Application Report
SWAA162A – July 2015 – Revised December 2015

# Precise Time Synchronization Over WLAN

*Oren Shani* *WL8 System Group*

**ABSTRACT**

Recently, there has been a big demand for connectivity in the consumer's and industrial's applications. The requirements and implementations are different, but the common denominator is reliability, robustness, performance and scalability, which permits further growth and development according to the industries' progress. The WiFi ® technology, contrary to the other connectivity solutions, has advanced a few major steps forward and continues advancing to meet the demanding requirements in different implementations. Most of these implementations are standalone and the only communication to the outer world is through wireless connection. Some applications just needs to be aware of a presence of another device, but other implementations will be strictly dependent on the operation of its counterpart device or devices located within the same network. Such neediness requires a high level of precise synchronization between the devices to permit them to operate independently on the one hand, but seamlessly and concurrently on the other hand.

A solution proposed in this document resolves this task by a usage of inherent features of the WiFi technology in addition to the proprietary add-ons.

**Contents**



**List of Figures**

# 1 Introduction

Precise time synchronization between devices is required for a variety of applications such as audio, industrial, medical and so forth. An example of such time synchronization necessity can be found in the audio market. The human brain is able to determine pretty precisely the direction of an audio source by taking advantage of only the slightly different distance that the sound waves have to pass in order to reach each ear. The brain is actually analyzing and comparing the amplitude vs. time deltas between the different waves. Unfortunately, in a case of time delta between different audio sources playing the same soundtrack, this amazing evolutionary capability might deceive our brain making it believe that there is a physical variance between the audio sources locations, even though there is none. Therefore, very accurate time synchronization is required for the channel-based audio, especially when maintaining separation between left and right stereo audio channels or recreating perfect 5.1 surround sound across wireless speakers (see Figure 1) .

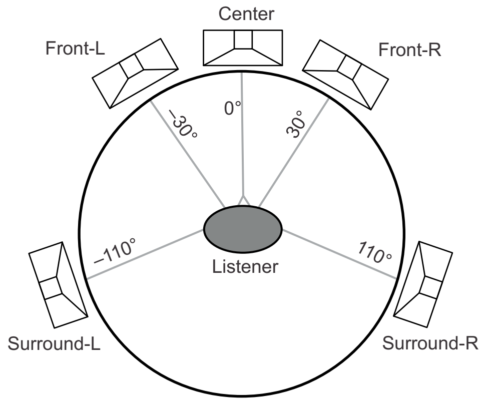


**Figure 1. Surround Sound System Based on Wireless Speakers**

Another implementation can be found in the industrial/medical market where two (or more) wireless controlled machines need to run surgical operations simultaneously. For a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems', see the IEEE 1588 standard.

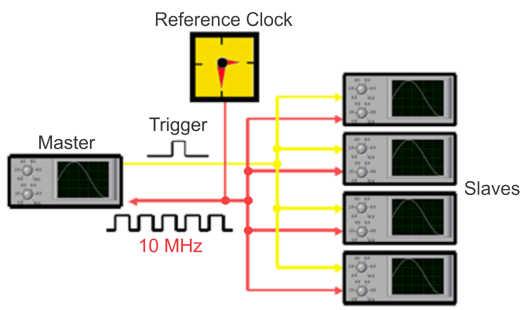


**Figure 2. Synchronizing Machines**

An example of a multi-camera system that creates a panoramic view of a dynamic vista or a 3D reconstruction from multiple images should also be precisely synchronized. Synchronization of frames, from many cameras, is typically employed at two levels: capturing and delivery to an algorithm on a different machine. Lack of capture synchronization would result in slight time offsets in the frame acquisition of each camera. Synchronization of capture is typically achieved by a common hardware trigger. Various triggering approaches have distinct levels of synchronization, a lack of which produces a relative temporal drift between frames. While high quality hardware triggering is a good and robust solution, it requires that cameras be equipped with a suitable interface for such signal, which rules out most commodity cameras. Such implementation will highly increase the overall cost of the system.

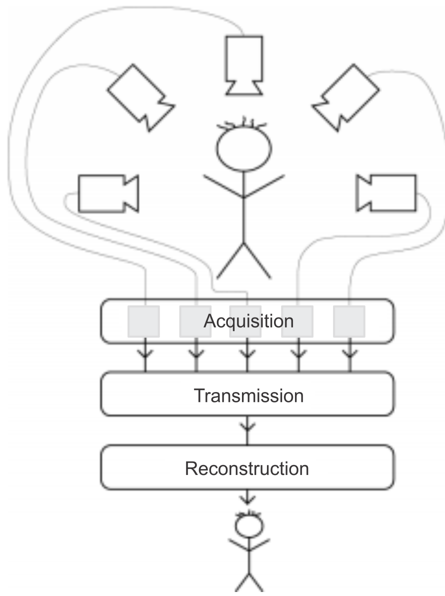


**Figure 3. 3D Reconstruction Process From Camera Image Acquisition to Reconstruction**

When a communication between those devices is based on the WLAN protocol (IEEE802.11), which, inherently, operates asynchronously and independently between devices, an additional synchronization mechanism is required.

According to the protocol, each WLAN capable device should "compete" to get an access to the air. Furthermore, the wireless communication embraces some level of collisions, which leads to data retransmission and creates delay and uncertainty in the packets' time of arrival.

