

# PLL Based Time Synchronization in Wireless Sensor Networks

Gang Zhou

College of William and Mary  
Williamsburg, VA

Email: gzhou@cs.wm.edu

Sachin Shetty

Tennessee State University  
Nashville, TN

Email: sachin.shetty@ieee.org

George Simmons

College of William and Mary  
Williamsburg, VA

Email: gsimms@cs.wm.edu

Min Song

Old Dominion University  
Norfolk, VA

Email: msong@odu.edu

**Abstract**—Time synchronization is a key component in numerous wireless sensor network applications. Most of the current software based time synchronization approaches suffer from communication overhead and lack of scalability. In this paper, we propose a hardware based approach based on voltage controlled crystal oscillator and phase locked loop techniques to achieve and maintain sub microsecond level time synchronization. Our approach does not require any exchange of synchronization messages with neighboring nodes. Performance evaluations in Matlab demonstrate sub microsecond accuracy and robustness to infrequent loss of WWVB signal. The principle advantages of our solution is scalability, accuracy, and low communication overhead.

## I. INTRODUCTION

Time synchronization is a critical component for many wireless sensor network applications [1]. For wired networks, various time synchronization mechanisms and algorithms have been proposed and extensively used over the past decade. But these synchronization techniques are not practical in WSNs for the following two reasons. First, time synchronization must be implemented in an energy efficient fashion due to the limited energy available to a battery operated sensor node. Second, limited bandwidth in WSNs renders frequent message exchanges for synchronization among sensor nodes impractical. So, synchronization schemes like Network Time Protocol and Global Positioning System are not suitable for WSNs because of complexity and high energy cost. In wireless sensor networks, software based time synchronization protocols have also been developed. Unfortunately, software based protocols usually require periodic exchange of messages to estimate clock drift. For example, FTSP [5] utilizes unidirectional broadcast messages for time synchronization, which generates  $2n$  messages in presence of a central node and  $n(n-1)$  messages without a central node in a  $n$  node network. RATS [8] improves on the performance of FTSP by minimizing the overhead at the expense of accuracy (synchronization in the order of 200 microseconds in an indoor environment). Also all software based synchronization protocols implement linear regression to estimate clock drift. The single precision of the emulated floating-point arithmetic on a sensor node results in large quantization errors that impact accuracy. For instance on mica2 motes, the truncation error in terms of clock ticks is  $\pm 128$ , and depending on the resolution of the local clock, the error can vary from  $\pm 17\mu s$  to  $\pm 17.7ms$ [8]. Double-

precision operations can provide higher accuracy but requires 64-bit arithmetic which is computationally intensive. Further, the software-based schemes are accurate only for networks with limited scale. A system wide time synchronization across many hops through a sensor network will inevitably lead to poorer phase error.

In this paper, we demonstrate the feasibility of achieving and maintaining sub microsecond level time synchronization using a Voltage Controlled Crystal Oscillator (VCXO) with frequency and phase lock loop (PLL) techniques implemented in hardware for wireless sensor nodes. The major contributions of this paper are:

- There have been very few hardware-based approaches to implement time synchronization in WSNs. Our hardware-based time synchronization scheme achieves better scalability, robustness, and accuracy than software based techniques.
- Our Voltage controlled crystal oscillator (VCXO) ensures long-term frequency stability and millisecond level accuracy in the absence of reference pulses for short duration of time. The timing module and the window control elements in the VCXO circuit ensure that the clock quickly regains microsecond level accuracy within a short period of time.
- Our novel Phase Detector component has been designed to take into account frequency scaling issues due to frequency mismatch between the VCXO and the reference pulse. Robustness to power supply and thermal noise have been implemented by a third-order loop filter.
- We implemented and evaluated the PLL/VCXO model in Matlab/Simulink. Extensive performance evaluation demonstrates that the PLL/VCXO model exhibit synchronization in the order of few microseconds and robustness to loss of reference pulse.

The organization of the remainder of this paper is as follows. In Section 2, we describe the PLL based time synchronization design. Simulation results are presented in Section 3. Finally, in Section 4, we draw our conclusions and describe future work.

