Control and Estimation Theory in Ranging Applications

by

Justin A. Echols

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2020 by the
Graduate Supervisory Committee:

Daniel Bliss, Chair
Konstantinos S. Tsakalis
Spring Berman
Hans Mittelmann

ARIZONA STATE UNIVERSITY

May 2020

ABSTRACT

For the last 50 years, oscillator modeling in ranging systems has received considerable attention. Many components in a navigation system, such as the master oscillator driving the receiver system, as well the master oscillator in the transmitting system contribute significantly to timing errors. Algorithms in the navigation processor must be able to predict and compensate such errors to achieve a specified accuracy. While much work has been done on the fundamentals of these problems, the thinking on said problems has not progressed. On the hardware end, the designers of local oscillators focus on synthesized frequency and loop noise bandwidth. This does nothing to mitigate, or reduce frequency stability degradation in band. Similarly, there are not systematic methods to accommodate phase and frequency anomalies such as clock jumps. Phase locked loops are fundamentally control systems, and while control theory has had significant advancement over the last 30 years, the design of time-keeping sources has not advanced beyond classical control. On the software end, single or two state oscillator models are typically embedded in a Kalman Filter to alleviate time errors between the transmitter and receiver clock. Such models are appropriate for short term time accuracy, but insufficient for long term time accuracy. Additionally, flicker frequency noise may be present in oscillators, and it presents mathematical modeling complications. This work proposes novel $\mathcal{H}^\infty$ control methods to address the shortcomings in the standard design of time-keeping phase locked loops. Such methods allow the designer to address frequency stability degradation as well as high phase/frequency dynamics. Additionally, finite-dimensional approximants of flicker frequency noise that are more representative of the truth system than the tradition Gauss Markov approach are derived. Last, to maintain timing accuracy in a wide variety of operating environments, novel Banks of Adaptive Extended Kalman Filters are used to address both stochastic and dynamic uncertainty.

To my mother: Susan Echols

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor Daniel Bliss. Professor Bliss regularly made time for me on short notice, and provided invaluable input and direction to my research. Professor Bliss is a true professional that provides his students with an environment that allows them to achieve things they otherwise could not. I would also like to thank my advisory committee, Dr. Konstantinos Tsakalis, who has provided me with guidance and insight since I was an undergraduate, Dr. Spring Berman and Dr. Hans Mittelmann. Moreover, I would like to thank my co-collaborators for their contributions to this thesis, including Sharanya Srinivas, who did the initial Kalman Filter modeling and coding for the HPPC project, which was used as a starting point in the work herein, Dr. Andrew Herschfelt, Dr. Srikanth Sridharan, Dr. Karan Puttannaiah and Kaustav Mondal. Additionally, I would like thank my academic advisor Lynn Pratte and Dr. Joseph Palais for their advise and guidance.

For a significant portion of my PhD journey, I worked full time as an engineer. It would be short sighted to not acknowledge my peers in industry that significantly influenced how I view algorithm design and model based systems engineering: Dr. Michael Whitehead, Dr. Steve Miller, Dr. Brad Badke, Mike King, Dr. Richard Bienz, Richard Rader and Borna Emami.

I would like to acknowledge my mother, Susan Echols, for always supporting and believing in me, as well as my grandparents, Jimmie Lee and Cynthia Echols, and the rest of my family: Savannah Echols, Alex Chiamakis, Allison Chiakmakis, Todd Snider, Jennifer Echols, Mike McGinnis and Karen McGinnis. I would like to acknowledge my friends Nathan Robb, Jefferey Brown and my partner Lindsay Mazzola, for their support and making life more enjoyable.

This research was supported in part by the National Science Foundation Graduate Research Fellowship Program. I am honored and grateful to have received this award.

TABLE OF CONTENTS

CHAPTER Page

APPENDIX Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

1.1 Introduction

With the advent of Global Navigation Satellite Systems (GNSS) [15], ranging receivers have received considerable attention over the last 30 years. Such systems allow for 24/7 full earth coverage, where a ground (or space) user can use a receiver to determine their current position, velocity and time. Spacecraft based navigation systems are fundamentally ranging systems - a system where the spacecrafts position, velocity, time and a known (sometimes) repeating sequence is appended to a radio-frequency (RF) signal. The ranging system searches the sky for such signals. Once a signal is found, a local time-stamp is computed and compared against the time-tag of the transmitted signal. This process is known as time of arrival (TOA) estimation. If time offsets between the transmitter system and the receiver system didn't exist, visibility of just 3 satellites would be necessary to produce a position, velocity and time (PVT) solution of the receiver system, using the distance formula. However, time-offsets between the time-keeping systems of both the transmitting spacecraft and receiver are always different, even if the atomic clocks or crystal oscillators driving the systems are physically identical. This is due to many reasons: (1) oscillators and atomic clocks are fundamentally driven by stochastic processes, even if environmental conditions are identical as well, from a mathematical perspective, different process and measurement noise seeding can create small differences, (2) environmental effects, such as temperature and vibration and (3) physically different clocks, which have varying degrees of accuracy. The presence of this time-offset makes it so that at least

4 navigation satellites must be present to determine a PVT solution, mathematically, this looks like

$$c\rho^i = \sqrt{(x^i_{Tx} - x_{Rx})^2 + (y^i_{Tx} - y_{Rx})^2 + (z^i_{Tx} - z_{Rx})^2} + \Delta t, \text{ for } i = 1, .., 4 \qquad (1.1)$$

Where c is the speed of light, and $c\rho^i$ is called the range, or pseudo-range, which just means the true range with a time-offset. Uncompensated time-offsets in range based navigation systems are catastrophic as a small time error leads a very large position error. For example, a nanosecond timing error leads to a 30 cm position error, whereas as a millisecond timing errors leads to a 300 km range error. Thus, the modeling and compensation of the atomic clocks and or oscillators noise characteristics is paramount, especially in applications that require high accuracy.

It must be stated that GNSS constellations are not the only ranging systems [34]. Ground based triangulation has been used for decades. One such example are pseudolites - ground based transmitters that transmit a signal either identical or similar in structure to that of the GNSS systems. The physics are fundamentally the same, and can be visualized in figure 1.1 Moreover, many different systems use the GNSS system for strictly time-keeping purposes. Gas stations, banks and registers at many retail companies use GPS time for transactions. With 5G technology on the horizon, GNSS "time-keeping" only receivers will become more prominent. Such receivers will be at every 5G cell-phone tower, and will need to compute time based off a ranging signal from a single satellite [7]. The GPS 1 PPS standard allows for nanosecond timing accuracy [35].

This thesis strives to provide systematic methods for modeling, simulating, and estimating atomic clocks, crystal oscillators and all local oscillators relevant to precision time-keeping and ranging. In addition to this, novel Kalman Filter methods are proposed which produce reliable estimates that are robust to environmental differences.

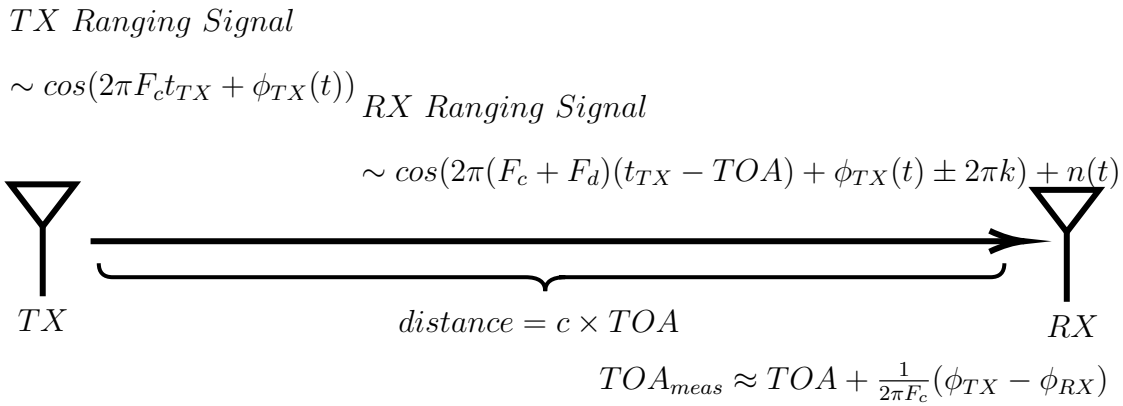

Figure 1.1: Standard Ranging System: The transmitted signal is initially offset in phase, and hence time because of the inherent noise characteristics of its Master Oscillator, denoted $\phi_{TX})(t)$. This problem is exacerbated at the receiver, and the time error between the systems oscillators appears in the TOA estimate as $\frac{\phi_{TX} - \phi RX}{2\pi F_c}$.

1.2 Contributions

The work herein this thesis strives to provide a holistic solution to timing degradation in ranging systems. In short, the novelty of this work is the following

- Systematic numerical procedures to generate realistic Oscillator noise
- Local Oscillators that minimize frequency stability degradation using the $\mathcal{H}^\infty$ control methodology
- Phase Locked Tracking Loops that are robust to high dynamics such as phase and frequency anomalies and clock jumps using the $\mathcal{H}^\infty$ control methodology
- Self Tuning Extended Kalman Filters with multiple layers of adaption, that allows for the maintenance of high accuracy in the presence of different noise characteristics and clock dynamics
- Systematic strategies for approximating and compensating colored noise constituents

this approach attacks timing and navigation accuracy from both a software and hardware perspective, empowering practitioners with previously unestablished computationally efficient "knobs" to achieve many different specifications. As such, the number of iterations in the design of range based navigation systems is decreased.

1.3 Background

Modern Control Theory is almost entirely untouched in wireless communications systems. While the work in this thesis focuses entirely on communications based navigation, the general communications systems community could benefit from the advances in control theory. Every communication system is driven by either an atomic clock or a crystal oscillator. This can be visualized in figure 1.2.

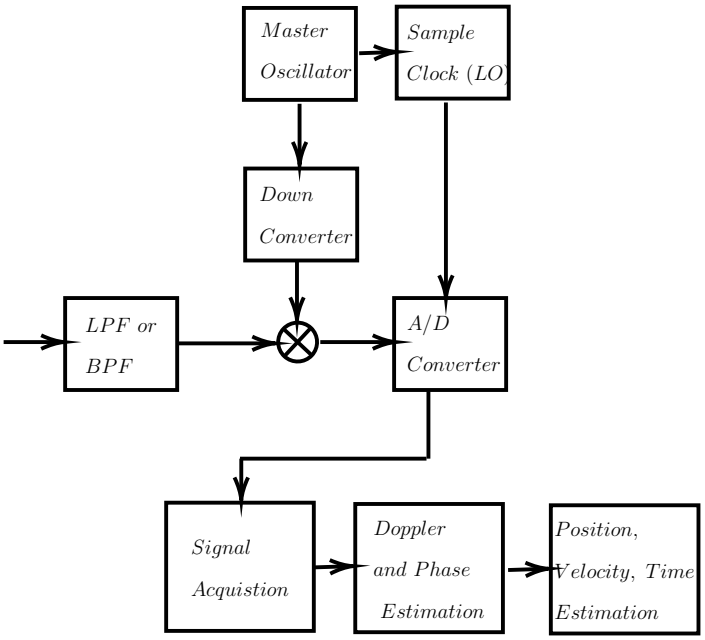

Figure 1.2: Topology of A Typical Wireless Communications System: The received signal is either bandpass or low-pass filtered. It then is downconverted to a lower frequency signal. This downconverted signal is generated by a phase locked loop driven by the Master Oscillator. Additionally, the master clock drives the sample clock, which then drives the analog-to-digital converter (ADC). Once the signal is in the digital domain, signal acquisition/detection algorithms are used to identify the signal, and doppler/phase tracking loops are used to remove the residual carrier wave, and potentially a pseudo-random code.

It must be noted that the sample clock, downconverter and tracking loops (doppler and phase estimation block) are all fundamentally control systems. In the case of the downconverter and sample clock, the Master Oscillator is converted to a typically higher frequency via a phase-locked loop. The topology of a typical phase locked loop is demonstrated in figure 1.3. From a control-systems perspective, the loop filter is the controller, traditionally denoted in the controls literature as K.

While the system to be controlled, the plant, P, is a Voltage Controlled Oscillator (VCO). In addition to inheriting all of the noise inherent to the reference frequency, the VCO has it's own noise constituents. These characteristics are typically that of a "flicker plus floor" term. To be more rigorous, these terms are known as White Frequency Noise (WPN), White Phase Noise (WPN), and Flicker Frequency Noise (FFN). These constituents will be discussed more thoroughly in Chapters 3 and 4. As such, the in-band noise of the system is *strictly worse*. The design standard has been, historically, to design for loop noise bandwidth. The work in [13] has provided formulas that allow for the achievement of a designer specified loop noise bandwidth, when the Loop Filter architecture is that of a PI, PID, or double integrator PI controller. Ad hoc methods are often used to add a notch filter in series with the loop filter, to address spurious modes. Alternatively, synthesizer designers often use classical control to design for bandwidth and phase margin. The limitation of both approaches is that many more performance metrics are paramount to phase locked loop performance and stability. There are many closed-loop transfer functions that impact the bottom-line synthesized frequency. For example, the transfer function from VCO input noise (q_{FFN} input) to synthesized frequency, and its corresponding frequency domain characteristics significantly impact the synthesized frequency - yet nothing in the literature addresses this. This transfer function has a bandpass shape, and if not addressed, can have very large amplification in the pass-band. If this transfer function is addressed explicitly, it can attenuate FFN for all frequency. Since the in-band noise of a frequency produced by a phase locked loop is always worse than it's reference frequency due to the VCO's noise characteristics, it is desirable to have a framework by which this degradation can be minimized, or traded-off against other design metrics. The work in this thesis provides a framework to shape any closed-loop transfer function using both the Generalized $\mathcal{H}^\infty$ Method, and the fixed

order $\mathcal{H}^\infty$ method. This allows the designer to tradeoff between different closed loop bandwidth, margins and filter properties. Since the loop filter order can be specified, it can be implemented using op-amps and RLC circuit components.

In the digital domain, doppler and phase estimates are computed using Frequency Locked Tracking Loops and Phase Locked Tracking loops. The purpose of such loops is to determine the residual doppler frequency and phase offsets due to relative dynamics between the receiver and the transmitter. In such cases, the VCO is replaced with a Numerically Controlled Oscillator (NCO). Once the tracking loop converges, it allows for the residual carrier wave to be wiped off the signal, such that only the transmitted data-bits or additional signal components are present. In high dynamics scenarios, such as space-to-space communications or space-to-aircraft, Doppler and phase-offset fluctuations are significant. If a frequency step (phase ramp) is present in the received signal, classical control-theory, via the internal model principle dictates that there must be a double integrator in the loop filter to track with zero steady-state error. This is highest dynamic scenario addressed in the literature [Gardner] - with a caveat that the bandwidth must be less than 25 Hz to maintain stability. However, there are situations with even more exotic dynamics, such as frequency accelerations, which lead to cubic phase dynamics. The literature does not currently address this scenario. Similarly, clock jumps and other anomalies may create even higher dynamics, which are also not addressed in the communications systems literature. The $\mathcal{H}^\infty$ methodology also allows for these instances to be directly addressed.

The previously described ranging system components work to produce a crude time-of-arrival, or ranging estimate. However, the in-band noise remains uncompensated. The fundamental tool used to address the remaining in-band noise sources, and produce a PVT solution is the Kalman Filter. Figure 1.4 shows a high-level block diagram of this process.The standard approach to address time-offsets is to append

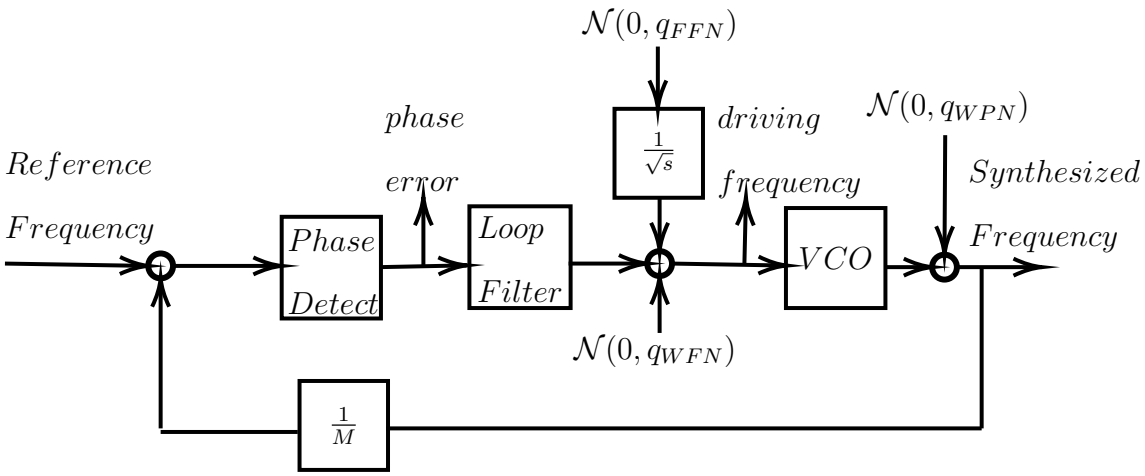

Figure 1.3: Topology of A Typical Phase Locked Loop: A reference frequency comes from the master oscillator, and goes into the closed loop system. This system is inherently non-linear, as two sinusoids enter the system, the phase detector performs a non-linear operation to compute the phase difference between the Voltage Controlled Oscillator (VCO) and reference frequency. The VCO is driven to a multiple of the frequency frequency, denoted by M.

one or two clock states - time offset and its derivative (can alternatively be interpreted as phase drift and fractional frequency off-set)to a Kalman Filter, in addition to the more deterministic ranging dynamics. Such an approach is valuable for short term stability, or short term timing accuracy, but is insufficient for longer term stability. The PVT solution from a Kalman Filter is only as good as its measurements, and the models therein.