This document discusses WL8 WLAN solutions for this fundamental issue. The solution is based on WL8 hardware and software implementation together with a software implementation on the compatible processor (the sample is based on Linux 3.14, TI AM335x ARM® Cortex®-A8 ). There are a few primary advantages of the WL8 solution:

- It accomplishes a time accuracy of less than 20 µsec between devices connected to the same AP/Router.
- On the one hand, it is a proprietary solution, which means that all devices to be synchronized should be based on WL8, but on the other hand any access point (AP) can be used for the synchronization. There is no need to support any special protocol (such as 802.11 V).
- Although it is a proprietary solution, it can be interoperable with other TSF providing systems. As the WiLink8 is providing high accuracy, using a system with Wilink8 devices will provide a robust and stable synchronized system.
- It does not overload the air as it does not require any kind of handshake for the synchronization itself. On top of that the application layer can use any control interface for utilizing the synchronization for its own needs.
- A number of synchronized devices might be as many as a specific AP supports (varies between 10 and 255 devices, depending on the AP's model).
- Two or more WL8 devices will also be synchronized when one is operating as access point and the other as station in the same BSS, meaning that WL8 station should be connected to the WL8 AP.

The solution is available starting WL8 R8.6 software release (July 2015).

# 2 Requirements

## 2.1 Hardware Requirements

The Time Synchronization present implementation was done using Sitara™ AM335 host processor.

A general-purpose input/output (GPIO) line should be connected between the AM335 device and the WL8 device:

- On WL8 side: COEX_MWS_FRAME_SYNC (GPIO11 on TI module)
- On AM335 side: GPIO 2_2 (TIMER4)

This GPIO line is responsible for the synchronizing between two different hardware devices.

## 2.2 Software Requirements

The time synchronization feature is fully supported starting from WL8 R8.6 software release. It includes modification on both firmware and driver codes.

# 3 Time Synchronization's Enabling

The following actions should be done in order to enable the time synchronization feature:

- Disable the ELP mode (should be done after connection or using configuration file)

```
'echo 0 > /sys/kernel/debug/ieee80211/phy0/wlcore/sleep_auth'
```

- Enable the feature (should be done only once):
  - For Station mode- will provide TSF only after connection

```
'wlconf-toggle-set.sh <wlconf.bin path> sync 1'
```

  - For AP mode

```
'wlconf-toggle-set.sh <wlconf.bin path> sync 2'
```

# 4 Description

The precise time synchronization can be achieved between all WL8 devices connected to the same AP, also when the AP itself uses the WL8 device.

The time synchronization is based on two major system's capabilities:

- WL8 WLAN ability to capture and register the precise arrival time of the connected AP's beacons. For this matter the packets propagation time over the air is negligible (1m = 3ns). The beacons' timestamp is used as a common reference for all devices to be synchronized.
- The synchronization between WL8 and AM335 devices' hardware, which are based on a different system clocks with a different precision level. The synchronization is based on a dedicated hardware GPIO line and software algorithm.

A combination of beacons' timestamp awareness by all devices connected to the same AP and the hardware, high priority, interface between two independent processors permits the host layer to "learn" and be synchronized to the WLAN physical layer on all devices.

The beacons' timestamp with a combination of the GPIO trigger time are uploaded to the host layer and translated to the host's clock. When all of the host's clock on all of the devices within the same BSS are synchronized, it can be used for any kind of application layer implementations, like synchronization between speakers within a home theater system, cameras' capturing with a precise intervals and so forth.

# 5 Integration

To integrate the software, the following modules will be needed:

- Kernel DTB modification, to set the AM335x GPIO as output, and allocate it for the WiLink Driver.
- Software release is based on R8.6 and later

# 6 Verification

The time synchronization feature has been tested and verified thoroughly towards WL8 R8.6 software release to assure its accuracy, reliability and robustness.

The following sections describes tests' results accomplished to verify the feature's implementation.

## 6.1 Accuracy

The Time Synchronization behavior has been tested using Cisco 1252 AP. A few use cases have been verified to permit a wide visibility on the system's behavior and provide the complete awareness on feature's usability and limitations to customers.

Since the mechanism is based on reading Beacons' data, received form the connected AP, on the one hand and the independent host's requests over SDIO hardware line on the other hand, a different combinations between Beacons' intervals and SDIO's request intervals have been tested to verify the synchronization accuracy between two connected devices, to the same AP.

Figure 4 shows the correlation between the Beacons' arrival intervals and SDIO request intervals. It can be seen that frequency of SDIO's request has a lower impact comparably to Beacons' intervals used by AP. As longer as Beacons' intervals will be, the time synchronization accuracy will slightly decrease. For the most common Beacon interval of 100TUs, the mean delta time between two WLAN devices is 1-2 µsec independently of SDIO request intervals that varies from 50 msec to 1000 msec (the mean duration was calculated using 1000 samples for each measurement combination).

To reflect the measured results, a few statistics parameters are presented:

- "Mean Duration" – the average of 1000 samples
- "Median Occurrences" – the delta time of the 500th measurement
- "Max Occurrences @" – the delta time at which the maximal number of occurrences occurred
- "95% Occurrences @" – Delta time at which 95% of total measurements was taken
- "Max Duration" – Maximal delta time result, even if occurred only once out of 1000 samples