## II. PLL BASED TIME SYNCHRONIZATION DESIGN

The goal of the PLL based time synchronization design is to achieve a network wide synchronization of the local

clocks of sensor nodes without exchanging synchronization messages. PLL based time synchronization schemes have been successfully implemented in military and aerospace programs [9]. VCXO based PLLs provided critical timing for all-weather formation flying. Time synchronization based on a PLL in a WSN works as follows: The hardware clock at each sensor node is an output of the VCXO. The voltage applied to the oscillator comes from a phase detector whose output is proportional to the phase error between the phase of its clock (the output of the VCXO it is controlling) and a reference signal generated by the WWVB atomic clock broadcast (an out-of-band time synchronization source) [11]. WWVB is a special NIST time signal radio station near Fort Collins, Colorado. WWVB is the station that radio-controlled clocks throughout North America use to synchronize themselves. The signal transmitted from WWVB is a continuous 60 kHz carrier wave, 50 KW signal transmitter accurate to less than 1 microsecond of Coordinated Universal time (UTC). Thus, by adjusting the frequency of each individual clock to the WWVB signal, the clocks can always be kept in lock-step with respect to one another. The main components of our PLL based design are Voltage Controlled Crystal Oscillator (VCXO) and Phase Locked Loop (PLL). While we are not aware of any studies of Voltage Controlled Crystal Oscillators(VCXO) in WSNs, they are the common element in PLLs and are not price prohibitive.

Fig 1 illustrates the three main components of our design: Phase Detector, Loop Filter, and VCO. In the next few sections, we will present the design of each of these three components.

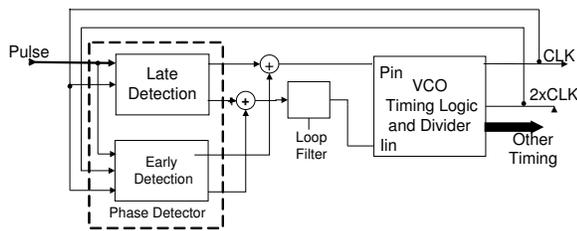


Fig. 1. Modified PLL Model

### A. Phase Detector Design

The phase detector operates by determining the relative time between the leading edge of the Reference Pulse and the falling edge (1 to 0 transition) of the 1 Hz CLK, and causing a corrective voltage to be applied to the VCO. If the Pulse occurs while the CLK is Hi (a logic 1), the CLK is considered Late; if the Pulse occurs while the CLK is Low (a logic 0), the CLK is considered Early. A late CLK is moved toward the Pulse by increasing the frequency of the VCO(increasing the voltage), and an early CLK is moved toward the Pulse by decreasing the frequency(decreasing the voltage) of the VCO. If the Pulse does not occur, no change is made to the VCO input voltage. The Phase Detector output is a sequence of CLK period pulses whose amplitude is a direct measure of the phase error; positive for Late CLK, and negative for Early CLK. The

timing differences between Late and Early CLK is shown in Fig 2 and Fig 3. Late detection begins with the Pulse leading edge, while early detection ends on the Pulse leading edge. Thus two circuits are required: one for measuring the Early condition, and a second distinct circuit for measuring the Late condition.