In this thesis the methods introduced in [5, 27, 25, 26] are combined to create an adaptive filtering framework to compensate for colored noise.

1.4 Organization

This thesis is organized as follows. Chapter 2 introduces a standard three-state model of an atomic clock or a crystal oscillator. Additionally, the Allan Variance (AV) characterization of oscillators is introduced and married with the state equations. Oscillators have a colored noise component that is difficult to approximate and hence model. As such Chapter 3 addresses the challenges and realizations associated with modeling and filtering of colored noise present in oscillators. Chapter 4 discusses a brief history of Kalman Filters, and marries existing techniques to create a new methodology that is insensitive to modeling and noise uncertainty. Chapter 5 shows how to integrate an oscillator model into a non-linear navigation filter. Monte Carlo simulations are used to validate the proposed adaptive filters.Chapter 6 applies the Generalized and Fixed Order $\mathcal{H}^{\infty}$ methodologies to address the design of phase locked loops. The previously described chapters are encapsulated in Chapter 7, which is the conclusion.

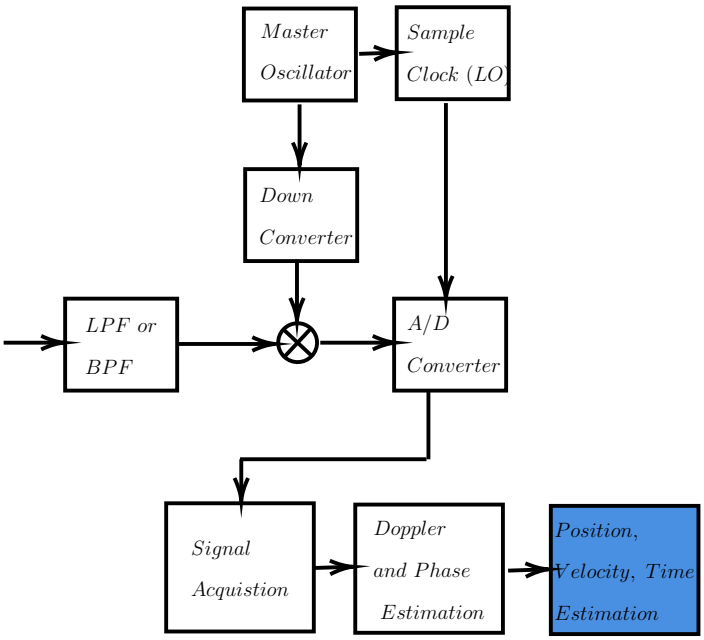

Figure 1.4: Topology of A Typical Communications Based Ranging System: Once the signal is downconverted and digitized, and residual frequency abnormalities due to relative dynamics are resolved, the measurements produced by the system - typically range or time-of-arrival, are fed into a Kalman Filter, which provides a PVT solution.

Chapter 2

OSCILLATOR MODELING

2.1 Standard Oscillator Model

Oscillator modeling has received considerable attention over the last 50 years [37, 36, 9, 32, 14]. Either an atomic clock or crystal oscillator is used as a timing reference. Pulses are measured and counted to create a precise time-keeping source. The fundamental mathematical description of a timing reference has historically been the following

$$V(t) = (V_o + \epsilon(t)) \sin(2\pi f_o t + \phi(t)) \tag{2.1}$$

where V_o is the actual amplitude, $\epsilon(t)$ is additive noise on the amplitude, and f_o is the actual frequency. From this relation, several fundamental quantities are derived. The first is the instantaneous phase ($\Phi(t)$)

$$\Phi(t) = 2\pi f_o t + \phi(t) \tag{2.2}$$

Frequency is the derivative of phase, hence the instantaneous frequency is

$$\frac{d\Phi}{dt} = 2\pi f_o + \frac{d\phi}{dt} \tag{2.3}$$

Now the accumulated timing error can be defined as $\frac{\phi(t)}{2\pi f_o}$. Moreover, $\phi(t)$ can be interpreted as time domain phase noise. In the case of ranging communications, f_o is typically the carrier frequency, F_c. In order to model, predict and estimate an oscillators phase degradation, state space methods are used. The standard model is

shown in block diagram form in figure 2.1. This is a 3-state model, which produces a quadratic function in time. Tables 2.1 and 2.2 summarize the states and noise constituents.

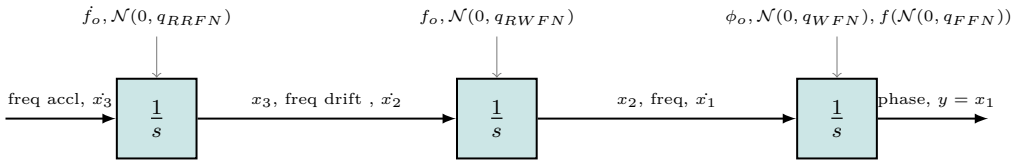

Figure 2.1: Three State Oscillator Drift Model - a standard model to characterize oscillator drift. The standard in GPS Kalman Filters is to use a lower order, two state model, emphasizing short term stability [19]. This model includes a third state, which can interpreted as frequency drift, and allows for the modeling of long term errors as well as short term [8].

Similarly, this model [8] can be captured in a state-space equation form, upon discretization, as in equation 2.5

$$\begin{bmatrix} x^1_{k+1} \\ x^2_{k+1} \\ x^3_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T_s & \frac{T_s^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^1_k \\ x^2_k \\ x^3_k \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{WFN} + u_{FFN} \\ u_{RWFN} \\ u_{RRFN} \end{bmatrix} \tag{2.4}$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x^1_{k+1} \\ x^2_{k+1} \\ x^3_{k+1} \end{bmatrix} + u_{WPN} \tag{2.5}$$

where the u's represent time series datum for the noise constituents present in both the process and the measurement. If the FFN term is included, the Kasdin Approximation [20] is typically used to generate, and hence simulate colored noise. The problem with such an approach is that it doesn't have a straight forward state-space realization and relies on batch process techniques. This allows for designers to accurately characterize a clock, but not reduce it's impact on timing degradation. Either vendor data on

the Allan Variance [2], or experiments using special test equipment can be used to generate Allan Variance data, which can then be used to compute the process noise covariance matrix through an iterative simulation process, which has the following form,

$$Q_k(T_s) = \begin{bmatrix} \sigma^2_{WFN}T_s + \frac{1}{3}\sigma^2_{RWFN}T_s^3 + \frac{1}{20}\sigma^2_{RRFN}T_s^5 & \frac{1}{2}T_s^2\sigma^2_{RWFN} + \frac{1}{8}\sigma^2_{RRFN}T_s^4 & \frac{1}{6}\sigma^2_{RRFN}T_s^3 \\ \frac{1}{2}T_s^2\sigma^2_{RWFN} + \frac{1}{8}\sigma^2_{RRFN}T_s^4 & \sigma^2_{RWFN}T_s + \frac{1}{3}\sigma^2_{RRFN}T_s^3 & \frac{1}{2}\sigma^2_{RRFN}T_s^2 \\ \frac{1}{6}\sigma^2_{RRFN}T_s^3 & \frac{1}{2}\sigma^2_{RRFN}T_s^2 & \sigma^2_{RRFN}T_s^2 \end{bmatrix} \tag{2.6}$$

where it is against noted that the noise constituents are described in table 2.2.

Table 2.1: Three State Oscillator Model Description.

State	**Physical Meaning**	**Unit**
x_1	Phase	cycle
x_2	Freq	Hz
x_3	Freq Drift	Hz/s

Table 2.2: Three State Oscillator Model Noise Constituents.

Constituent	**Physical Meaning**	**Unit**
WFN	White Frequency Noise	Hz
WPN	White Phase Noise	Cycle
FFN	Flicker Frequency Noise	Hz/s
RWFN	Random Walk Frequency Noise	Hz/s
RRFN	Random Run Frequency Noise	Hz/s/s

The modeling of the FFN requires special attention and more discussion. Flicker Frequency Noise (FFN) is fundamentally a medium to long term instability that is

not physically understood, but consistently observed in clocks [30]. The FFN noise constituent is characterized by colored noise with transfer function $\frac{1}{\sqrt{s}}$ and power $\approx \frac{1}{|\omega|}$. This noise source is often called pink noise, or $\frac{1}{f}$ noise.The transfer function is infinite dimensional, irrational, and thus must be approximated in order to be simulated and incorporate in a state estimator, like a Kalman Filter [32]. Additionally, this transfer function has singularities at $\omega = 0$ (infared catastrophe) and $\omega = \infty$ (ultraviolet catastrophe). The work in [32] has described these theoretical as nothing more than "mental annoyances" - precision of instruments prevents the observation $\omega = 0$ and $\omega = \infty$. So the behavior at singularities only matter if you want a model accurate for infinite time. The most common state-space approximant is the Gauss-Markov stochastic differential equation [32, 20, 9]. These issues will be discussed further in Chapter 3. However, it's worth stating that colored noise states are typically not modeled in a Kalman Filter, and the consensus among academics is that including such states in a Kalman Filter model doesn't make a difference in timing/positioning accuracy [32, 20].

2.2 Allan Variance

The initial characterization of oscillator instability posed serious mathematical challenges as the flicker noise ($\frac{1}{f}$) constituent has an unbounded variance/standard deviation as a function of averaging time [2]. An atomic clock physicist named David Allan discovered this problem [2], and came up with a weaker definition of variation, that converges for noise processes with frequency spectrum slopes steeper than $\frac{1}{f}$. This formula is known as the Allan Variance and is a specific case of the M-sample variance, which is defined below 2.7 The M-sample Variance is defined as

$$\sigma_y^2(M, T_s, \tau) = \frac{1}{M-1}\left\{\sum_{k=0}^{M-1}\left(\frac{x(kT_s+\tau)-x(kT_s)}{\tau}\right)^2 - \frac{1}{M}\left(\sum_{k=0}^{M-1}\frac{x(kT_s+\tau)-x(kT_s)}{\tau}\right)^2\right\} \tag{2.7}$$

Table 2.3: M-Sample Variance Equations

Parameter	**Definition**	**Unit**
M	Number of Frequency Samples used	n/a
T_s	Sampling Time	(s)
τ	frequency estimate averaging time	(s)
x	phase angle	radians
$\mathbb{E}$	Expectation Operator	n/a

The parameters in equation 2.7 are defined in table 2.3 . The Allan Variance is the expected value of the M-sample variance in the case where $T_s = \tau$ and $M = 2$. As previously stated, this metric converges for processes steeper than $\frac{1}{f}$. This quantity can be seen in equation 2.8 and visualized in figure 2.2.

$$\begin{aligned}\sigma_y^2(\tau) &= \mathbb{E}\left[\sigma_y^2(2,\tau,\tau)\right] \\ &= \mathbb{E}\left[\frac{1}{2-1}\left\{\sum_{k=0}^{2-1}\left(\frac{x(k\tau+\tau)-x(k\tau)}{\tau}\right)^2 - \frac{1}{2}\left(\sum_{k=0}^{2-1}\frac{x(k\tau+\tau)-x(k\tau)}{\tau}\right)^2\right\}\right]\end{aligned} \tag{2.8}$$

Power Law Theory [31] presents a convenient characterization relating Allan Variance with Single Side Band Power Spectral Density of the phase fluctuation, which is shown in figure 2.3. The Power Law approach will not be used or implemented in this work. In practice, power spectral density plots do not possess slopes identical to power law theory, nor do the slopes necessarily occur in the proposed order shown in figure 2.3. The same can be said of the Allan Variance. The consequence of this is that the power law equations do not result in PSD/AV profiles that accurately approximate physically observed PSD and the AV profiles simultaneously [23]. The

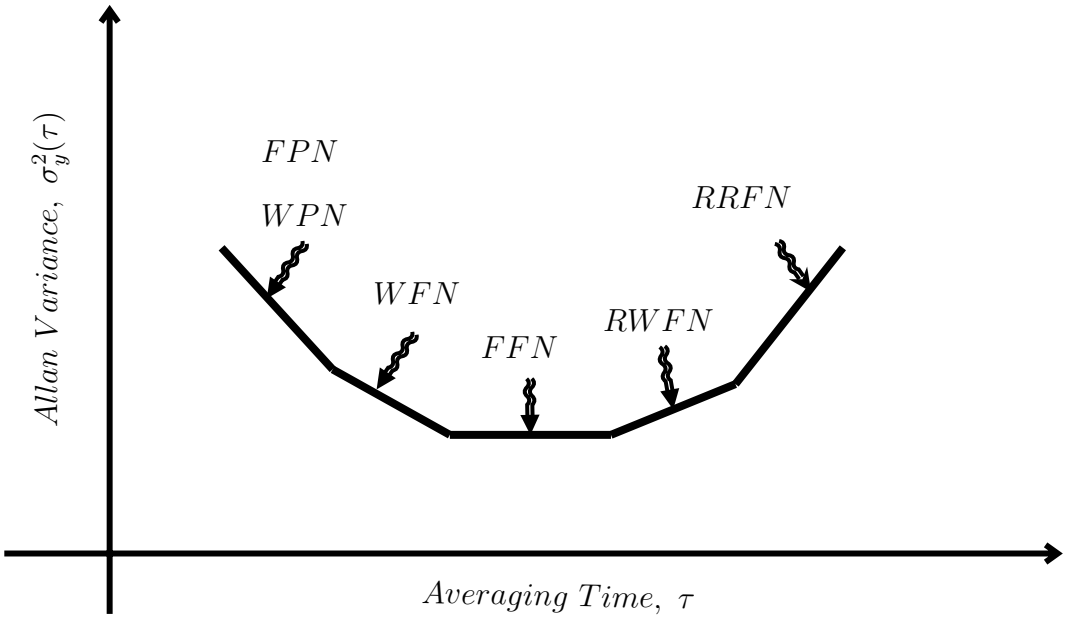

Figure 2.2: Allan Variance Noise Contributors - RWFN and RRFN dominate long term fluctuations, WPN/WFN dominate the short term, and FFN dominates "medium-to-long-term" fluctuations

work in [23] has proposed a new power law theory that attempts to address its shortcomings. As such, models and processes will be used that accurately reflect the Allan Variance profile of a clock. Moreover, the PSD measurement is typically made with sampling (measurement) frequencies consistent with short term stability characterization. The Allan Variance is a long-term metric. The Allan Variance is the standard for characterizing timing sources, and is now used in other applications that involve quantifying systems driven by integrated white noise, such as inertial measuring units [10].

2.2.1 Other Variance Estimators

The Overlapping Allan Variance averages blocks of size n before processing. This is the preferred Allan Variance by NIST, and as such will be the only Allan Variance metric used in this thesis. The Overlapping Allan Variance is superior to the standard Allan Variance because poses a significant improvement in resolution of frequency readings.

$$\sigma_y^2(n_o, N) = \frac{1}{2n^2\tau_o^2(N-2n)} \sum_{k=0}^{N-2n-1} (x(k\tau+2n) - 2x(k\tau+n) + x(k\tau)) \tag{2.9}$$

Another method for quantifying frequency stability is the Hadamard Variance [6]. The Hadamard Variance can be interpreted as a 3-sample variance.

$$\sigma_H^2(\tau) = \frac{1}{\tau^2(N-3)} \sum_{N-3}^{k=1} (x(k\tau+3) - 3x(k\tau+2) + 3x(k\tau+1) - x(k\tau)) \tag{2.10}$$

Last, there is the Modified Allan [22] and Hadamard Variance. Both of these methods are attractive because it allows for WPN and FPN to be distinguishable from one another.