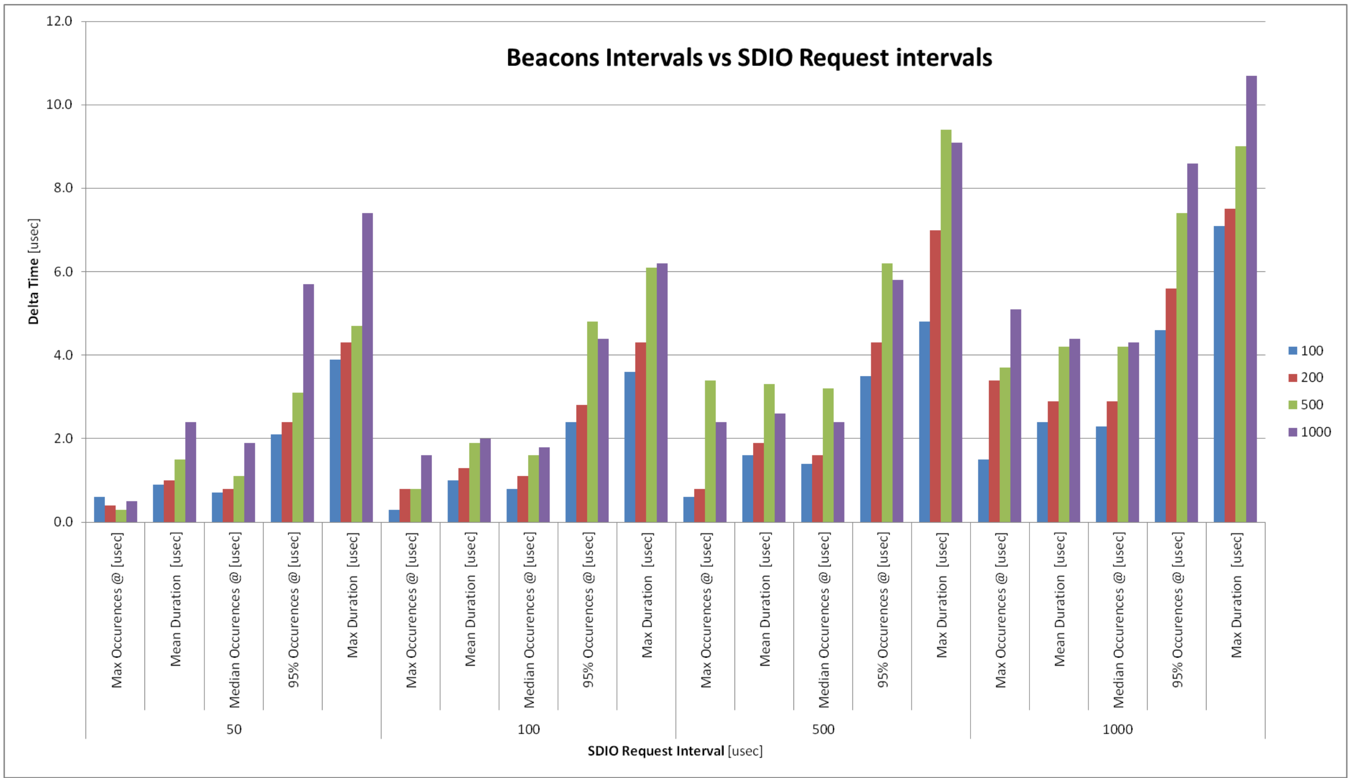


**Figure 4. Beacons Intervals vs SDIO Request Intervals**

### 6.2 Reliability

To verify the feature's reliability and accuracy, it has been tested with a wide range of APs and routers from different vendors and models. In total, about 160 APs have been tested. Each AP has its own level of accuracy in Beacons' delivery, Beacons' time interval and access to the air. Each AP has been tested in three scenarios as listed below:

- 2 STAs in idle connection mode
- 1 STA in idle connection mode, 1 STA is running TCP Rx, max TP, from the AP
- 2 STA running TCP Rx max TP from the AP

The obtained measurement results are presented in a few manners to show the statistics results' values and reflect the time synchronization reliability over the wide range of APs.

Figure 5 shows statistics results for each AP out of 160 tested APs measured when both WL8 devices are connected idle without traffic. The results are shown using five statistics calculations, as described above. Each bar represents result for a single AP.

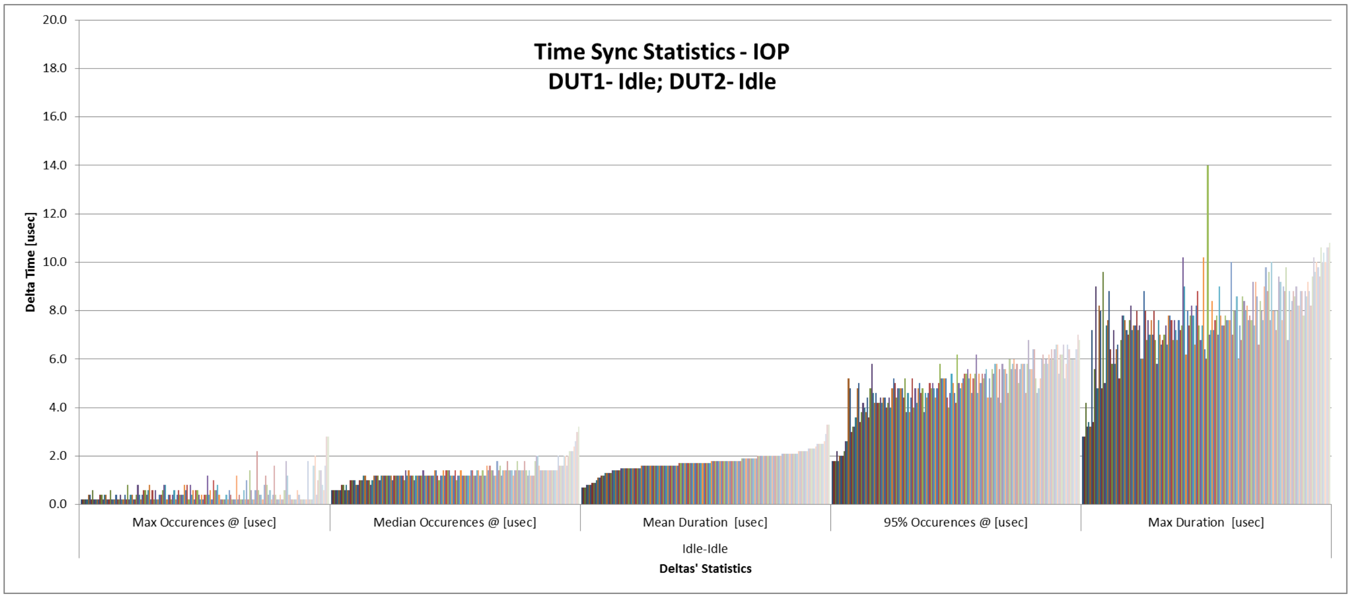


**Figure 5. IOP Results – Both Devices Connected Idle**

Figure 6 and Figure 7 show statistics results for scenarios when only one device is running TCP Rx and the other is idle connected and when both devices are running TCP Rx.

