NetDev 0x15

Timing and Synchronization

Maciej Machnikowski

intel®

Legal Disclaimers

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS.

Intel may make changes to specifications and product descriptions at any time, without notice.

All products, dates, and figures specified are preliminary based on current expectations, and are subject to change without notice.

Intel processors, chipsets, and desktop boards may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Alta, Red Rock Canyon, Seacliff Trail, and other code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees, and other third parties are not authorized by Intel to use code names in advertising, promotion, or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Go to: www.intel.com/products/processor_number.

Intel product plans in this presentation do not constitute Intel plan of record product road maps. Please contact your Intel representative to obtain Intel's current plan of record product roadmaps.

Intel, the Intel logo, the Intel. Experience What's Inside logo, and Intel. Experience What's Inside are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

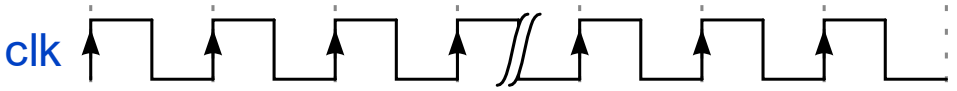
*Other names and brands may be claimed as the property of others.

Intel does not control or audit the design or implementation of third-party benchmark data or websites referenced in this document. Intel encourages all of its customers to visit the referenced websites or others where similar performance benchmark data are reported and confirm whether the referenced benchmark data are accurate and reflect performance of systems available for purchase.

Agenda

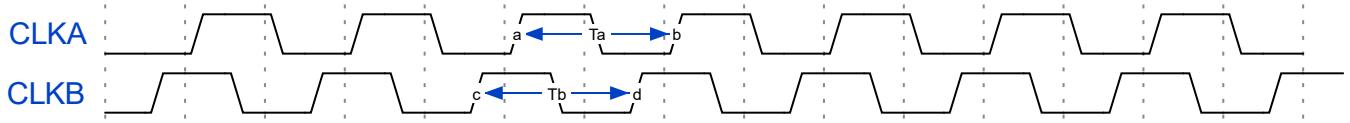
- Definitions
- Packet-based time and phase Distribution
- Physical Layer Frequency Distribution
  - SyncE introduction
  - ESMC messaging
  - Frequency synchronization states
- Traceability
- Summary

What is clock?

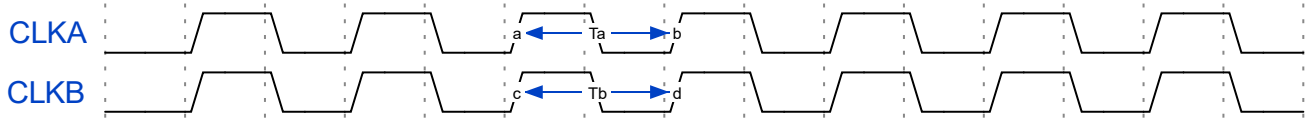


Three things to synchronize

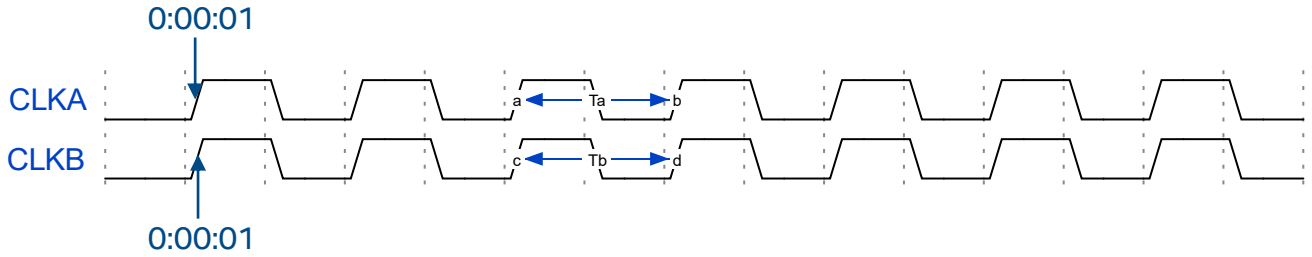
- Frequency



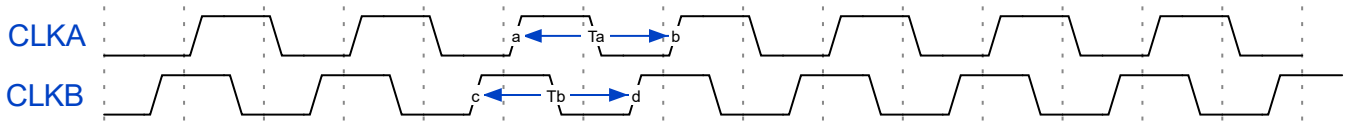
- Phase



- Time (of day)

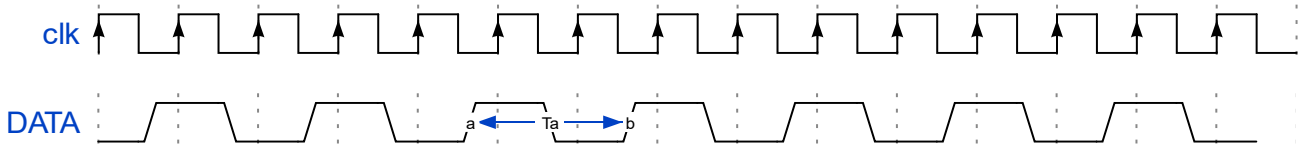


Frequency “synchronization”

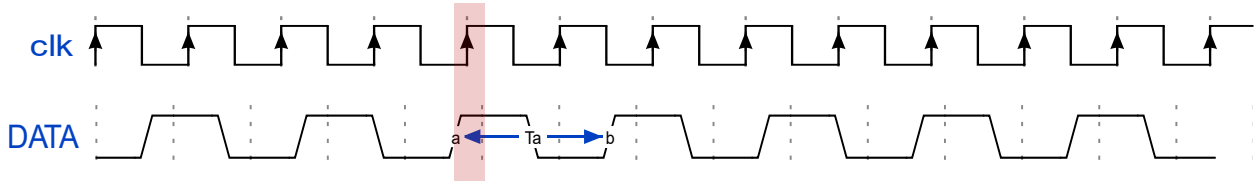


- Syntonization
- Events happen at the same rate
- Signals are sampled correctly:

In sync:



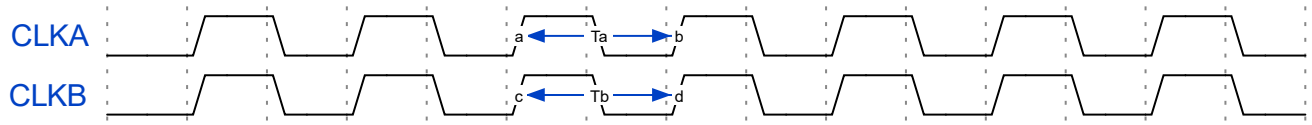
out of sync:



Frequency synchronization - why

- TX and RX need to run at the same frequency
- 2G/3G/4G –required for RAN (Radio Access Network)
  - FDD (Frequency-division duplexing)
- Backhaul needs to run at the same rate as the RU / BTS

