Fundamentals of Synchronization & Timing
WSTS 2021 Tutorial Session
March 24, 2021
Kishan Shenoi
kshenoi@qulsar.com
Fundamentals of Timing and Synchronization

► Basic Principles
  ▪ Time and Frequency
  ▪ Alignment (frequency, phase, time)

► Fundamental need for Synchronization
  ▪ Coordinated Signal Processing requires phase alignment
  ▪ Time-stamping events (in geographically separated locations) requires time alignment
  ▪ Buffer read/write requires frequency alignment

► Transfer methods for frequency/time
  ▪ Transfer methods (one-way and two-way)
Time and Frequency

- A clock is a frequency device based on physics
  - Provides “ticks” at precise intervals (period);
    Frequency is reciprocal of period

- Electronic systems count “ticks” for time interval
  - “Time-Clock” provides the elapsed time from “start”
  - Granularity of time related to tick period
  - PLL…reduce tick interval;
    Divider…increase tick interval

- Time is a combination of a signal (event) and a label (time value) and is always considered in terms of elapsed time from an agreed-upon reference
Alignment in Frequency, Phase & Time

Aligning (or Synchronization) of two Time Clocks implies:

<table>
<thead>
<tr>
<th></th>
<th>Frequency B</th>
<th>Frequency A</th>
<th>Syntonization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase B</td>
<td>=</td>
<td>Phase A</td>
<td>Roll-over instant</td>
</tr>
<tr>
<td>Seconds B</td>
<td>=</td>
<td>Seconds A</td>
<td>Elapsed time equal</td>
</tr>
</tbody>
</table>

“Time”: Same formatting convention, time-zone, etc.

Clock is a frequency device, provides “ticks”
Electronic Systems count “ticks” for time interval
Time is a combination of a signal (event) and a label (time value)

Clock A

10MHz
Tick Generation

10,000,000 Counter

10,000,000 Counter

1Hz

Phase alignment (roll-over coincident)

Time alignment (equal # of seconds)

Seconds Counter

Seconds Counter

“Time”

Time alignment (“local time”)

Time alignment (UTC)

Difference = time-error

Clock B
Clock Metrics

**Time Error**
- Clock signals are (almost) periodic (nominal period ~ T)
- Time Error (Phase Error): Edge does not line up – phase error (expressed in time units)
- Time Error is the basis for all other metrics

**Time Interval Error (TIE)**
- Consider an interval of interest
- Start: “n”; Stop: “m”
- Duration measured by ideal clock (“truth”): \((m - n) \cdot T_S\)
- Error in measurement of same interval by clock being analyzed: \(TIE(m, n) = x(m) - x(n)\)

---

**MTIE and TDEV**

**Maximum Time Interval Error (MTIE):** A measure of peak-to-peak excursion expected within a given interval, \(\tau\) (\(\tau\) is a parameter). The observation interval is scanned with a moving window of duration \(\tau\) and \(MTIE(\tau)\) is the maximum excursion. MTIE is a useful indicator of the size of buffers and for predicting buffer overflows and underflows.

**Time Deviation (TDEV):** A measure of stability expected over a given observation interval, \(\tau\) (\(\tau\) is a parameter). TDEV provides guidance on the noise process type.
Accuracy and Stability

► **Accuracy**: Maximum (freq., phase or time) error over the entire life of the clock

► **Stability**: (Frequency, phase or time) change over a given observation time interval

► **Stability** is expressed with some statistical dispersion metric as a function of observation interval (e.g. ADEV, TDEV, MTIE, etc.)