2.3 Sigma Diffusion Parameter Identification

Due to the limitations of the previously discussed power-law theory, and also to avoid brute force identification of clocks, an optimization based method was developed to determine the sigma diffusion parameters. Hence, the following constrained normalized norm minimization problem is solved

$$\min_{\sigma_{RRFN,RWFN,FFN,WPN,WFN}>0} \left\{ \left\| \frac{\sigma^2_{y\ simulated}(\tau) - \sigma^2_{y\ target}(\tau)}{\sigma^2_{y\ target}(\tau)} \right\| \right\} \tag{2.11}$$

Where

- $\sigma^2_{RRFN,RWFN,FFN,WPN,WFN}$ are the sigma diffusion parameters/variances
- $\sigma^2_{y\ target}(\tau)$ is Allan Variance data provided by a clock vendor
- $\sigma^2_{y\ simulated}(\tau)$ is simulated Overlapping Allan Variance data
- $\sigma^2_{y\ simulated}(\tau)$ is obtained from simulated phase data using the three state clock model in equation 2.4.

Both process and measurement noise seeds must be fixed when propagating simulated noise within the optimizer, so that the stochastic differential equations, or, in optimization terms, the objective function evaluations, are identical. This allows for the simulation and filtering of realistic clock noise. The process of solving problem 2.11 can be observed in Algorithm 1. Note that this can be used not only for simulating clock noise, but for tuning Kalman Filters that seek to estimate phase fluctuations. Moreover, the result can be used to initialize non-linear and adaptive Kalman Filters. This procedure was carried out with Stratum 3E High Stability Stability Oven Stabilized Oscillator OH300 Series where $T_s = 10ms$ and $F_c = 40MHz$, $100s$ of data,2 norm, Sequential Quadratic Program, 4 logical cores, 4 physical on a intel i7 4790 using Matlabs fmincon. The results can be seen in figure 2.4. Several other approaches exist for determining sigma diffusion coefficients, and can be found in [8, 11, 37].

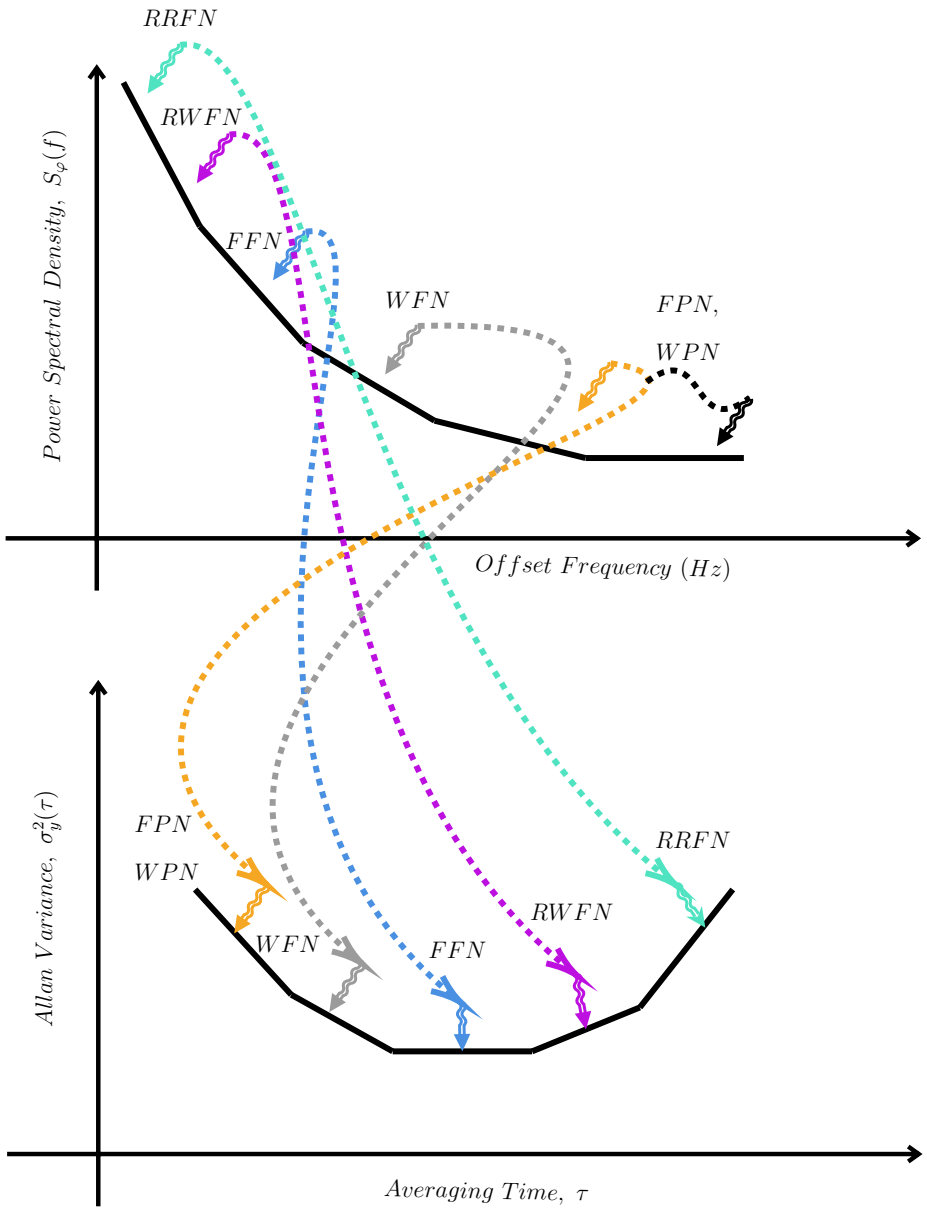

Figure 2.3: Relationship between Allan Variance and Power Spectral Density - Short averaging time contributors are equivalent to high frequency PSD, and large averaging time contributors are equivalent to low frequency PSD

Algorithm 1: Sigma Diffusion Parameter Identification Algorithm

Initialize $\sigma_{RRFN,RWFN,FFN,WPN,WFN}$ through iterative procedures;

Call Matlabs fmincon using the Sequential Quadratic Program Solver ;

Initial states to 0, or to initial phase and frequency offsets if known ;

for *For each function call* **do**

Propogate the clock states ;

for *For all time* **do**

$$\begin{bmatrix} x_{k+1}^1 \\ x_{k+1}^2 \\ x_{k+1}^3 \end{bmatrix} = \begin{bmatrix} 1 & T_s & \frac{T_s^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k^1 \\ x_k^2 \\ x_k^3 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{WFN} + u_{FFN} \\ u_{RWFN} \\ u_{RRFN} \end{bmatrix} \tag{2.12}$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{k+1}^1 \\ x_{k+1}^2 \\ x_{k+1}^3 \end{bmatrix} + u_{WPN} \tag{2.13}$$

Create the noise time series with the process noise covariance Q_k (sigma parameters selected by optimizer) ;

$$Q_k(T_s) = \begin{bmatrix} \sigma_{WFN}^2 T_s + \frac{1}{3}\sigma_{RWFN}^2 T_s^3 + \frac{1}{20}\sigma_{RRFN}^2 T_s^5 & \frac{1}{2}T_s^2\sigma_{RWFN}^2 + \frac{1}{8}\sigma_{RRFN}^2 T_s^4 & \frac{1}{6}\sigma_{RRFN}^2 T_s^3 \\ \frac{1}{2}T_s^2\sigma_{RWFN}^2 + \frac{1}{8}\sigma_{RRFN}^2 T_s^4 & \sigma_{RWFN}^2 T_s + \frac{1}{3}\sigma_{RRFN}^2 T_s^3 & \frac{1}{2}\sigma_{RRFN}^2 T_s^2 \\ \frac{1}{6}\sigma_{RRFN}^2 T_s^3 & \frac{1}{2}\sigma_{RRFN}^2 T_s^2 & \sigma_{RRFN}^2 T_s^2 \end{bmatrix} \tag{2.14}$$

And the measurement noise covariance $R = \sigma_{WPN}^2$;

end

Compute the Overlapping Allan Variance of x;

$$\sigma_y^2(n_o, N) = \frac{1}{2n^2\tau_o^2(N-2n)} \sum_{k=0}^{N-2n-1} (x_1(k\tau + 2n) - 2x_1(k\tau + n) + x_1(k\tau)) \tag{2.15}$$

Compute the objective function value $\frac{\left\|\sigma_{y\ simulated}^2(\tau) - \sigma_{y\ target}^2(\tau)\right\|}{\left\|\sigma_{y\ target}^2(\tau)\right\|}$

end

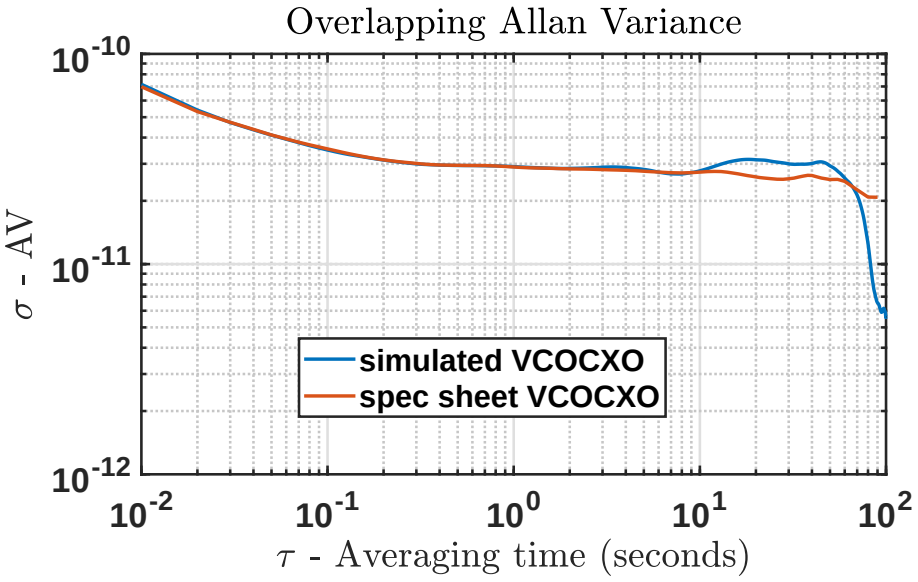

Figure 2.4: Vendor Provided Allan Profile and the Allan Profile obtained by optimization

Chapter 3

COLORED NOISE MODELING

It's standard to include the finite-dimensional states when modeling clock fluctuations, as shown in Chapter 2. However, Flicker Frequency Noise (FFN) is present in many oscillators. As previously mentioned, FFN has a transfer function of $\frac{1}{\sqrt{s}}$, which cannot be realized identically by a rational finite-dimensional model. Such a transfer function is infinite dimensional, unstable and irrational. To further complicate matters, the transfer function $\frac{1}{\sqrt{s}}|_{s=j\omega}$ has singularities at $\omega = 0$ and $\omega = \infty$, which *cannot* be approximated [30, 20, 9, 32]. This seemingly creates challenges as it's desirable to include such states in a state-estimator such as a Kalman Filter. However, it is possible to approximate $\frac{1}{\sqrt{s}}|_{s=j\omega}$ for ω in a compact sub-set of the frequency domain, $[\omega_b, \omega_h]$. The FFN constituent can be interpreted as white frequency acceleration entering a system described by $\frac{1}{\sqrt{s}}$. Figure 3.1 is an oscillator model that includes the FFN constituent explicitly. This chapter presents several methods for obtaining finite-dimensional state-space models which approximate the FFN constituent. A strong approximant should both mimic the behavior of $\frac{1}{\sqrt{s}}$ on compact sub-sets of the frequency domain, as well as generate a profile similar, if not identical to the Allan Variance profile of the oscillator of interest.

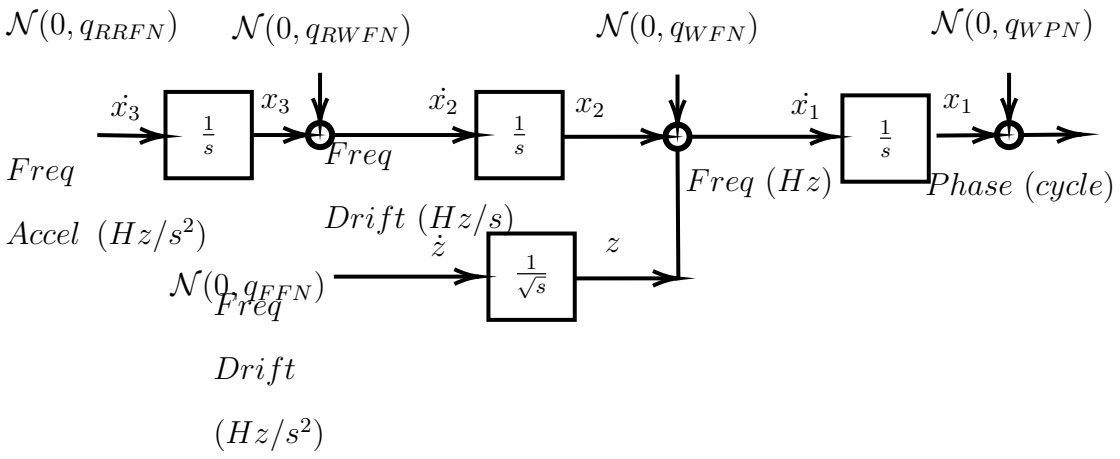

Figure 3.1: True Master Oscillator Model

One interpretation of Flicker Frequency Noise is that of Fractional Brownian motion. White noise can be interpreted as the derivative of Brownian motion [20]:

$$\frac{d}{dt}x(t) = w(t) \tag{3.1}$$

This expression can be extended to the fractional case:

$$\frac{d^\alpha}{dt^\alpha}x(t) = w(t), \alpha \in (0, 1) \tag{3.2}$$

Taking Laplace transforms of both sides of equation 3.2 gives the transfer function $\frac{1}{\sqrt{s}}$.

3.1 The Kasdin Approximation

The most computationally straight forward process for Flicker Frequency Noise is the Kasdin Approximation [20]. Consider the following discrete time fractional integrator in the z-domain:

$$H(z) = \frac{1}{(1-z^{-1})^{\frac{\alpha}{2}}}, z > 1 \tag{3.3}$$

Note that the singularities associated with Laplace domain equivalent transfer function are removed by the condition that $z > 1$ and the Nyquist frequency of the discretization. A power series expansion can be used to approximate 3.3:

$$H(z) = 1 + \frac{\alpha}{2}z^{-1} + \frac{\frac{\alpha}{2}(\frac{\alpha}{2}+1)}{2!}z^{-2} + \dots \tag{3.4}$$

Note that this expression corresponds to a sum of pulses. The pulse coefficients, for an order k approximant, can be computed as

$$h_o = 1 \tag{3.5}$$

$$h_k = (\frac{\alpha}{2} + k - 1)\frac{h_{k-1}}{k} \tag{3.6}$$

This is fundamentally a FIR filter (MA), that can be easily realized with a batch inverse FFT. Similarly, an IIR (AR) realization is made by possible by the following equations

$$H(z) = \frac{1}{1 - \frac{\alpha}{2}z^{-1} - \frac{\frac{\alpha}{2}(1-\frac{\alpha}{2})}{2!}z^{-2} + \dots} \tag{3.7}$$

with coefficients a_k,

$$a_o = 1 \tag{3.8}$$

$$a_k = (k - 1 - \frac{\alpha}{2})\frac{a_{k-1}}{k} \tag{3.9}$$

The benefit of the MA approach is that it consumes less memory, can be written in state-space and doesn't require the simulated sequence to be a strict power of 2.