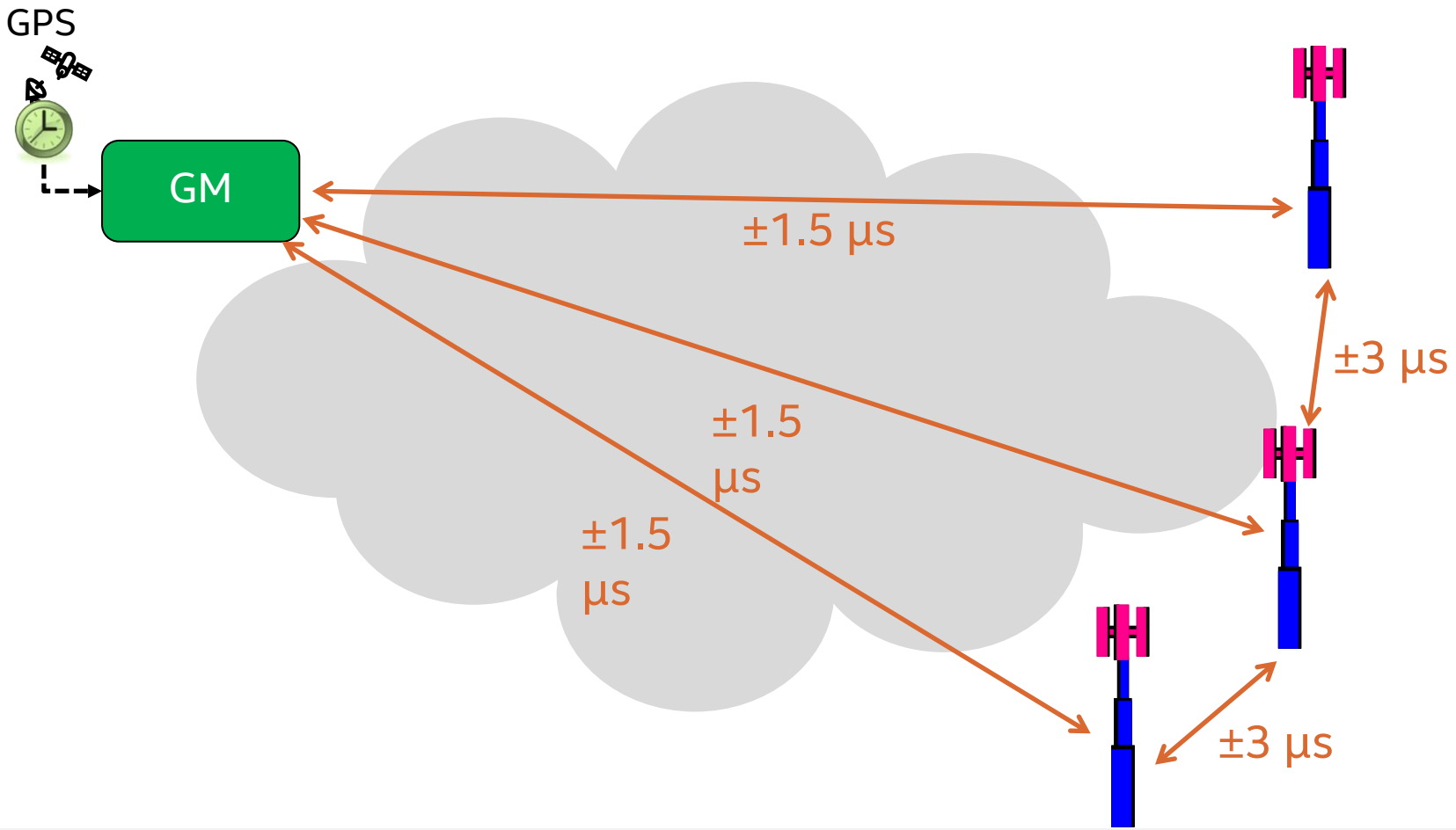
Phase synchronization



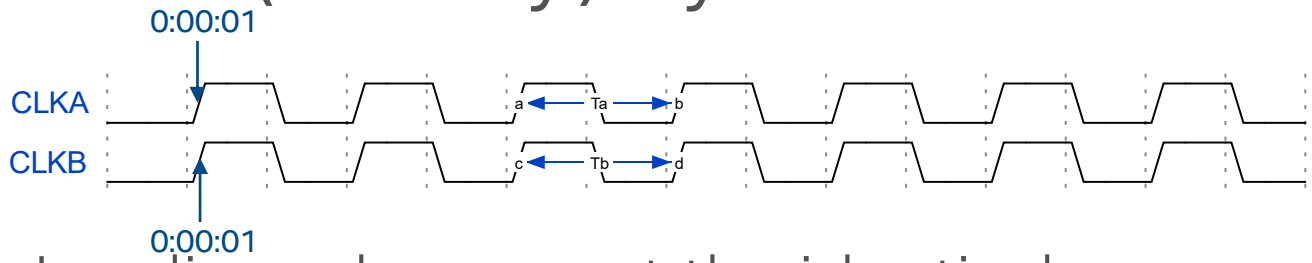
- Leading edges are at the identical moments
- Events start at the exact same point in time
- And at the same rate

Phase synchronization - why

- 3GPP ±3 µs between BS (WCDMA and LTE TDD)



Time (of Day) synchronization

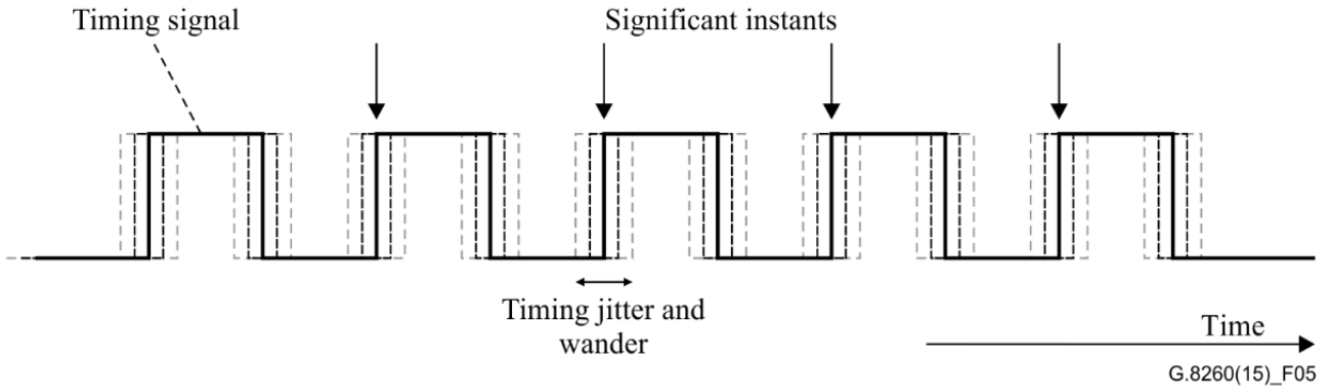


- Leading edges are at the identical moment and indicate the same time
- Each period is marked and dated
- And at the same rate

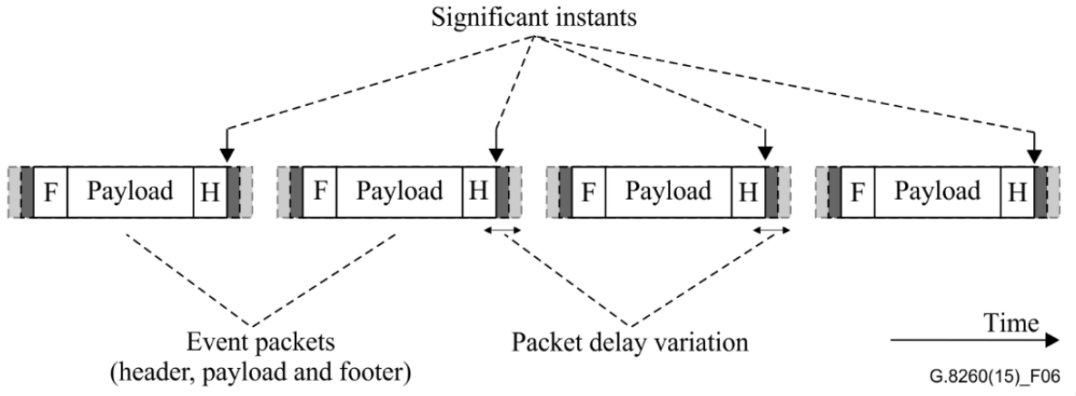
- Billing
- Quality of Service (Y.1731)
- Logging
- Synchronizing events across the network

Timing signals (G.8260)

- Physical layer timing signals:

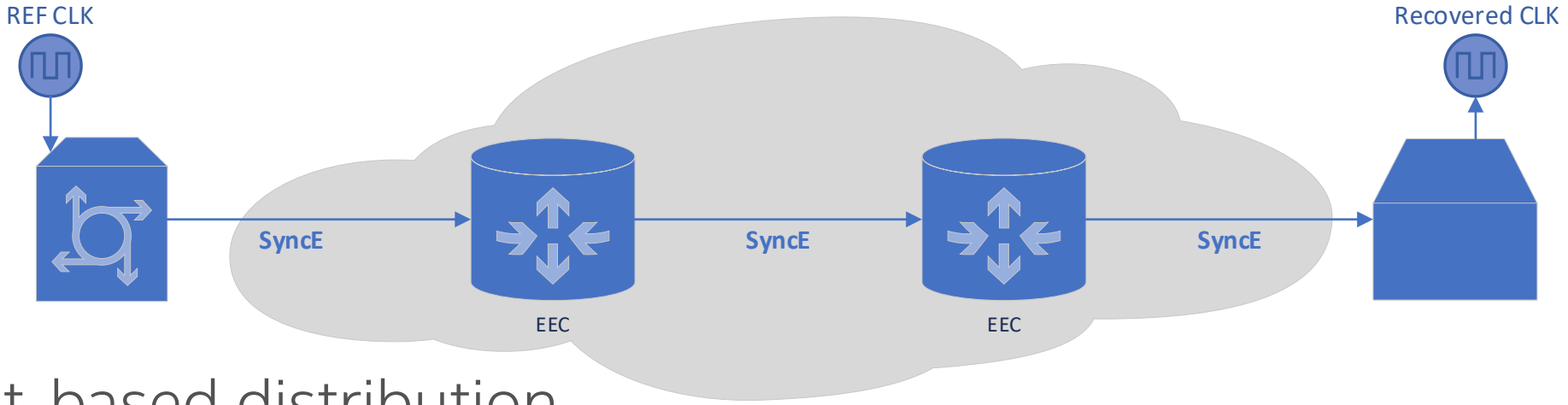


- Packet timing signal:

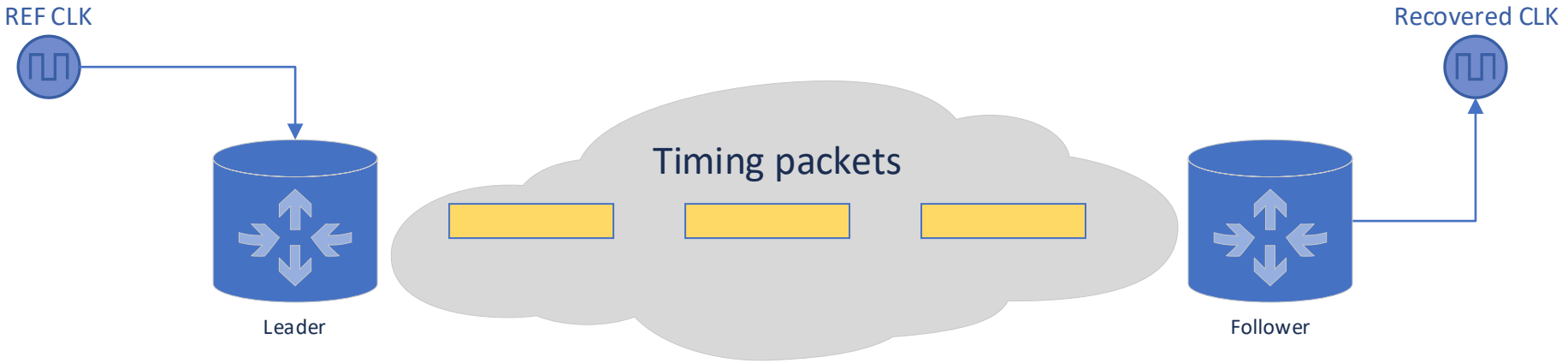


Transmission of frequency

- Physical Layer distribution:



