Addition of p802.11-MC Fine Timing Measurement (FTM) to p802.1AS-Rev:

Tradeoffs and Proposals

Rev 0.9

March 9, 2015

Authors:
Kevin Stanton, Intel
Carlos Aldana, Qualcomm

Supporters:
Ashley Butterworth, Apple
Matt Mora, Apple
Michael Johas Teener, Broadcom
Geoff Garner, Consultant
Craig Gunther, Harman International
Daron Underwood, Interval Zero
Don Pannell, Marvell
Rodney Cummings, National Instruments

Overview

- **The time to add *Fine* Timing Measurement (FTM) is now**
- **Some additions are straight forward, some additions require decisions that are outlined in this presentation**

Background

- **Regarding adding FTM to 802.1AS: Kevin, in July 2014, recommended:**

http://www.ieee802.org/1/files/public/docs2014/asbt-kbstanton-use-of-timing-measurement-0714-v02.pdf

- **There's been solid progress (elsewhere) in use of TM with 802.1AS-2011**
- **We now believe that the time to add FTM to 802.1AS has come**

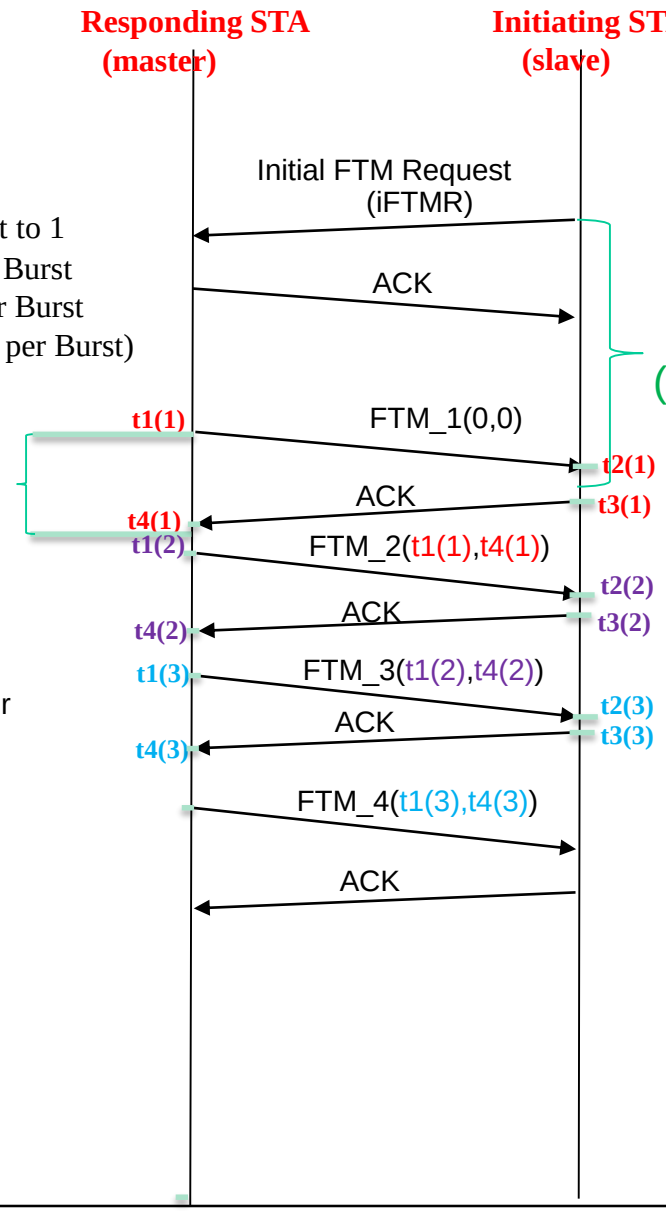
Timeline to FTM standardization

TGmc Timeline

	Open	Close
Fourth recirculation (TGac D5.0)	17-March-15	01-April-15
First sponsor ballot	*07-April-15*	*07-May-15*
Second sponsor ballot	*27-Sept-15*	*11-Oct-15*
Third sponsor ballot	*5-Jan-16*	*20-Jan-16*
Fourth sponsor ballot (unchanged)	*04-Feb-16*	*19-Feb-16*
EC to RevCom Jan 16		
RevCom to Standards Board Mar teleconference-16		

https://mentor.ieee.org/802.11/documents?is_dcn=0287

FTM Mechanism Overview with ASAP=1



ASAP set to 1
e.g. Single Burst
4 FTMs per Burst
(max 31 FTMs per Burst)

<=10ms
(recommended)

Min_Delta FTM

Minimum spacing between consecutive FTM frames

(min=100us, max = 25.5 ms)

Note that FTM retransmissions can occur when an Ack is not received.