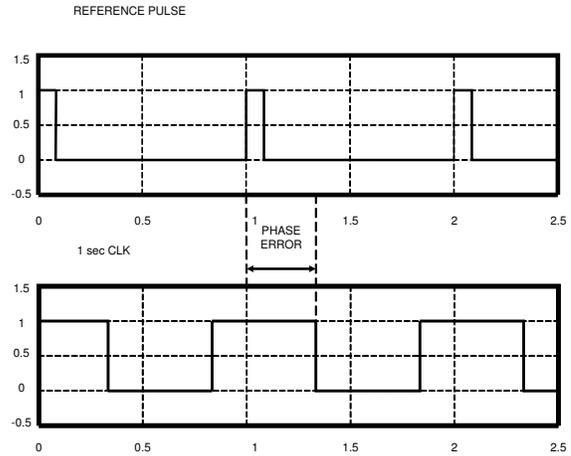


Fig. 2. Late CLK Timing

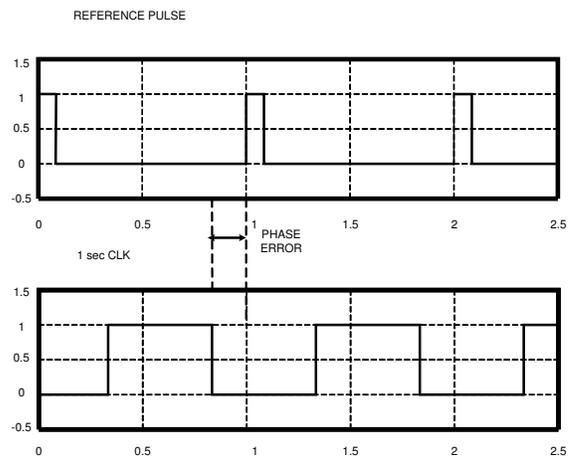


Fig. 3. Early CLK Timing

While the Late Detection and Early Detection circuits do not share individual components, they use a similar methodology. A latch is set to a 1 by the Pulse leading edge; on the next CLK transition, the 1 is shifted to a second latch using the 2xCLK, and the first latch is cleared. The second latch provides the 'enable' for the corrective signal. On the next 2xCLK 1 to 0 transition, the second latch is cleared. Thus, the corrective signal is enabled during the CLK half-cycle following the Pulse. As the circuit exists, time intervals are measured by integrating a constant voltage. In the late detector, the integrator is started by the Pulse and stops on the next CLK 1 to 0 transition. In the early detector, the integrator is started on the CLK 1 to 0 transition, and stops on the Pulse. Two corrective signals are provided by each circuit:

a direct 'proportional' signal; and a signal which is integrated by the 'Error Integrator'. The direct signal corresponds to the 'P' in PID controller, and the output of the error integrator corresponds to the 'I' in PID controller. Both the P and Error Integrator input signals are derived from the same time-interval error voltage and differ only by a scale factor.

1) *Late Detection*: Fig 4 illustrates the block diagram for the late detector. A late CLK means the VCO frequency must be increased, so the detector output is positive. When the leading edge of the Pulse occurs while CLK is a 1, a latch is set starting the late detection integrator in the positive direction. When the CLK transitions to 0, the latch is reset to 0, stopping the integrator. The state of the latch is shifted into the 'Late Enable' Flip-Flop using the 2xCLK. By using the 2xCLK to shift state into the Late Enable Flip-Flop, a signal race condition with the latch is avoided. The integrator maintains a constant value until the next CLK 0 to 1 transition, when it is reset to 0 V. The enable flip-flop switches the outputs from 0 V to the positive value of the integrator. If a pulse does not occur while the CLK is a 1, no action occurs.

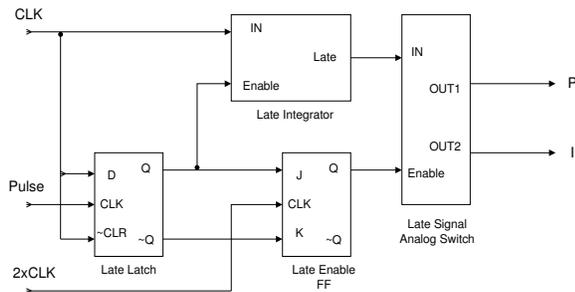


Fig. 4. Late Detector

Fig 5 shows the action of the PLL when the CLK is late. As shown in the first two traces, the Pulse occurs while the CLK is high, setting the Late Latch to a 1. The third trace shows the Late Latch, which is a 1 for the time difference between the leading edge of the pulse and the 1 to 0 transition of the CLK. The time difference is converted to a voltage by the Late Integrator, which is enabled by the Late Latch. The integrator output is shown in the fifth trace. When the Late Latch goes to 0, the integrator stops and holds its voltage until it is reset. The integrator 'reset' is edge triggered, and is caused by the 0 to 1 transition of the CLK. The sixth trace shows the Late Detection circuit positive output signal which is enabled by the Late Enable Flip-Flop. Fig 5 also shows how the PLL corrects the phase error. At the first Pulse, the CLK is late by approximately 250 msec causing a 500 ms, +1 volt signal to be applied to the Error Integrator and the VCO, thus increasing the frequency. At the second Pulse, the CLK is approximately 200 ms late, causing a 500 ms, +0.8 volt signal. And so forth.

