10GE WAN PHY:
Physical Medium Attachment (PMA)

IEEE 802.3 Meeting, Albuquerque
March 6-10, 2000

Norival Figueira, Paul Bottorff, David Martin,
Tim Armstrong, Bijan Raahemi ..................... Nortel Networks
Enrique Hernandez-Valencia ......................... Lucent (Bell Labs)
Nevin Jones .............................................. Lucent Microelectronics
Pankaj Kumar ............................................ Level One/Intel
Bjørn Liencres ......................................... Juniper Networks
Tom Palkert .............................................. AMCC
Iain Verigin, Stuart Robinson, Tom Alexander ... PMC Sierra
Nader Vijeh ............................................... Lantern Communications
Frederick Weniger ...................................... Vitesse
Based on Posted Document

- “Proposal for a 10 Gigabit Ethernet WAN PHY”

Agenda

- PMA/PMD interface
  - PCS/PMA interface is conceptual
- PMA frame and overheads
- PMA framing functions
  - Transmit and Receive PMA frame
- PMA frame synchronization process
- $x^7 + x^6 + 1$ frame-synchronous scrambler
Functional Block Diagram

10GMII

TXD<31:0>
10GTX_CLK
TXC<3:0>

Flow control

RXD<31:0>
RX_CLK
RXC<3:0>

312.5 M
TXD<31:0>/s

Conceptual Interface

622.08 M
tx_bit<15:0>/s

10GE WAN PHY: PMA, March 2000-3
Possibly Better Terminology

10GE WAN PHY: PMA, March 2000

10GMII

TXD<31:0>  
10GTX_CLK
TXC<3:0>

Flow control

RXC<3:0>
RX_CLK
RXD<31:0>

312.5 M
TXD<31:0>/s

622.08 M
rx_bit<15:0>/s

Conceptual Interface

PCS 1

TRANSMIT

RXD<31:0>

PMA/PMD

Transmit

MDI

Receive

PCS 2

RECEIVE

TXD<31:0>

TXC<3:0>

10GTX_CLK

tx_bit<15:0>

rx_bit<15:0>

tx_data

tx_control

rx_data

rx_control

TRANSMIT PMA Frame

RECEIVE PMA Frame

10GMII

Possibly Better Terminology
PMA Interfaces

- **PCS/PMA ➔ conceptual interface**

- **PMD interface**
  - `tx_bit<15:0>`
    - 16-bit vector representing two octets received from the PMA
    - transitions synchronously with `tx_bit_clk`
  - `tx_bit_clk`
    - 622.08 MHz clock generated by the PMA
  - `rx_bit<15:0>`
    - Most recently received 16 bits (MSB first) from the MDI. It is a continuous and unaligned sequence of octets
    - transitions synchronously with `rx_bit_clk`
  - `rx_bit_clk`
    - 622.08 MHz clock generated by the PMD
  - all LVDS
PMA/PMD Interface

Serialized octets (MSB first)

Transmit process

Receive process

PMA
(Conceptual view)

tx_bit_clk
tx_bit<15:0>

rx_bit_clk
rx_bit<15:0>

16-bit word

15
0

1

15
0

622.08 MHz

PMD

Transmit

Receive

MDI

10GE WAN PHY: PMA, March 2000- 6
PMA Framing Functions

- **Transmit PMA Frame**
  - PMA framing of octet stream
  - Scrambling of PMA frames using the \(x^7+x^6+1\) frame-synchronous scrambler
  - Transmission of resulting data stream to the PMD sublayer
    - depends on the PMD interface
PMA Framing Functions (cont.)