Station Management Entity (SME) at Initiating STA estimates timeOffset of the local clock relative to that at the responding STA using

$$\frac{\frac{1}{3}\left(\sum_{k=1}^{3} t2(k) - \sum_{k=1}^{3} t1(k)\right) - \left(\sum_{k=1}^{3} t4(k) - \sum_{k=1}^{3} t3(k)\right)}{2}$$

and can compute RTT (Round Trip Time) = 2*linkDelay

$$\frac{1}{3}\left(\sum_{k=1}^{3} t4(k) - \sum_{k=1}^{3} t1(k)\right) - \left(\sum_{k=1}^{3} t3(k) - \sum_{k=1}^{3} t2(k)\right)$$

TM / FTM Deltas

Timing Measurement (TM)	Fine Timing Measurement (FTM)
10 ns timestamp resolution	0.1 ns timestamp resolution
Class 3 frame (post-association)	Class 1 frame (both pre and post-association)
Timing Measurement Request is optional	Initial Fine Timing Measurement Request (iFTMR) is Mandatory
Is already incorporated in 802.1AS	Is not yet incorporated in 802.1AS
No indication of when TM frame will be sent	FTM_1 should to be sent within 10ms of reception of iFTMR.
No concept of burst and no control of either the spacing between consecutive Timing Measurement frames or the number of Timing Measurement frames to be received.	FTMs per Burst, Min Delta FTM, Number of Bursts can be negotiated.

TM Request and TM Frame Format

Timing Measurement ***Request***
Action Frame

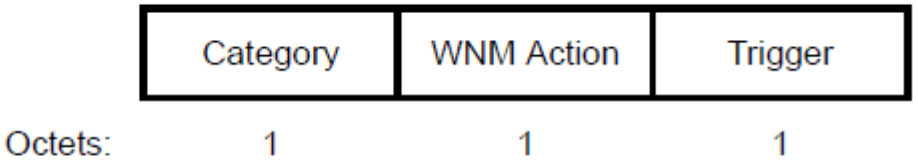
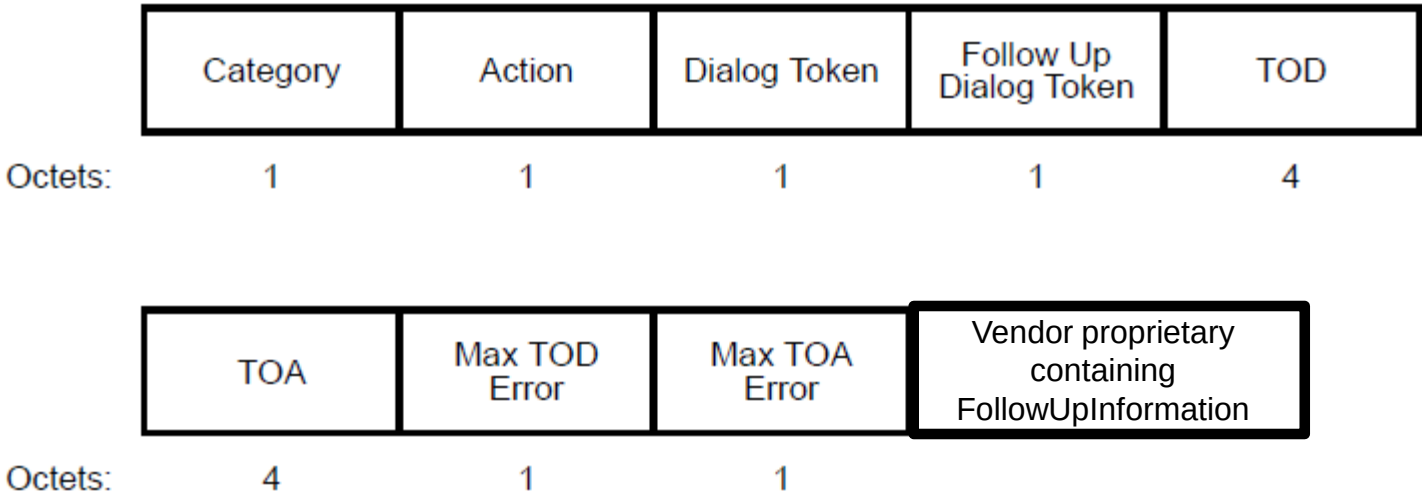


