFO[2]
       …
    FIFO[N-1] = FIFO[N]

    // receive next 66-bit block from scrambler
    FIFO[N] = tx_block<65:0>
}
```

Please, note that this particular representation of FIFO is only a model, not a guideline for implementation.

State Machine Description (3)

- **State LASER_IS_OFF**

  This state is entered when laser is off (outside a burst).

  In LASER_IS_OFF state, the state machine passes a 66-bit value (0x555…) to GearBox for further transmission to PMA sublayer (using *TransmitBlock()* function). It then increments *UnprotectedBlockCount* counter to account for this transmitted block. When the value of this counter reaches 28, two additional blocks with the same value (0x555…) are sent to the PMA . This is done to ensure that the PCS/PMA rate increase remains constant regardless of whether the laser is turned on or off. After two additional blocks are sent, the *UnprotectedBlockCount* is decremented by the size of full payload (28 blocks).

  Note: It doesn't matter what is transmitted to the PMA when laser is turned off, since the PMD will not transmit any data at all. For simplicity, 0x555… is transmitted (as in preamble), though it may be an implementation decision.

  Following this, the state machine waits for the next block received from scrambler (by calling *ReceiveNextBlock()*).

  Upon exiting this state, the state machine checks the sync header of the last block in the FIFO. If this is a data block (SH = 01), a transition is made to state TURN_LASER_ON. Otherwise, the state machine remains in state LASER_IS_OFF.

State Machine Description (4)

- **State TURN_LASER_ON**

  In this state a command is issued to the PMD sublayer to start turning the laser on. Also a variable *SyncBlockCount* is reset to 0. This variable will be used to determine the completion of synchronizing preamble.

  From this state the control is unconditionally passed to state TRANSMIT_BURST_PREAMBLE.

State Machine Description (5)

- **State TRANSMIT_BURST_PREAMBLE**

  In this state, a preamble synchronization sequence (0x555…) is passed to GearBox for further transmission to PMA sublayer (using *TransmitBlock()* function). The *UnprotectedBlockCount* and *SyncBlockCount* counters are incremented to account for this transmitted block. When the value of this counter reaches 28, two additional blocks with the same value (0x555…) are sent to the PMA .

  Following this, the *ReceiveNextBlock()* is invoked to receive the next 66-bit block from the scrambler.

  Finally this state updates the *IdleBlockCount* value, based on the just-received block. If this block is a non-data block (SH=10), the *IdleBlockCount* is incremented by 1. If this is a data block (SH=01), the counter is set to -1. The value of -1 is chosen to account for the fact that a block containing /T/ also has SH=10. Therefore, the first IDLE block following the /T/-block will be counted as 1.

Block:	data	data	/T/-block	idle	idle	…
IdleBlockCount:	-1	-1	0	1	2	…

  The state machine remains in this state until the required number of sync blocks has been transmitted (SYNC_LENGTH). When the condition *SyncBlockCount == SYNC_LENGTH* becomes true, a transition to state TRANSMIT_BURST_DELIMITER is made.

State Machine Description (6)

- **State TRANSMIT_BURST_DELIMITER**

  This state is entered after the requires number of syncronization blocks (0x555…) has been transmitted.

  In this state the *BURST_DELIMITER* value is sent to the GearBox and *UnprotectedBlockCount* is incremented by one.

  Please, note that even though the *UnprotectedBlockCount* may have reached 28 at this moment, no additional overhead blocks are inserted here. This is because the first FEC codeword should start immediately after the delimiter.

  However, to keep overhead constant, the two overhead blocks should be inserted eventually. This happens in state LASER_IS_OFF after the laser is turned off .

  As an example, if after the last overhead blocks were inserted, the burst preamble transmitted 20 more blocks 0x555… followed by the burst delimiter, then the additional two blocks will be inserted following 8 blocks 0x555… sent after the laser is turned off. Overall, the PCS will maintain the ratio of exactly 2 overhead blocks per every 28 blocks received from XGMII.

  From this state, the state machine unconditionally transitions to FEC_IS_ON state.