- Packet-based distribution



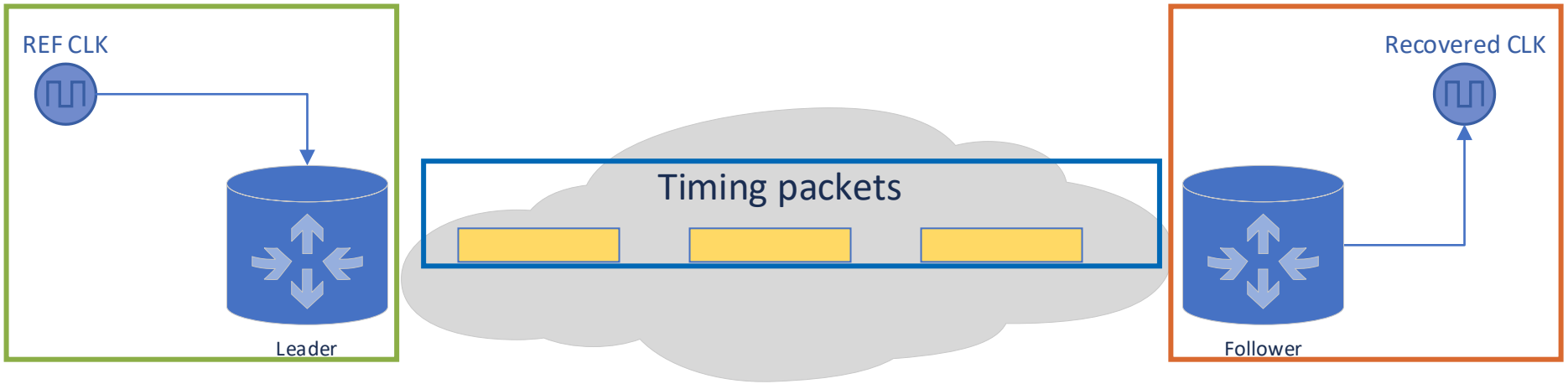
Summary

- There are three aspects of synchronization:
  - Frequency
  - Phase
  - Time
- Frequency can be transmitted in different ways
  - Physical layer
  - Packet-based
- More areas needs synchronization:
  - LTE-FDD
  - LTE-TDD

Packet-based time and phase Distribution

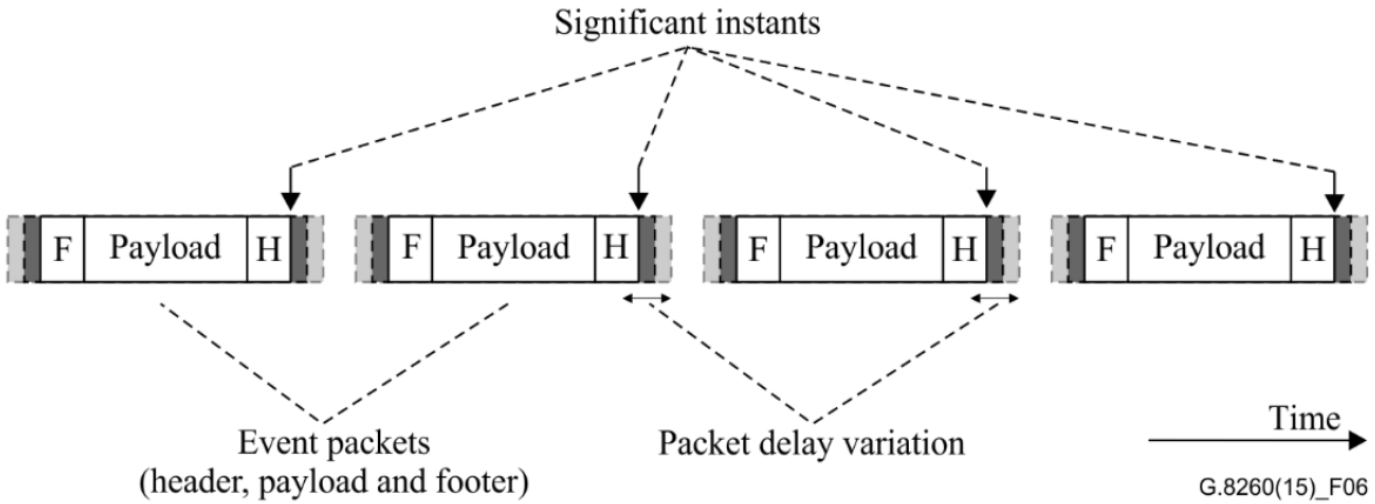
IEEE 1588

Frequency transfer



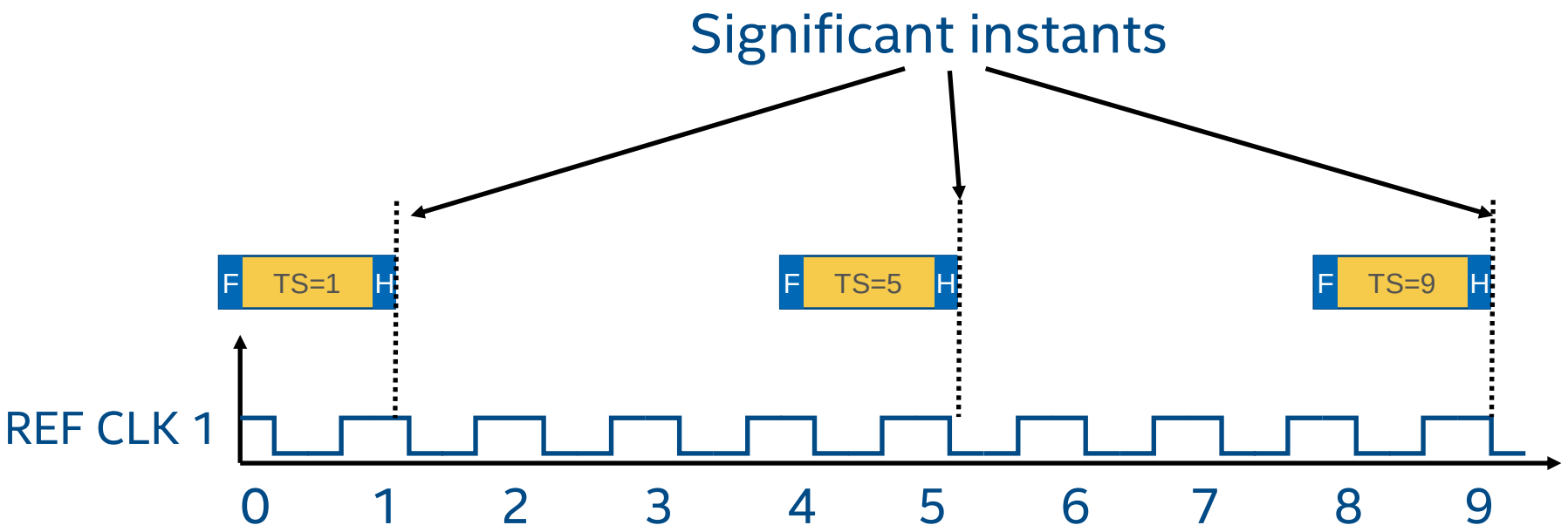
- Generation: create packets from physical reference source
- Transfer: packet transmission over packet network
- Recovery: regenerate the physical frequency from the received packets

Packet-based distribution

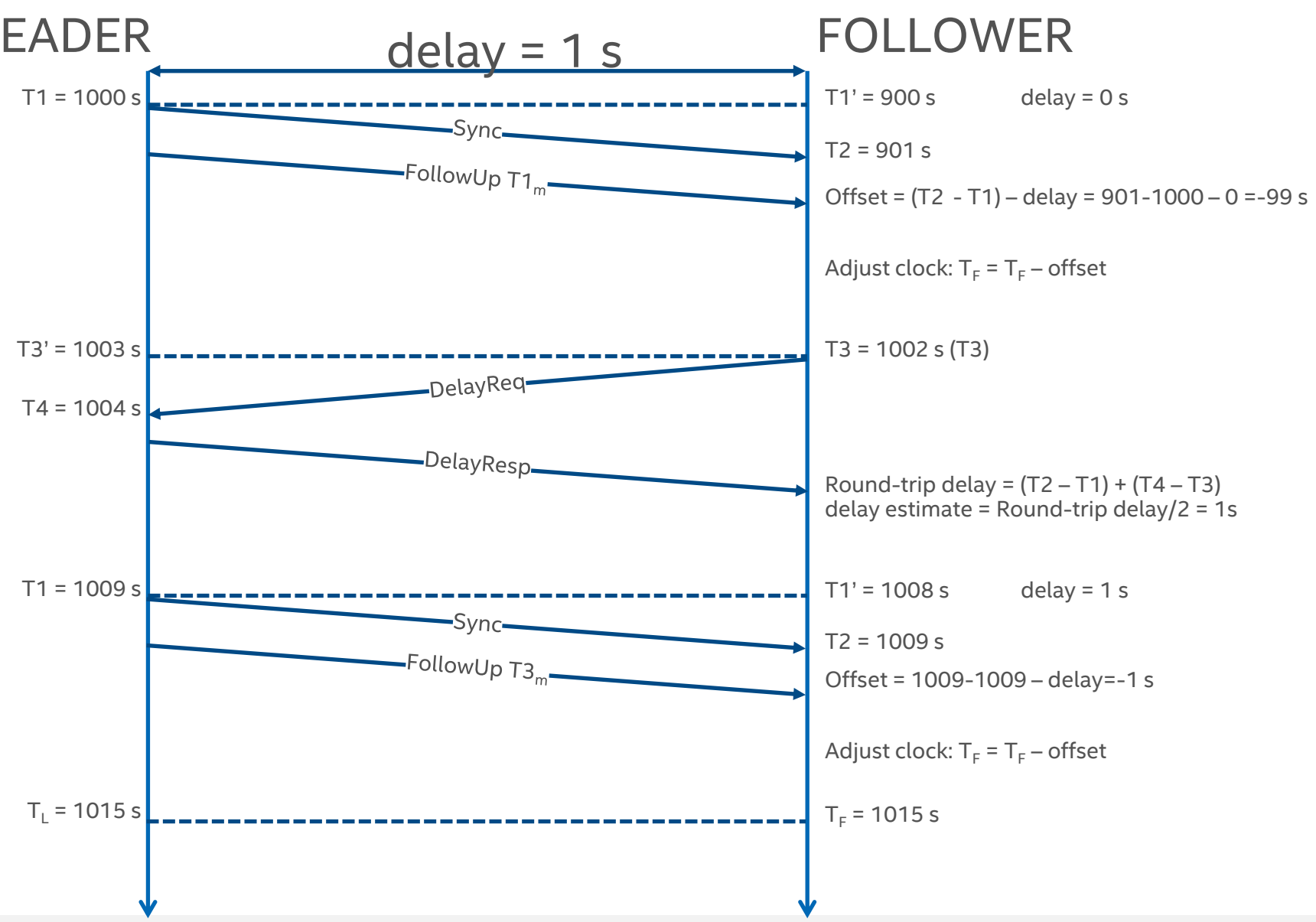


- Packets sent at a specified rate (typically 1-128 pps)
- Timing signals can be periodic (CES) or aperiodic (NTP, PTP)
- Can carry additional information:
  - Timestamps defining ideal position of significant instant in the defined time scale
  - Quality of the reference signal