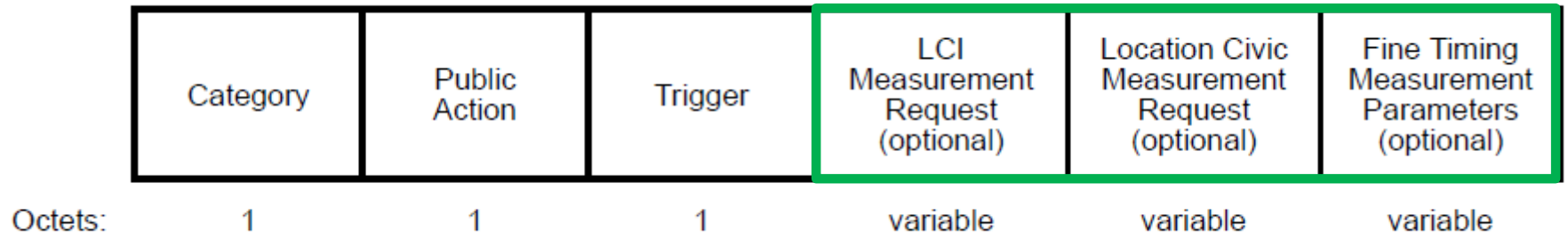
Figure 8-705—Timing Measurement Request Action field format

Timing Measurement
Action Frame

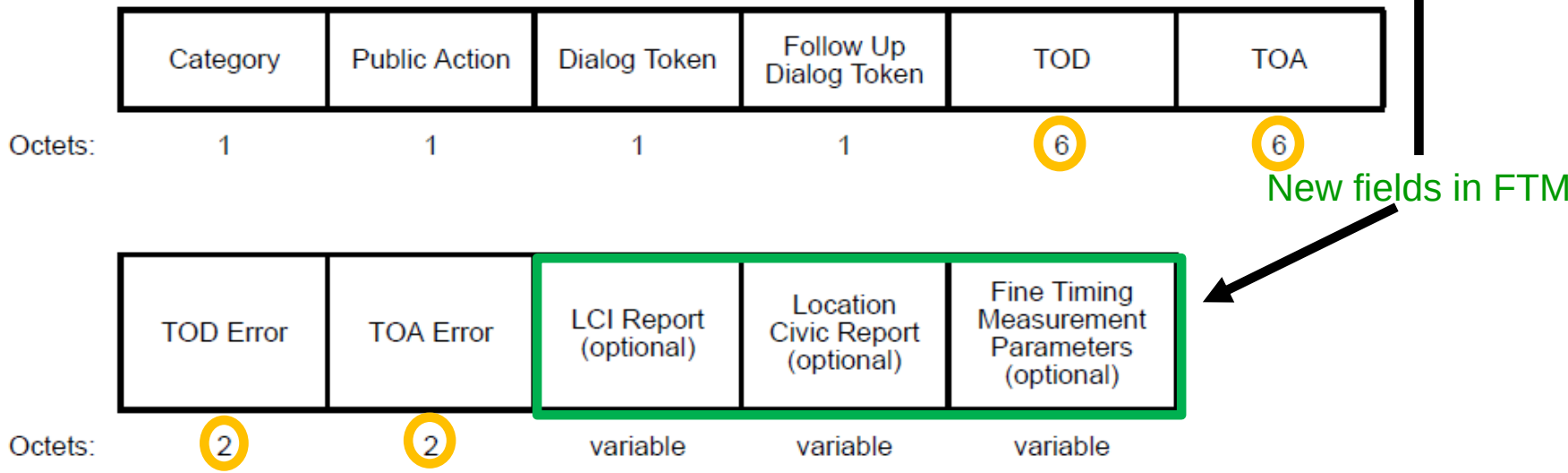


FTM Request and FTM Frame Format

Fine Timing Measurement ***Request*** Action Frame



Fine Timing Measurement Action Frame



FTM Changes to TM to account for timestamp resolution change from 10 ns to 0.1 ns

Assumptions

- **Maintain backward compatibility for TM while transitioning to FTM**
- **Time-sensitive applications will benefit from the improved accuracy delivered by FTM**
- **Compatibility with the direction of the next edition of IEEE P1588 is desired**

Rational Approaches

1. **Make FTM act just like TM, with adjustments applied as needed (e.g., timestamp resolution, etc.)**
   - Low-touch approach; very simple changes to 802.1AS-REV
   - Would achieve better accuracy
2. **Extend the MD state machines to take advantage of FTM features beyond improved timestamp accuracy**
   - Require quick response to request for TM/FTM from slave port
   - FTM frames per burst
   - Min Delta FTM (spacing out requests by some interval)
   - Etc.