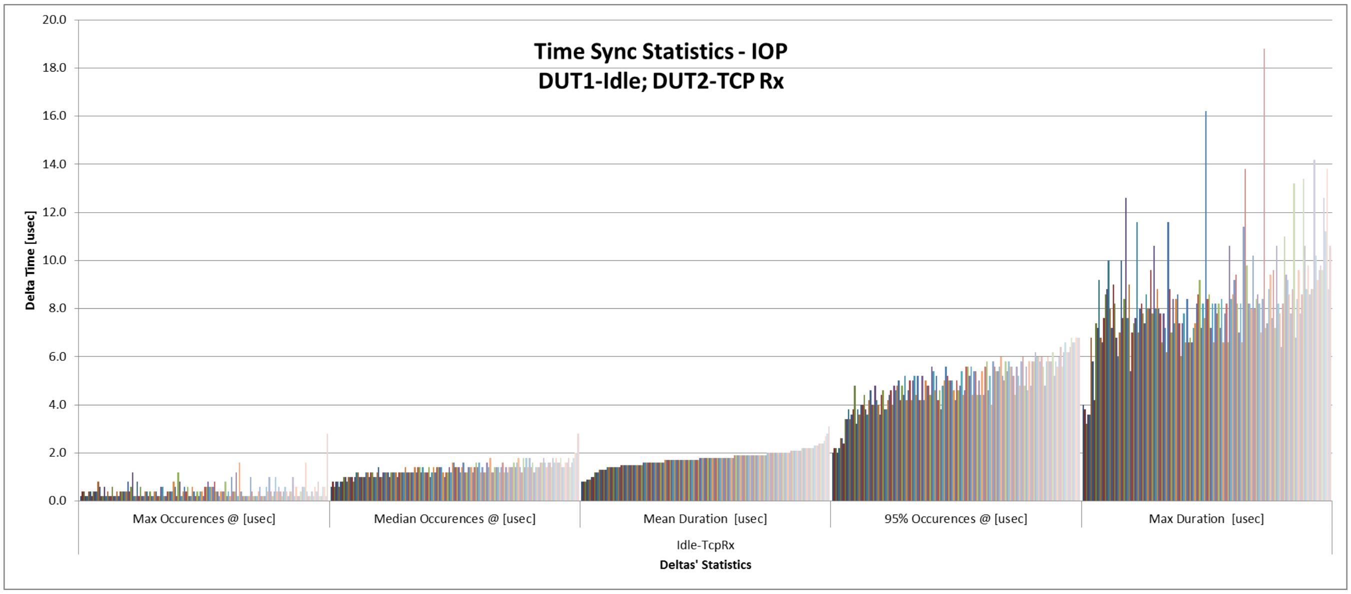


**Figure 6. IOP Results –Device 1 Connected Idle, Device 2 is Running TCP Rx**

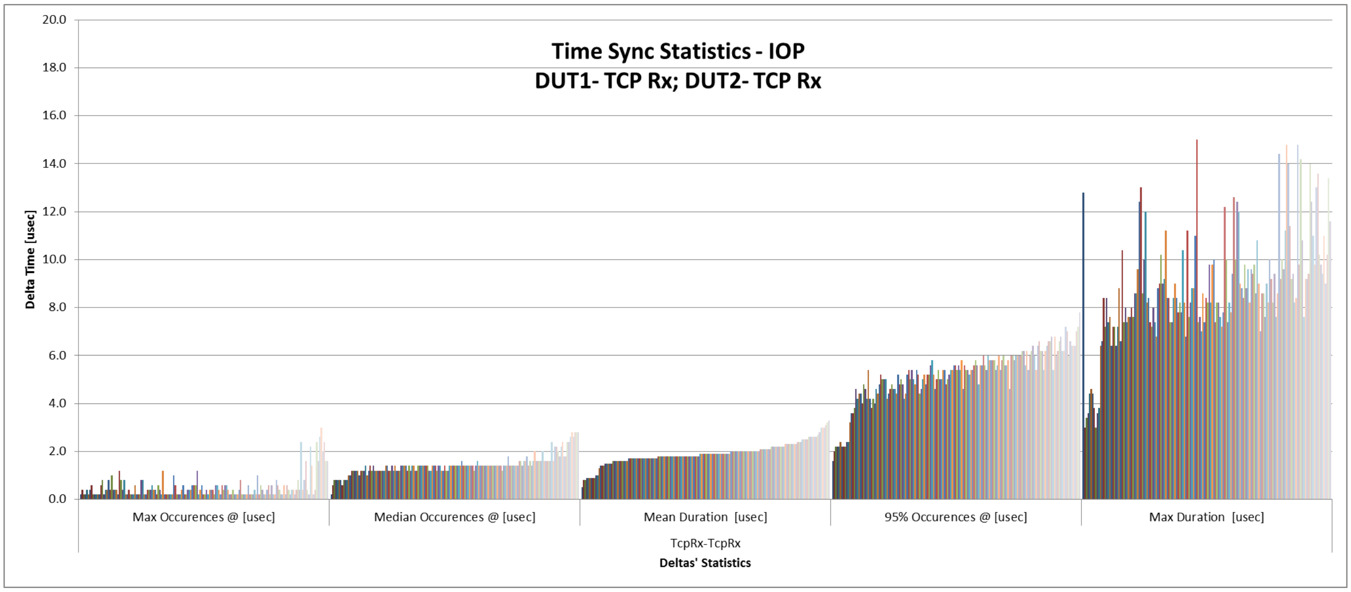


**Figure 7. IOP Results – Two Devices are Running TCP Rx**

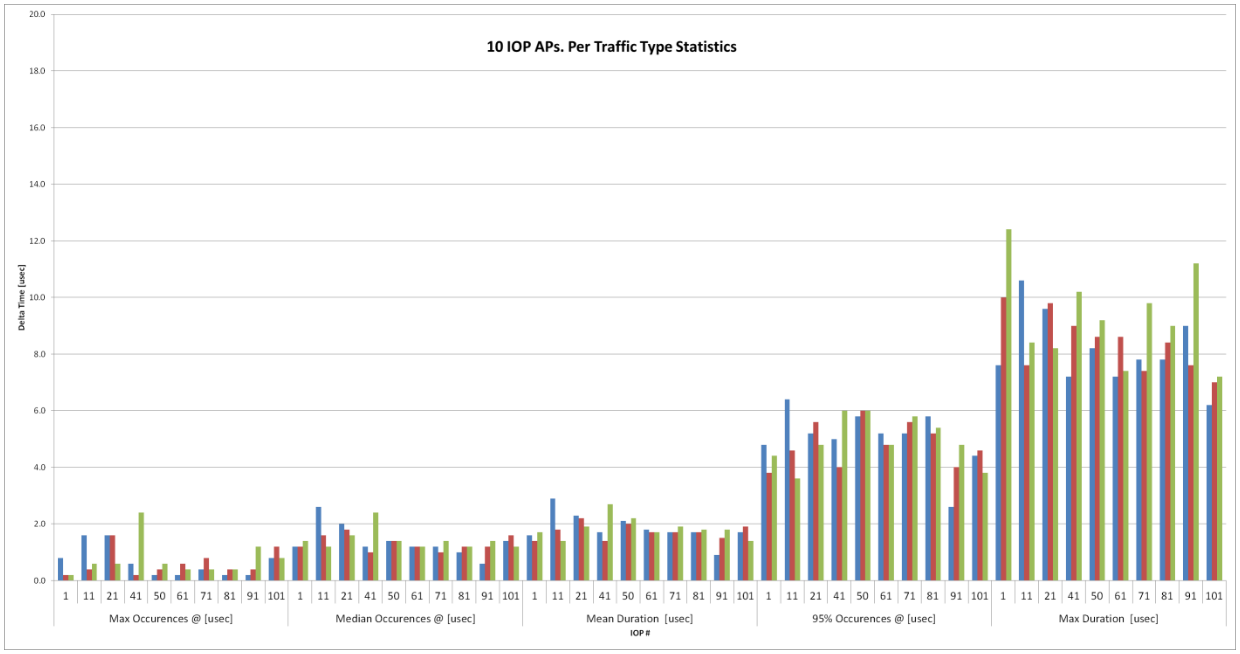


**Figure 8. IOP Results – Per AP Representation for Three Scenarios**

Figure 9 through Figure 11 show a single statistics result for all tested APs.