Recovering the frequency from packets



IEEE 1588

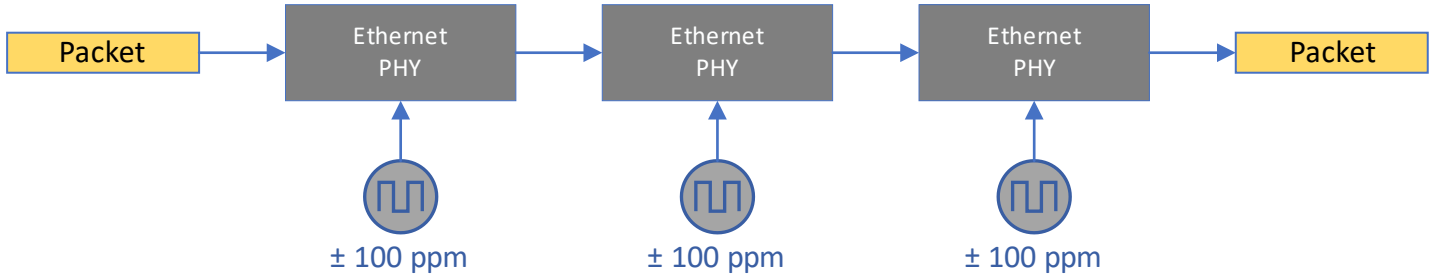

LEADER
FOLLOWER
delay = 1 s
T1 = 1000 s
T1' = 900 s
delay = 0 s
Sync
T2 = 901 s
FollowUp T1m
Offset = (T2 - T1) – delay = 901-1000 – 0 =-99 s
Adjust clock: TF = TF – offset
T3' = 1003 s
T3 = 1002 s (T3)
DelayReq
T4 = 1004 s
DelayResp
Round-trip delay = (T2 – T1) + (T4 – T3)
delay estimate = Round-trip delay/2 = 1s
T1 = 1009 s
T1' = 1008 s
delay = 1 s
Sync
T2 = 1009 s
FollowUp T3m
Offset = 1009-1009 – delay=-1 s
Adjust clock: TF = TF – offset
TL = 1015 s
TF = 1015 s

Physical Layer Frequency Distribution

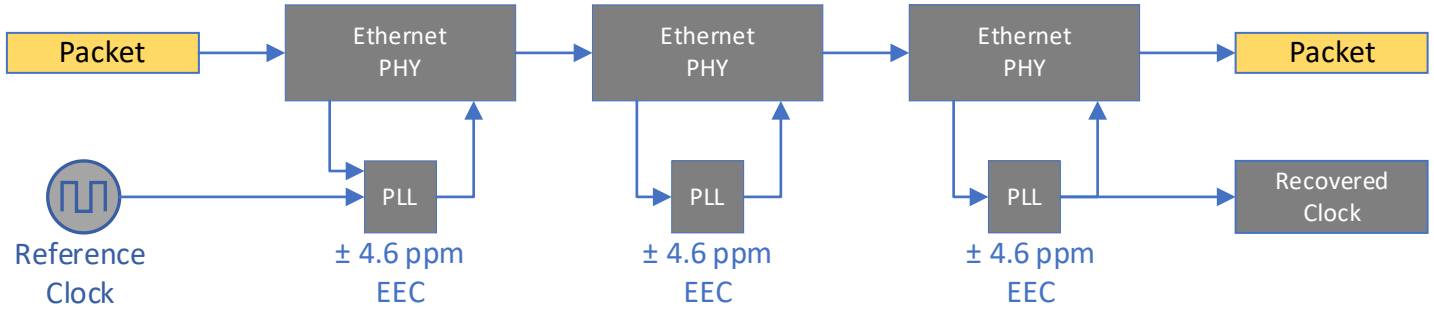
Synchronous Ethernet (SyncE) Introduction

SyncE

- Use Ethernet fabric to transfer frequency over physical layer
- Use high stability oscillators to generate line frequency:
  - Traditional 802.1 Ethernet: ±100 ppm



  - SyncE: ±4.6 ppm in free-run mode

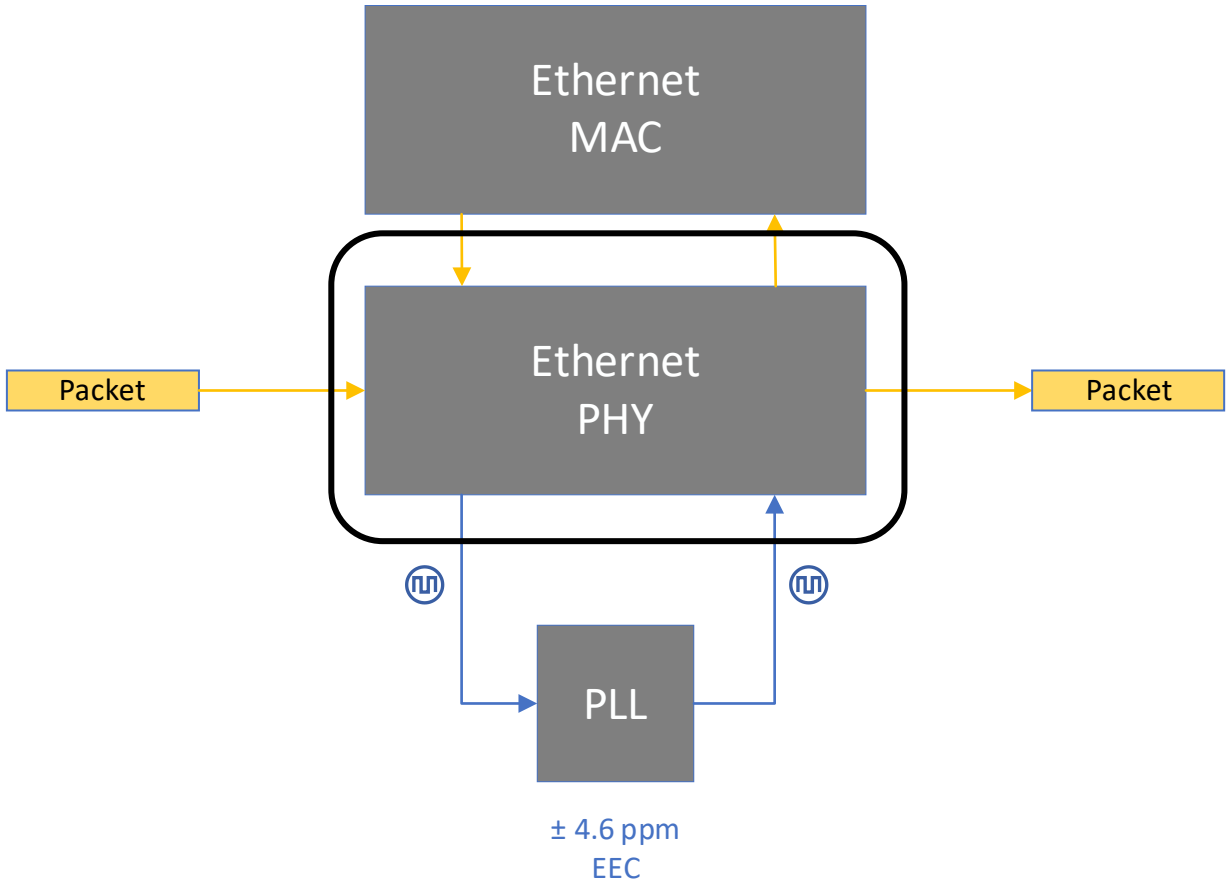


SyncE

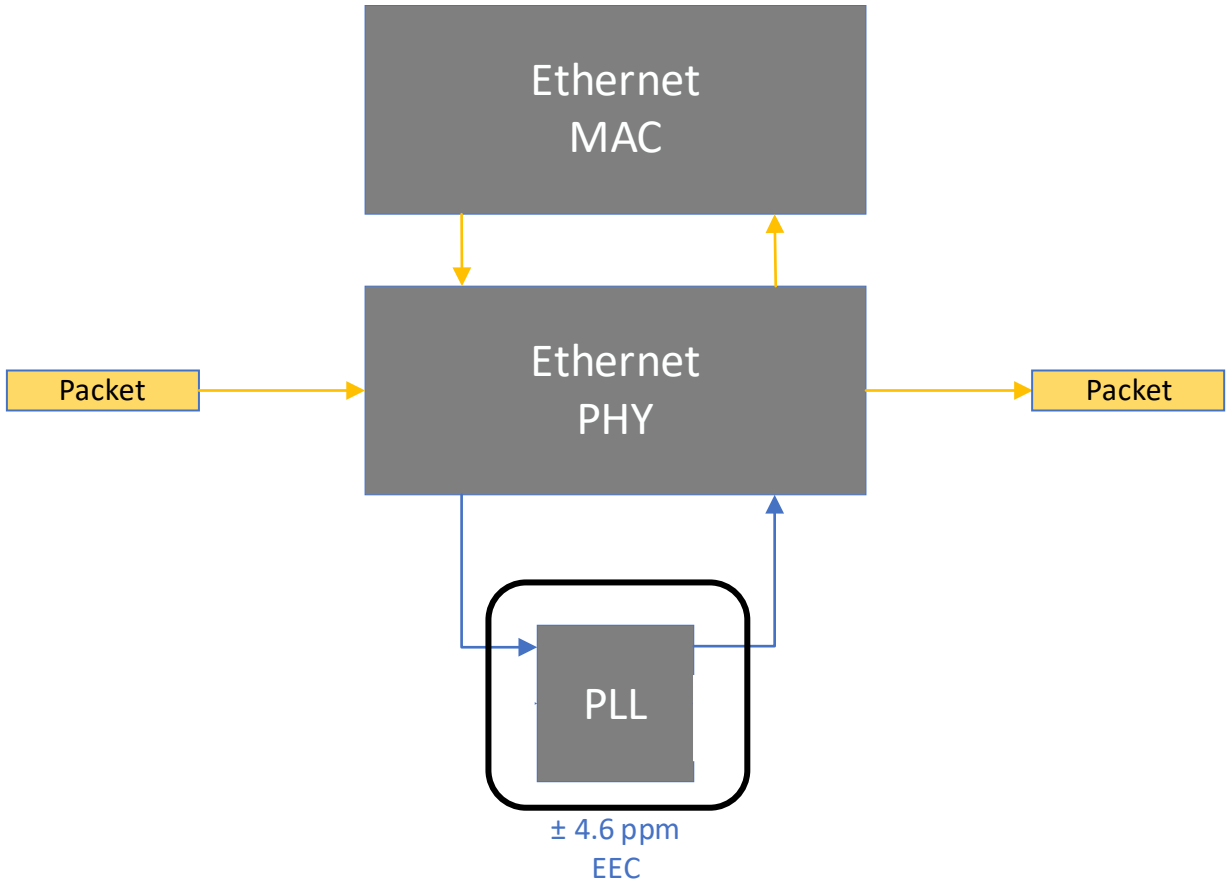
- Every EEC locks itself to the incoming frequency
- Clocks are locked and traceable to the source
- When locked can achieve ±10 ppt
- No impact to the packet layer