3.2 Gauss Markov Approach

The most common approach is that of a stochastic differential equation described by a first-order Gauss-Markov approach [32, 9]. One can arrive at the same stochastic differential equation, by deterministically approximating $\frac{1}{\sqrt{s}}$ by using a continued fraction expansion in conjunction with the Pade table [32]. From a systems perspective, this approximation scheme can be interpreted as a family of low-pass filters in a parallel architecture. This topology can be observed in figure 3.2. Similarly, this system is realized in equation form

$$\begin{bmatrix} x_{k+1}^1 \\ x_{k+1}^2 \\ x_{k+1}^3 \\ x_{k+1}^4 \\ \dots \\ x_{k+1}^n \end{bmatrix} = \begin{bmatrix} 1 & T_s & \frac{T_s^2}{2} & \frac{1-e^{-p_1 T_s}}{p_1} & \dots & \frac{1-e^{-a_n T_s}}{p_n} \\ 0 & 1 & T_s & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & e^{-p_1 T_s} & \ddots & \ddots \\ \vdots & \vdots & \vdots & 0 & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & e^{-p_n T_s} \end{bmatrix} \begin{bmatrix} x_k^1 \\ x_k^2 \\ x_k^3 \\ x_k^4 \\ \dots \\ x_k^n \end{bmatrix} + \begin{bmatrix} u_{WFN} \\ u_{RWFN} \\ u_{RRFN} \\ u_{FFN}^1 \\ \vdots \\ u_{FFN}^n \end{bmatrix} \tag{3.10}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_{k+1}^1 \\ x_{k+1}^2 \\ x_{k+1}^3 \\ x_{k+1}^4 \\ \vdots \\ x_{k+1}^n \end{bmatrix} + u_{WPN} \tag{3.11}$$

Where the u's represent time series datum for the different noise constituents. The process noise covariance matrix is populated by the following equations

$$Q_A = \begin{bmatrix} \sigma^2_{WFN}T_s + \frac{1}{3}\sigma^2_{RWFN}T_s^3 + \frac{1}{20}\sigma^2_{RRFN}T_s^5 + q_{11} & \frac{1}{2}T_s^2\sigma^2_{RWFN} + \frac{1}{8}\sigma^2_{RRFN}T_s^4 & \frac{1}{6}\sigma^2_{RRFN}T_s^3 \\ \frac{1}{2}T_s^2\sigma^2_{RWFN} + \frac{1}{8}\sigma^2_{RRFN}T_s^4 & \sigma^2_{RWFN}T_s + \frac{1}{3}\sigma^2_{RRFN}T_s^3 & \frac{1}{2}\sigma^2_{RRFN}T_s^2 \\ \frac{1}{6}\sigma^2_{RRFN}T_s^3 & \frac{1}{2}\sigma^2_{RRFN}T_s^2 & \sigma^2_{RRFN}T_s^2 \end{bmatrix} \tag{3.12}$$

$$q_{11} = \sigma^2_{FFN}\sum_{j=1}^{n}\sum_{m=1}^{n}\left(\frac{1}{p_m p_j}\right)\left(Ts - \frac{(1-e^{-p_j T_s})}{p_j} - \frac{(1-e^{-p_m T_s})}{p_m} + \frac{1-e^{-(p_m+p_j)T_s}}{(p_m+p_j)}\right) \tag{3.13}$$

$$Q_B^m = \sigma^2_{FFN}\sum_{j=1}^{n}\left(\frac{1-e^{-p_m T_s}}{p_m} - \frac{1-e^{-(p_m+p_j)T_s}}{(p_m+p_j)}\right) \tag{3.14}$$

$$Q_D^{m,j} = \sigma^2_{FFN}\frac{1-e^{-(p_j+p_m)T_s}}{p_j+p_m} \tag{3.15}$$

Thus, the process noise covariance matrix is

$$Q(T_s) \quad = \begin{bmatrix} Q_A & Q_B \\ Q_B^T & Q_D \end{bmatrix} \tag{3.16}$$

This approach is attractive for many reasons: explicit state-transition matrix, measurement matrix, and process noise covariance, which can conveniently be placed in a Kalman Filter model. However, it has many parameters: the 5 noise constituents, pole locations in the low-pass filters, and order of approximation. Moreover, it does not contain an integrator, which limits the fidelity of the approximation in the frequency domain. Moreover, the driving noise source must be interpreted as a white frequency drift, as opposed to a frequency acceleration. The designer or simulator must juggle $5 + N$ parameters, where N is the order of the Gauss-Markov approximant. While not explored in this thesis, higher order Gauss-Markov schemes can be used.

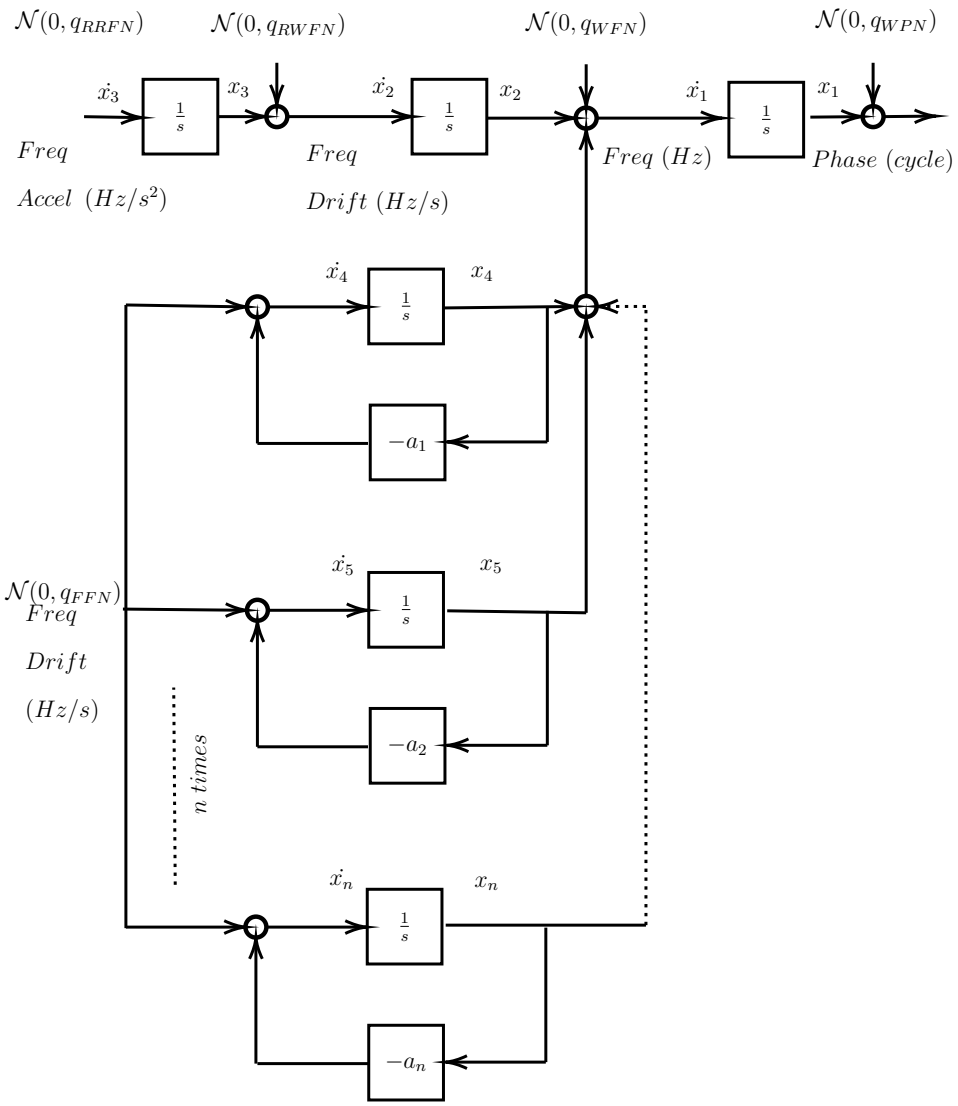

Figure 3.2: Master Oscillator with GM FFN Approximant

3.3 Oustaloup Approximation

An approximant that is a more explicit representation of $\frac{1}{\sqrt{s}}$ is the Oustaloup fractional order differentiation approach [24]. This presents advantages over the traditional Gauss-Markov framework. The Oustaloup approximation is described by the following equations

$$s^{\alpha} \approx K \prod_{k=-N}^{k=N} \frac{s+\omega_k^{'}}{s+\omega_k} = K \prod_{k=-N}^{k=N} \left(1 + \frac{\omega_k - \omega_k^{'}}{s+\omega_k}\right) \tag{3.17}$$

Where

- $w_k^{'} = \omega_b(\frac{\omega_h}{\omega_b})^{\frac{k+N+0.5(1-\alpha)}{2N+1}}$
- $w_k = \omega_b(\frac{\omega_h}{\omega_b})^{\frac{k+N+0.5(1+\alpha)}{2N+1}}$
- $K = \omega_h^{\alpha}$

This infinite product approximates s^{α} for $\omega \in (\omega_b, \omega_h)$. Thus, by realizing that $\frac{1}{\sqrt{s}} = \frac{1}{s}s^{\frac{1}{2}}$, an approximation for $\frac{1}{\sqrt{s}}$ can be constructed as

$$\frac{1}{\sqrt{s}} = \frac{1}{s}s^{\frac{1}{2}} = \frac{1}{s}\left(K \prod_{k=-N}^{k=N} \left(1 + \frac{\omega_k - \omega_k^{'}}{s+\omega_k}\right)\right) \tag{3.18}$$

As demonstrated by equation 3.18, this approximation scheme is a series configuration of low-pass filters, which can be seen in figure 3.3. The Oustaloup Approximation method can combined with the standard finite dimensional clock states, and turned into a stochastic differential equation, which can be expressed in discrete time as seen in equation 3.19

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dots \\ \dot{x}_{n+4} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & K & K & \dots & \dots & K \\ 0 & 0 & 1 & 0 & 0 & & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 \\ 0 & 0 & 0 & -\omega_n & \omega_n^{'} - \omega_n & \dots & \dots & \omega_n^{'} - \omega_n \\ 0 & 0 & 0 & 0 & -\omega_{n-1} & \omega_{n-1}^{'} - \omega_{n-1} & \dots & \omega_{n-1}^{'} - \omega_{n-1} \\ \dots & \dots & \dots & \dots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & -\omega_1 & \omega_1^{'} - \omega_1 \\ 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ \dots \\ x^{n+4} \end{bmatrix} \tag{3.19}$$

$$+ \begin{bmatrix} \sigma_{WFN} & 0 & 0 & 0 \\ 0 & \sigma_{RWFN} & 0 & 0 \\ 0 & 0 & \sigma_{RRFN} & 0 \\ \vdots & \vdots & 0 & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \sigma_{FFN} \end{bmatrix} \begin{bmatrix} u_{WFN} \\ u_{RWFN} \\ u_{RRFN} \\ u_{FFN} \end{bmatrix} \tag{3.20}$$

The standard for discretizing higher order stochastic systems is to create a block matrix populated by the F and G state matrices

$$M = \begin{bmatrix} -F & GG^T \\ 0 & F^T \end{bmatrix} T_s \tag{3.21}$$

The matrix exponential of A is the following

$$B = expm(M) = \begin{bmatrix} \dots & \phi_k^{-1} Q_k \\ 0 & \phi_k^T \end{bmatrix} \tag{3.22}$$

Thus, Q_k and ϕ_k can be computed for simulation and estimation purposes

$$x_{k+1} = \phi_k x_k + w_k \tag{3.23}$$

$$y_k = x_k^1 \tag{3.24}$$

where $w_k \sim \mathcal{N}(0, Q_k)$. The advantages of this approach is that pole locations are given and it contains an integrator inherent to the process.

Additionally, it is explicitly related to equation 3.2 and hence is better equipped to be an approximate of Fractional Brownian motion. Moreover, the designer only needs to decide the order of approximation and the compact sub-set of the frequency domain where the expression is valid. The selection of ω_b and ω_l can be based on the precision of the measurement instrument. A disadvantage of the approach is that the order of approximation grows as $2N + 1 \times 2N + 1$. The 5 noise constituents still must be determined. The parameters used in this thesis are defined in table 3.1. Figures 3.4 and 3.4 demonstrate accurate approximation at crossover, and that the estimation error Bode plot, as expected, converges to a bowl shape, because of the singularities at $\omega = 0$ and $\omega = \infty$.

Table 3.1: Oustaloup Approximation Parameters

Parameter	**Value**
ω_b	1e-7
ω_h	100
b	10
d	9
α	$\frac{1}{2}$
N	10

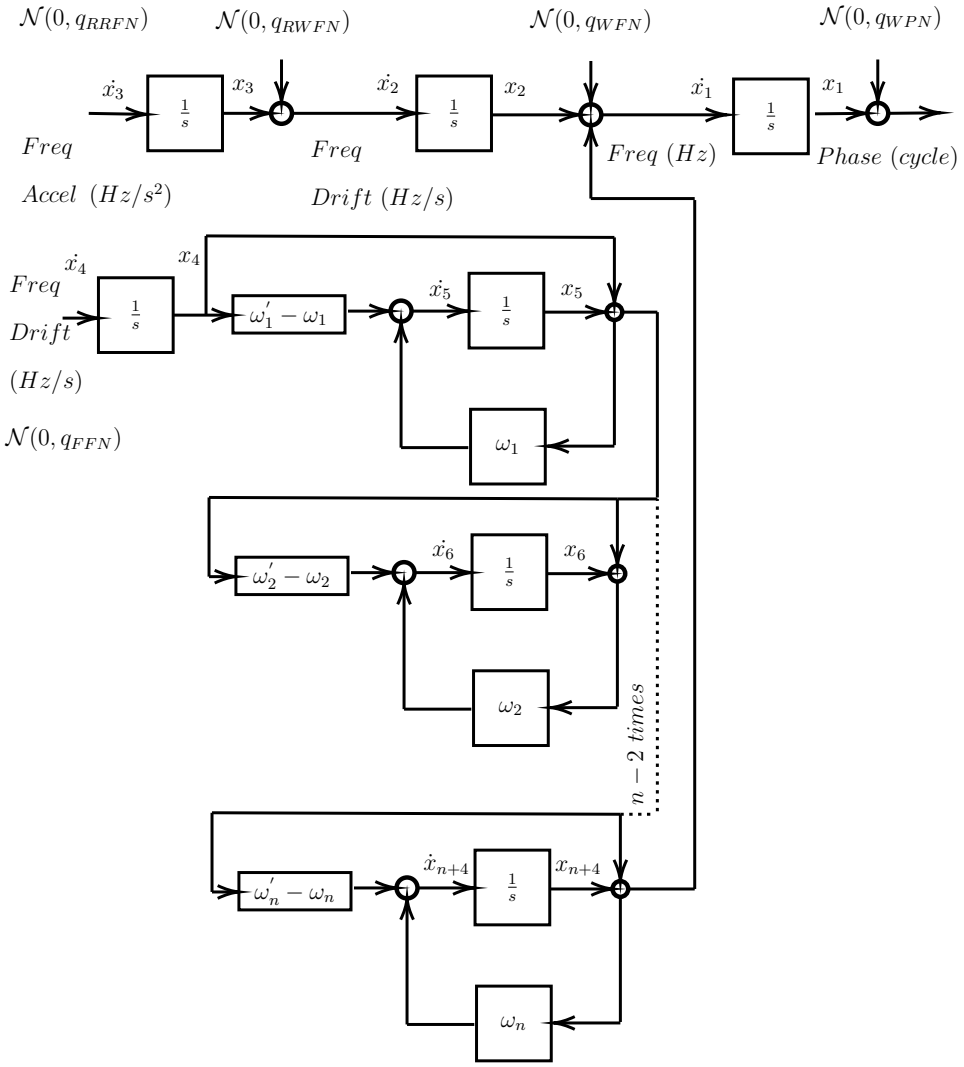

Figure 3.3: Master Oscillator with Oustaloup Aproximants

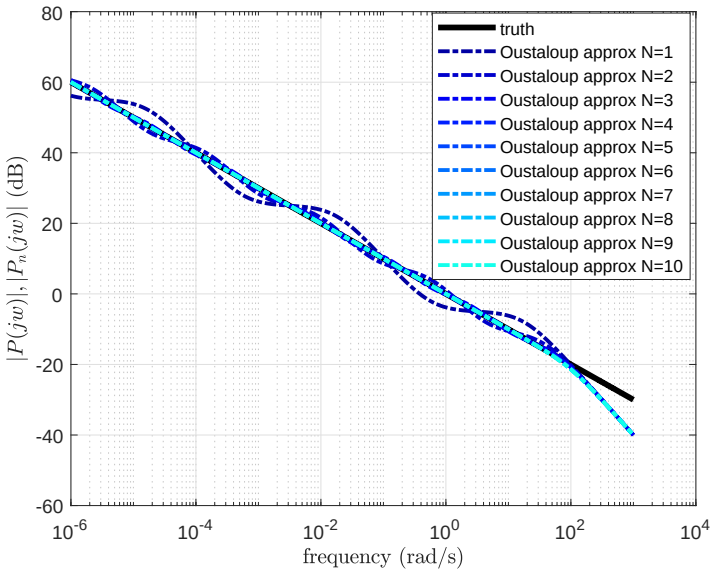

Figure 3.4: Bode Magnitude Plot for $\frac{1}{\sqrt{s}}$ and Oustaloup Approximants of increasing order. It appears that all approximants for $N > 1$ are accurate at crossover. Note that $\frac{1}{\sqrt{s}}$ crosses over at -10dB per decade.