2) *Early Detection*: Fig 6 illustrates the block diagram for the early detector. An early CLK means the VCO frequency needs to be reduced, so the detector output is negative. The early integrator is enabled by (Not-CLK), and (Not-Early-Enable-FF) AND (Not-Early Latch). On the 1 to 0 transition of

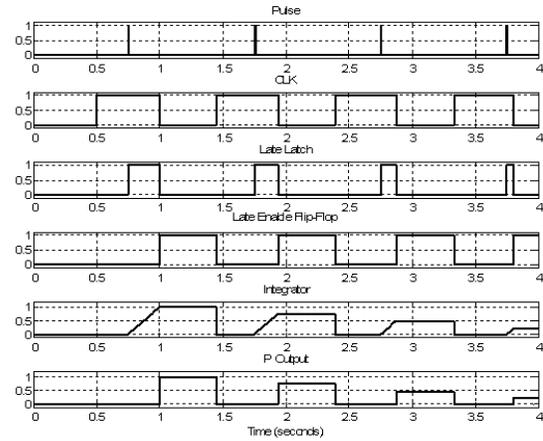


Fig. 5. Late CLK Signals

the CLK, the early integrator is reset to 0v and is enabled (the early-latch and FF were already reset to 0 ). When the leading edge of the Pulse occurs while the CLK is a 0, the early latch is set to 1 stopping the integrator. On the next CLK 0 to 1 transition, the state of the latch is shifted into the enable flip-flop using the 2xCLK, and the early latch is reset. The integrator maintains a constant value until the next CLK 1 to 0 transition, when it is reset to 0v. If a pulse does not occur while CLK is a 0, the integrator runs until the CLK changes to a 1. But, since the early latch was not set, the enable flip-flop is not set, and the outputs remain at 0v.

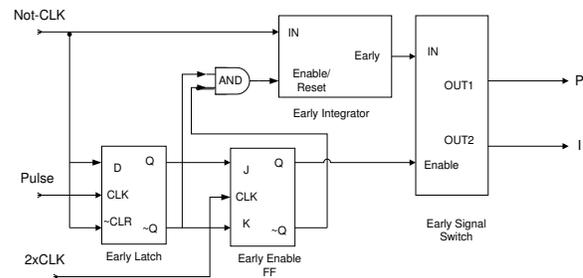


Fig. 6. Early Detector

Fig 7 shows the action of the PLL when the CLK is early. As shown in the first two traces, the Pulse occurs while the CLK is low, setting the Early Latch to a 1. The third trace shows the Early Latch, which is a 0 for the time difference between the 1 to 0 transition of the CLK and the leading edge of the pulse. The time difference is converted to a voltage by the Early Integrator, which is enabled by ANDing Not-Early Latch with Not-CLK. The integrator output is shown in the fifth trace. When the Early Latch goes to 0, the integrator stops and holds its voltage until it is reset. The integrator 'reset' is edge triggered, and is caused by the 1 to 0 transition of the CLK. The sixth trace shows the Early Detection circuit negative output signal which is enabled by the Late Enable Flip-Flop. Fig 7 also shows how the PLL corrects the phase error. At the first Pulse, the CLK is early by approximately

250 msec causing a 500 ms, -1 volt signal to be applied to the Error Integrator and the VCO, thus decreasing the frequency. At the second Pulse, the CLK is approximately 200 ms early, causing a 500 ms, -0.8 volt signal. And so forth.