- ITU standards:
  - G.8261 Timing and synchronization aspects in packet networks
  - G.8262 Timing characteristics of a synchronous equipment slave clock
  - G.8264 Distribution of timing information through packet networks

SyncE – clock recovery

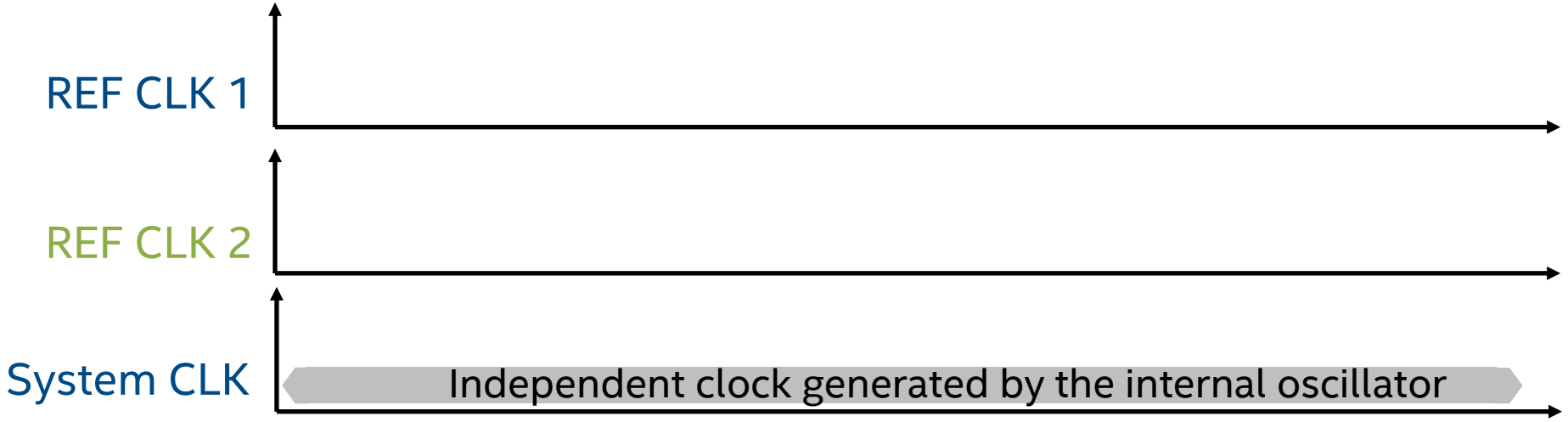


SyncE – PLL



Frequency synchronization states

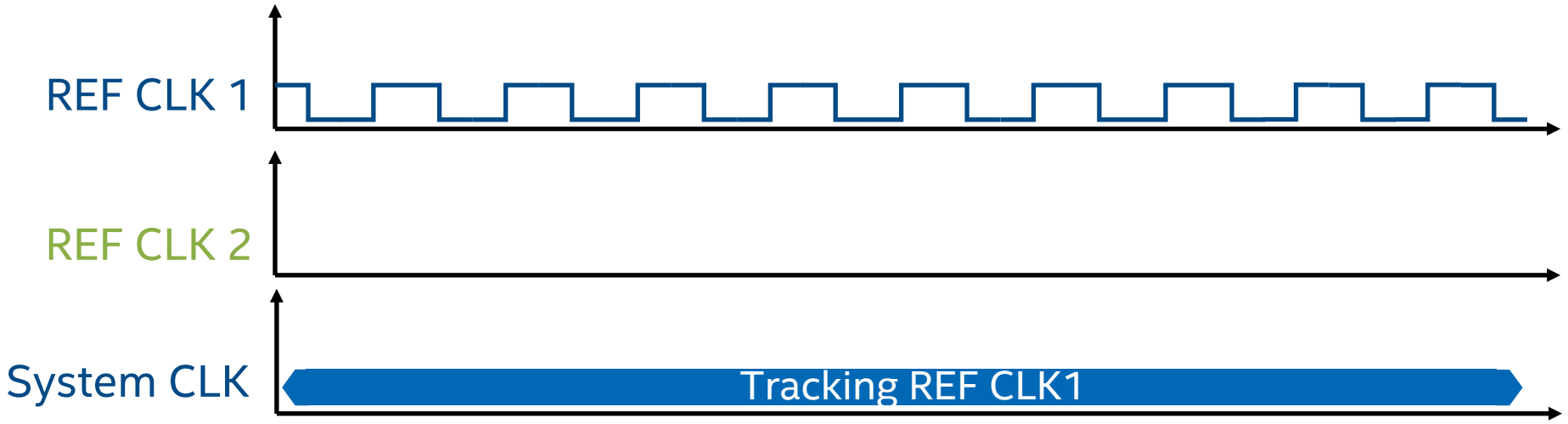
- Free-running



- Never synchronized to any reference
- Holdover time expired

Frequency synchronization states

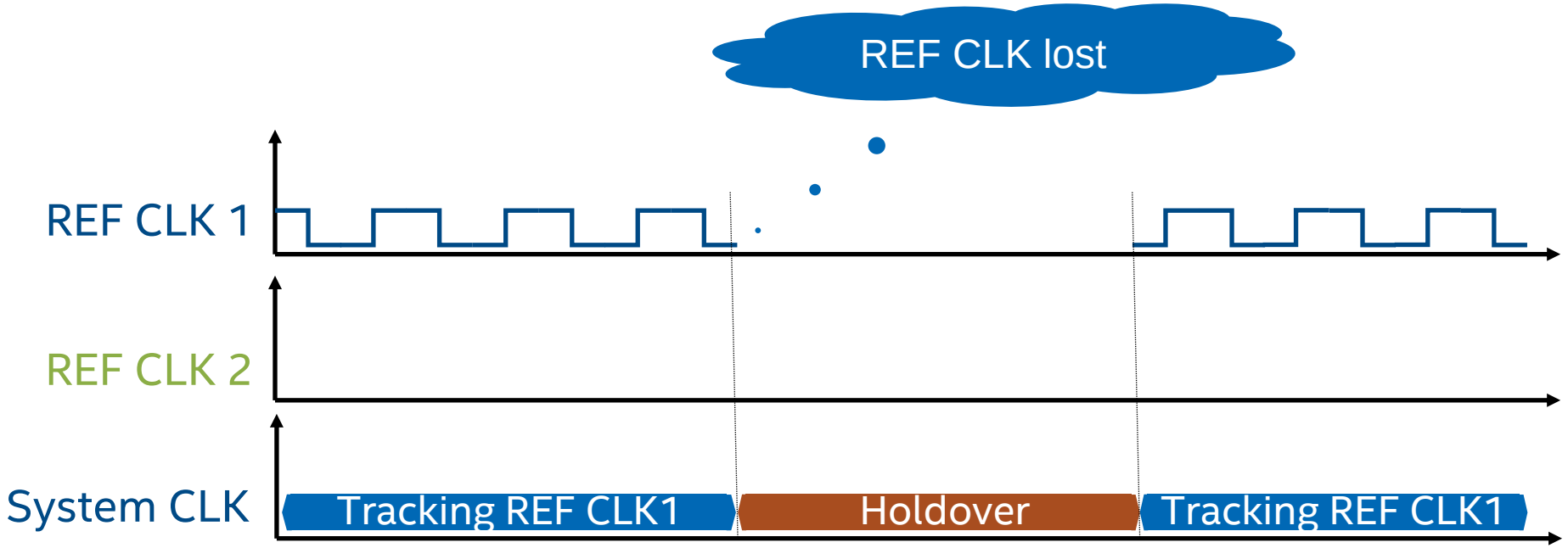
- Locked



- Synchronized to a valid reference clock source
- Tracked clock can be selected