- **Receive PMA Frame**
  - Receiving of data stream from PMD sublayer
    - depends on PMD interface
  - PMA frame synchronization and octet delineation
  - Descrambling of PMA frames with the $x^7 + x^6 + 1$ frame-synchronous scrambler
PMA Frame

PMA Frame = STS-192c Frame

STS-192c = Synchronous Transport Signal – level 192, c = concatenated
SPE = Synchronous Payload Envelope
SPE Position

17280 columns

Transport Overhead

Transport Overhead

9 rows

9 rows

Start of SPE

pointer

Path Overhead

SPE

125 µs
Octet Transmission Order

• Top to bottom, row-by-row, left to right
Overhead Layers

- Payload
  - Map payload and Path Overhead into SPE
  - Map SPE and Line Overhead into PMA frame
  - Map Section Overhead into PMA frame
  - PMA Frame

- Path
- Line
- Section

- PMA Frame
Transport Overhead

Section Overhead

Line Overhead

STS-1#  1  2  3  ...  192  1  2  3  ...  192  1  2  3  ...  192
Column#  1  2  3  ...  192  193  194  195  ...  384  385  386  387  ...  576

- **A1, A2, A3**
- **B1, H1, H2, H3, S1**
- **J0, Z0, Z1**

**Columns:**
- **K1, K2, K3** defined overhead octets (B2, E1-2, F1, D1-12, M1, Z1-2), unused by 10GE WAN PHY (set to zero)
- **Undefined overhead octets** (set to zero)

---

10GE WAN PHY: PMA, March 2000-13
Section Overhead: A1 and A2

• “Framing octets”

• Used by the PMA frame synchronization process to determine where octets and the PMA frame start
  — Transition from A1 to A2 octets is used for synchronization

• Fixed value:
  — A1 = 11110110
  — A2 = 00101000
Section Overhead: J0 and Z0

• J0 (“Section Trace”)
  — Allows a receiver to verify its continued connection to the intended transmitter
  — Provisioned Value
    – When no value is provisioned, J0 shall be set to 00000001

• Z0 (“Section Growth”)
  — Fixed value: 11001100
Section Overhead: B1

• “Section BIP-8”

• Used as a Section error monitoring function

• Calculated value:
  — BIP-8 code (using even parity) over all the bits of the last transmitted PMA frame after scrambling

NOTE

BIP-8 (Bit-Interleaved Parity-8) with even parity: The $i^{th}$ bit of the code provides even parity over the $i^{th}$ bit of all the covered octets.

BIP-8 of the bit sequence 11110000 00001111 is 11111111.
Line Overhead: H1 and H2

- “Payload Pointer”
  - Allows the SPE to be dynamically aligned within the Envelope Capacity

- Values:
  - All H1 octets after the first one are set to the fixed value 10010011
  - All H2 octets after the first one are set to the fixed value 11111111
Line Overhead: H1 and H2 (cont.)

- **First H1 and H2**
  - 16-bit word containing an NDF field and a 10-bit STS pointer in the range of 0 to 782
  - Fixed values:
    - 10GE WAN PHY transmits H1 = 01100010 and H2 = 00001010, i.e., “normal” STS pointer = 522
    - Receiver 10GE WAN PHY shall be able to process arbitrary NDF and STS pointer values (which may be changed by a transport network)

![Diagram showing the structure of H1 and H2 frames](image-url)
Line Overhead: H1/H2 and SPE Position

- H1 – – H2 – – H3 – 0 1 2 … Start of SPE

10-bit pointer (first H1 and H2)

192 octets (not to scale)

Transport Overhead

Path Overhead

SPE

522 523

782

17280 columns

125 µs

9 rows

9 rows

192 octets (not to scale)
Line Overhead: H3

- “Pointer Action Bytes”
- Used for SPE frequency justification
  - Allows LTE to have slightly different clocks at the receiver and transmitter paths
- Content:
  - Carries 192 extra SPE octets in the event of a “negative pointer adjustment,” i.e., which may be required when the receiver clock is faster than the transmitter clock
  - Set to zero when not used
Line Overhead: K1, K2, and S1

- **K1 and K2**
  - Fixed values: K1 = 00000001, K2 = 00010000
  
  - K1 and K2 are used on the protection line for automatic protection switching signaling. Above settings indicate a working channel rather than the protection channel.