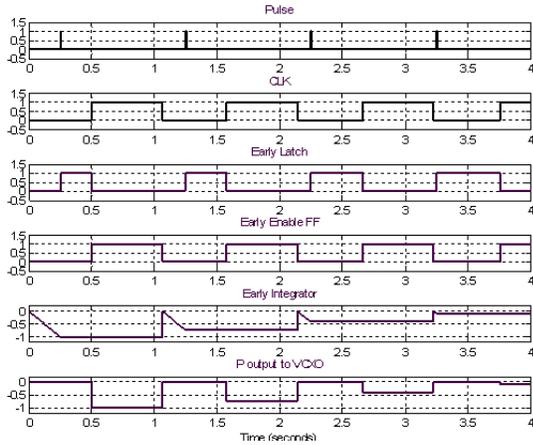


Fig. 7. Early CLK Signals

### B. VCXO

The long term frequency stability of our design can be dramatically improved by replacing the VCO with a VCXO. Changing the VCO to a VCXO requires additional circuitry to reset the Divider so that the CLK is coarsely set within 1 ms of the external Pulse. The block diagram for the changed circuitry is shown in Fig 8. The Early and Late Detectors are unchanged. The oscillator block includes the VCXO, a circuit to sum the P and I inputs; and a sine to square wave converter. In addition to the nominal 1 sec CLK, the Timing Module generates a “window” control signal, +/- 2 ms, centered on the CLK 1 to 0 transition. Pulses which occur during the window are passed to the Early/Late circuits. Pulses occurring outside the window are counted by the “miss” counter. Four consecutive pulses outside the window causes the CLK to reset within 1 ms of the pulse. (A higher frequency oscillator will allow resetting to 10 us.). Any Pulse occurring inside the window resets the “miss” counter. Fig. 9 shows the details of the coarse sync timing process.

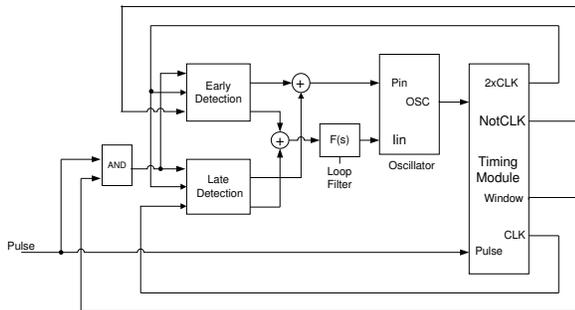


Fig. 8. PLL with VCXO

Fig. 9 shows Coarse Sync timing signals when the CLK is 200 ms late. After 5 consecutive Pulses occur outside of the window, the entire timing chain is reset. The effect is seen in trace 2. As identified by the pointer, the CLK cycle is reduced, bringing the CLK 1 to 0 transition to within 2 ms of the Pulse leading edge. Thereafter, all Pulses occur in the Window. (During later testing, the number of ‘out of window’ pulses was changed from 5 to 4). Fig. 10 shows timing expanded at the Reset. Trace 2 shows that one CLK cycle has been shortened by 200 ms. Trace 3 is the Reset signal. Trace 4 expands the reset signal. The timing chain is a synchronous counter. The Reset signal is generated by gating the VCO output.

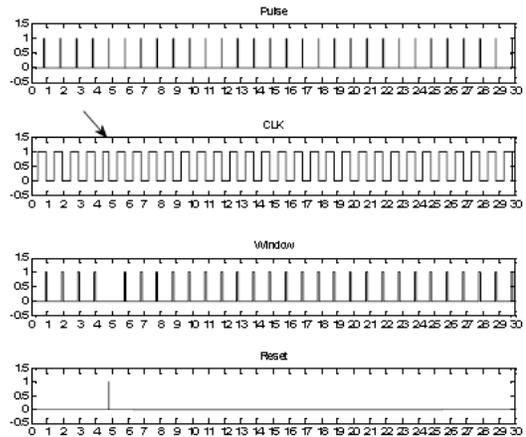


Fig. 9. Coarse Sync Timing

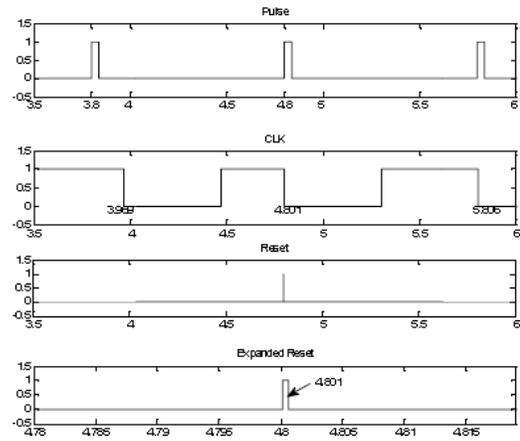


Fig. 10. Coarse Sync Timing Reset

### III. PERFORMANCE EVALUATION

To validate the effectiveness of our time synchronization approach, the PLL model was simulated using Matlab and Simulink. The effectiveness of the PLL model was measured using the following three performance metrics: time taken