Frequency synchronization states

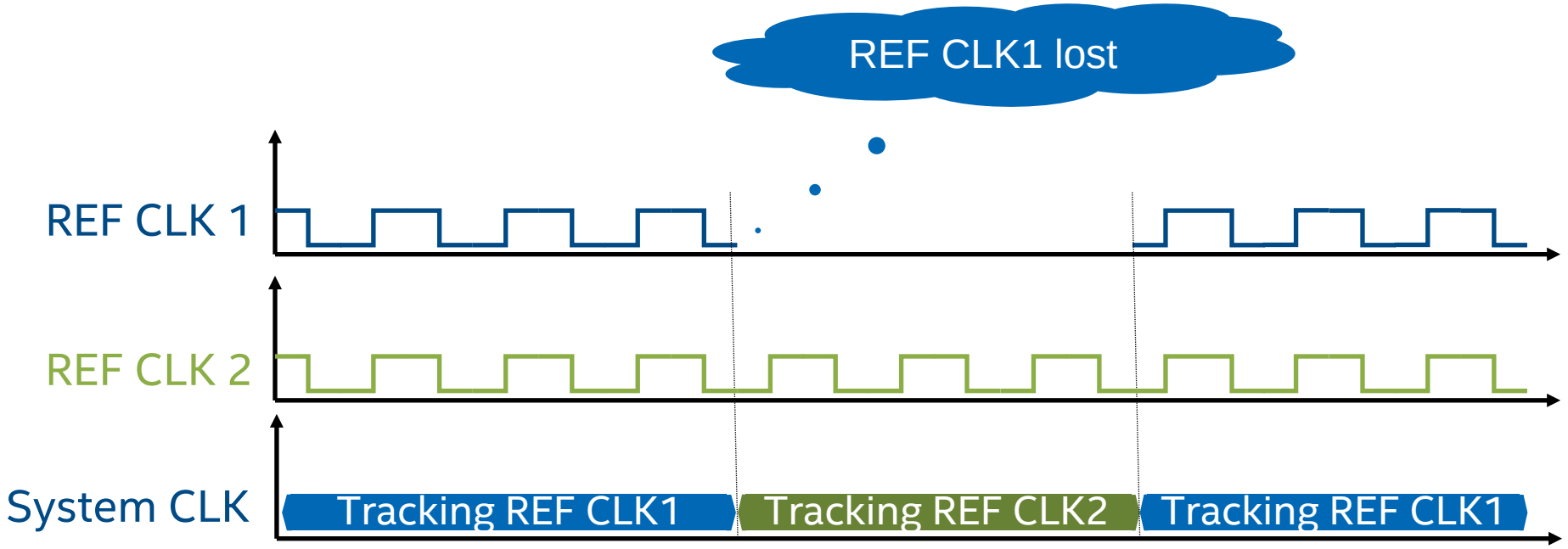
- Holdover



- Was synchronized to a valid reference clock source but now is not

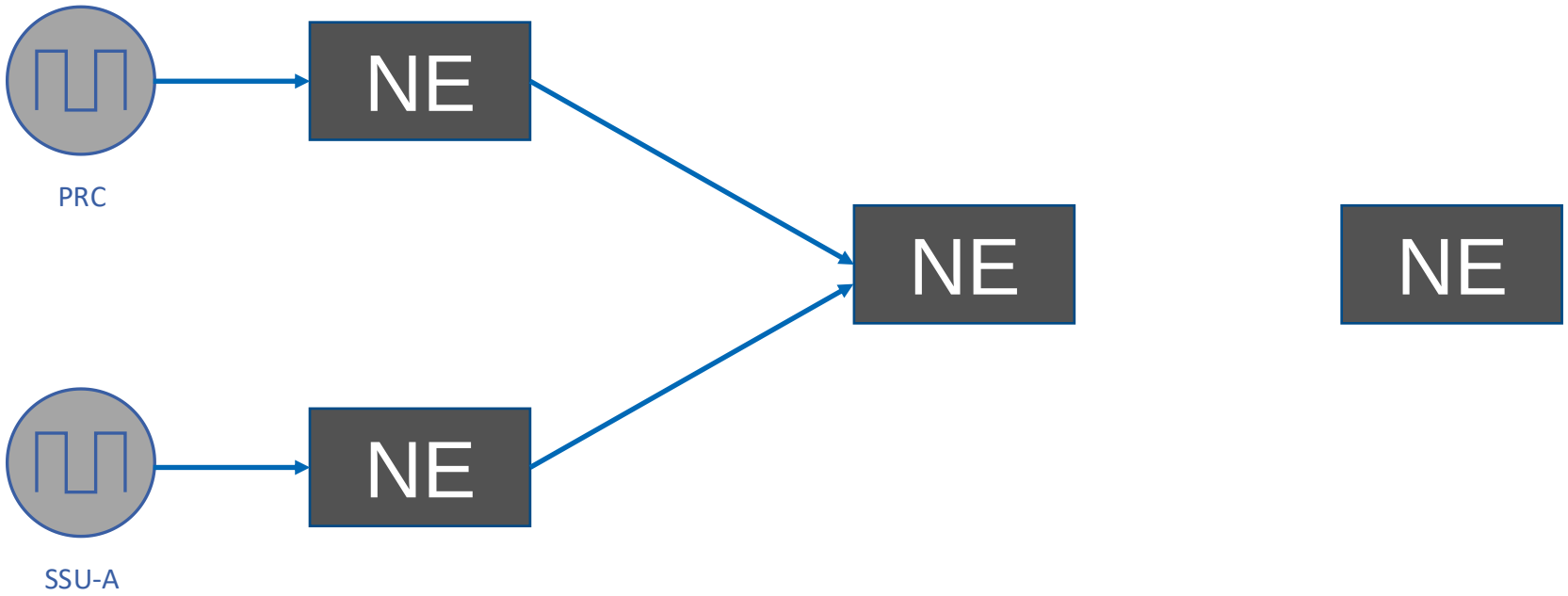
Frequency synchronization states

- Automatic reference mode



- Automatic selection of the best reference

Determining the reference

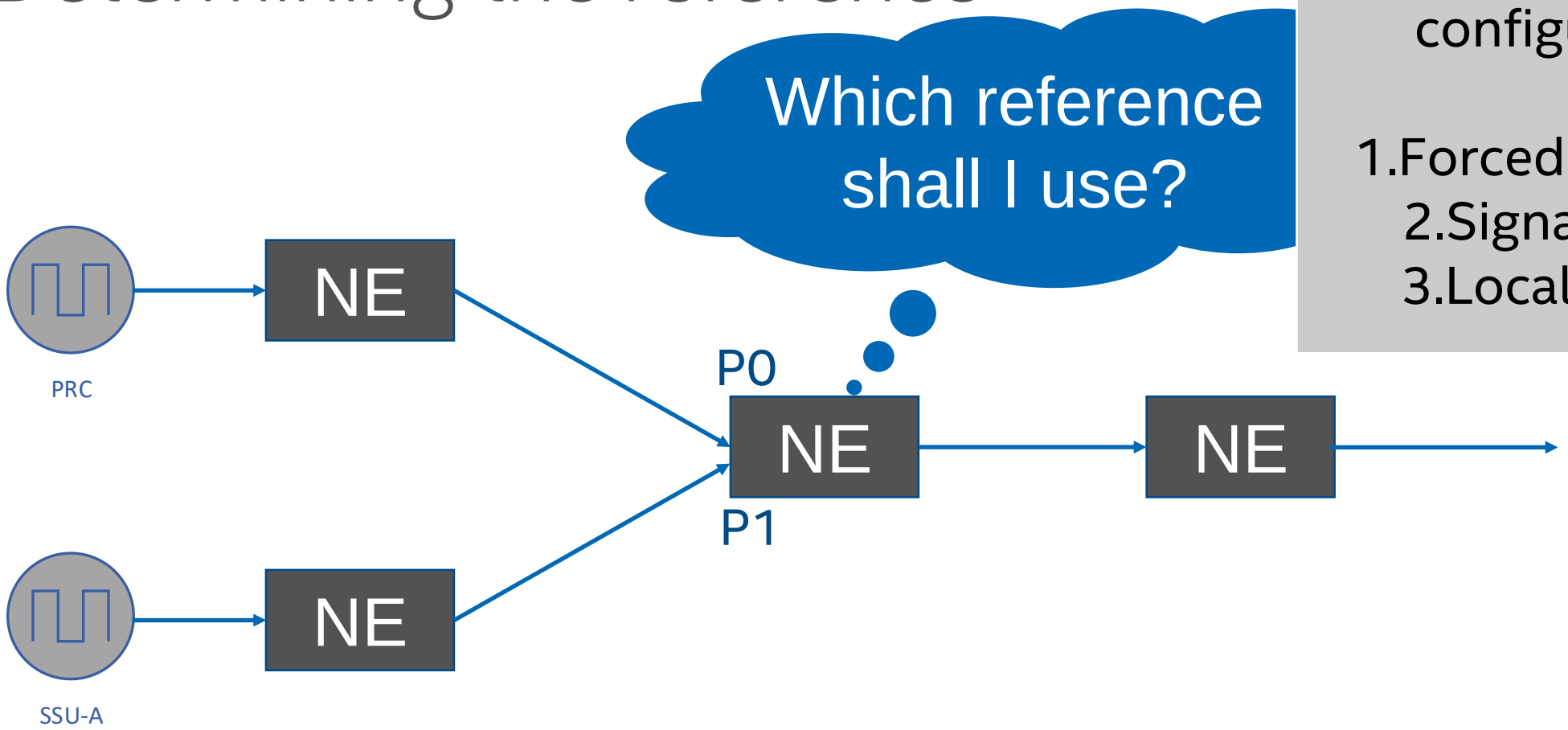


Determining the reference

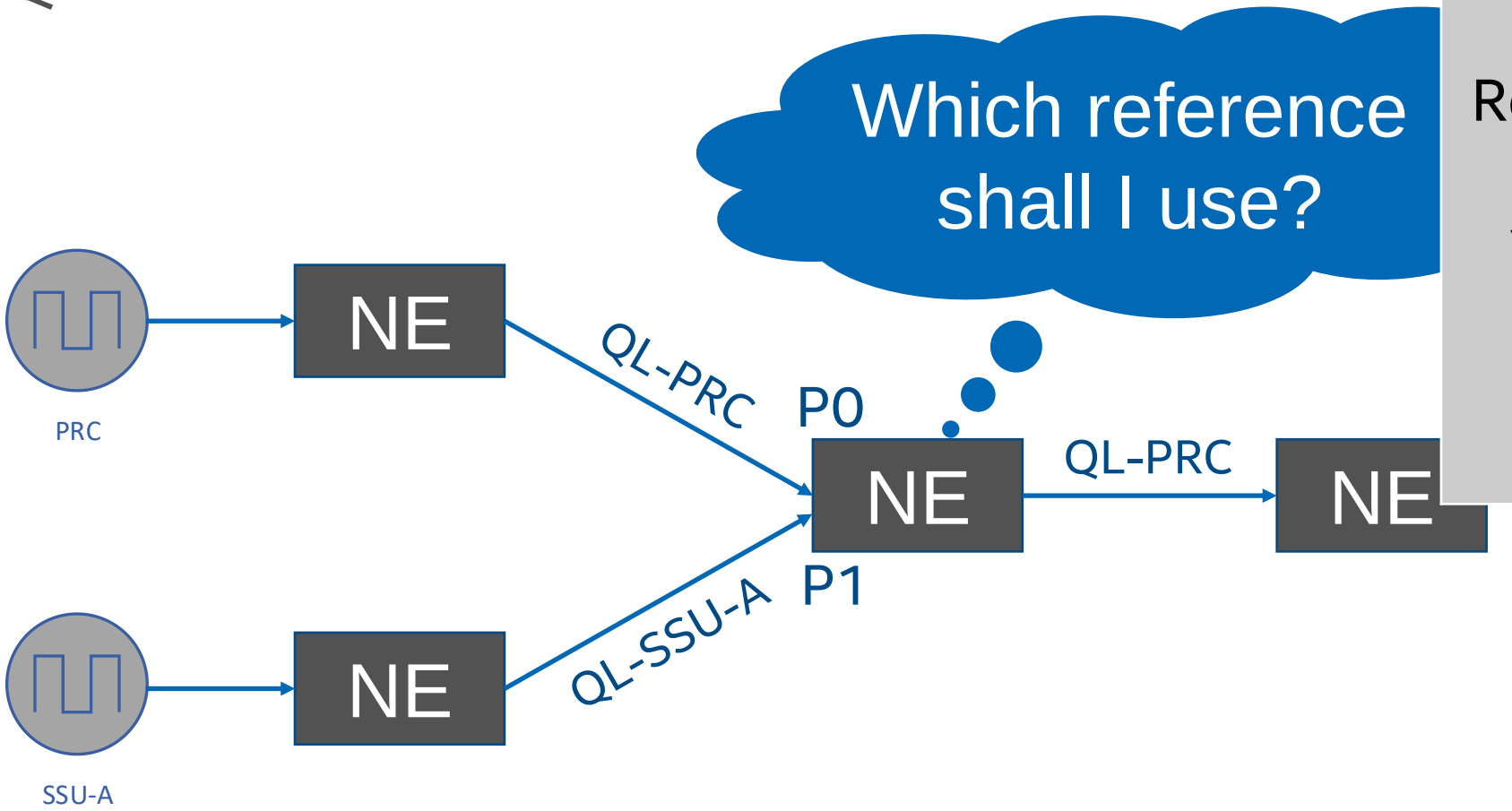
QL-Disabled

Fixed provisioned configuration:

1.Forced reference
2.Signal failure
3.Local priority



QL-Enabled



QL-Enabled

Dynamic configuration:
Reference chosen by:

