AV Time Synchronization for Wired and Wireless 802 LANs

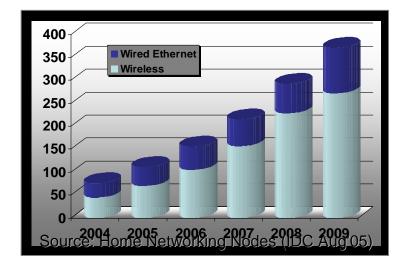Kevin Stanton

Intel Corporation

5/16/2006

1

Objective

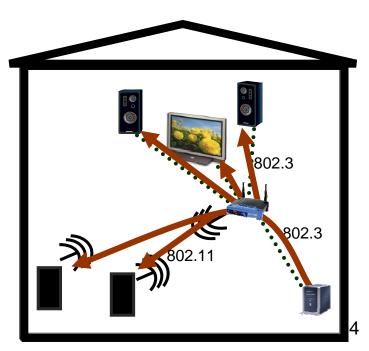Ensure 802.1 time sync architecture is specified for both 802.3 and 802.11 MACs

Agenda

- Motivation and background
- Wireless time sync challenges
- Why Wireless might want something different than Ethernet
- Proposal / approach

Motivation

Source: Home Networking Nodes (IDC Aug 05)

- 802.3 AND 802.11 are important to home customers
- Many home AV networks will require both wired and wireless hops
- Time sync must be end-to-end
- Considering wireless now will improve probability of end-to-end interoperability

802.3

802.3

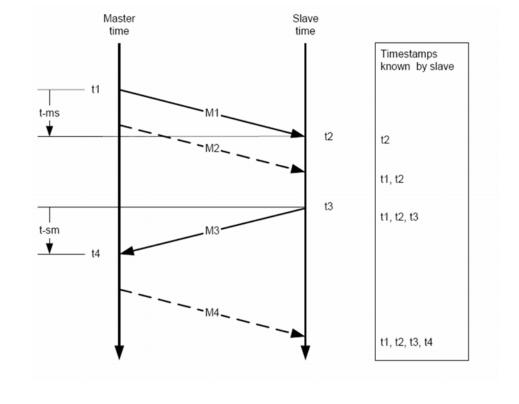802.11

4

Wireless time sync challenges

- Link delay changes over time
  - Multipath delay spread
    - *Is statistically zero mean*
  - Changes in link distance (mobility)
    - This effect is negligble
      - » *100ns in 100 ms → 682 MPH (MACH 0.9)*

- Wireless data rate is lower than Ethernet
  - Must ensure small time sync overhead
    - *Wireless clocks have 5x better PPM (20PPM vs. 100PPM)*
      - » *Sync interval can be longer*

- Link errors in .11 are very likely (10% drop rate)
  - Retransmissions result

- Effective data rate changes
  - *Should only impact bandwidth – not delay of 1st octet*

5

Time sync with IEEE 1588
[Also a proposed method for 802.1as]

Goal: Synchronize clocks of networked nodes

1. Master schedules M1 for Tx
2. As it passes from MAC to PHY, t1 captured
   - Using master clock
3. Time t2 captured as passes from PHY to MAC
   - Using slave clock
4. M2 carries t1 to slave
5. Slave schedules M3 for Tx
6. t3, t4 captured as above
7. M4 carries t4 to slave

If link delay is fixed & symmetric:

Clock offset between master and slave
= [(t2-t1) – (t4-t3)] / 2

Will work ONLY if link delay is Fixed & Symmetric

6

Location estimation with 802.11v

Goal: Measure distance between 802.11 entities in ns

1. Requester schedules M1 for Tx
2. As it passes through the PHY, t1 captured
   - Using requester clock
3. Time t2 captured in PHY on Rx
   - Using slave clock
4. Responder MAC automatically sends M1 ACK very quickly (a control frame)
5. t3, t4 captured as above
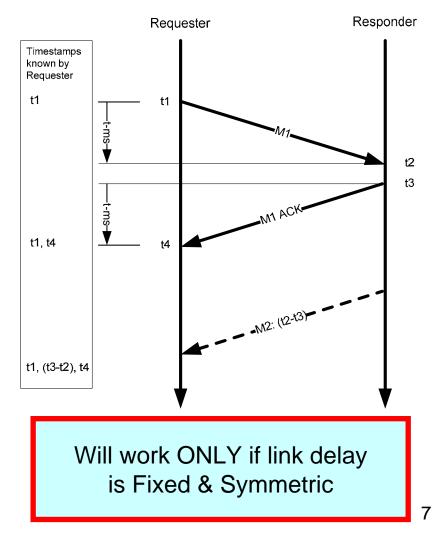6. M2 carries (**t3-t2**) to requester