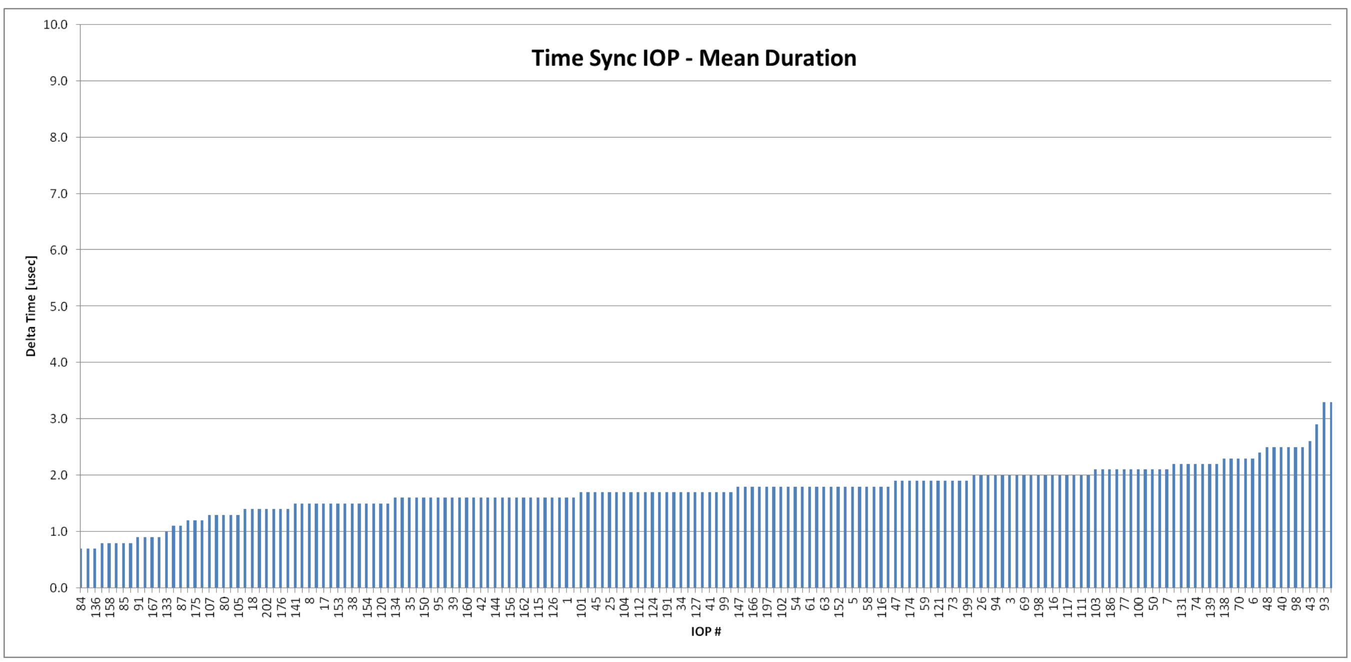


**Figure 9. IOP – Mean Delta Time Between Two Devices for all APs**

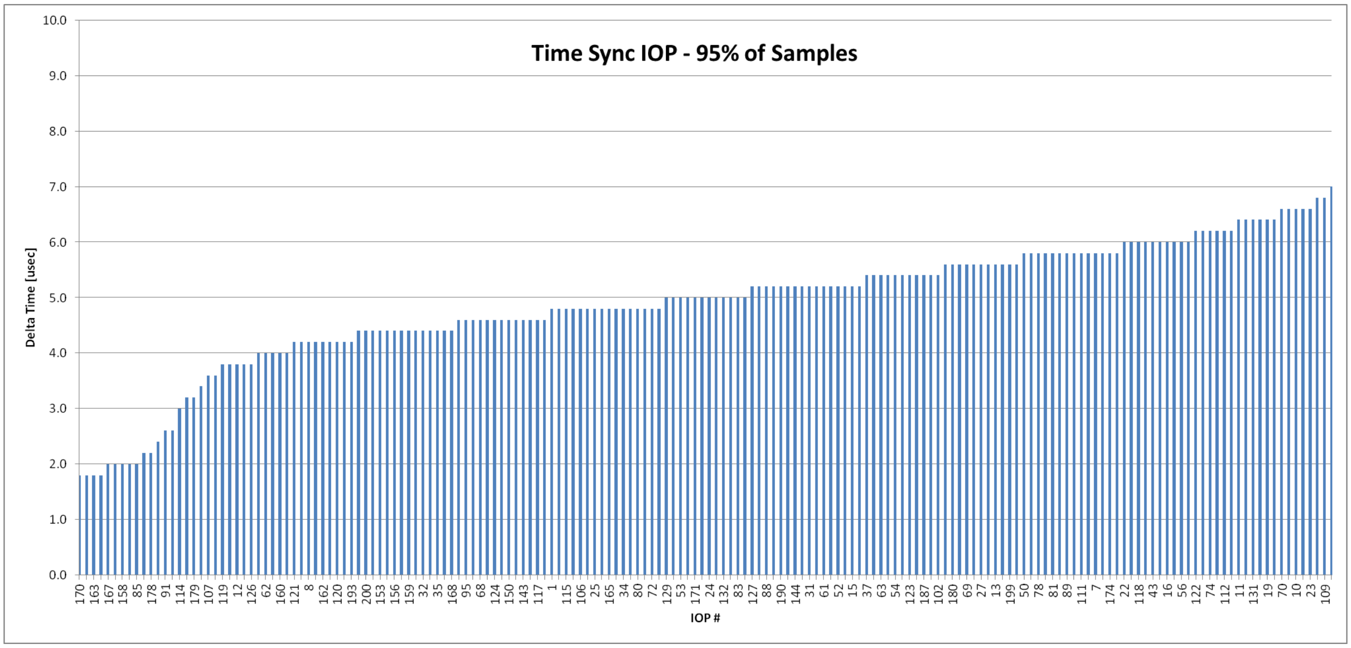


**Figure 10. IOP – 95% of all Delta Times Instances Measured Between Two Devices for all APs**

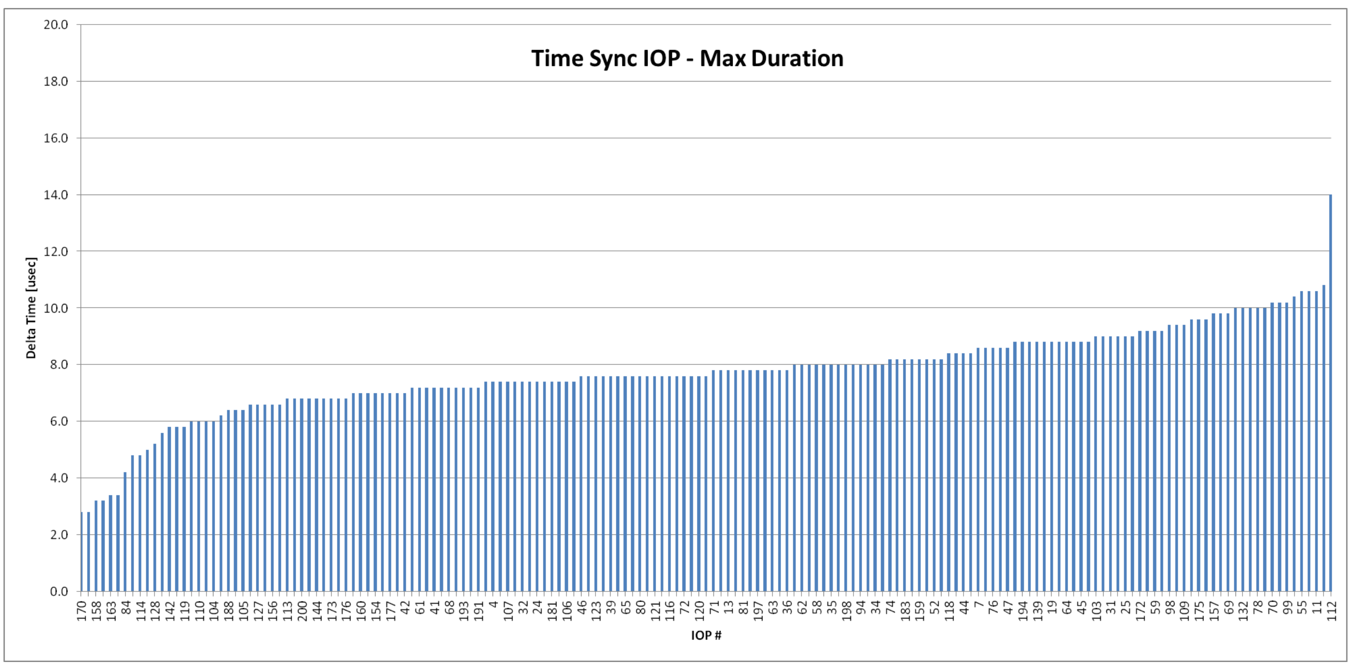


**Figure 11. IOP – Max Delta Time Measured Between Two Devices for all APs**

### 6.3 Robustness

All results shown above have been obtained in the clean environment without any interferers on the tested channel. However, the open air environment with a presence of some additional WLAN devices cause has an impact on WLAN devices behavior in terms of air access and accuracy on transmission.