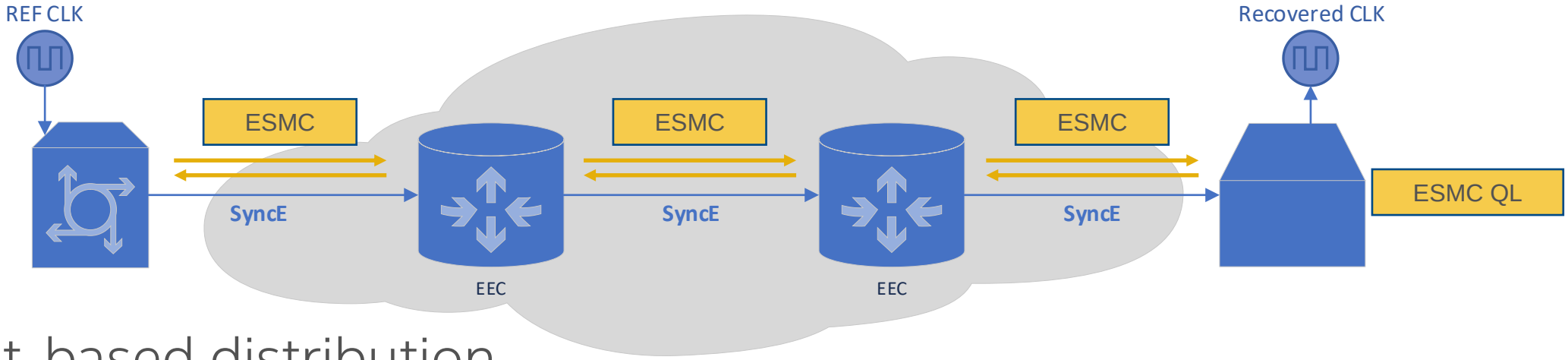
1.Forced reference
2.Quality level
3.Signal failure
4.Local priority

Traceability

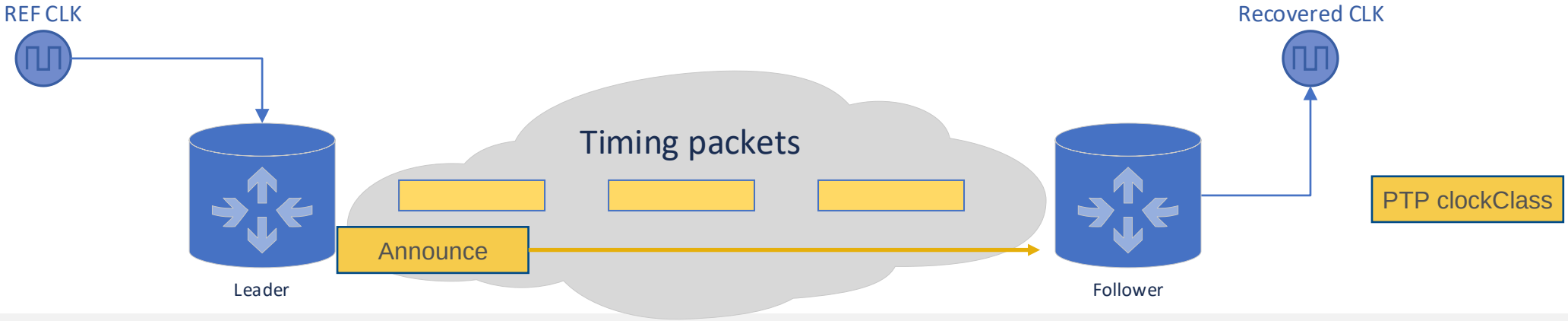
Quality of clock

Frequency source traceability

- Physical Layer distribution:



- Packet-based distribution



ESMC – G.8264

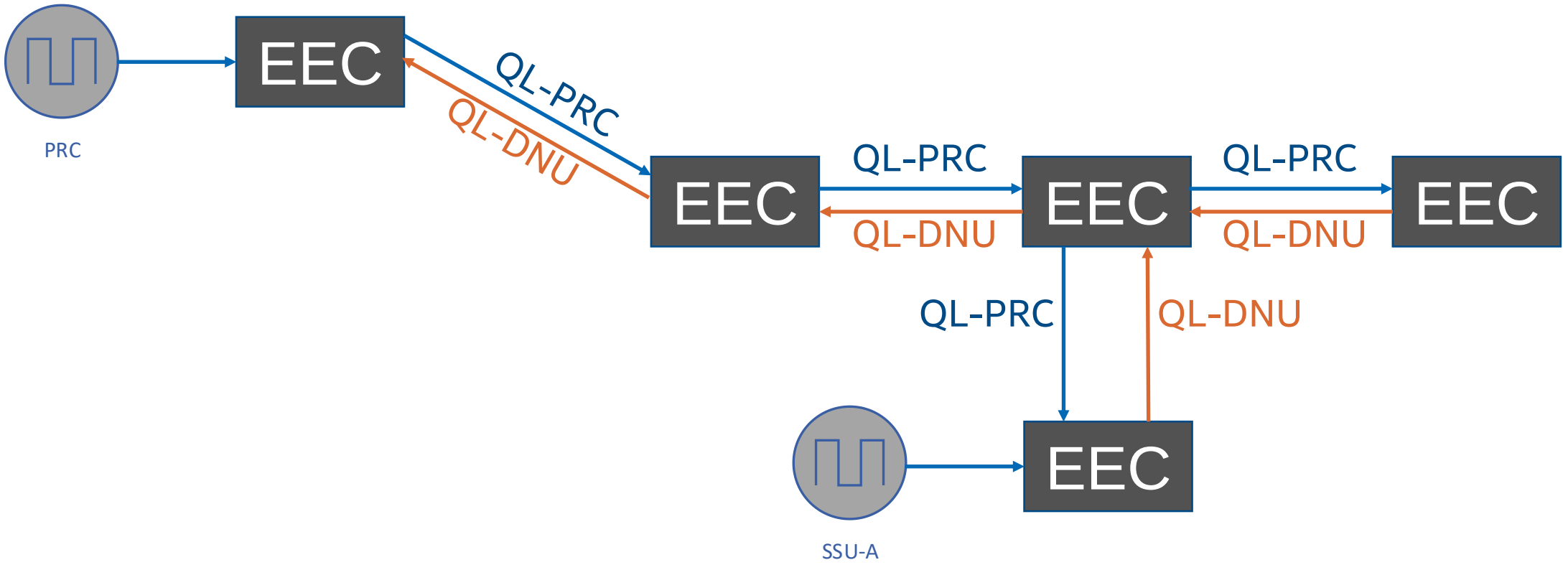
- Ethernet Synchronization Messaging Channel
- Based on slow protocol
- Transmit Synchronization Status Message (SSM) QL
- Two types of messages:
  - Heartbeat – sent every 1s
    - No Heartbeat for 5 seconds = ESMC failed
  - Event message – sent when QL value changes
- Data mapped into a TLV