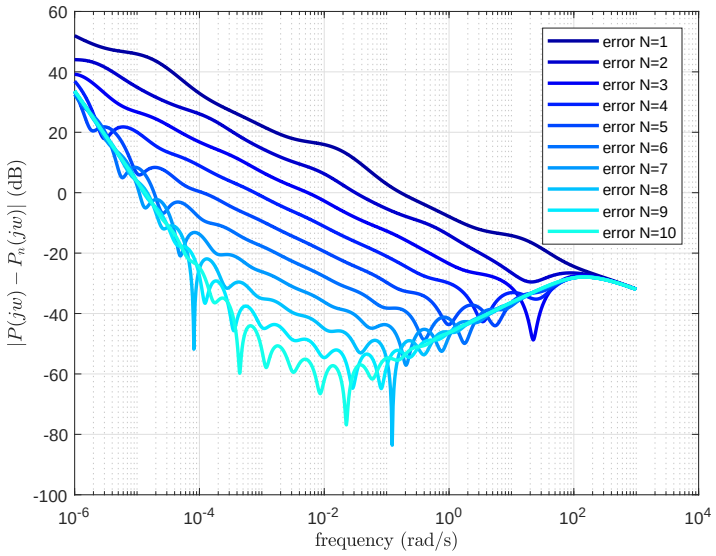

Figure 3.5: Bode Magnitude Plot of the error between $\frac{1}{\sqrt{s}}$ and Oustaloup Approximants of increasing order. Note that the approximation error transfer function converges to a bowl shape because of the singularities at $\omega = 0$ and $\omega = \infty$.

3.4 Colored Noise Linear Filtering For Time Keeping

It has been suggested in the literature that FFN states do not lead to a notable increase in timing accuracy. In order to gain insight into whether or not FFN states should be included in a Kalman Filter model, a numerical experiment was conducted. The experiment assumes that the phase of a Master Oscillator can be measured directly, which is not realistic, but is an academic exercise that can serve as a crude lower bound in timing accuracy. The phase measurement is fed into a Kalman Filter, and the noise constituents are estimated, and issued as corrections to the Master Oscillator. This is shown in figure 3.6. The standard Iterative Kalman Filter [18] was used, as can be seen in algorithm 2. Since the state estimates $\hat{x}_k$ are T_s seconds ahead of the measurement, they are issued as corrections to the Master Oscillator.

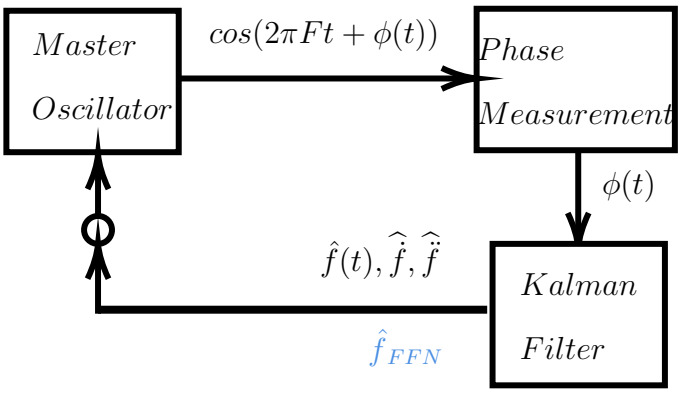

Figure 3.6: Block Diagram Representation of a Kalman Filter issuing corrections to a Master Oscillator. This is an ideal situation for illustrative purposes. The phase measurement allows for the Kalman Filter to measure the phase of the Master Oscillator directly. The Kalman Filter estimates the noise constituents, and issues corrections.

Algorithm 2: Iterative Kalman Filter Update

Initialize $P_0 = A\epsilon IA^T + Q_o$ and $x_o = \overrightarrow{0}$;

for *all k (time)* **do**

$\hat{x}_k^- = A\hat{x}_{k-1}^+$;

$P_k^- = AP_{k-1}^+A^T + Q_k$;

$\tilde{y}_k = z_k - h(x_k^-)$;

$S_k = H_k P_k^- H_k^T + R_k$;

$K_k = P_k H_k^T S_k^{-1}$;

$\hat{x}_k^+ = \hat{x}_k^- + K_k\tilde{y}$;

$P_k^+ = (I - K_k H_k)P_k^-$;

end

It was found that there exists scenarios where including FFN states in a Kalman Filter leads to an order of magnitude improvement in timing accuracy. The parameters that dictate whether or not such an improvement will occur are the sampling time, and the ratio between σ_{FFN} and σ_{RWFN}, or the ratio between σ_{FFN} and σ_{RWFN}. The three scenarios are the following

- (I) There is so much flicker noise relative to random walk and random run that including colored noise states in a Kalman Filter will lead to an order of magnitude increase in timing, and hence ranging accuracy, which can be seen in figures 3.7 and 3.8

- (II) There is a decent amount of flicker noise, but also a decent amount of random walk/random run - the consequence of this is that including colored noise states in the Kalman Filter lead to an order of magnitude increase in timing accuracy over a period of hours, but don't make a difference in a long term average sense. May be useful when you need high accuracy immediately, which can be seen in figures 3.9 and 3.10

- (III) The RRFN and RWFN dominate such that modeling FFN states in the Kalman Filter leads to no improvement, which can be seen in figures 3.11 and 3.12

The σ diffusion coefficients used in these numerical experiments can be found in table 3.2

Table 3.2: 3+2N+2 State Oscillator Model Noise Constituents.

Constituent	**Case I**	**Case II**	**Case III**
WFN	2.6e-4 Hz	2.6e-4 Hz	2.6e-4 Hz
WPN	0.02 Cycle	0.02 Cycle	0.02 Cycle
FFN	9e-4 Hz/s	9e-4 Hz/s	9e-4 Hz/s
RWFN	1e-14 Hz/s	1e-13 Hz/s	1e-11 Hz/s
RRFN	1e-15 Hz/s/s	1e-14 Hz/s/s	1e-12 Hz/s/s

These σ diffusion coefficients were nominally found using algorithm 1, and the Allan Variance data comes from the Stratum 3E High Stability Stability Oven Stablized Oscillator OH300 Series. The sampling time is $T_s = 10$ ms and the carrier frequency, F_c is 40 MHz.

3.4.1 Determining Order of Approximation

A critical question when dealing with infinite-dimensional systems, such as the Flicker Frequency noise constituent, is what the order of the approximation scheme needs to be. Figures 3.13, 3.14, 3.15 and 3.16 demonstrate that at most $N = 2$ is necessary to get roughly an order of magnitude increase in timing accuracy in a Kalman Filter. This corresponds to a $3 + 2(1) + 2 = 7$ order Kalman Filter model. However, it's possible that $N = 1$ is enough, which corresponds to a $3 + 2(2) + 2 = 9$

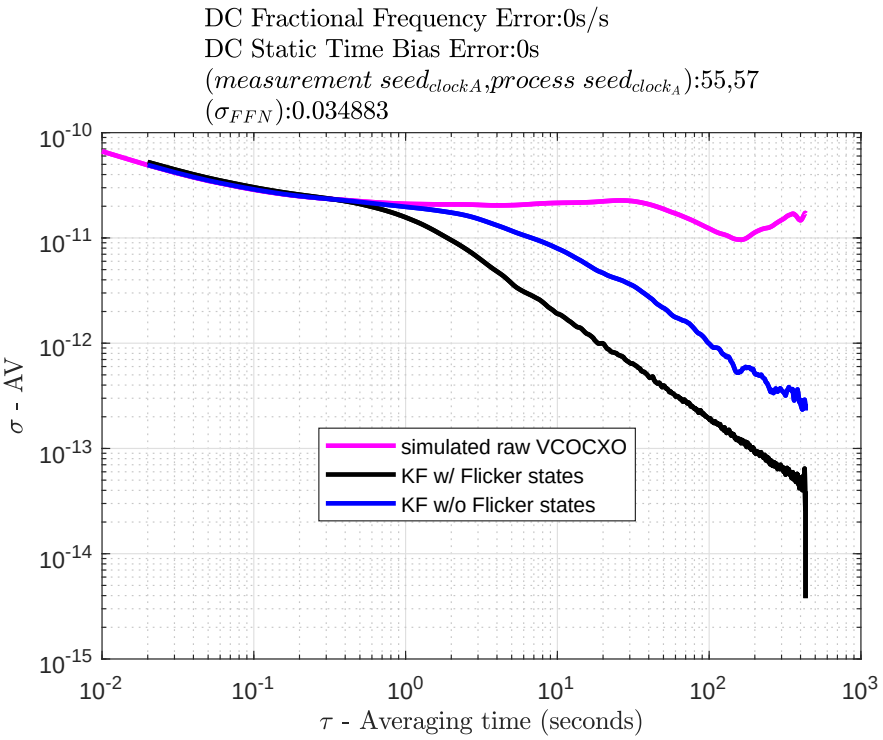

Figure 3.7: Allan Variance Plot of the Nominal Master Clock, the Master Clock with corrections from the finite-dimensional clock model in the Kalman Filter, and the master Clock with corrections from the Kalman Filter including FFN states. Here the order of the Kalman Filter is the same as that of the generating Master Clock model, $N = 10$. It's seen that the corrected clocks have significantly better frequency stability than the nominal clock over a day.

order Kalman Filter model. It's worth noting that an exceptional amount of Flicker Frequency noise is present in the examples shown in figures 3.15 and 3.16. This example, while perhaps not realistic, was selected as it serves as a worst scenario. If more Flicker Frequency noise is present, it stands to reason that a higher order model may be necessary, as a wider array of averaging times are effected.

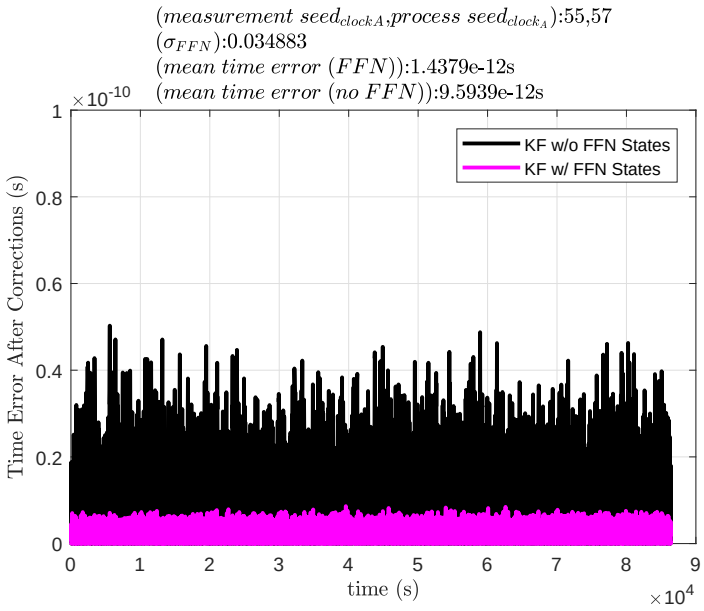

Figure 3.8: Accumulated Timing Error of the the Master Clock with corrections from the finite-dimensional clock model in the Kalman Filter, and the master Clock with corrections from the Kalman Filter including FFN states. Here the order of the Kalman Filter is the same as that of the generating Master Clock model, $N = 10$. It's seen that the Kalman Filter that includes colored noise states has an order of magnitude improvement in mean accumulated timing error over a day.

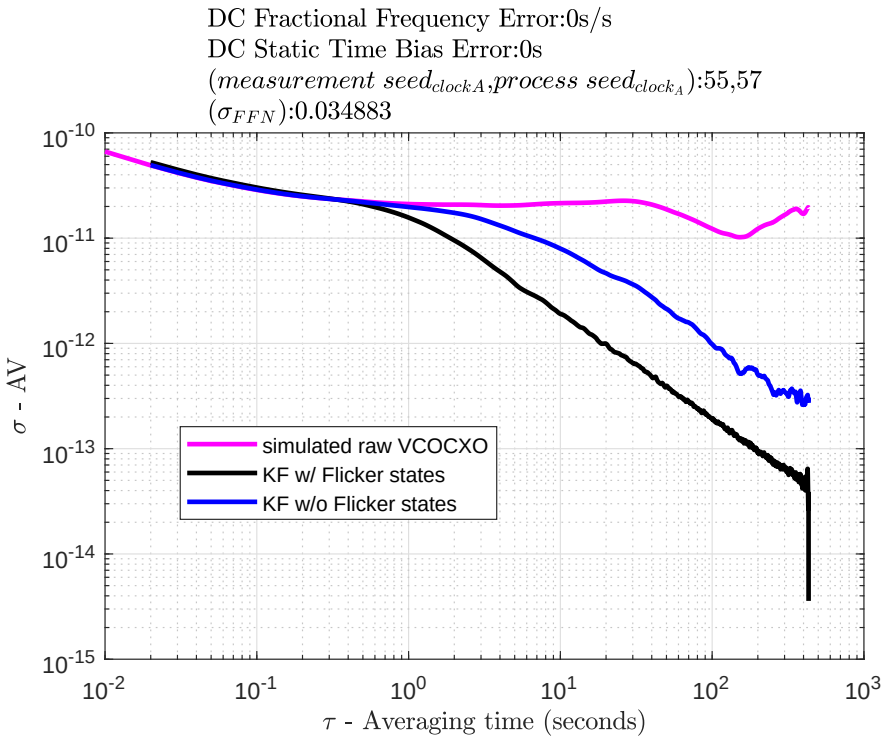

Figure 3.9: Allan Variance Plot of the Nominal Master Clock, the Master Clock with corrections from the finite-dimensional clock model in the Kalman Filter, and the master Clock with corrections from the Kalman Filter including FFN states. Here the order of the Kalman Filter is the same as that of the generating Master Clock model, $N = 10$. It's seen that the clock containing colored noise corrections has significantly better frequency stability than the clock without for roughly 2.5 hours, but over a day the average is roughly the same.

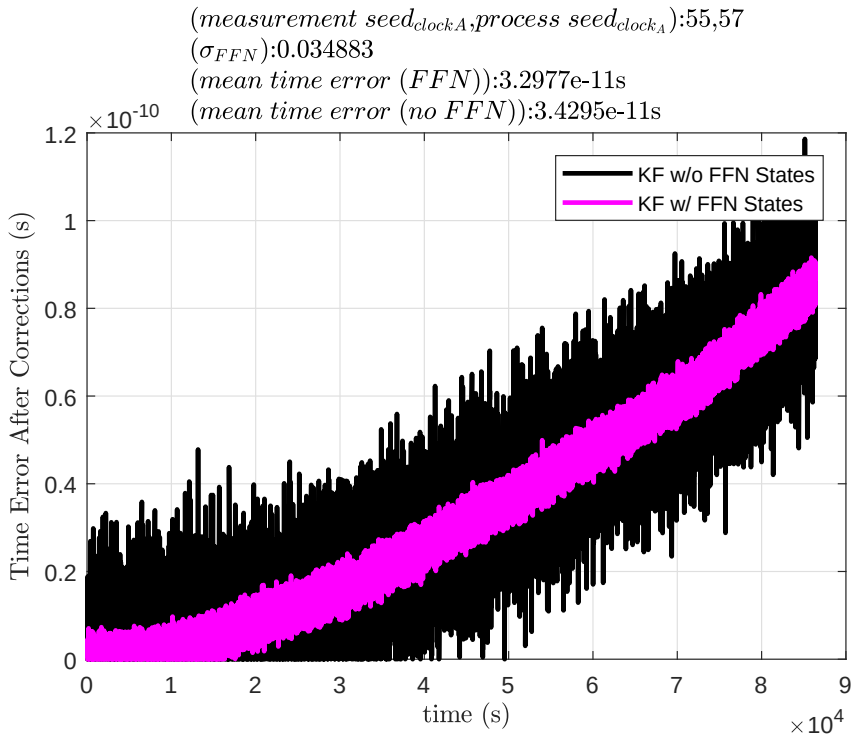

Figure 3.10: Accumulated Timing Error of the the Master Clock with corrections from the finite-dimensional clock model in the Kalman Filter, and the master Clock with corrections from the Kalman Filter including FFN states. Here the order of the Kalman Filter is the same as that of the generating Master Clock model, $N = 10$. It's seen that the clock containing colored noise corrections has significantly better average accumulated timing error over the clock without for roughly 2.5 hours, but over a day the average is roughly the same.