If link delay is fixed & symmetric:

Link delay = [(t4-t1) − (**t3-t2**)] / 2

Clock offset between master and slave
   = [(t2-t1) − (t4-t3)] / 2
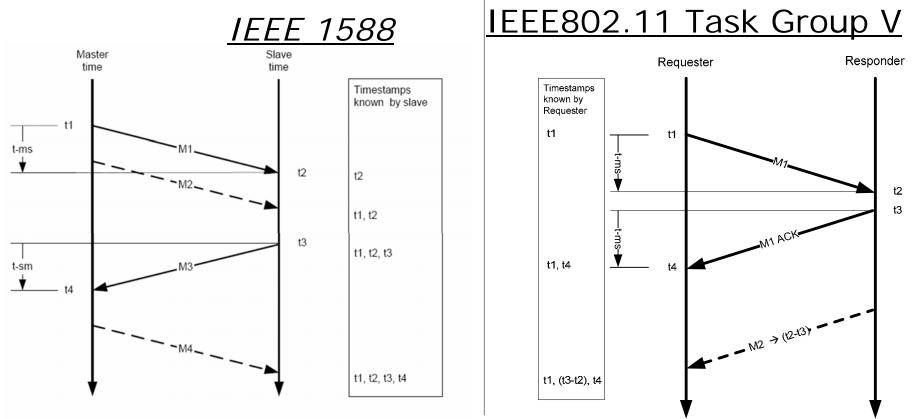
BUT Requester doesn't know t3 and t2!

Will work ONLY if link delay is Fixed & Symmetric

7

We need to consider these separately:

1. Protocol
2. Measurement (frame timestamp)

1. Protocol differences

IEEE 1588

Master time

Slave time

t1

t-ms

M1

M2

t2

t-sm

M3

t4

M4

t3

Timestamps known by slave

t2

t1, t2

t1, t2, t3

t1, t2, t3, t4

IEEE802.11 Task Group V

Requester

Responder

Timestamps known by Requester

t1

t1, t4

t1, (t3-t2), t4

t1

t-ms

M1

t2

t3

M1 ACK

t4

t-ms

$M2 \rightarrow (t2-t3)$
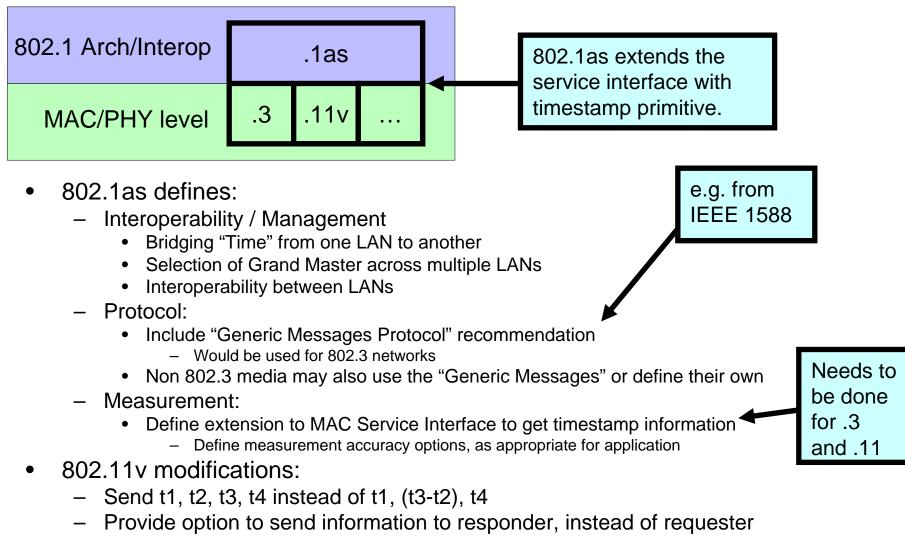
Differences (compared to IEEE 1588):