ESMC – G.8264

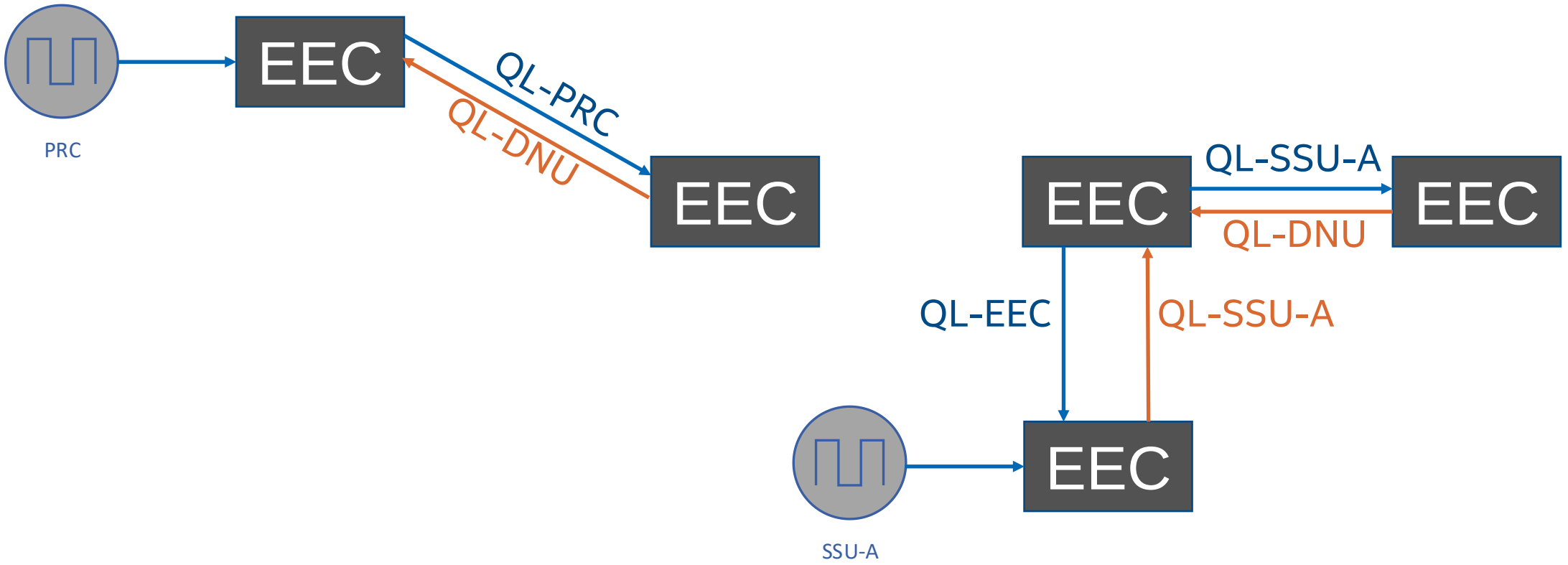
Field	Value
Dest. MAC Address	01:80:C2:00:00:02 (Slow-Protocols)
Source MAC Address	XX:XX:XX:XX:XX:XX (Source-MAC)
EtherType	0x8809 (Slow Protocols)
Slow Protocols subtype	0x0A (Organization Specific Slow Protocol)
ITU-OUI	00:19:A7 (ITU-T)
ITU-Subtype	0x0001 ESMC
Version + Flags	Version
	Event type (0: Information; 1: Event)
TLV(s)	
Padding	
Frame Check Sequence	

Field		Octet
Type (0x01)		1
Length(0x00-0x04)		2
		3
Reserved	SSM Code	1

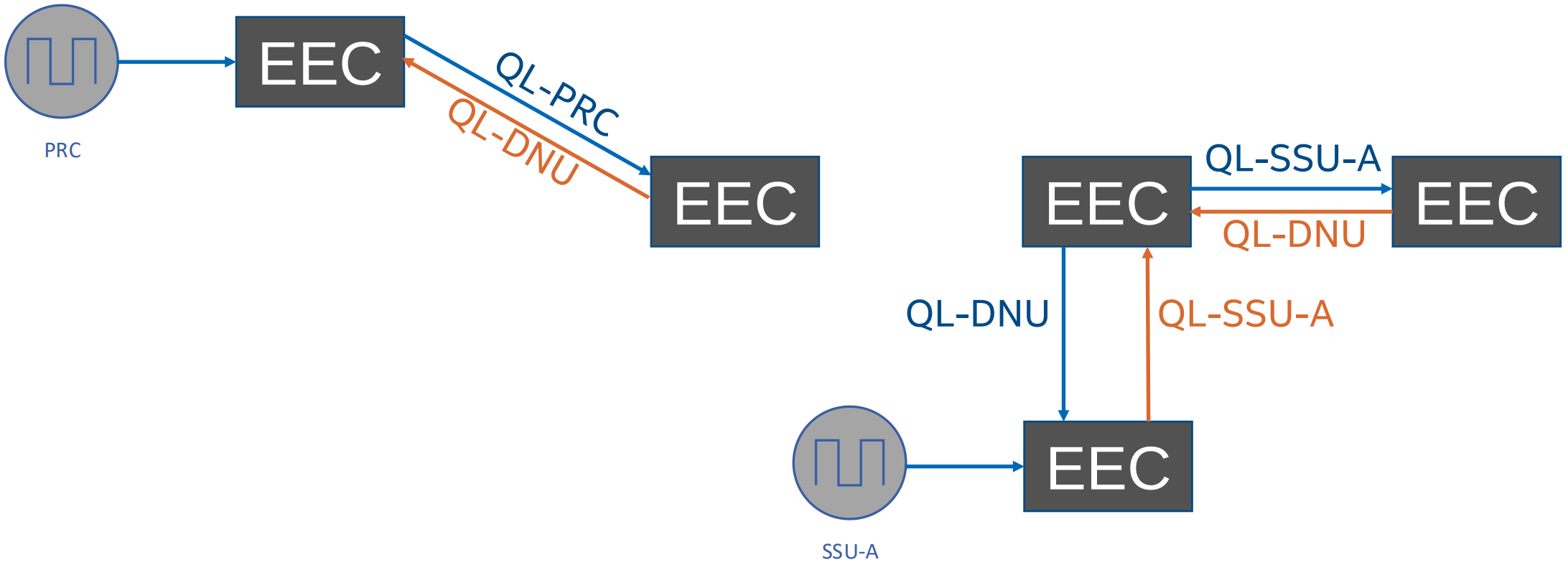
PRC traceable



Failure – transient mode



Failure – Stable mode



SyncE recap

- SyncE transfer frequency over physical layer
- Doesn't rely on packets
- Can work in QL-Enabled or QL-Disabled mode
- Use ESMC messages to trace clock class and events

Summary

Technology comparison

Parameter	GPS	PTP	SyncE
Frequency synchronization	Yes	Yes	Yes
Phase synchronization	Yes	Yes	No
Time of Day Synchronization	Yes	Yes	No
Frequency Accuracy	Very high	Medium/High	High
Time and Phase accuracy	Very high	High	No
Holdover	No	Yes	Yes
Connection	RF	Ethernet	Ethernet

Summary

- Frequency can be distributed across network using
  - Physical layer (SyncE)
  - Packet-based (IEEE-1588)
- Quality of frequency can be traced back to the source
- For the best results:
  - IEEE 1588: Phase and time of day
  - SyncE: Frequency
- IEEE 1588 is prone to PDV and delay asymmetry

Linux Kernel interfaces for SyncE

SyncE in Linux 5.11.2

```
[root@maciek-pc-1 linux-5.11.2]# grep --color -rn "\bSyncE\b"
Documentation/devicetree/bindings/mips/mscc.txt:48:configuration and status of PLL5, RCOMP, SyncE, SerDes configurations and
drivers/net/phy/mscc/mscc_main.c:2126:  .name           = "Microsemi GE VSC8502 SyncE",
drivers/net/phy/mscc/mscc_main.c:2150:  .name           = "Microsemi GE VSC8504 SyncE",
drivers/net/phy/mscc/mscc_main.c:2175:  .name           = "Microsemi GE VSC8514 SyncE",
drivers/net/phy/mscc/mscc_main.c:2246:  .name           = "Microsemi FE VSC8540 SyncE",
drivers/net/phy/mscc/mscc_main.c:2270:  .name           = "Microsemi VSC8541 SyncE",
drivers/net/phy/mscc/mscc_main.c:2294:  .name           = "Microsemi GE VSC8552 SyncE",
drivers/net/phy/mscc/mscc_main.c:2318:  .name           = "Microsemi GE VSC856X SyncE",
drivers/net/phy/mscc/mscc_main.c:2340:  .name           = "Microsemi GE VSC8572 SyncE",
drivers/net/phy/mscc/mscc_main.c:2365:  .name           = "Microsemi GE VSC8574 SyncE",
drivers/net/phy/mscc/mscc_main.c:2390:  .name           = "Microsemi GE VSC8575 SyncE",
drivers/net/phy/mscc/mscc_main.c:2413:  .name           = "Microsemi GE VSC8582 SyncE",
drivers/net/phy/mscc/mscc_main.c:2436:  .name           = "Microsemi GE VSC8584 SyncE",
[root@maciek-pc-1 linux-5.11.2]# grep -r "Synchronous Ethernet"
[root@maciek-pc-1 linux-5.11.2]# grep --color -rn "\bESMC\b"
[root@maciek-pc-1 linux-5.11.2]# grep --color -rn "\bQL\b"
arch/m68k/Kconfig.machine:126:    The Q40 is a Motorola 68040-based successor to the Sinclair QL
drivers/gpu/drm/radeon/radeon_agp.c:107:        /* VIA VT8363 Host Bridge / R200 QL [Radeon 8500] Needs AGPMode 2 (lp #141551) */
drivers/infiniband/hw/qedr/main.c:137:  return sysfs_emit(buf, "FastLinQ QL%x %s\n", dev->pdev->device,
drivers/infiniband/hw/qib/qib_iba7322.c:7379:           { 0x00, 0x90, 0x65 }, "FCBG410QB1C03-QL",
drivers/infiniband/hw/qib/qib_iba7322.c:7383:           { 0x00, 0x90, 0x65 }, "FCBG410QB1C30-QL",
drivers/net/ethernet/hisilicon/hns3/hns3_debugfs.c:358: dev_info(&h->pdev->dev, "support INT QL: %s\n",
drivers/net/ethernet/hisilicon/hns3/hns3_debugfs.c:390: dev_info(priv->dev, "MAX INT QL: %u\n", dev_specs->int_ql_max);
drivers/net/ethernet/hisilicon/hns3/hns3_enet.c:272:     * 3. QL (Interrupt Quantity Limiter)
[root@maciek-pc-1 linux-5.11.2]#
```

Required interfaces

- PHY clock redirection*
- Clock Generator state monitoring
- Clock Generator configuration

PHY clock redirection

- Scope: Per physical port
- Support multiple output pins
- Map physical port to the pin
- Allow to enable/disable clock
- May read the clock frequency

Clock Generator state monitoring

- Scope: device
- Return the state of the Clock Generator
- Return events on Clock Generator state change
  - loss of signal
  - holdover

Clock Generator configuration

- Scope: device
- Advanced clock generator configuration
  - Pin priority
  - Pin frequency
  - Additional features

Interface considerations

- IOCTL
  - Linuxptp uses it as a standard interface
- sysfs
  - Standard for ptp_clock
- devlink
  - Could be used to configure advanced features (pin priorities, expected frequency etc.)
  - Not supported by the linuxptp stack

Proposed interfaces

- PHY clock redirection*
  - Add IOCTL and sysfs file per port
- Clock Generator state monitoring
  - Add event queue and sysfs file in the ptp_clock device
- Clock Generator configuration
  - Add devlink parameters to allow advanced configuration


intel