► All metrics are computed on the *time-error* sequence

Samples of measurements of time-error or frequency offset
Fundamentals of Timing and Synchronization

► Basic Principles
  ▪ Time and Frequency
  ▪ Alignment (frequency, phase, time)

► Fundamental need for Synchronization
  ▪ Coordinated Signal Processing requires phase alignment
  ▪ Time-stamping events (in geographically separated locations) requires time alignment
  ▪ Buffer read/write requires frequency alignment

► Transfer methods for frequency/time
  ▪ Transfer methods (one-way and two-way)
Fundamental Need for Synchronization: Signal Processing

- Combining signals from different sources necessitates that the signals be in proper “phase”
  - Example: Interference cancellation involves subtracting the “known” interference from the received signal (e.g. EICIC, echo cancellation)
  - Analysis is application specific

- In interference cancellation, the received signal, \( y(t) \), contains an interfering signal, \( x(t) \), which is “known”...imperfect representation of \( x(t) \) results in degraded performance that can be quantified in terms of signal-to-noise ratio (SNR):
  - Proper signal: \( x(t) \); *Synchronization* error manifests as a delay: \( x(t + \delta) \)
  - “Noise” resulting just from synchronization error is
    \[
    \epsilon(t) = x(t) - x(t + \delta)
    \]

- Synchronization error can be quantified in terms of *Signal-to-Noise Ratio* (SNR)
Fundamental Need for Synchronization: Signal Processing

- “Noise” resulting just from synchronization error of $\delta$ is
  \[ \epsilon(t) = x(t) - x(t + \delta) \]

SNR drops to ~25dB just due to 0.1 UI time error; impact increases with signal bandwidth.

Signal Processing requires good synchronization.
Fundamental Need for Synchronization – More

► Time-Stamping Events
- Required if events occur “simultaneously” in separate equipment/locations
- Ordering of events established by time-stamping using a common clock (e.g. traceable to UTC or TAI or GPS, etc.).
- Requires end-point synchronization to this common clock.
- Many examples (distributed database, shared documents, stock trades, sensor fusion, multi-player gaming, etc., etc.)

► How can an action or event be verified or validated?
- Time-stamp using a common clock (usually UTC)
- Important in Blockchains, crypto-currency, etc.
- Important for stock market to chronologically order trading activities

► Synchronous multiplexing (“TDM”)
- Lack of synchronization (syntonization) results in buffer overflow/underflow events (aka slips)
Fundamentals of Timing and Synchronization

► Basic Principles
  ▪ Time and Frequency
  ▪ Alignment (frequency, phase, time)
► Fundamental need for Synchronization
  ▪ Coordinated Signal Processing requires phase alignment
  ▪ Time-stamping events (in geographically separated locations) requires time alignment
  ▪ Buffer read/write requires frequency alignment
► Transfer methods for frequency/time
  ▪ Transfer methods (one-way and two-way)
Transfer of frequency – *Timing Signal (one-way)*

- A timing signal is a signal that inherently includes the clock properties of the source, allowing the destination to extract a timing reference.
- Using this timing reference the destination can construct a (near) replica of the source clock.
- Example: the transmit waveform used to deliver digital information can provide a frequency reference.
Transfer of Time (e.g. Precision Time Protocol: IEEE 1588™)

- Transfer of time and/or phase requires two-way exchange to determine round-trip delay
- Utilizes time-stamped packets to provide a timing reference
- Transfer quality affected by variable transmission delay and asymmetry
- PTP (aka IEEE 1588™):
  - Master sends `Sync_Message` (with $T_1$)
  - Slave time-stamps arrival ($T_2$)
  - Slave sends `Delay_Request`; time-stamps departure ($T_3$)
  - Master time-stamps arrival ($T_4$)
  - Master sends `Delay_Response` (with $T_4$)

1. Round Trip Delay (RTD) = ($T_4 - T_1$) + ($T_2 - T_3$)
2. Assuming symmetric delays upstream and downstream, One Way Delay (OWD) = (1/2) RTD
3. Slave Offset from Master, OFM = (1/2)($T_4 + T_1 - (T_3 + T_2)$)
Fundamentals of Timing and Synchronization

Topics Addressed

► Basic Principles
  ▪ Time and Frequency
  ▪ Alignment (frequency, phase, time)

► Fundamental need for Synchronization
  ▪ Coordinated Signal Processing requires phase alignment
  ▪ Time-stamping events (in geographically separated locations) requires time alignment
  ▪ Buffer read/write requires frequency alignment

► Transfer methods for frequency/time
  ▪ Transfer methods (one-way and two-way)
Thank You
Questions, comments, suggestions?
kshenoi@qulsar.com