To simulate an equivalent open air environment, 35 APs that operate on the same channel have been activated. A presence of such high number of APs on the same channel caused a hardness to access the air, which lead to Beacons' jittering from the target transmitting time and some Beacons to be not transmitted at all.

Five APs have been tested in two conditions: clean environment and congested environment. The measurement itself was the same: 1000 delta time measurement for each scenario. The results did not show any worsening in comparison between the two scenarios as can be shown in Figure 12.

Figure 13 and Figure 14 show examples of APs' beacons jittering and non-transmitted beacons in the congested environment.

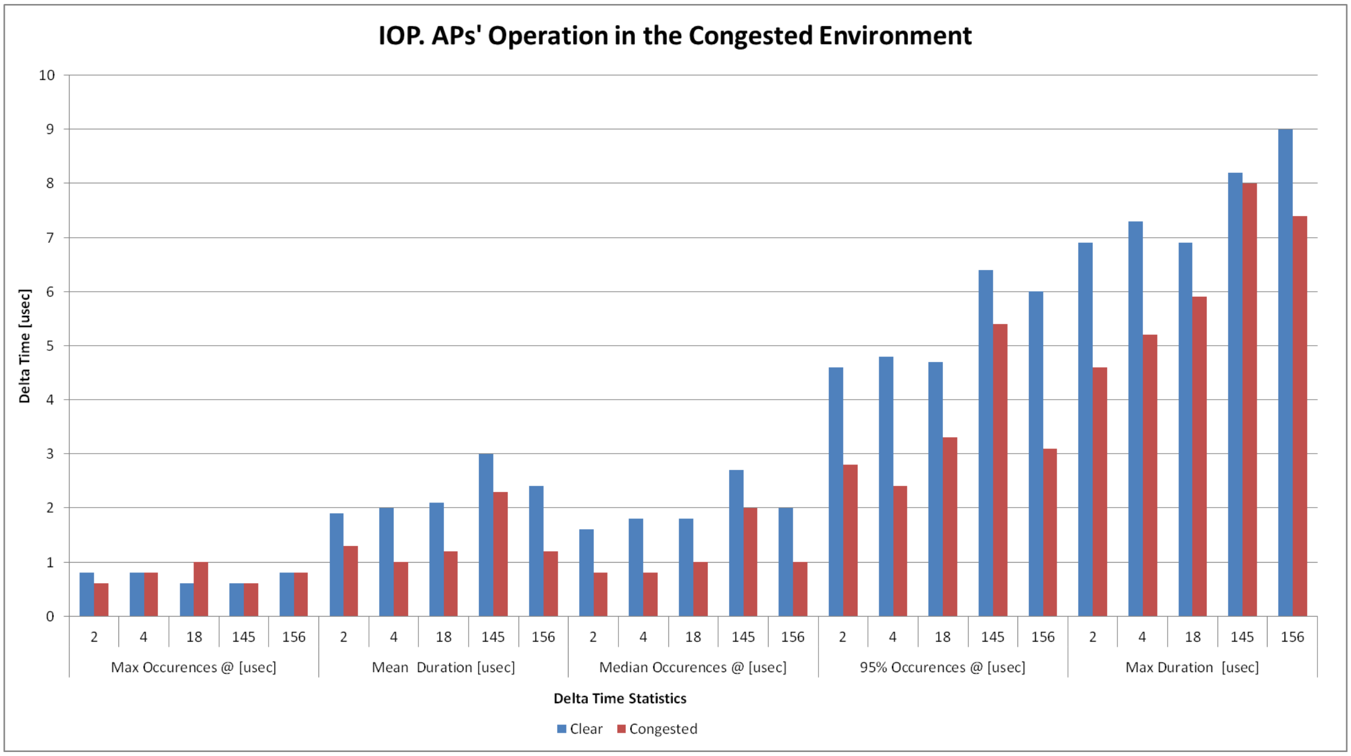