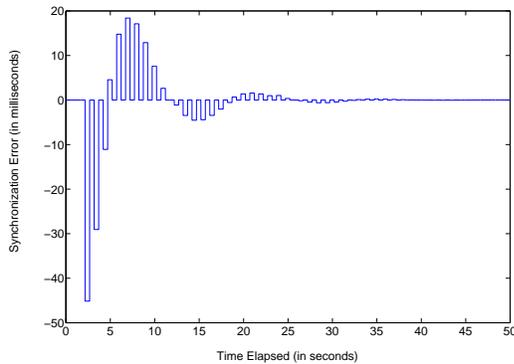


Fig. 11. Synchronization error for CLK 50 ms early.

for synchronization, estimation of the synchronization error, and robustness to loss of reference pulse. To measure the synchronization error, the voltage at the Pin input of the VCO was measured. The pin voltage provides the amount of synchronization error. The error detector gain for the Pin input is 8 volts per second. The window control signal generated by the Timing module is set to  $\pm 50$  ms. The conversion between the Pin Volts(V) to Error Time (s) is given by the following relationship:  $ErrorTime = PinVolt/8$ . For instance, if the observed pin voltage is 4 V, the corresponding synchronization error is 500 ms, 125 ms for 1V, 500 ns for  $4 \mu V$ , and so on.

In Fig. 11, we illustrate the synchronization error achieved by our PLL model when the CLK starts 50 ms early relative to the first pulse. The pulses occur at exactly 1 second intervals. The first CLK 1 to 0 transition occurs 50 ms early relative to the first pulse. The synchronization error was computed by measuring the voltage across the Pin. Fig. 11 illustrates the synchronization accuracy of our model during the entire time period of 50 s. In presence of a Early CLK, the early and late detectors provide two signals for correcting the VCXO frequency. The two signals are applied to the loop filter and the VCXO respectively. The Early CLK detector provides a negative voltage signal to decrease the VCXO frequency. When the CLK is ahead of the pulse, the Late CLK detector generates a positive signal to increase the VCXO frequency. The vertical bars in the figure correspond to the voltage (proportional to the synchronization error) applied to the VCXO to correct the clock phase. During the first 5 seconds of simulation time, the Early CLK Detector is generating a negative signal and a positive signal is generated by the Late CLK detector during the next 5 seconds. This process alternates throughout the rest of the simulation. It is clear from the figure that our model achieves millisecond level accuracy within the first 20 seconds of the simulation. Fig. 11 b demonstrates that our model synchronizes the clock to 1 microsecond level accuracy within 50 seconds of simulation time.

Table 1 compares the performance of our model with FTSP. We measure the amount of time and overhead to achieve synchronization from 1 millisecond to 1 microsecond. The message overhead in FTSP is the amount of resynchronization

TABLE I  
SYNCHRONIZATION ERROR

Accuracy ( $\mu s$ )	FTSP		PLL	
	Time (hh:mm)	Energy (mJ)	Time (s)	Energy (mJ)
100	00:08	835	37	171.97
10	3:10	19830	46	172.21
5	3:20	20604	51	172.21
1	4:00	25050	65	172.21

messages sent by each node every 30 seconds. The energy overhead in FTSP is only due to the transmission of the synchronization messages by a single sensor node. The energy computations were derived from the Micaz Crossbow motes characteristics. We considered a transmission power of 0dBm, a power supply of 3 V, and a current draw of 17.4 mA [12]. The energy overhead in our model is computed by measuring the voltage applied to the VCXO. The current consumed by the VCXO is 4 mA. We observe that our model is capable of achieving microsecond level accuracy within a minute, while it takes 4 hours to achieve the same level of synchronization in FTSP. The energy consumption of a single sensor node in FTSP is very high due to the excess power consumed during the transmission of resynchronization messages every 30 seconds. Also, in FTSP, the energy consumption will be higher, if the received and transmission messages are considered for a network of sensor nodes. As there are no message exchanges in our model, energy consumption for our model will remain constant for a large network of nodes. Our model is also capable of providing nanosecond level accuracy within 200 seconds. But for most sensor network applications which desire at most 1 millisecond - 1 microsecond level accuracy, our model is capable of providing the required accuracy within a minute.

