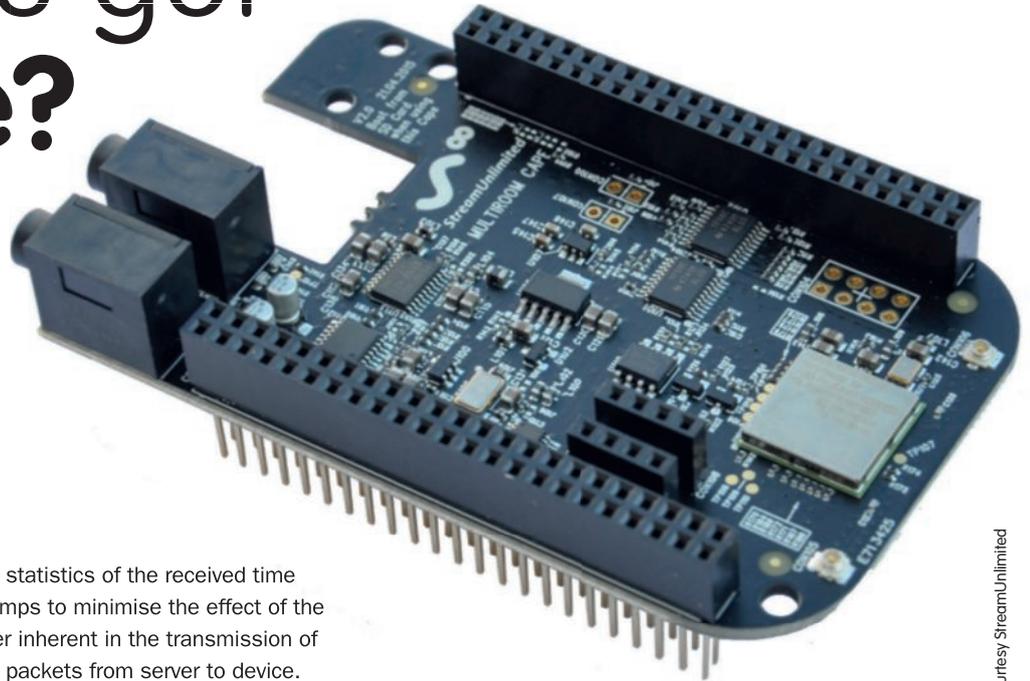# Have you got the time?

Using WiLink 8 as a reference clock for the network time protocol. By **Iain Hunter**.

One of the challenges facing those developing networks featuring wirelessly connected devices is maintaining a common clock time. A popular way of doing this is by using Network Time Protocol (NTP), a standard tool used in all types of computing platforms.

Devices in a system drift apart over time because their clocks are all slightly different. While the crystals that provide their main clock are very accurate, they are specified with a variation in parts per million (PPM). So, even if crystals only differ by 1PPM, the time calculation on each device could differ by 3.6ms after just one hour.

NTP works by querying several highly accurate NTP servers on the Internet. Over time, the device's NTP daemon uses these periodic time updates to improve the operating system's calculation of local device time, either by:
• applying a step change to device time so it matches server time. This infrequent occurrence corrects larger time differences, or.
• monitoring the drift between device time and server time as the crystals on each will have slightly different characteristics and thus frequency. It uses many samples to understand

the statistics of the received time stamps to minimise the effect of the jitter inherent in the transmission of the packets from server to device. It prioritises accuracy over speed of convergence and so can take several hours to lock. The Linux kernel provides specific APIs for NTP to update the system clock on the device based upon the calculated 'drift' between the device crystal and the server crystal.
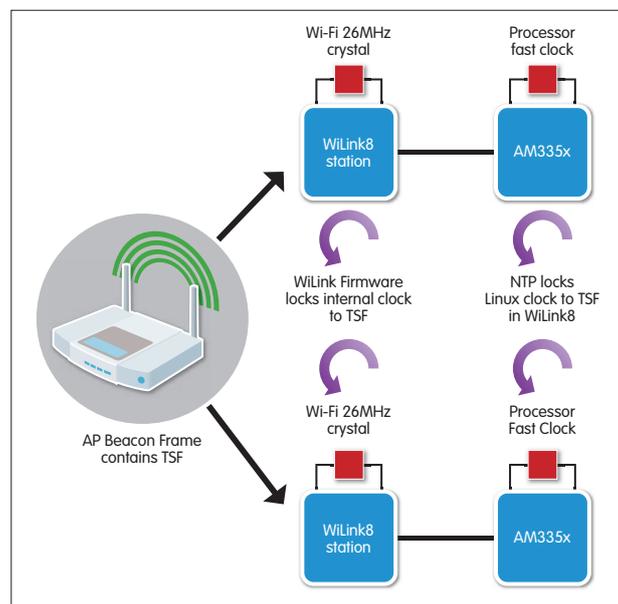
When connected by Ethernet, there is very little variation in the time taken

Above: TI's WiLink8 module features in the BeagleBone Wi-Fi audio cape from StreamUnlimited
Fig 1: How Wi-Fi connected devices can calculate their own real time

for updates to arrive at the embedded device. This means that NTP can achieve sub millisecond accuracy. Typically, when a Wi-Fi connection is used, it can take longer for packets to arrive over the wireless link and this will degrade the accuracy of the updates. For some applications, this could mean the clock accuracy will not be good enough and it will not be possible to include Wi-Fi in the system.

But NTP also provides a method for a device to get its time from a local reference clock, rather than from a server over the Internet, avoiding the problem of variable transmission times. These reference drivers are typically radio receivers connected to the device that acquire time information from GPS or national clock broadcasts.