- Timestamp information sent to requester not to responder (slave)
- $(t3-t2)$ is sent, not both $t3$ and $t2$
- Data frame and MAC-generated control frame (an ACK) are used, not two data frames – IMPORTANT for Location accuracy
- 802.11v doesn't define the precise timestamp point in the frame

9

2. Measurement Differences

- AV time sync requires 100s of ns of accuracy
  - Measurement probably taken between MAC and PHY

- Location estimation requires 1s of ns of accuracy
  - Measurement probably taken in the PHY

NOTE: Both measurements require an extension to the respective MAC service interfaces
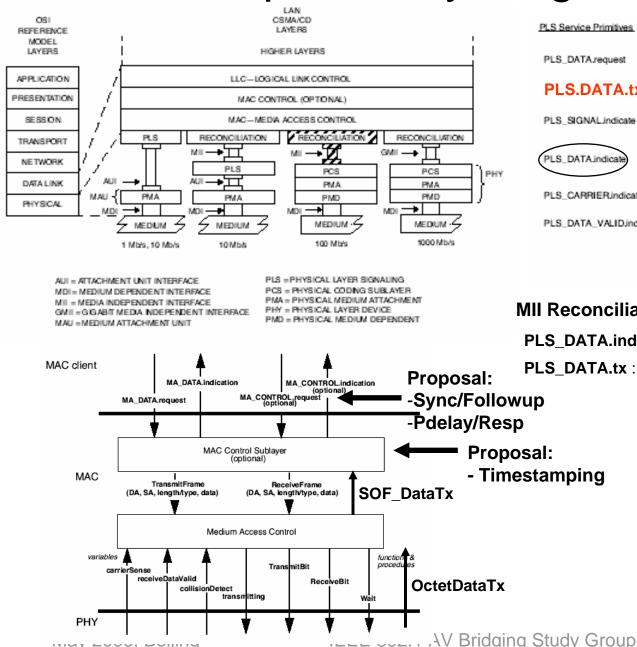
Putting it all together

802.1 Arch/Interop	.1as
MAC/PHY level	.3 / .11v / …

802.1as extends the service interface with timestamp primitive.

- 802.1as defines:
  - Interoperability / Management
    - Bridging "Time" from one LAN to another
    - Selection of Grand Master across multiple LANs
    - Interoperability between LANs
  - Protocol:
    - Include "Generic Messages Protocol" recommendation
      - Would be used for 802.3 networks
    - Non 802.3 media may also use the "Generic Messages" or define their own
  - Measurement:
    - Define extension to MAC Service Interface to get timestamp information
      - Define measurement accuracy options, as appropriate for application
- 802.11v modifications:
  - Send t1, t2, t3, t4 instead of t1, (t3-t2), t4
  - Provide option to send information to responder, instead of requester
  - Define timestamp capture point in frame

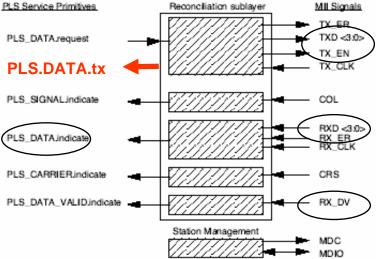**e.g. from IEEE 1588**

Needs to be done for .3 and .11

11

Summary

- Residential AV users will benefit from synchronized playback over wired & wireless
- Time Sync interoperability is needed across 802 LANs
- 802.11 TGV approach is very similar to current p802.1as proposal to use IEEE p1588 version 2
- Separating interoperability features from the actual protocol and measurement method allows flexibility per media type.