Next, we evaluate the robustness of our approach to loss of reference pulse. As our model relies on the WWVB signal for the reference pulse, a hardware implementation of our model will use a off-the shelf WWVB receiver to detect the broadcast signal. In indoor and rugged environments, it is possible that the WWVB receiver fails to detect any signal in certain areas (in a hardware design this issue can be circumvented by using a carrier-current AM broadcast). Fig. 12 (a) illustrates the synchronization error after the pulse was lost for varied periods of time. For the four scenarios depicted in the figure, pulse was lost when the synchronization error was 0.1 milliseconds. In the absence of a reference pulse, the Early and Late CLK detectors will not be able to generate any signal as input to the VCXO. The absence of the pulse fails to generate a positive or negative signal to correct the VCXO frequency drift in the feedback loop. Fig. 12 (b) depicts the amount of time taken to recover from a lost reference pulse. From the figure it is clear for all the four scenarios, the model regains 1 microsecond accuracy approximately within 30 seconds. Once the pulse is regained, the Early and Late detectors promptly create the signals to correct the VCXO frequency drift. The ability to promptly regain 1 microsecond accuracy is independent of the amount of time with no pulse.

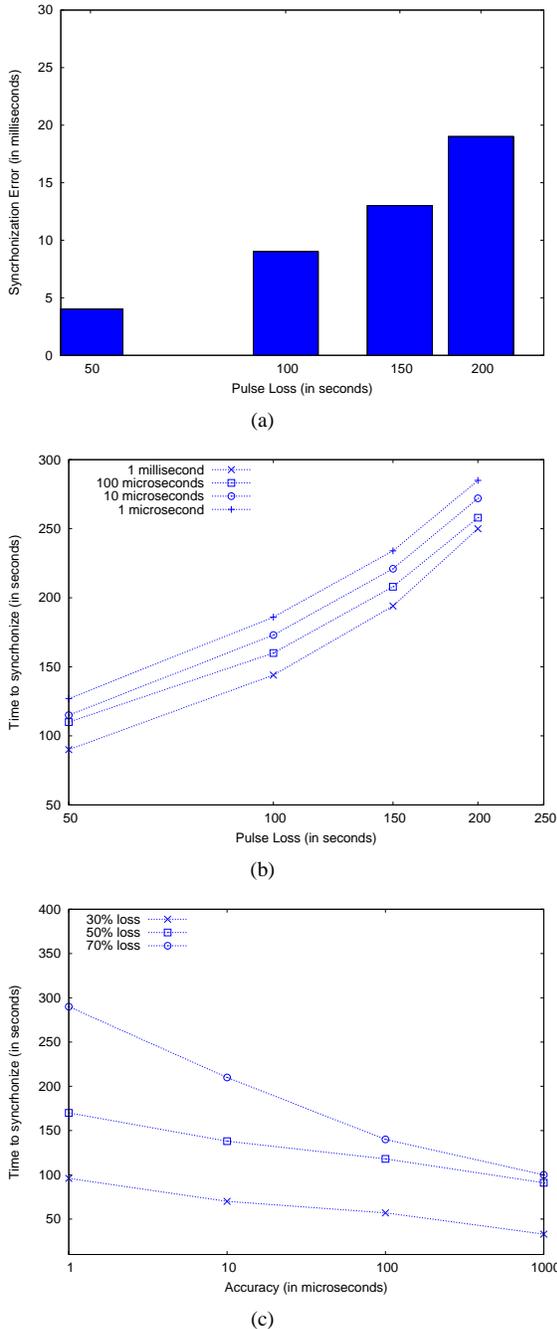


Fig. 12. Robustness to loss of lock.