A reference clock driver for NTP uses timing information received by Texas Instruments' WiLink 8 from a Wi-Fi access point (AP). This Time Sync feature can be used to provide a common clock for all stations connected to the AP, with an error of
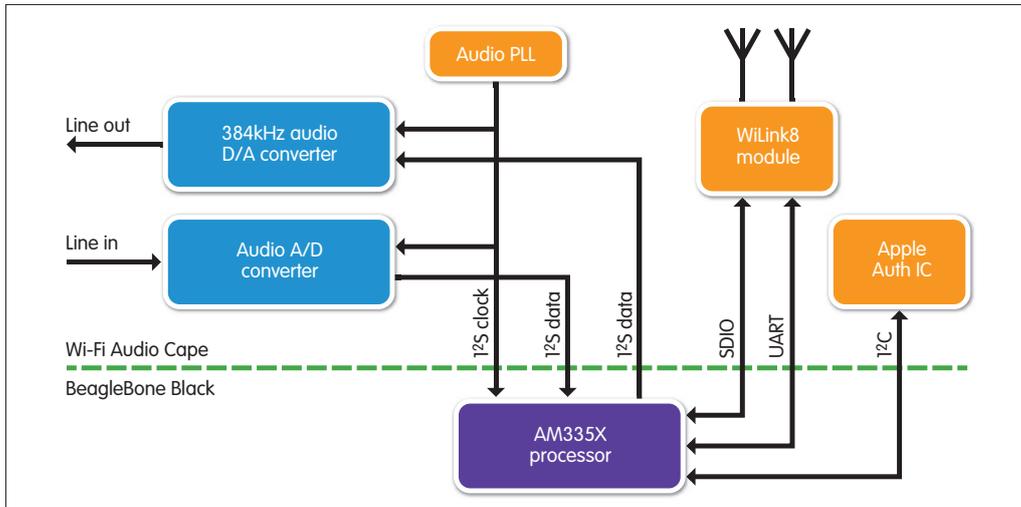
Line out

Audio PLL

384kHz audio
D/A converter

Line in

Audio A/D
converter

WiLink8
module

Apple
Auth IC

I²S clock | I²S data | I²S data | SDIO | UART | I²C

Wi-Fi Audio Cape
BeagleBone Black

AM335X
processor

Fig 2: Block diagram
of StreamUnlimited's
Wi-Fi Audio Cape

less than 100μs between devices. For some applications, Wi-Fi connected devices will now be accurate enough and provide a viable alternative to Ethernet.

Each Wi-Fi beacon transmitted by the AP includes a Time Synchronisation Function (TSF) value, which is the timestamp for that beacon. However, while this only represents the time since the AP was switched on, it does provide a common time base for all the stations connected to it and can be used as the basis for a reference clock. In this scenario, while the devices do not share an accurate real time, they share the AP's clock as a common reference, including any drift or error. This means host processors connected to WiLink8 devices associated to the AP can be synchronised.

The next step is to synchronise all devices connected to the AP to real time. To do this, the first device to link to the AP runs 'ntpdate'. This connects to an internet based public NTP server in order to acquire the current real time. It then calculates the offset between the real time and the TSF from the AP. This offset is then sent to each device as it connects to the AP. As each device shares the same TSF derived time, it is possible for each to add the offset and thus calculate its own real time (see fig 1).

The WL18XX driver is updated every 100ms with the current AP time derived from the TSF in the beacon, as well as the current Linux time. These two values are written into a file in the Linux filesystem so they can be accessed by user space applications. The NTP Reference Clock driver reads these values and uses them as the next samples to update its statistical estimation of real time. The output from this algorithm is a periodic update of the kernel's clock calculation to maintain synchronisation with the estimation of real time.

As noted, NTP prioritises accuracy of estimate over speed of convergence. If the application requires tighter and faster control of drift, then the WL18XX driver's samples of real time and device time can be used as the inputs for a clock estimation algorithm that meets the system requirements. In this case, NTP was chosen as a well known and accepted algorithm to control the Linux kernel clock.

An example of a different clock estimator is the Multi-room Wi-Fi audio speaker reference design, which uses a custom timing algorithm from StreamUnlimited to maintain the drift between speakers to typically

less than 20μs and to have fast convergence (see fig 2). This is essential to ensure the audio does not sound as though it is moving around the room.

As an example, imagine a board defined as the master is the first to connect and calculate the offset between TSF and NTP derived real time. A slave device will then receive the TSF offset from the master. If the master is known to have drift of 1200ppm and the slave has a drift of 1180ppm, there is a drift of 20PPM between them. However, drift can converge to less than 1ms within 40 minutes. Once it has converged, drift between the devices will generally stay at less than 100μs; if it goes out of this range, the NTP algorithm quickly corrects this.

As the subnet's concept of real time will drift from actual real time, there is a need to decide whether to maintain a link to actual real time and, if so, how this should be done to maintain the integrity of system synchronisation. In the example above, the AP has a drift of more than 1000PPM, which means it drifts from actual real time by more than 3s per hour.

Summary

The Time sync feature allows WL18xx devices to act as highly accurate reference clocks for NTP at no additional cost by reusing information that is already available in the Wi-Fi communication. NTP provides an industry standard method of managing the Linux system clock to maintain synchronisation between stations connected to the AP in the range of ±100μs. If more accurate or faster convergence is required, then the same basic information can be used to drive an application specific custom clock estimator to achieve an accuracy of ±20μs.

Author profile:
Iain Hunter is a European connectivity system applications engineer with Texas Instruments.

The source code to replicate this work and the hardware modification required are available at http://processors.wiki.ti.com/index.php/WL18xx_Clock_Synchronisation_with_NTP.