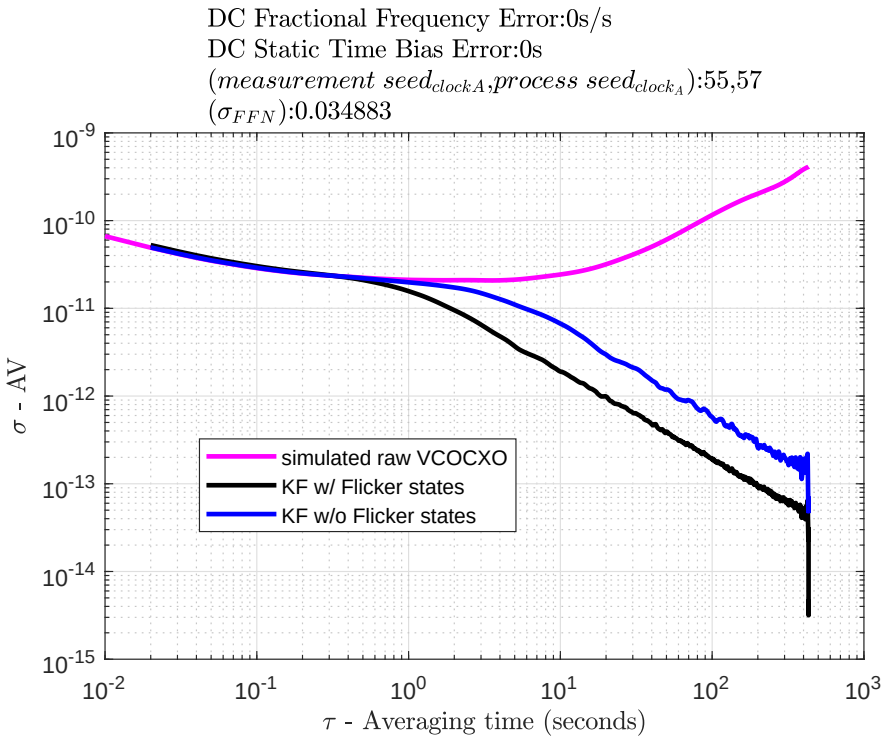

Figure 3.11: Allan Variance Plot of the Nominal Master Clock, the Master Clock with corrections from the finite-dimensional clock model in the Kalman Filter, and the master Clock with corrections from the Kalman Filter including FFN states. Here the order of the Kalman Filter is the same as that of the generating Master Clock model, $N = 10$. It's seen that including colored noise states in the Kalman Filter makers no difference in frequency stability. The reason is because the long term stability terms, RRFN and RWFN dominate early on.

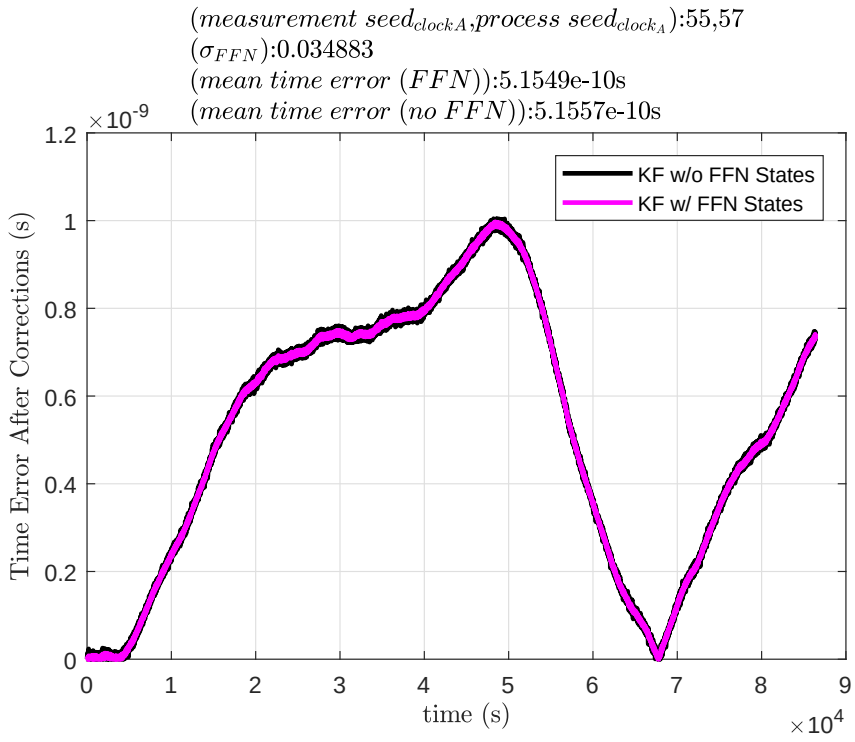

Figure 3.12: Accumulated Timing Error of the the Master Clock with corrections from the finite-dimensional clock model in the Kalman Filter, and the master Clock with corrections from the Kalman Filter including FFN states. Here the order of the Kalman Filter is the same as that of the generating Master Clock model, $N = 10$. It's seen that including colored noise states in the Kalman Filter makers no difference in average accumulated timing error. The reason is because the long term stability terms, RRFN and RWFN dominate early on.

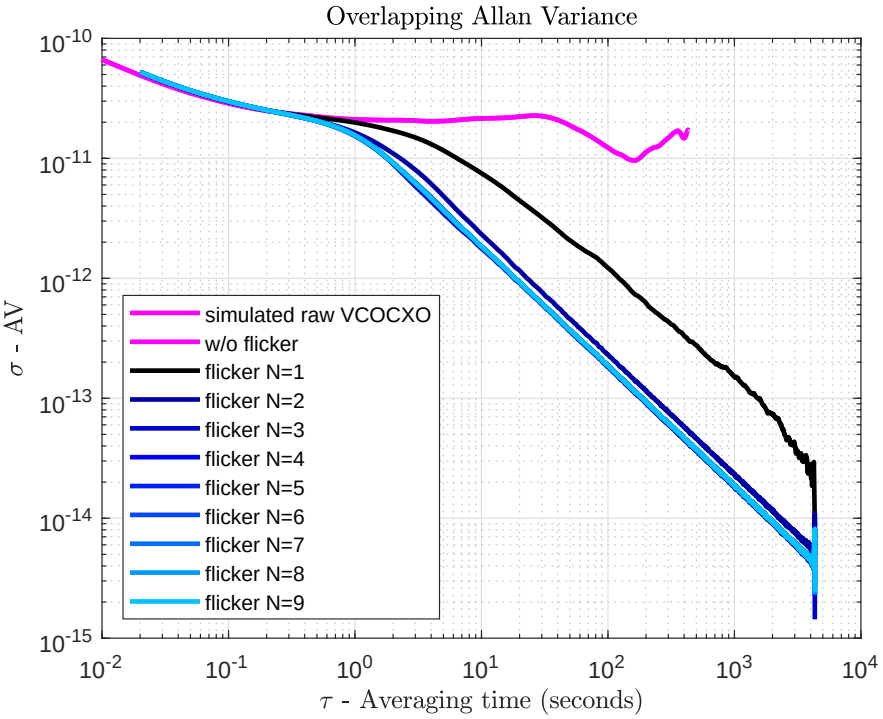

Figure 3.13: Order of Approximation Experiment. From a computational perspective, it's beneficial to have as low-order a model of colored noise states as possible, while still maintaining accuracy. It's seen that that for N>1 there are diminshing returns in frequency stability improvement, so $N = 2$ allows for the realization of the desired tradeoff.

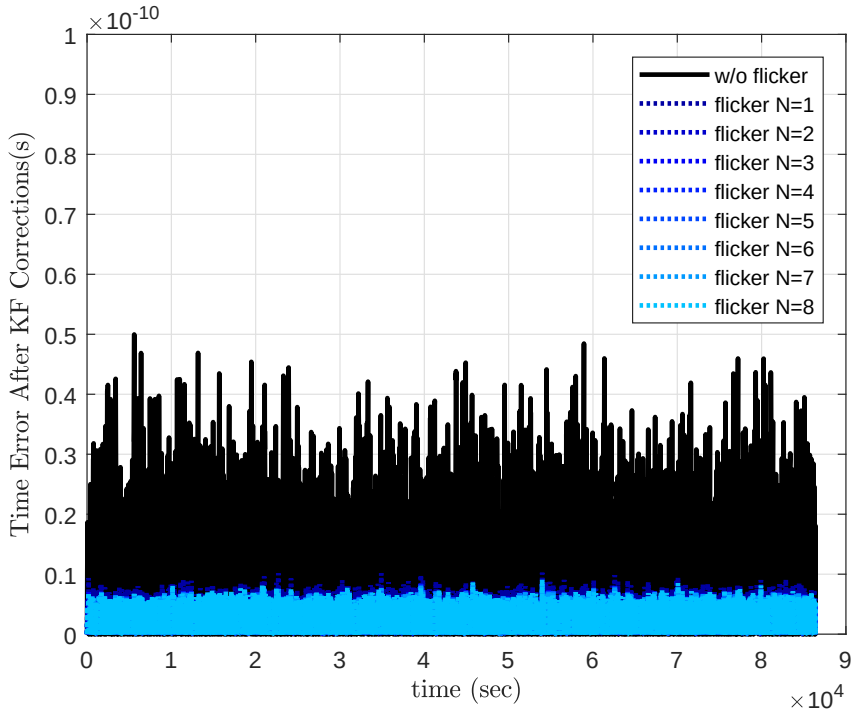

Figure 3.14: Average Accumulated Timing Error of Corrected Clocks. Order of Approximation Experiment. From a computational perspective, it's beneficial to have as low-order a model of colored noise states as possible, while still maintaining accuracy. It's seen that that for N>2 there are diminshing returns in average accumulated timing error, so $N = 2$ allows for the realization of the desired tradeoff.

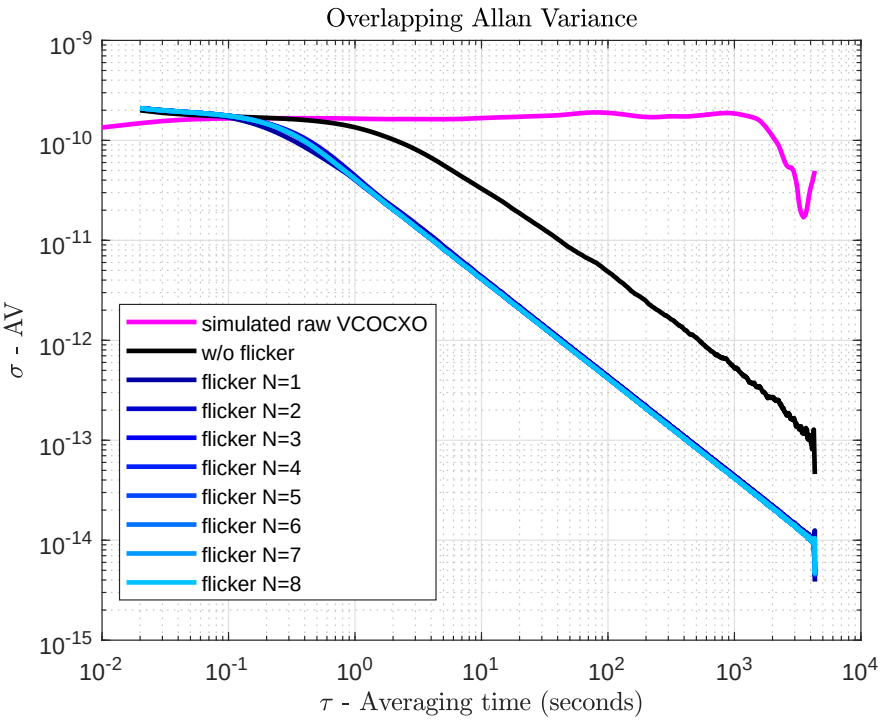

Figure 3.15: Order of Approximation Experiment. From a computational perspective, it's beneficial to have as low-order a model of colored noise states as possible, while still maintaining accuracy. It's seen that that for $N \geq 1$ there are diminshing returns in frequency stability improvement, so $N = 1$ allows for the realization of the desired tradeoff.

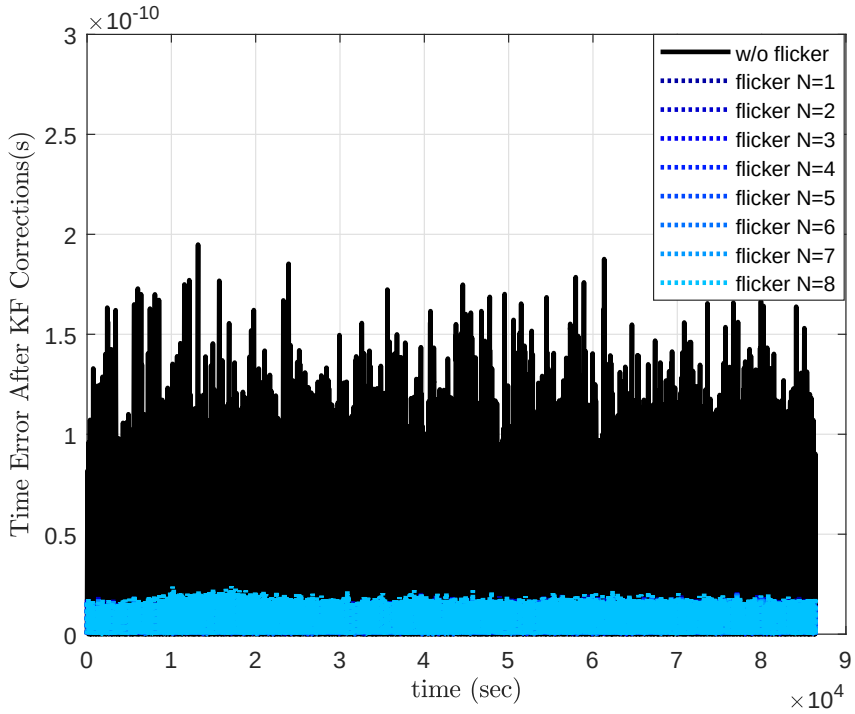

Figure 3.16: Average Accumulated Timing Error of Corrected Clocks. Order of Approximation Experiment. From a computational perspective, it's beneficial to have as low-order a model of colored noise states as possible, while still maintaining accuracy. It's seen that that for N>1 there are diminshing returns in average accumulated timing error, so $N = 1$ allows for the realization of the desired tradeoff.

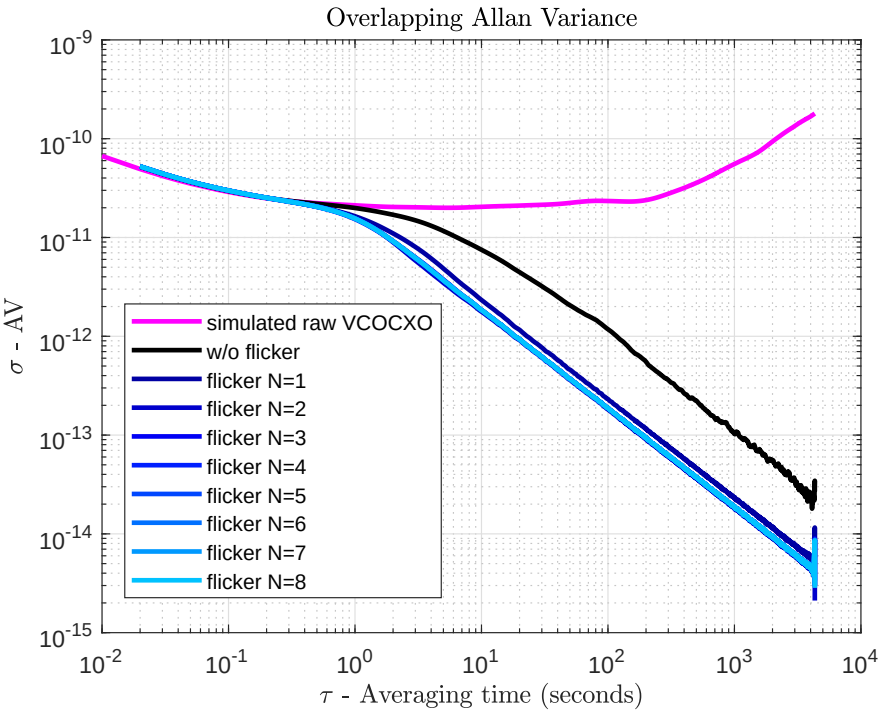

Figure 3.17: Order of Approximation Experiment. From a computational perspective, it's beneficial to have as low-order a model of colored noise states as possible, while still maintaining accuracy. It's seen that that for $N \geq 1$ there are diminshing returns in frequency stability improvement, so $N = 2$ allows for the realization of the desired tradeoff.

