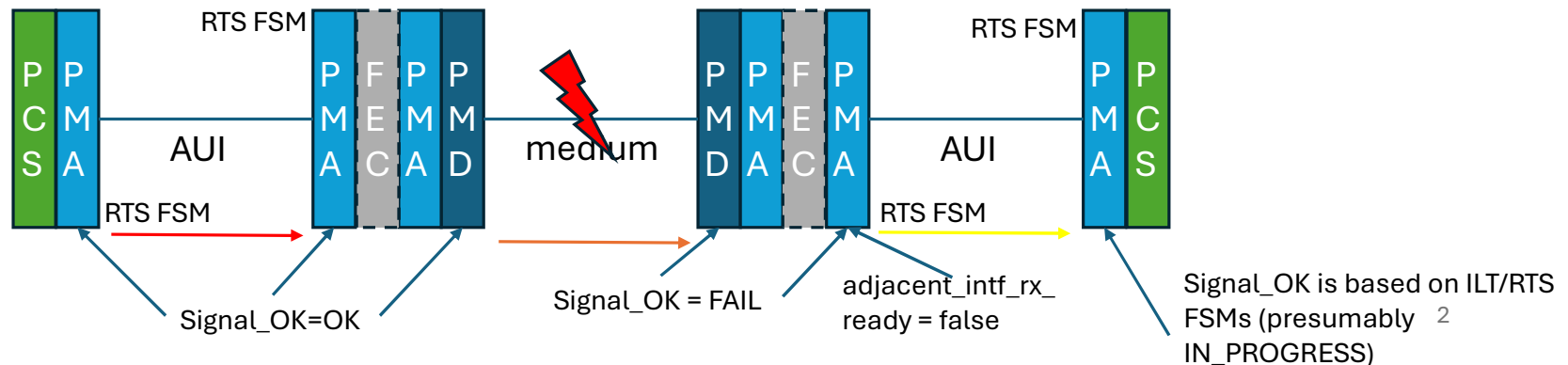


# An alternative to refactoring APSU FSMs

Tom Huber, Nokia  
Marco Mascitto, Nokia  
Luz Osorio, Nokia

# Fundamental issue with ER1 support in current proposals

- It is awkward to say ER1 PMDs do not participate in APSU and instead tunnel the signal\_OK status
  - When the PMD is failed, there is still a signal mapping into the downstream service interface that is explicitly not the signal\_OK status of the upstream interface of the peer
  - This is effectively an RTS function, just not the one currently defined in 178B (because ILT is not being used)
- huber\_01a proposed a refactored RTS FSM that would work the same way for all PMDs



# Alternative approach overview

- In 178B, add a definition of APSU as a process used to accomplish orderly path startup rather than the outcome of using the RTS and ILT functions currently defined in 178B
  - Change description of the existing ILT and RTS functions in 178B to indicate that they are for PAM4 PMDs/AUIs
  - Modify introductory material in 178B to align with that new definition
- Add a new subclause in 178B with the refactored RTS FSM from huber\_01a as the RTS FSM to be used for LR1 and ER1 PMDs
  - Align proposed text for APSU subclauses in the LR1 PMD and ER1 FEC sublayers with this new view
- Modify the APSU subclauses in all the PMDs to align with this new definition
- Modify introduction clauses to align with this

# Annex 178B changes

# Renumber subclauses in Annex 178B

- Make 178B.6 through 178B.8 subclauses of 178B.5
- Add new 178B.6 for the RTS FSM from huber\_01a
- Renumber 178B.9 and higher starting at 178B.7
- 178B.5 ILT and RTS functions for ISLs using PAM4 signaling
  - 178B.5.1 RTS function
  - 178B.5.2 ILT function
  - 178B.5.3 State diagrams
- 178B.6 RTS function for ISLs not using PAM4 signaling
- 178B.7 Training retimers
- 178B.8 Management variables
- 178B.9 PICS

# Annex 178B.2 - Overview

## 178B.2 Overview

APSU is the coordinated, orderly initialization of all ISLs in a path (Figure 178B-1). ~~APSU is facilitated by the combination of the ready-to-send (RTS) function and the inter-sublayer link training (ILT) function.~~ APSU is described in 178B.4. ~~For ISLs that use 200G/lane PAM4 signaling, APSU is facilitated by the combination of the ready-to-send (RTS) function and the inter-sublayer link training (ILT) function specified in 178B.5. For other ISLs, APSU is facilitated by the RTS function specified in 178B.6~~

The RTS function is an internal function of an AUI component or a PMD, collectively referred to as interfaces (see 178B.3), that propagates the readiness of each ISL to transition to DATA mode. The RTS function propagates the RTS status between peer and adjacent interfaces. RTS status indicates whether an ISL is ready to transition to DATA mode, or not. The RTS status is transferred between adjacent interfaces via the service interface primitives and between peer interfaces via ~~an ISL-specific mechanism the ILT function. The RTS function is described in 178B.6.~~

The ILT function is an internal function of an AUI component or a PMD that controls the establishment of communication with a peer interface within an inter-sublayer link (ISL). The ILT function provides optimization of data signaling and coordinates the transition to DATA mode on an ISL. The ILT function is defined in 178B.5.x7.