Other potential enhancements

- **Specify the desired timestamp accuracy in a request**
- **Add the use of [FINE_]TIMING_MSMT_REQ to MD state machines**
- **More completely specify determination of asCapable**
- **Better align 802.1AS with the wakeup schedule of 802.11 power-management**

Questions to answer

- **How do we handle 0.1ns timestamps?**
  - 1588 represents time in ns and 16-bits of sub-ns
    - 802.1AS handle the sub-nanoseconds
- **Determination of asCapable (which implies "Actively send me time information")**
  - Proposal: Use [FINE_]TIMING_MSMT_REQ frame from the Slave device
    - Question: What does receipt of the _REQ frame mean?
      1. **Every time it wants to receive time – CONSISTENT WITH FTM**
      2. When it wants to start receiving time regularly
- **What about the PUBLIC nature of FTM Action Frames**
  - How do I determine the address of my "peer"?
    - In 802.1AS-2011, Announce frames are sent to all devices in the [W]LAN via one of the 16 reserved mcast mac address
    - If I can send Announce, I can address the [F]TM frame
    - FTM use by 802.1AS does not require association but in practice will always be associated—RIGHT?!
  - FTM frames are not automatically encrypted
- **Should 802.1AS propagate the optional FTM LCI and Civic reports to higher-layer applications?**
  - Lat/Long, street addresses, floor number in the building, retransmission rules (restricting propagating this info further)
  - Is there any e911 opportunity here?
- **Dealing with timestamp wrap-around**
  - NOTE: it's 7.8 hours with FTM vs. 43 seconds with TM

Questions to answer

- **For burst mode:**
  - Should the FTM burst be averaged and a single MDSyncReceive structure passed up from the slave MD to the slave PortSync entity—Proposed: **YES**
    - FTM Frames of a burst can be sent as close as 100us together
    - neighborRateRatio influenced by the ensemble of FTM measurements (e.g., first and last)
    - Bursts can happen with a burst period ranging from 100ms to every 2^16*100ms (1.8 hours). It is ultimately decided by the master.
    - Consecutive FTM frames can be separated by as much as 25.5ms and as little as 100us
    - Bursts can contain up to 31 FTM frames
    - Min Delta FTM requested is sent in the iFTMR frame. Min Delta FTM assigned is sent in FTM_1.
      - "Please send me a burst, but don't space the FTM frames more closely spaced than this within a burst"
    - Note: Consecutive FTM frames 100us apart with +/-0.1ns timestamp accuracy → +/-1PPM resolution
    - Should allow gPTP slave station to request a burst, for cases where the innate accuracy of the timestamping is not sufficient, and averaging is desired.
  - Is PTP time recovered in the same manner as individual TM measurements today
  - Should FollowUpInformation (see 12.5) be in every FTM frame of a burst?
  - In the SLAVE state machine
    - Compute and average NeighborPropDelay for every FTM of the burst
    - Compute and average (actual upstream t1 - MDSyncReceive.VendorSpecific.upstreamTxTime), called OffsetFromMaster in 1588, for every FTM of the burst
      - The raw timestamps could be combined in a simple way (e.g., average all t1s, t2s, t3s, t4s), subtract aggregate [(t2-t1) – (t4-t3)] / 2 = time offset between master and local clock in the slave, corrected for neighbor rate ratios.
      - The result of this calculation does not exist in Clause 12 today.
        - Possible solution is to add a statement to the Clause 10 stating that: computations on MDSync, for simplicity, assume a single set of values, but if an set of values is passed, the computations to combine or select among the values is up to the implementation, e.g., averaging.