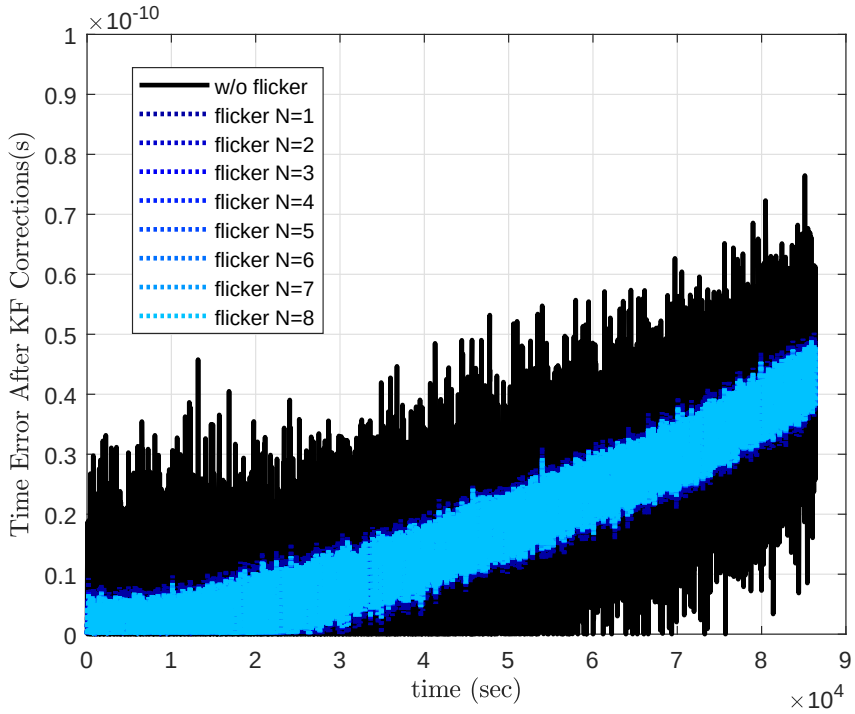

Figure 3.18: Average Accumulated Timing Error of Corrected Clocks. Order of Approximation Experiment. From a computational perspective, it's beneficial to have as low-order a model of colored noise states as possible, while still maintaining accuracy. It's seen that that for N>1 there are diminishing returns in average accumulated timing error, so $N = 2$ allows for the realization of the desired tradeoff.

Chapter 4

KALMAN FILTER ARCHITECTURES FOR TIMING ERROR COMPENSATION

To estimate the navigation relevant quantities - effectively range and ranging error constituents, several Kalman Filter approaches are proposed and compared [17, 18]. It is well known that the Kalman Filter is the optimal state estimator with respect to the expectation of the estimation error in the presence of white Gaussian noise [18]. The Extended Kalman (EKF) expands on the Kalman Filter and extends the ideas to non-linear systems using Taylor Series arguments. While Extended Kalman Filters are widely used in many navigation applications, it is not without its technical pitfalls - dependence on the state trajectory being close to the point of linearization, and the impact of non-linearities on noise characteristics. The Unscented Kalman Filter (UKF) softens the assumptions of the EKF by way of the Unscented Transform. The Unscented Transform allows for the underlying Gaussian distribution of the measurement to be preserved when passing through a non-linear function [17].
All Kalman Filters have something in common - dependence upon how well the model for the process dynamics and measurement model as well as their noise covariance matrices (Q, R) effectively capture physical reality. Thus, Adaptive Kalman Filtering has been an active field of study in the control and estimation community for over 40 years. The two prevailing schools of thought in adaptive estimation are Multiple Model Adaptive Estimation (MMAE) [5] and Innovations Based Adaptive Estimation (IAE) [25, 26]. In the MMAE framework, a topology composed of a bank of Kalman Filters is used - each with their own (Q, R, P) - process, measurement and estimation error covariance matrices, respectively. Knowledge of the underlying probability

distribution of the measurement and state is used to determine which filter produces the "best" estimate. The Innovations Based Adaptive Filtering framework uses the Kalman Filter residuals to compute the pair (Q, R) on the fly.

In this section, these approaches are compared and systematic approaches are proposed to decrease the number of iterations in the filter design process. Moreover, a novel algorithm is presented which marries the MMAE and IAE approach, as well as modifications to the measurement noise covariance matrix to ensure a more stable estimate. This algorithm allows for a more robust adaptive law as it addresses the weaknesses of both methods. The IAE approach relies on an initial triplet (Q, R, P), and if the initial guess is too far off, estimates will not be accurate, and filter divergence is possible. Combining the two methods allows for the MMAE framework to find which Innovations based Adaptive Extended Kalman Filter produces the "best" estimate. In this section, the following nomenclature will be used

- (Q_{act}, R_{act}) - the actual process and measurement noise covariance
- $(\tilde{Q}, \tilde{R})$ - the adaptively estimated process and measurement noise covariance matrix
- (Q_0, R_0) - initial guess for process and measurement noise covariance matrix

4.1 Innovations Based Adaptive Extended Kalman Filter

The IAE approach was first proposed in [25, 26] and naturally extends to the EKF [27]. The utility of this approach is that it changes the filter design problem from selecting an appropriate pair (Q, R) for all time and conditions, to selecting an appropriate initial (Q_0, R_0) such that over time the estimated covariance matrix $(\tilde{Q}, \tilde{R})$ converges, in some sense, to (Q_{act}, R_{act}), while also achieving accurate estimates. Many iterations of this framework have been proposed over the years. One

approach suggests a forgetting factor on $(\tilde{Q}, \tilde{R})$ [1]. Some studies suggest using the measurement pre-fit residual innovation [1], while other propose the measurement post-fit residual [27]. The standard has been to update $(\tilde{Q}, \tilde{R})$ every update instant. The approach adopted herein is shown explicitly in Algorithm 3. The novelty of this approach is to only update the estimates of $(\tilde{Q}, \tilde{R})$ periodically. This is implemented using a design parameter T_s^{adapt}, the adaptation update rate - a parameter characterizing how often $(\tilde{Q}, \tilde{R})$ should be estimated. Larger values of T_s^{adapt} translate to longer averaging times in the calculation of $\tilde{C}_k$. Numerical studies have demonstrated that averaging times longer than a few samples translate to more informed calculations of $(\tilde{Q}, \tilde{R})$, and hence better estimates. Similarly, it has been observed that averaging times of a single sample, or just a few samples translates to inaccurate estimates, before the filter finally converges. Averaging times that are very long makes it so previously unseen dynamics and noise statistics will be uncompensated a larger amount of time. Thus, engineering judgment must be exercised when using this framework.

4.2 Multiple Model Adaptive Estimation

The Multiple Model Adaptive Estimation frame was originally applied to Kalman Filters and Extended Kalman Filters by Athans in [5]. In equation form, the algorithm can be seen in algorithm 4, or in block diagram form in figure 4.1.

4.3 A Bank of Adaptive Kalman Filters: MMAE Plus Innovations Based Adaptive Extended Kalman Filter

As the previous approach is still sensitive to an initial estimate, (Q_0, R_0). The use of a Filter Bank makes the filter less sensitive to initial guesses, provided at least one element of the bank contains initial covariance matrices "close" the distribution underlying the process and measurement. Here, "close" means that an initial pair

Algorithm 3: Innovations Based Adaptive Extended Kalman Filters

Initialize ($P_{0,j} = \sigma_j^P I, Q_{0,j} = \sigma_j^Q I, R_{0,j} = \sigma_j^R I$) and $m_0 = 0$, where I is an identity matrix of appropriate dimension;

Correct initial (Q, R) using the standard IAE with the first few measurements, leave initial P as is;

Select the Adaptation update rate T_s^{adapt} - how frequently the pair (Q_k, R_k) are updated;

for *all k (time)* **do**

$\hat{x}_k^- = A\hat{x}_{k-1}^+$;

$P_k^- = AP_{k-1}^+A^T + Q_k$;

$\tilde{y}_k = z_k - h(x_k^-)$;

$S_k = H_k P_k^- H_k^T + R_k$;

$K_k = P_k H_k^T S_k^{-1}$;

$\hat{x}_k = \hat{x}_k^- + K_k\tilde{y}$;

$P_k^+ = (I - K_k H_k)P_k^-$;

$\tilde{y_k^{prod}} = \tilde{y}_k\tilde{y}_k^T + \tilde{y}_{k-1}^{prod}$;

$m = m + 1$;

if $\mod\left(m, \frac{T_s^{adapt}}{T_s}\right) = 0$ **then**

$\tilde{C}_k = \frac{\tilde{y}_k^{prod}}{m}$;

$\tilde{Q}_k = K_k\tilde{C}_k K_k^T$;

$\tilde{R}_k = \tilde{C}_k + H_k P_k^+ H_k^T$;

$Q_{k+1} = \tilde{Q}_{k,j}$;

$R_{k+1} = \tilde{R}_{k,j}$;

$m = 0$;

end

end

(Q_0, R_0) results in an estimated pair (Q_k, R_k) such that $\mathcal{N}(0, \tilde{Q}) \approx \mathcal{N}(0, Q_{act})$ and $\mathcal{N}(0, \tilde{R}) \approx \mathcal{N}(0, R_{act})$. Moreover, the use of the IAE approach for each filter allows for each filter to estimate a more informed (Q, R). The approach adopted herein is shown explicitly in Algorithm 5. The novelty of this approach is that it transforms the filter design problem from one of populating two symmetric positive semi-definite matrices, to selecting the initial posterior probability distribution of the filter banks state estimates, and the size of the filter bank.

Algorithm 4: A Bank of Kalman Filters

Initialize ($P_{0,j} = \sigma_j^P I, Q_{0,j} = \sigma_j^Q I, R_{0,j} = \sigma_j^R I$) and $m_0 = 0$ for each j, where I is an identity matrix of appropriate dimension;

Initialize posterior probabilities of estimates of each filter bank - if nothing is known about the distribution, then select $p_{0,j}^{norm} = \frac{1}{j}$;

Correct $(Q_{0,j}, R_{0,j})$ using the standard IAE approach with the first measurement, leave $P_{0,j}$ as is;

Select the Adaptation update rate T_s^{adapt} - how frequently the pair $(Q_{k,j}, R_{k,j})$ are updated;

for *all k (time)* **do**

  for *every filter and, hence initial guess j* **do**

$$\hat{x}_{k,j}^- = A\hat{x}_{k-1,j}^+ \ ;$$

$$P_{k,j}^- = AP_{k-1,j}^+ A^T + Q_{k,j} \ ;$$

$$\tilde{y}_{k,j} = z_{k,j} - h(x_{k,j}^-) \ ;$$

$$S_{k,j} = H_{k,j} P_{k,j}^- H_{k,j}^T + R_{k,j} \ ;$$

$$K_{k,j} = P_{k,j} H_{k,j}^T S_{k,j}^{-1} \ ;$$

$$\hat{x}_{k,j}^+ = \hat{x}_{k,j}^- + K_{k,j}\tilde{y}_{k,j};$$

$$\beta_{k,j} = \frac{1}{(2\pi)^{\frac{n}{2}} \det S_{k,j}};$$

$$p_{k,j} = \beta_{k,j} e^{-\frac{1}{2}\tilde{y}_{k,j}^T S_{k,j}^{-1} \tilde{y}_{k,j}};$$

$$P_k^+ = (I - K_k H_k) P_k^- \ ;$$

$$\tilde{y_{k,j}^{prod}} = \tilde{y}_{k,j}\tilde{y}_{k,j}^T + \tilde{y}_{k-1,j}^{prod} \ ;$$

$$m_j = m_{j-1} + 1 \ ;$$

$$\underbrace{p_{k,j}^{net} = p_{k,j-1}^{net} + p_{k,j} p_{k,j-1}^{norm}}_{\text{Sum of the Weights}} \ ;$$

  end

  for *every j* **do**

$$\underbrace{p_{k,j}^{norm} = \frac{p_{k,j} p_{k,j-1}^{norm}}{p_{k,j}^{net}}}_{\text{Posterior Probability}} \ ;$$

$$\underbrace{\hat{\mathbf{x}}_{k,j}^+ = \hat{\mathbf{x}}_{k,j-1}^+ + p_{k,j}^{norm} \hat{x}_{k,j}^+}_{\text{Filter Bank Solution}};$$

  end

  $p_{k,j}^{net} = 0 \ ;$

end

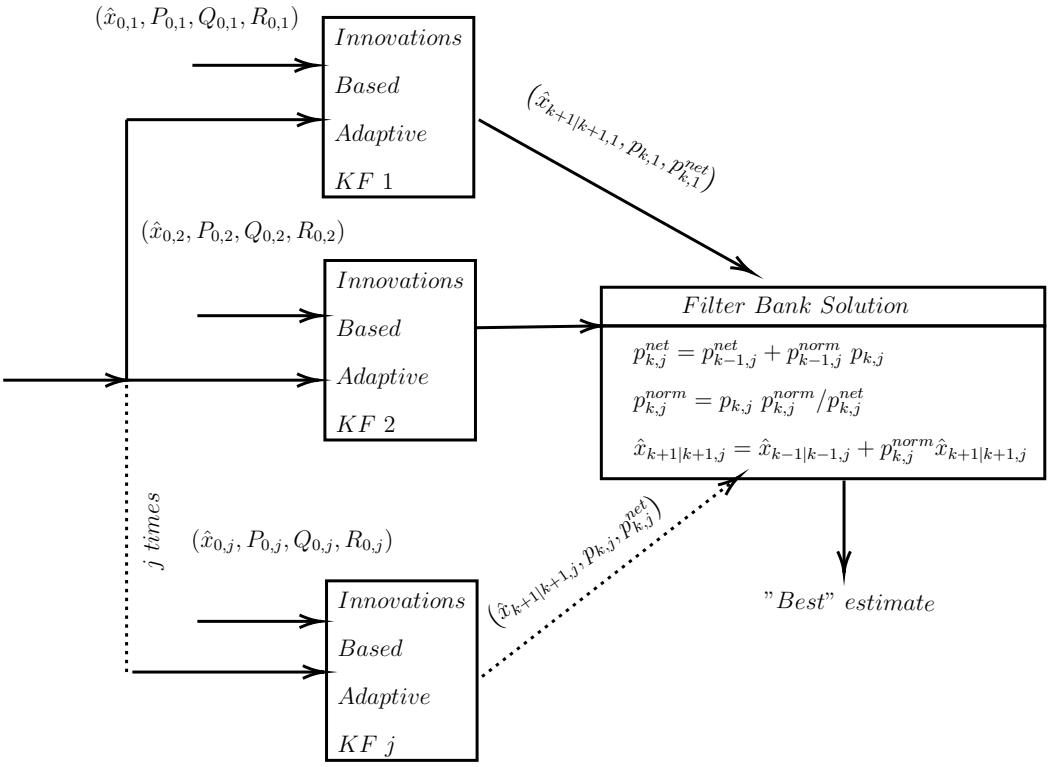

Figure 4.1: Block Diagram Representation of Bank Of Kalman Filters

Algorithm 5: A Bank of Innovations Based Adaptive Extended Kalman Filters

Initialize ($P_{0,j} = \sigma_j^P I, Q_{0,j} = \sigma_j^Q I, R_{0,j} = \sigma_j^R I$) and $m_0 = 0$ for each j, where I is an identity matrix ;

Initialize posterior probabilities of each filter- if nothing is known about the distribution, then $p_{0,j}^{norm} = \frac{1}{j}$;

Select the Adaptation update rate T_s^{adapt} - how frequently the pair $(Q_{k,j}, R_{k,j})$ are updated;

for *all k (time)* **do**

  for *every filter and, hence initial guess j* **do**

    $\hat{x}_{k,j}^- = A\hat{x}_{k-1,j}^+$;

    $P_{k,j}^- = AP_{k-1,j}^+ A^T + Q_{k,j}$;

    $\tilde{y}_{k,j} = z_{k,j} - h(x_{k,j}^-)$;

    $S_{k,j} = H_{k,j} P_{k,j}^- H_{k,j}^T + R_{k,j}$;

    $K_{k,j} = P_{k,j} H_{k,j}^T S_{k,j}^{-1}$;

    $\hat{x}_{k,j}^+ = \hat{x}_{k,j}^- + K_{k,j}\tilde{y}_{k,j}$;

    $\beta_{k,j} = \frac{1}{(2\pi)^{\frac{n}{2}} \det S_{k,j}}$;