**Figure 12. IOP – Clean vs Congested Environment Results**

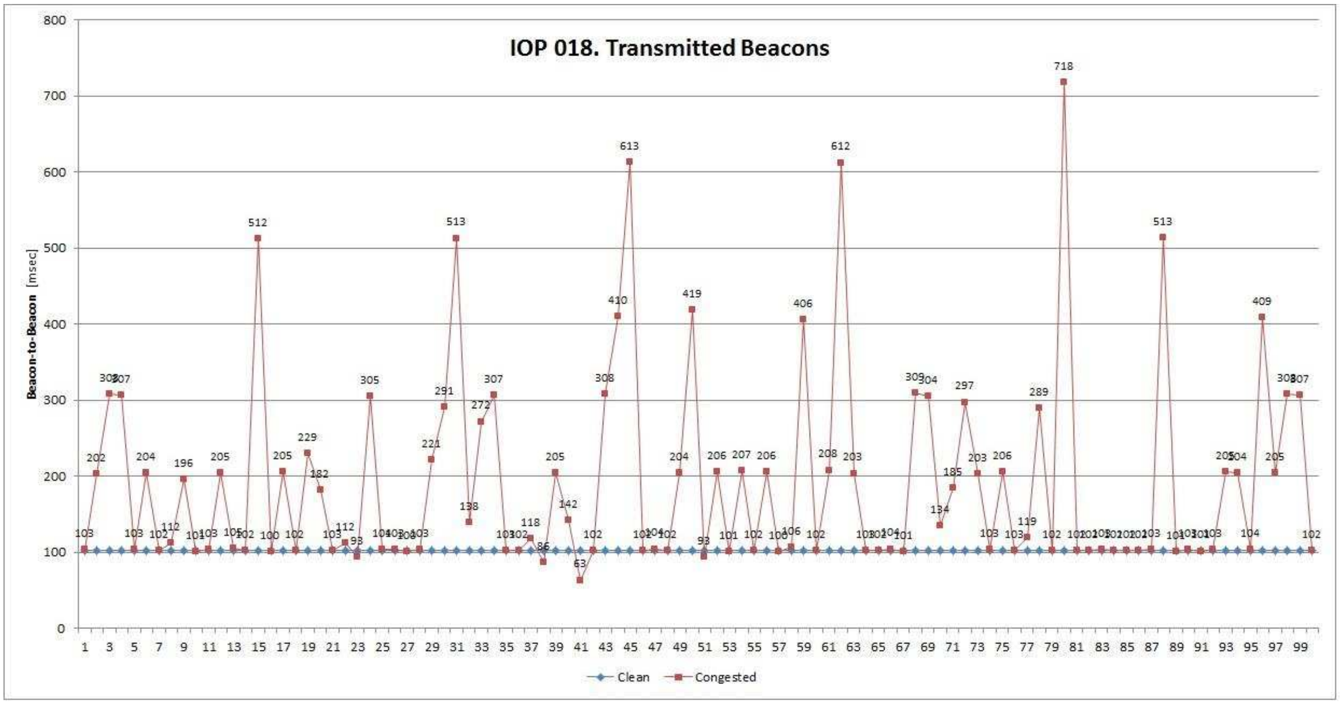


**Figure 13. IOP 018 – Beacons Behavior in the Congested Environment**

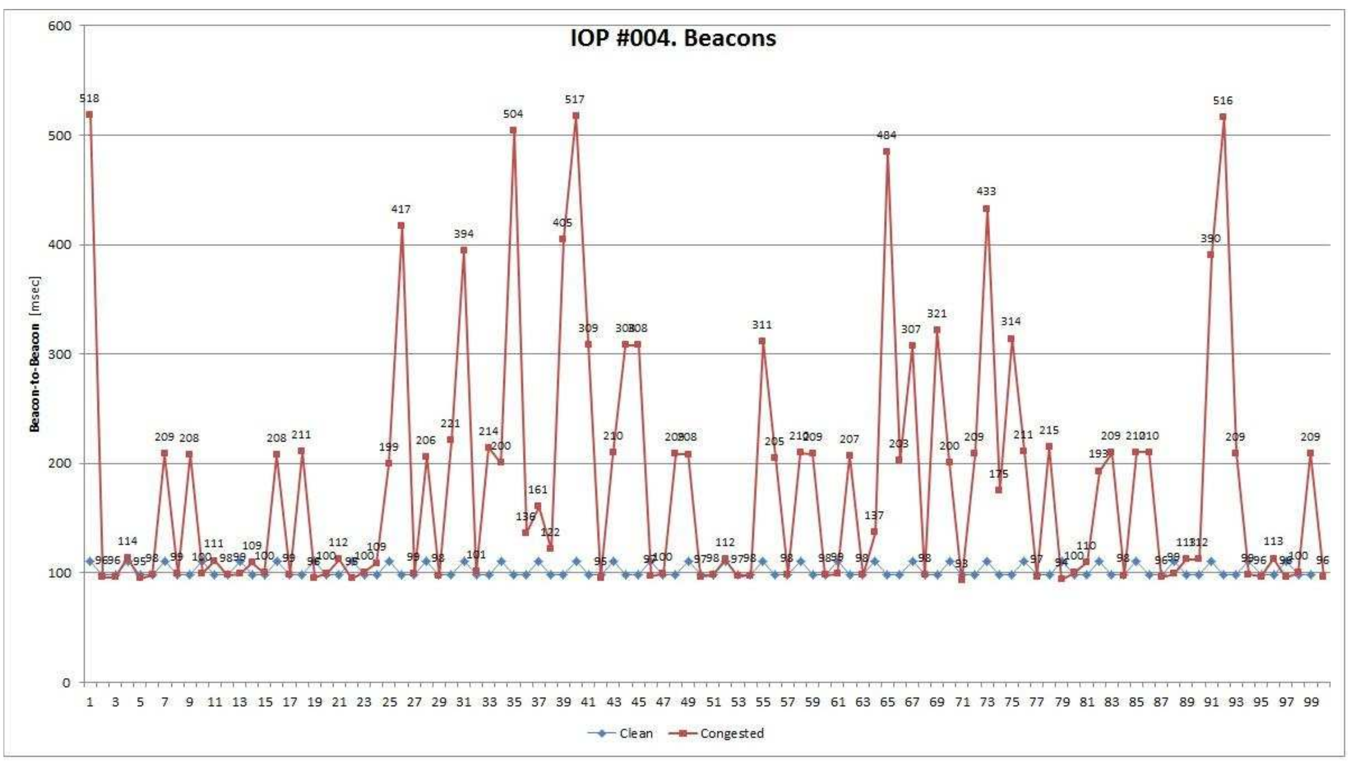


**Figure 14. IOP 004 – Beacons Behavior in the Congested Environment**

### 6.4 Stability

To verify devices' synchronization over the time, the devices have been tested over weekend. The tested scenario was TCP Rx to both devices. The results show that:

- TCP Rx run continuously without throughput jittering or degradation
- No disconnects from the connected AP
- No other abnormal behavior
- The delta time between two devices has remained stable around 1-2 µsec

# Revision History

| Changes from Original (July 2015) to A Revision | Page |
|---|---|
| • Update was made in Section 1. | 3 |

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.