NOTE—AUI components and PMDs that are not specified to support the ILT function or the RTS function as defined in this annex may include equivalent capabilities, however this is beyond the scope of this standard.

# Annex 178B.4 – APSU (1)

## 178B.4 Autonomous path startup (APSU)

APSU is not intrinsically a function, it is an externally observable behavior resulting from the RTS and ILT functions. Support for APSU is defined as follows:

~~— An ISL can be activated using APSU if the two interfaces and the associated sublayers (e.g. PMA, Inner FEC), implement the RTS function (see 178B.6) and the ILT function (see 178B.7), or have equivalent functions.~~

— A PHY can be activated using APSU if every ISL within the PHY can be activated using APSU.

— An xMII Extender can be activated using APSU if every ISL within it can be activated using APSU.

— A Physical Layer can be activated using APSU if the PHY and xMII Extender (if implemented) can be activated using APSU.

— A path can be activated using APSU if the Physical Layer at each end can be activated using APSU.

~~An ISL, PHY, Physical Layer, or path that cannot be activated using APSU may be activated using management or other means beyond the scope of this annex.~~

~~Whether an ISL supports activation via APSU is specified in the clause that defines the interfaces for the ISL. In order to support APSU, the interface must provide a mechanism for conveying Signal\_OK status of the upstream interface across the ISL, which means the service interface must include an IS:SIGNAL\_OK.request primitive.~~

## Annex 178B.4 – APSU (2)

APSU is the result of each ISL in the path transitioning to DATA mode using the functions in 178B.5 or 178B.6. using the RTS and ILT functions as follows:

- The local\_rts variable indicates that an interface is ready to send data and propagates from the RS at one end of the path towards the RS at the other end of the path.
- The remote\_rts variable indicates that the peer interface is ready to send and receive normal data. It propagates from RS to RS in both directions independently of each other.
- The local\_rts and remote\_rts variables are propagated only across an ISL that is ready to send data.
- When an interface sends local\_rts and receives remote\_rts, it means all the ISLs in the same path (see 178B.3) are ready and it switches to DATA mode (tx\_mode = data, see 178B.8.3.1).
- When all interfaces in the path are in DATA mode, communication on the path is established.

# Annex 178B.5

## 178B.5 RTS and ILT functions ~~within an interface~~ for ISLs using PAM4 signaling

The RTS function and ILT functions ~~within an interface~~ are illustrated in Figure 178B–3. The ILT function is composed of one per-interface ILT function for the entire interface and one per-lane ILT function for each lane associated with the interface. The RTS function uses inputs from the adjacent interface (via the service interface) and the peer interface (via the ILT function) to permit the ILT function to transition to DATA mode. The signals between the RTS function and ILT function are illustrated in Figure 178B–4.

# New subclause 178B.6

## **178B.6 RTS function for ISLs not using PAM4 signaling**

The RTS function uses inputs from the adjacent interface (via the service interface) and the peer interface to enable the ISL to transition to DATA mode.

### 178B.6.1 RTS function

<see next slide>

### 178B.6.2 State diagrams

<incorporate material from huber\_01a>

# New clause 178B.6.1

The RTS function facilitates the indication of the local interface readiness to switch to DATA mode (`local_rts`) to other interfaces (peer and adjacent) and controls the switch over of the transmitter clock source to its DATA mode clock (when necessary). The RTS update state diagram (Figure 178B-x) and its associated variables define this process. An overview of the behavior of the RTS function is as follows:

- Initially, the `local_rts` variable is set false.
- Complete any training procedures for the ISL.
- Wait for the adjacent interface to be ready for DATA mode, as indicated by the `SIGNAL_OK` parameter from the adjacent interface being equal to `READY` or `OK`
- If the interface is implemented in a retimer that uses the recovered clock in DATA mode, initiate the swap to the recovered clock from the adjacent interface
- Set the `local_rts` variable to true, indicating readiness to move to DATA mode. The state of the `local_rts` variable is provided to the peer interface via an ISL-specific mechanism. The state of the `remote_rts` variable is provided by the interface to the adjacent interface via the service interface `SIGNAL_OK` parameter.

There is no specified timeout when waiting for either the `local_rts` or the `remote_rts` variables to change.

# Update clause 178B.9

- References to figure 178B-9 (RTS FSM for PAM4 ISLs) need to be generalized to also reference the RTS FSM for non-PAM4 ISLs.

# PMD clause changes

# PMD clause updates: 178 and 179

- Change clause 179.8.9 as shown
- Clause 178 simply points to 179, so no changes needed

## 179.8.9 Autonomous path startup (APSU) functions

The PMD shall **provide support the APSU function by providing** the RTS function and the ILT function with training frame format E1, as specified in Annex 178B.5

When the variable `mr_training_enable` is true, the ILT function is used to request changes to the peer transmitter state (modulation, training pattern, and precoder state), control the PMD transmitter output on each lane based on requests from the peer, indicate the receiver state, and coordinate the transition of the PMD transmit function to DATA mode. The PMD transmit function is configured in response to the requests in the incoming training frames as specified in 179.9.4.2.3 through 179.9.4.2.5. The PMD transmit function supports the coefficient indexes `k_list = {-3, -2 -1, 0, 1}` and the initial conditions preset 1 through preset 6 and initialize (see Table 179-8).