Finally, we evaluate the robustness of our approach to probabilistic loss of pulses. This scenario could take place if the sensor node is placed in a mobile environment where the strength of the received WWVB signal varies depending on the spatio-temporal dynamics of the surrounding environment. In Fig. 12 (c), we analyze the time taken to achieve different levels of synchronization accuracy when the reference pulse is lost for infrequent periods to very frequent periods of time. The pulse losses were generated by a Bernoulli random process. When the pulse is lost for 30 % of the time, the number

of times VCXO frequency drift is corrected decreases. But this decrease is not significant, as the time taken to achieve 1 microsecond synchronization only increases by 34 %. But as the pulses become more infrequent, the time taken to achieve 1 microsecond accuracy increases exponentially. This is due to the fact that the VCXO frequency drift corrections take place sporadically leading to higher synchronization errors. However, it should be noted that our model can still achieve 1 microsecond level accuracy within 300 seconds in spite of losing pulses for 70 % of the time. We have also observed that losing pulses beyond 70 % of the time adversely affects the ability of our model to achieve 1 microsecond accuracy.

#### IV. CONCLUSIONS

This paper presents a PLL/VCXO based time synchronization model to achieve microsecond level accuracy in WSNs. This method was inspired by the success achieved by atomic clocks which are synchronized by a WWVB atomic broadcast pulse. We adapt to the realistic parameters of the nonideal wireless environment by implementing a novel Phase Detector and Voltage Controlled Crystal Oscillator module. These modifications allow the sensor node to have frequency stability for long term synchronization and robustness to loss of WWVB signal. Through Matlab simulations, we demonstrate that our model is capable of achieving sub microsecond level accuracy within 65 seconds in spite of a 50 ms LATE CLK. Our model also demonstrates the capability to quickly recover in the absence of reference pulses. We plan to implement a hardware prototype of our model to verify that the accuracy, energy consumptions, and size is applicable to be implemented on a wireless sensor node.

#### REFERENCES

- [1] T. He, S. Krishnamurthy, L. Luo, T. Yan, and B. Krogh L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, T. F. Abdelzaher, J. Hui. VigilNet: An Integrated Sensor Network System for Energy-Efficient Surveillance. ACM TOSN, 2006.
- [2] J. Elson, L. Girod, and D. Estrin. Fine-Grained Network Time Synchronization Using Reference Broadcasts. ACM OSDI, 2002.
- [3] H. Dai and R. Han. Tsync: A Lightweight Bidirectional Time Synchronization Service for Wireless Sensor Networks. ACM MC2R, 2004.
- [4] S. Ganeriwal, R. Kumar, and M. B. Srivastava. Timing-Sync Protocol for Sensor Networks. ACM SenSys, 2003.
- [5] M. Maroti, B. Kusy, G. Simon, A. Ledeczi. The Flooding Time Synchronization Protocol. ACM SenSys, 2004.
- [6] A.-S. Hu and S. D. Servetto. Asymptotically Optimal Time Synchronization in Dense Sensor Networks. ACM WSNA, 2003.
- [7] Y. W. Hong and A. Scaglione. A Scalable Synchronization Protocol for Large Scale Sensor Networks and Its Applications. IEEE JSAC, 2005.
- [8] S. Ganeriwal, D. Ganesan, H. Shim, V. Tsitsis, M. Srivastava. Estimating clock uncertainty for efficient duty-cycling in sensor networks. ACM SenSys, 2005.
- [9] P. Johannes. A Stationkeeping System for IFR Formation Flying. Journal of The American Helicopter Society, 1961.
- [10] A. Rowe, R. Mangharam, R. Rajkumar. FireFly: A Time Synchronized Real-Time Sensor Networking Platform. Wireless Ad Hoc Networking: Personal-Area, Local-Area, and the Sensory-Area Networks, CRC Press Book Chapter, 2006
- [11] <http://tf.nist.gov/stations/wwvb.htm>
- [12] [http://www.xbow.com/products/Product\\_pdf\\_files/Wireless\\_pdf/MICAz\\_Datasheet.pdf](http://www.xbow.com/products/Product_pdf_files/Wireless_pdf/MICAz_Datasheet.pdf)
- [13] J. Allison, R. Hormigo, E. Jovanov. A Low-Power Geographically Distributed Data Acquisition System with WWVB Synchronization. IEEE SSST 2003.