- **S1**
  - Fixed value: 00001111

  - Indicates quality clock information to receiver. Above setting indicates “don’t use for synchronization”
Path Overhead and “Fixed Stuff”

Defined overhead octets (F2, H4, Z3-5), unused by 10GE WAN PHY (set to zero)

Fixed Stuff columns provide compatibility with SONET/SDH byte-interleaving and concatenation rules (set to zero)

Path Overhead

- J1
- B3
- C2
- G1

Fixed Stuff

63 columns

Path

Overhead

Fixed Stuff

“Fixed Stuff” columns provide compatibility with SONET/SDH byte-interleaving and concatenation rules (set to zero)
Path Overhead: J1, B3, and C2

- **J1** ("Path Trace")
  - Fixed value: 00000000

- **B3** ("Path BIP-8")
  - Used as a Path error monitoring function
  - Calculated value: BIP-8 code (using even parity) over all the octets of the last transmitted SPE before \((x^7+x^6+1)\) scrambling

- **C2** ("Path Signal Label")
  - Identifies the contents of the STS SPE (i.e., 10GE WAN PHY)
  - Fixed value: 00011010 (provisional value assigned to 10 GE)
Path Overhead: G1

- **“Path Status”**
  - Conveys the Path terminating status and performance back to the transmitter (i.e., a PTE)

- **Calculated value:**
  - REI-P field = number of bit errors detected with the B3 octet of the last received SPE
  - RDI-P field = Detected defects on the received signal (values are TBD)
    - Propose to support:
      - Loss of Packet Delineation (LPD-P)
      - Loss of Pointer (LOS-P)
      - Payload Mismatch (PLM-P)

```
<table>
<thead>
<tr>
<th></th>
<th>REI-P</th>
<th>RDI-P</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>
```

REI-P = Path Remote Error Indication
RDI-P = Path Remote Defect Indication

REI-P field
0000 to 1000 = 0 to 8 errors when received, 1xx1 = 0 errors
**Reference Diagram: Transmit PMA Frame**

- **Functional View**
- **PMA frame formation (stages)**
  - (1) Path Overhead and fixed stuff columns
  - (2) Line Overhead
  - (3) Section Overhead
  - (4) Scramble with $x^7 + x^6 + 1$ (first row of Section Overhead, i.e., $A1/A2, J0,$ and $Z0,$ is not scrambled)
  - (5) 16-bit words are transmitted to PMD (depends on PMD interface)
Reference Diagram: Receive PMA Frame

- **Functional View**
- **PMA frame processing (stages)**
  - (1) “Serialize” received PMD signal
  - (2) PMA frame synchronization and octet delineation
  - (3) Descramble with $x^7 + x^6 + 1$ (first row of Section Overhead is not descrambled)
  - (4) Extract Section Overhead, Line Overhead, Path Overhead, Fixed Stuff columns
  - (5) Remaining octets = payload
Reference Diagram

PMA Service Interface

Flow Control

Path Overhead

Fixed stuff

for next SPE

B3 (BIP-8)

Path Overhead

Fixed stuff

Columns

Section Overhead

Inhibit scrambling first row of Section OH

Line Overhead

x^7+x^6+1

scrambler

Synchronization

Information required to calculate G1

PMA frame

SPE

for next frame

16-bit word assembly

B1 (BIP-8)

for next frame

622.08 Mbaud

rx_bit_clk

rx_bit<15:0>

“Serializer”

rx_control

rx_data

PMD Service Interface

1

x^7+x^6+1

Descrambler

inhibit descrambling of first row of Section Overhead

rx_bit<15:0>

rx_bit_clk

PMA Service Interface

Flow Control

Conceptual Interface

PMD Service Interface

Transmit PMA Frame

Receive PMA Frame

PMD Service Interface

Reference Diagram
PMA Frame Synchronization

- Uses A1/A2 transition (i.e., frame marker) for frame and octet delineation

- Looks for the A1/A2 framing pattern consistently
  - Expects it to appear once every 155520 octets (155520 = length of the PMA frame)
  - When the framing pattern appears in the right place enough times, correct frame synchronization is assumed
PMA Frame Synchronization (cont.)