BACKUP

802.1AS-2011 General Layering

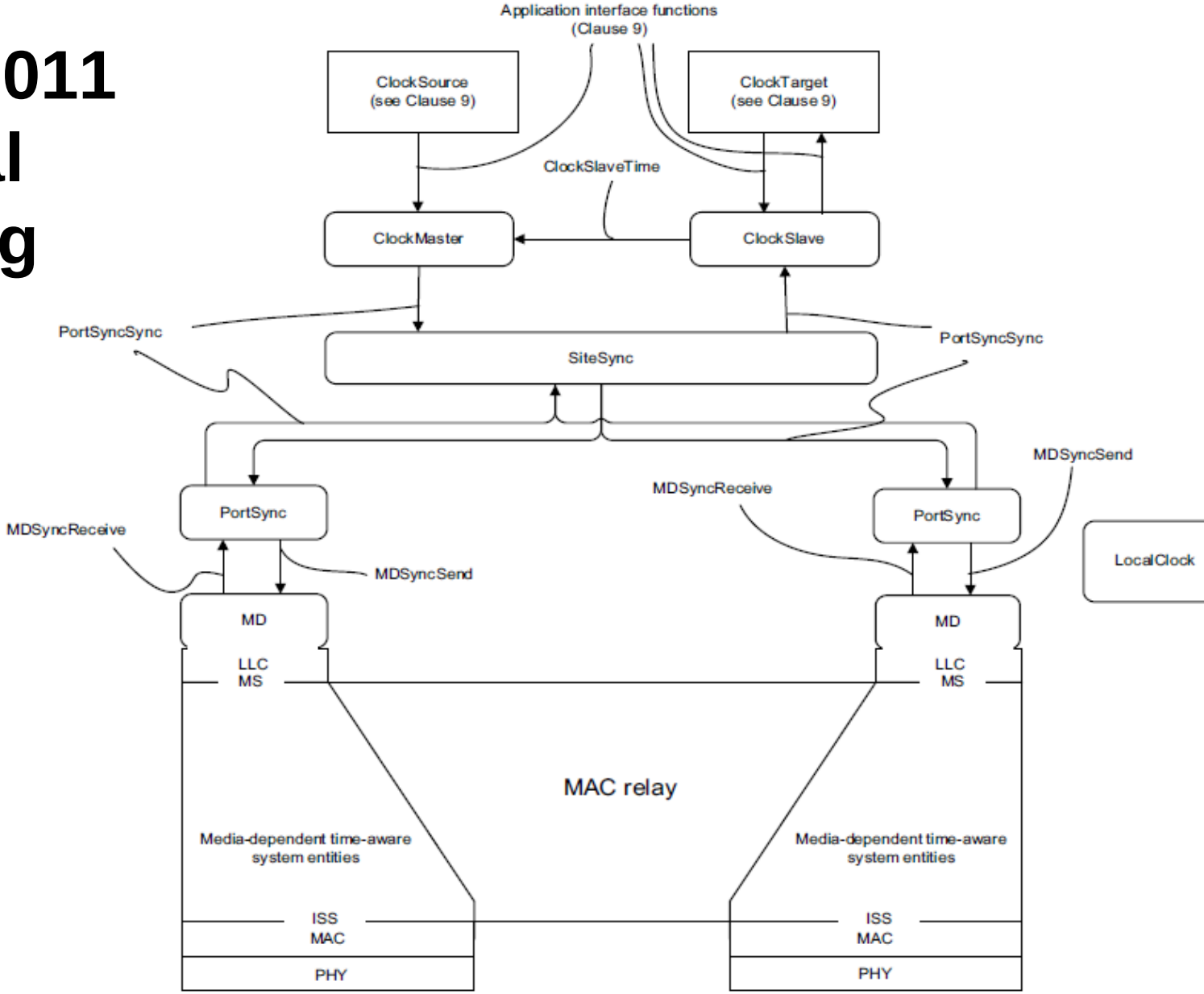


Figure 10-1—Model for media-independent layer of time-aware system

802.1AS-2011 Layering for 802.11 TM

(Clause 12)