- Next steps:
  - Specify service interface for packet timestamp
  - Recommend 802.11 TGV changes
  - Ensure maximum interoperability between .3 and .11 networks

Backup

IEEE 802.1 AV Bridging Study Group

Proposed layering in 802.3

MII Reconciliation sublayer (similar to GMII)

PLS_DATA.indicate : used for reception timestamp

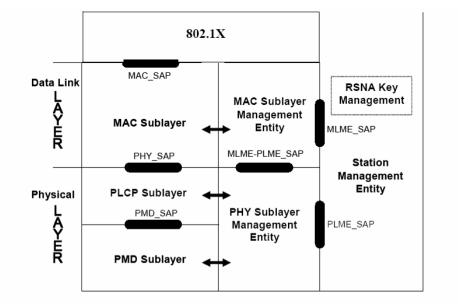**PLS_DATA.tx** : new primitive, for transmission timestamp

Proposal:
-Sync/Followup
-Pdelay/Resp

Proposal:
- Timestamping

802.11 Layering

Figure 11—Portion of the ISO/IEC basic reference model covered in this standard

IEEE 802.1 AV Bridging Study Group

802.11 Synchronization

11.6 Higher layer timer synchronization

11.6.1 Introduction

Higher layer timer synchronization is beyond the scope of this amendment. However, explanation on how the constructs in this amendment can be used to support such capabilities may be useful to the designer. One way to accomplish synchronization across a BSS is by multicasting synchronization (Sync) packets from the higher layers containing a time stamp and a sequence number. These packets would be opaque to the MAC and would be transported in the same way as any other MSDU (most likely addressed to the multicast address). Sync packets would be treated as a type of management packet by the higher layers. The time stamp in the Sync packet would contain the higher layer clock value at the time when the previous Sync packet was transmitted. The sequence number parameter has a value equal to the sequence number of the MSDU in which the time stamp is provided.

The reason the packet would contain the time stamp for the previous Sync packet (rather than the current packet) is that hardware and layering constraints would prohibit the ability to provide a time stamp for the exact instant the current packet is transmitted within that packet. However, a MLME-HL-SYNC.indication primitive allows the transmitting QSTA to know exactly when each Sync packet is transmitted. A receiving QSTA can similarly note the time when each Sync packet is received as well as the sequence number for that frame. The receiving QSTA would save this receive time indication for each packet along with its sequence number and compare the indication of the previously received Sync packet to the time stamp in the most currently received packet. This comparison allows the STA to compute an offset, which can be used to adjust its time reference to match that of the synchronization source. The sequence number would ensure that the correct packet is being used for time stamp comparison. It is possible a packet is lost; in this case, the received time stamp for the lost packet should be discarded.

The last symbol of the Sync frame is indicated by the PHY using the PHY-TXEND.confirm and PHYRXEND. indication primitives in the transmitter and receiver of the Sync frame, respectively. Practical limits on the coincidence of this indication and the last symbol of the Sync frame are implementation dependent. The accuracy of this technique also depends on the propagation delay between the source and receiving channel. However, both the time difference (between the PHY indication and the last symbol of the Sync frame) and the propagation delay can be considered as fixed-delay components. Therefore, they contribute only to the fixed time difference between the transmitter and receiver STAs' clocks and do not contribute to jitter. An implementation-dependent scheme can be used to cancel or minimize this fixed time difference. The Sync frame may also be relayed through the AP. In this case, the non-AP QSTA that generates the time stamps notes the reception of the multicast Sync frame from the AP as the indication to save the higher clock value for the next Sync frame. Receiving QSTAs would also similarly note the time when each Sync packet is received from the AP. The sequence number would include a value corresponding to the frame received from the AP.

Other implementations (aside from what is described here) may also be supported.

Ethernet Timestamp Accuracy

- MAX RX+TX PHY processing delay :
  - 100Mbps: 480ns
  - 1000Mbps: 328ns

  - This gives maximum variation allowed by standard
    - Assumes 0ns minimum

- We assume frame-to-frame variation in a specific PHY will be very small
  - Not true between PHYs from different vendors

Table 40–14—MDI to GMII delay constraints (full duplex mode)

Sublayer measurement points	Event	Min (bit times)	Max (bit times)	Input timing reference	Output timing reference
GMII ⇔ MDI	TX_EN Sampled to MDI Output	—	84	GTX_CLK rising	1st symbol of SSD/CSReset/ CSExtend/ CSExtend_Err
	MDI input to RX_DV de-assert	—	244	1st symbol of CSReset	—