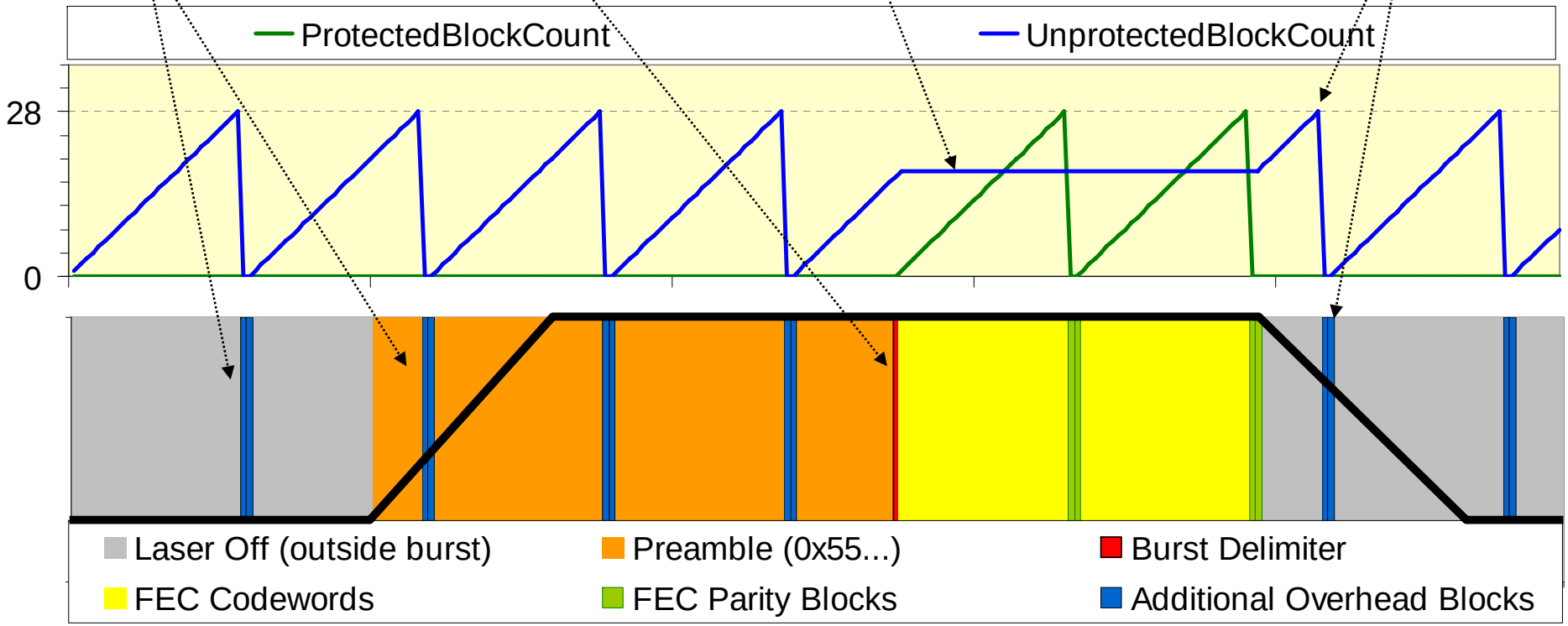
Maintaining Constant Overhead

Outside of FEC codewords, two overhead blocks are inserted after every 28 blocks

Delimiter can start in the middle of a 28-block segment

The block count of an interrupted 28-block segment is carried to the end of grant

The overhead blocks are inserted when the count reaches 28



State Machine Description (7)

- **State FEC_IS_ON**

  In this state a 66-bit block at the head of FIFO is passed to FEC encoder for parity calculation. The block is then sent to the GearBox for transmission to PMA and *ProtectedBlockCount* is incremented.

  When value *ProtectedBlockCount* reaches 28, two parity blocks are sent to the GearBox and the counter is reset to 0.

  Following this, the *ReceiveNextBlock()* is invoked to receive the next 66-bit block from the scrambler.

  Finally this state updates the *IdleBlockCount* value, based on the just-received block, as was explained in state TRANSMIT_BURST_PREAMBLE.

  Upon exiting this state, the state machine checks whether this was the last FEC codeword in a burst. This would be the last codeword if the entire FIFO is filled with IDLE blocks (i.e., *IdleBlockCount* >= N) and if the current codeword was transmitted entirely (i.e., if *ProtectedBlockCount* == 0). If this is the case, a transition to state TURN_LASER_OFF occurs, otherwise the state machine re-enters the same state.

State Machine Description (8)

- **State TURN_LASER_OFF**

  In this state a command is issued to the PMD sublayer to start turning the laser off.

  From this state the control is unconditionally passed to state LASER_IS_OFF.