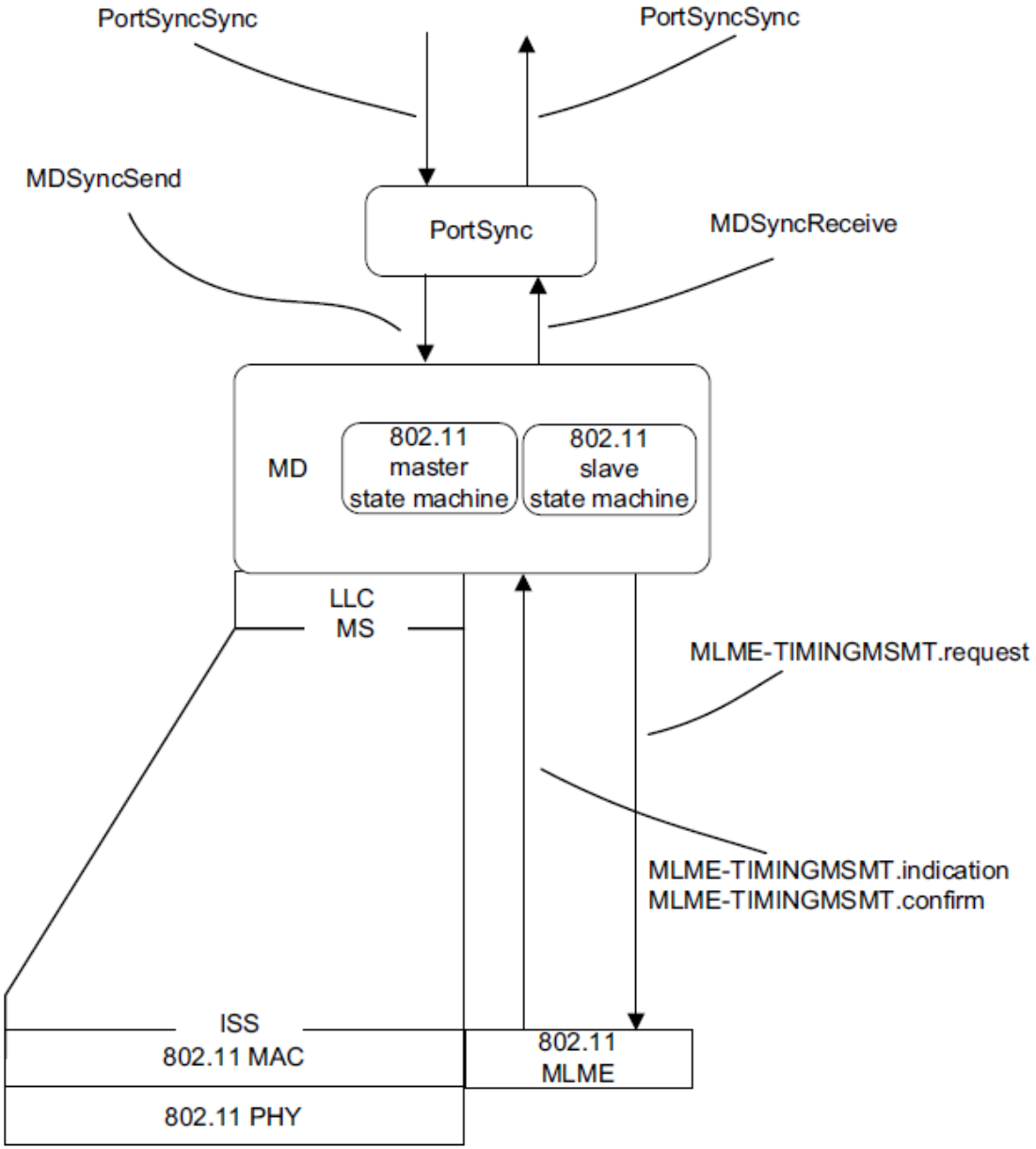


Figure 12-2—Media-dependent and lower entities in stations with IEEE 802.11 links

BACKUP

The following are from:

http://www.ieee802.org/1/files/public/docs2014/asbt-kbstanton-use-of-timing-measurement-0714-v02.pdf

Determination of asCapable for dot11 links

> "The per-port global variable asCapable shall be set to FALSE if the *timing measurement* bit … indicates that the peer IEEE 802.11 station is incapable of participating in the timing measurement protocol. Otherwise, asCapable *may* be set to TRUE."