~~When `mr_training_enable` is false and `tx_mode = local_pattern` (see 178B.8.3.1), the PMD transmits The local pattern for the PMD is PRBS31Q (see 176.7.4.2).~~

The default value of `max_wait_timer_duration` is 30, corresponding to a duration of 30 seconds for `max_wait_timer` (see 178B.8.3.3).

# PMD clause updates 180 and 181 (text of 181.5.12 is the same as 180.5.12)

## 180.5.12 Autonomous path startup functions (APSU) functions

A PMD shall **support the APSU function by providing** ~~provide~~ the ready-to-send (RTS) function and the ILT function with a training frame of O1 format, specified in Annex 178B.5

When the variable `mr_training_enable` is true, the ILT function is used to request changes to the peer transmitter state (modulation, training pattern, and precoder state), indicate the receiver state, and coordinate the transition to DATA mode.

~~When `mr_training_enable` is false and `tx_mode` is `local_pattern` (see 178B.8.3.1) The local pattern for the PMD **transmits** is PRBS31Q (see 176.7.4.2).~~

The default value of `max_wait_timer_duration` is 60, corresponding to a duration of 60 seconds for `max_wait_timer` (see 178B.8.3.3).

# PMD clause updates 182 and 183 (183.5.12 has the same text as 182.5.12)

## 182.5.12 Autonomous path startup functions (APSU) functions

A PMD shall **support the APSU function by providing** ~~provide~~ the ready-to-send (RTS) function and the ILT function with a training frame of O1 format, specified in Annex 178B.5

When the variable `mr_training_enable` is true, the ILT function is used to request changes to the peer transmitter state (modulation, training pattern, and precoder state); indicate the receiver state, and coordinate the transition to DATA mode.

~~When `mr_training_enable` is false and `tx_mode` is `local_pattern` (see 178B.8.3.1) The local pattern for the PMD **transmits** is PRBS31 encoded by inner FEC (see 177.6.1.1).~~

The default value of `max_wait_timer_duration` is 60, corresponding to a duration of 60 seconds for `max_wait_timer` (see 178B.8.3.3).

# Clause 185

- Changes to the proposal in brown\_04 for 185.x.x:

185.x.x Autonomous path startup functions (APSU) functions

A PMD shall ~~support the APSU function by providing~~ ~~provide~~ the ready-to-send (RTS) function ~~and the ILT function~~ as specified in Annex 178B.6

~~The variable mr\_training\_enable is always set to false.~~

The variable isl\_ready is always set to true.

When in the notRTS state, the PMD output is squelched.

~~When tx\_mode is local\_pattern (see 178B.8.3.1),~~ When in the RTS state, the PMD transmits PRBS31 encoded by the 800GBASE-LR1 Inner FEC as defined for the test pattern generator in 184.6.1.

# Clause 186

- New clause 186.2.6:

## 186.2.6 Autonomous path startup functions (APSU)

The ER1 FEC sublayer shall support the APSU function by providing the ready-to-send (RTS) function specified in Annex 178B.6.

The signal\_OK parameter from the service interface is mapped to the MNT field as specified in 186.2.x.

# Introductory clause changes

# Example of current text of introduction clauses

## **116.2.9 Autonomous path startup (APSU)**

The autonomous path startup (APSU) functions are defined in Annex 178B. The inter-sublayer link training (ILT) function facilitates the orderly startup of an inter-sublayer link (ISL). The ready-to-send (RTS) function coordinates the startup of a series of ISLs along a path. Path and ISL are defined in 178B.3.

ILT and RTS functions are specified for use by the following PMD and AUI types:

<bullet list of PMD and AUI types>

- The same text is in 169.2.10 and 174.2.12

# Proposed replacement text for 116.2.9, 169.2.10, 174.2.12

Autonomous path start-up (APSU) is the orderly startup of the path, which occurs when SIGNAL\_OK reaches the OK status at every sublayer interface in both directions. This requires communication to be established across all ISLs within the path (see 178B.3) and all sublayers transmit and receive paths to be operating as expected. PHYs and Physical layers composed entirely of sublayers that provide SIGNAL\_OK in both the transmit and receive directions are able to make use of APSU.

Annex 178B defines the RTS and ILT functions to facilitate establishing the communication and the transfer of SIGNAL\_OK across ISLs found within a path. These functions are supported by many PMD and AUI components that utilize various types of medium.

Details of how each PMD or AUI supports APSU are found in the relevant clauses.

APSU is specified for use by the following PMD and AUI types:

<bullet list>

(for 169.2.10, add 800GBASE-LR1 and 800GBASE-ER1 to the list)