- **Posted document**
  - Provides a set of rules to be satisfied by a PMA frame synchronization process
  - Does not provide specific details on how a PMA frame synchronization process works
  - Does not imply any specific implementation. Any PMA frame sync procedure that complies with the defined set of rules is acceptable

- This presentation shows the state diagram of a frame synchronization processes similar to the ones used in typical OC-192 equipment
PMA Frame Sync: START State

- Initial state
- Searches bit by bit for \(i\) correct A1 octets
- Moves to A1_ALIGN state on an exact match
PMA Frame Sync: A1_ALIGN State

- Confirms byte alignment
- Moves to PRESYNC state on at least \( j \) correct A1 octets followed by \( k \) correct A2 octets
- Moves to START state if pattern is not found

\[
\begin{align*}
\text{PRESYNC} & \quad \geq j \text{ correct A1s followed by } k \text{ correct A2s} \\
\text{START} & \quad \text{Did not find } i \text{ correct A1s} \\
& \quad \text{Pattern not found}
\end{align*}
\]
PMA Frame Sync: PRESYNC State

- Checks frame for correct A1/A2 transition pattern at correct place
- Moves to SYNC state on “n” correct A1/A2 transition patterns
- Moves to START state on an incorrect A1/A2 transition pattern
PMA Frame Sync: SYNC State

**SYNC**
[frame by frame]

- Correct A1/A2 transition pattern or < *m* consecutive incorrect transitions
- *m* consecutive incorrect A1/A2 transition patterns

**PRESYNC**
[octet by octet]

- *n* correct A1/A2 transition patterns
- Incorrect A1/A2 transition pattern

**START**
[bit by bit]

- Did not find *i* correct A1s
- *i* correct A1s
- ≥ *j* correct A1s followed by *k* correct A2s

- Checks frame for A1/A2 transition pattern at correct place
- Moves to START state with “*m*” consecutive frames with incorrect A1/A2 transition patterns
PMA Frame Sync: State Diagram

- **SYNC** [frame by frame]
  - Correct A1/A2 transition pattern or < m consecutive incorrect transitions
  - m consecutive incorrect A1/A2 transition patterns

- **START** [bit by bit]
  - Did not find i correct A1s
  - Pattern not found

- **PRESYNC** [octet by octet]
  - n correct A1/A2 transition patterns

- **A1_ALIGN** [octet by octet]
  - ≥ j correct A1s followed by k correct A2s
  - Incorrect A1/A2 transition pattern
PMA Frame Sync. Performance

- Example for \( m = 4 \), A1/A2 transition pattern = 2 A1/A2s
  - Probability of frame loss \( \approx 1.049 \times 10^6 \times \text{BER}^4 \)
    \[ = 1.049 \times 10^{-42} \ (@ \text{BER} = 10^{-12}) \]
  - Average interval to frame loss
    \[ \approx 3.7 \times 10^{30} \text{ years} \ (@ \text{BER} = 10^{-12}) \]
    (> estimated age of observable universe, i.e., \( \sim 10^{10} \text{ years} \))

- More robust implementations are possible, e.g., see
  - “10GE WAN PHY Delineation Performance”
$x^7 + x^6 + 1$ Frame-Synchronous Scrambler

- **Purpose**
  - Assures that the optical interface signal has an adequate number of transitions for line rate clock recovery at the receiver

- **Scrambles**
  - All the octets of the “PMA frame” with the exception of the first row of the transport overhead

- **State is periodically resynchronized**
  - Scrambler state is reset to $1111111$ on the most-significant bit of the octet following the last Z0 octet
Use of $x^7 + x^6 + 1$ Scrambler

Reset scrambler state to 1111111

PMA frame

Transport Overhead

Envelope Capacity

Scrambled

Not scrambled
$x^7 + x^6 + 1$ Scrambler/Descrambler

(Functional Diagram)

- Scrambled/descrambled bit stream
- XOR
- 7-bit shift register
- Reset to 1111111
- Clock
- Scrambler/descrambler state = content of the 7-bit shift register
Bit Order of Scrambling/Descrambling

- Most significant bit (LSB) first
Summary

- **PMA/PMD interface**
  - 16-bit LVDS

- **PMA frame and overheads**
  - Described proposed minimum set of overheads

- **PMA framing functions**
  - Described Transmit and Receive PMA frame processes

- **PMA frame synchronization process**
  - Described a typical frame synchronization process

- **$x^7 + x^6 + 1$ frame-synchronous scrambler**
  - Described functional diagram and resynchronization scheme