This was intentionally ambiguous, to reflect special needs of battery-powered wireless devices

There are multiple approaches for a mobile device to advertise its need for precision time:

- Use an existing SIGNALING message
  - Redefine the default TimeSyncInterval as 127 (stop)
  - A subsequent SIGNALING message then set the TimeSyncInterval to something else, e.g. -3 or 1/8s
- Timing Measurement Req Frame
- Fine Timing Measurement Req Frame

This deserves further discussion and involvement of other experts

BACKUP

The following are from:

http://www.ieee802.org/1/files/public/docs2013/ASbt-kbstanton-TimingMeasurements-status-1113-v01.pptx

Residence Time of a Station-Bridge

- **P802.11ak introduces WLAN Stations that Bridge**
- **Residence Time of a WLAN Station-Bridge is large**
  - FOLLOWUP information for message `n` sent in message `n+1`
  - With fixed Sync Interval, Residence Time ≥ sync interval
- **Implications to P802.1ASbt, if we do nothing:**
  - Introduces small lag in time propagation through Station Bridge
  - And this grows linearly with # of Station-Bridge

    += Sync Interval * # of Station-Bridges
- **Approaches**
  1. Bi-modal sync interval (“burst-of-two”)
  2. Increased sync rate
  3. Live with the lag
     - Probably fine for Stations, less fine for long chains of Station-Bridges

IEEE Std. 802.11™-2012 Timing Measurement

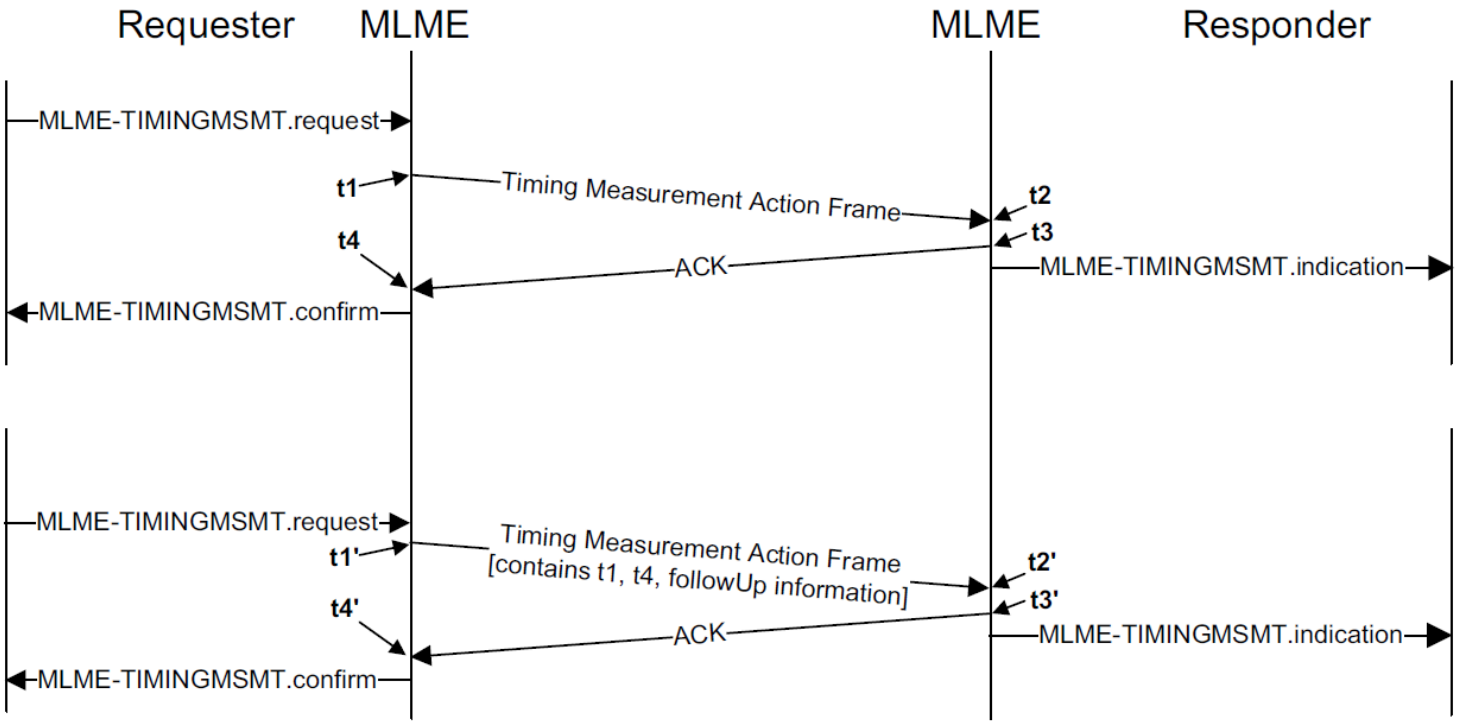


Figure 12-1—Timing measurement procedure for IEEE 802.11 links

IEEE p802.11-MC Fine Timing Measurement Protocol