    $p_{k,j} = \beta_{k,j} e^{-\frac{1}{2}\tilde{y}_{k,j}^T S_{k,j}^{-1} \tilde{y}_{k,j}}$;

    $P_k^+ = (I - K_k H_k) P_k^-$;

    $\tilde{y_{k,j}^{prod}} = \tilde{y}_{k,j}\tilde{y}_{k,j}^T + \tilde{y}_{k-1,j}^{prod}$;

    $m_j = m_{j-1} + 1$;

    $\underbrace{p_{k,j}^{net} = p_{k,j-1}^{net} + p_{k,j} p_{k,j-1}^{norm}}_{\text{Sum of the Weights}}$;

  end

  for *every j* **do**

    $\underbrace{p_{k,j}^{norm} = \frac{p_{k,j} p_{k,j-1}^{norm}}{p_{k,j}^{net}}}_{\text{Posterior Probability}}$;

    $\underbrace{\hat{\mathbf{x}}_{k,j}^+ = \hat{\mathbf{x}}_{k,j-1}^+ + p_{k,j}^{norm} \hat{x}_{k,j}^+}_{\text{Filter Bank Solution}}$;

  end

  if $\mod\left(m_j, \frac{T_s^{adapt}}{T_s}\right) = 0$ **then**

    $\tilde{C}_{k,j} = \frac{\tilde{y}_{k,j}^{prod}}{m_j}$;

    $\tilde{Q}_{k,j} = K_{k,j}\tilde{C}_{k,j}K_{k,j}^T$;

    $\tilde{R}_{k,j} = \tilde{C}_{k,j} + H_{k,j}P_{k,j}^+ H_{k,j}^T$;

    $Q_{k+1,j} = \tilde{Q}_{k,j}$;

    $R_{k+1,j} = \tilde{R}_{k,j}$;

    $m_j = 0$;

  end

  $p_{k,j}^{net} = 0$;

end

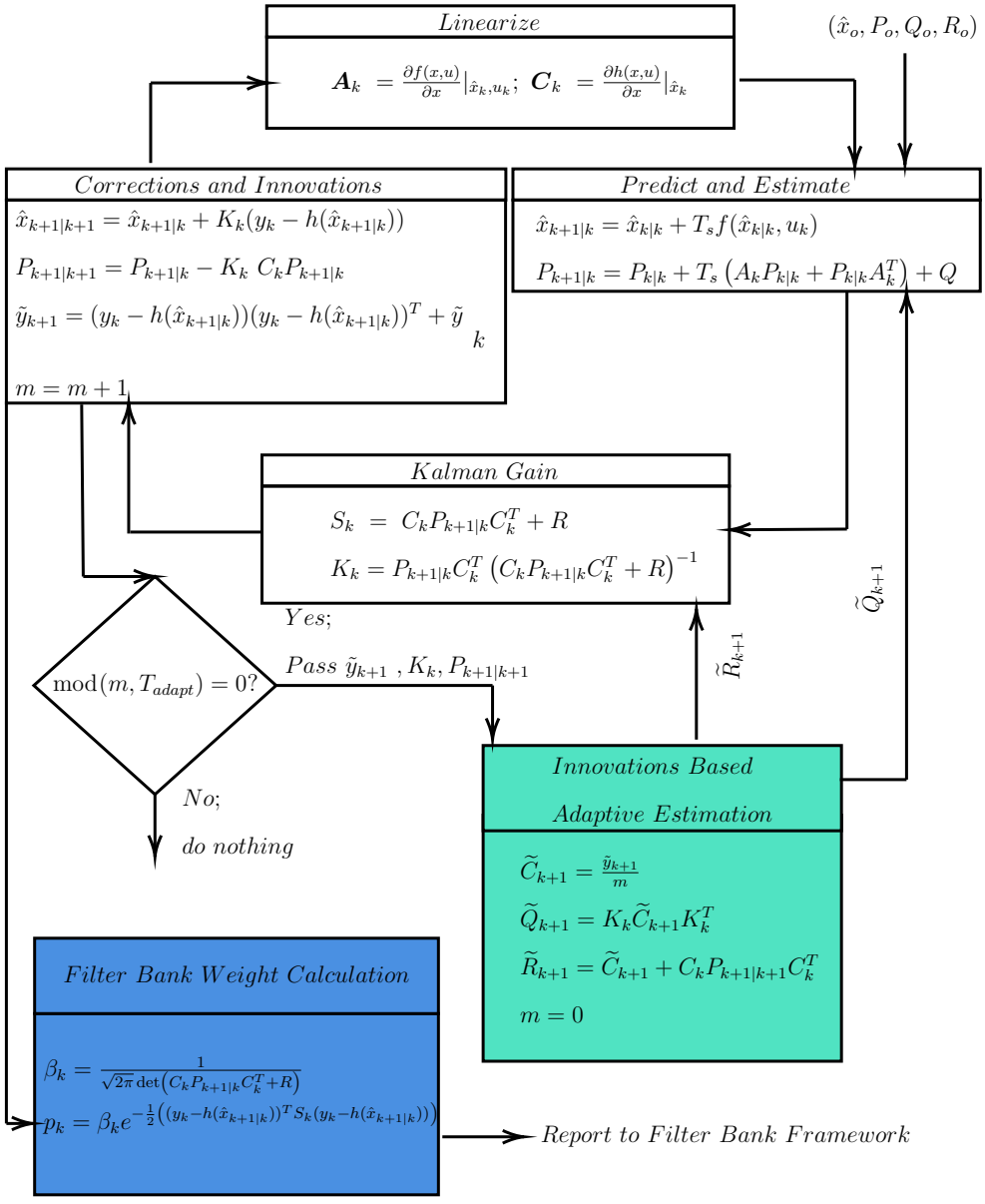

Figure 4.2: Block Diagram Representation of Each member of the Bank of Kalman Filters

Chapter 5

THE HYPER-PRECISE POSITIONING AND COMMUNICATIONS TIMING PROTOCOL

The Hyper-Precise Positioning and Communications Project [16] is a ground based 4 antenna communications ranging system that achieves distributed coherence for a network of users through novel system architecture and estimation algorithms. The timing protocol for a single antenna within the network can be seen in figure 5.2 and the geometry can be observed in figure 5.1.

From figure 5.2, a state space model can be derived as in equation 5.2, which is a 6-state model, linear in the state transition, and non-linear in the measurement:

$$\begin{bmatrix} \tau \\ \dot{\tau} \\ \ddot{\tau} \\ T \\ \dot{T} \\ \ddot{T} \end{bmatrix}^{(n-1)} = \begin{bmatrix} 1 & L_A & \frac{1}{2}L_A^2 & 0 & 0 & 0 \\ 0 & 1 & L_A & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & L_A & \frac{1}{2}L_A^2 \\ 0 & 0 & 0 & 0 & 1 & L_A \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{(n-1)} \begin{bmatrix} \tau \\ \dot{\tau} \\ \ddot{\tau} \\ T \\ \dot{T} \\ \ddot{T} \end{bmatrix}^{(n-3)} \tag{5.1}$$

$$\begin{aligned} \begin{bmatrix} t_{B,Rx}^{(n-1)} \\ t_{A,Rx}^{(n)} \end{bmatrix} &= \begin{bmatrix} t_{A,Tx}^{(n-1)} \\ t_{B,Tx}^{(n)} \end{bmatrix} + \begin{bmatrix} \tau^{(n-1)} - T^{(n-1)} \\ \tau^{(n-1)} + l_A\dot{\tau}^{(n-1)} + \frac{1}{2}l_A^2\ddot{\tau}^{(n-1)} + \\ T^{(n-1)} + l_A\dot{T}^{(n-1)} + \frac{1}{2}l_A^2\ddot{T}^{(n-1)} \end{bmatrix} \\ &= \begin{bmatrix} t_{A,Tx}^{(n-1)} \\ t_{B,Tx}^{(n)} \end{bmatrix} + \mathbf{h_A} \begin{pmatrix} \tau^{(n-1)}, \dot{\tau}^{(n-1)} \\ T^{(n-1)}, \dot{T}^{(n-1)} \end{pmatrix} \end{aligned} \tag{5.2}$$

Where L_A^n is the measurement update time, and l_A^n is the frame length. The physical description of the states can be found in table 5.2. A more thorough exploration of

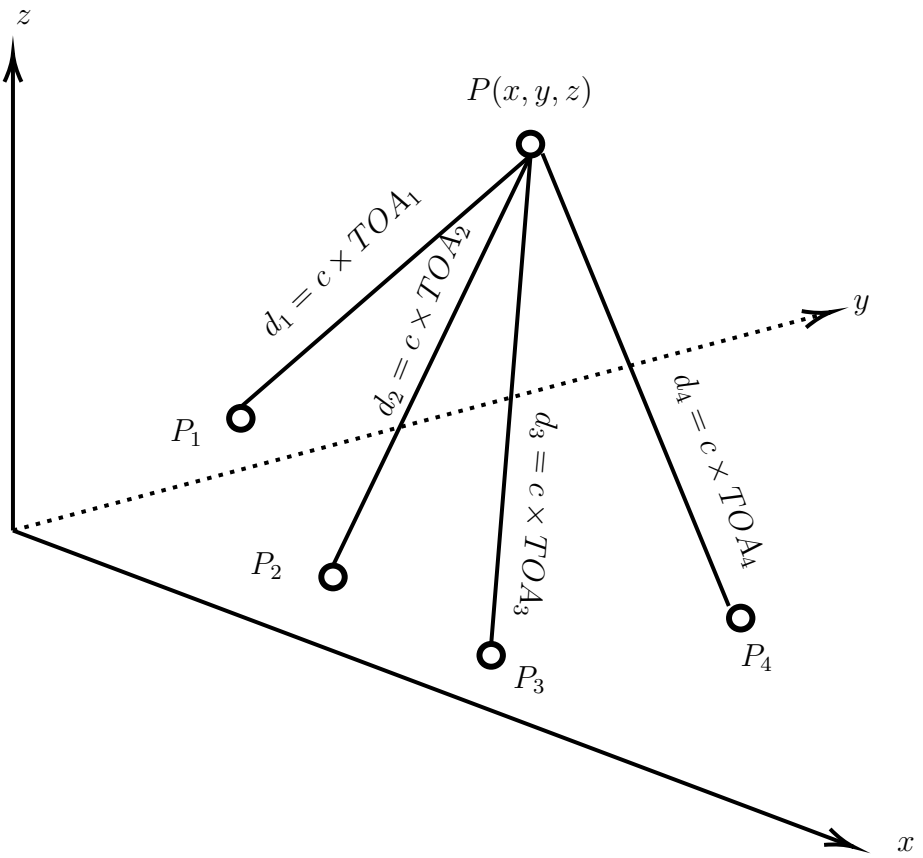

Figure 5.1: Ranging Diagram for the HPPC System.

such a system lies outside the scope of this thesis, but can be found in [16]. The primary purpose of this chapter is to determine: (1) how and if the benefits of including colored noise states in a simple Linear Kalman Filter in Chapter 3 carry over to a non-linear problem, (2) if adaptive architectures such as Innovations Based Adaptive Estimation and Multiple Model Adaptive Estimation are used, what are the benefits, if any and (3) how to embed colored noise states into a non-linear navigation filtering problem. Since the Extended Kalman Filter is the workhorse of navigation algorithms, such a topology will be used in conjunction with IAE, MMAE and combined IAE/MMAE methods.

seen in 5.2, the frame length, which is a non-linear function of the states, can be crudely calculated as

$$l_A^{(n-1)} = \tilde{t}_{A,Tx}^{(n)} - t_{A,Tx}^{(n-1)} \tag{5.3}$$

where the timestamps are (equation 5.4)

$$\tilde{t}_{A,Tx}^{(n)} = t_{B,Tx}^{(n)} + T^{(n-1)} + \dot{T}^{(n-1)} l_A^{(n-1)} + \frac{1}{2}\ddot{T}^{(n-1)} (l_A^{(n-1)})^2 \tag{5.4}$$

which then reduces to a quadratic equation in $l_A^{(n-1)}$

$$\frac{1}{2}\ddot{T}^{(n-1)} l_A^{(n-1)2} + (\dot{T}^{(n-1)} - 1) l_A^{(n-1)} + (T^{(n-1)} + t_{B,Tx}^{n} - t_{A,Tx}^{(n-1)}) = 0 \tag{5.5}$$

which then provides an explicit non-linear relation between $l_A^{(n-1)}$ and the states. This model and derivations were originally carried out in [33]. There is a direct relationship between the states $(T, \dot{T}, \ddot{T})$ and (x_1, x_2, x_3) from Chapter 2, as defined in table 2.1. These relationships can be seen in table 5.1. Expressions for the measurement Jacobian matrix can be found in [33]. These expressions can be used in an Extended or Unscented Kalman Filter framework, and will be in the subsequent sections. Note that in all examples in this section, the dynamics are linear, but the measurement is non-linear. The subsequent sections contain the following information (1) How to include a standard state-space oscillator model into the HPPC framework, (2) appending colored noise states to the HPPC framework, not including time of arrival estimates, with (adaptive) filtering results, (3) applying MMAE and IAE methods to the nominal HPPC model and (4) appending colored noise state to the HPPC framework, including time of arrival estimates.

Table 5.1: Relationship between HPPC Timing Protocol Model and Standard Clock Model.

HPPC State	**Physical Meaning**	**Unit**	**Standard Clock State**
T	Timing Drift	s	$x_1/(2\pi F_c)$
$\dot{T}$	Time Drift Derivative	s/s	x_2/F_c
$\ddot{T}$	Time Drift Acceleration	s/s^2	$x_3/(T_s F_c)$

Table 5.2: HPPC Timing Protocol For a Single Antenna State Space Model Description.

State	**Physical Meaning**	**Unit**
τ	Time of Arrival (range)	s
$\dot{\tau}$	Time of Arrival Derivative (range rate)	s/s
$\ddot{\tau}$	Time of Arrival Acceleration	s/s^2
T	Timing Drift	s
$\dot{T}$	Time Drift Derivative	s/s
$\ddot{T}$	Time Drift Acceleration	s/s^2

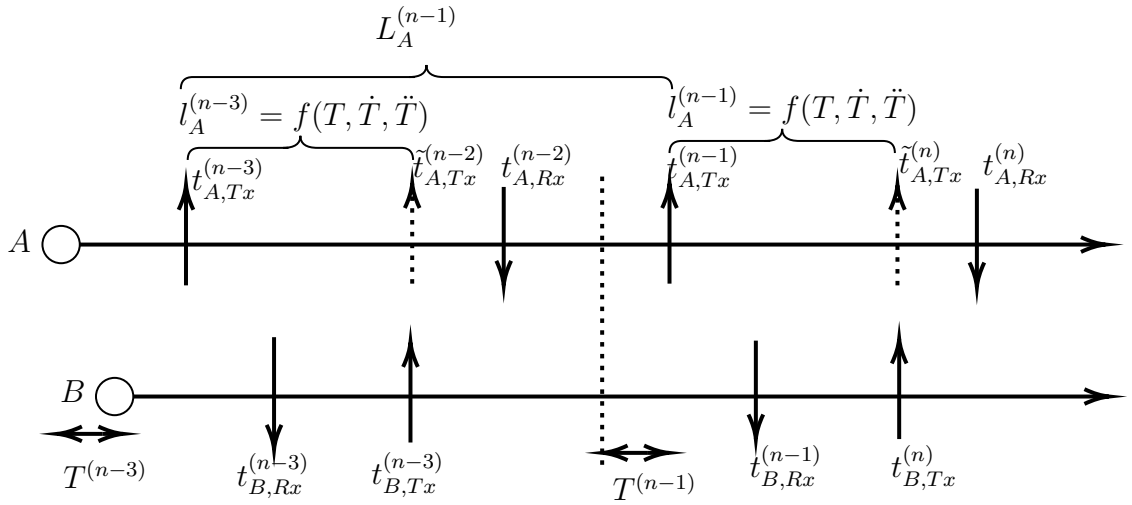

Figure 5.2: Non-linear Timing Protocol Used in Measurement Model For HPPC system.

5.1 Incorporating Oscillator Models Into the Timing Protocol

This section takes the models in Chapters 2 and 3, in equations 2.5, 3.7,3.10 and 3.21 and uses them to generate realistic timing drift. The example in Chapter 3, where an ideal phase measurement is available is not realistic, as a reference oscillator is always necessary to directly measure the phase of an oscillator. In the HPPC navigation system, all transmitters and receivers have the same oscillator, nominally a Stratum 3E High Stability Stability Oven Stabilized Oscillator OH300 Series. While each transmitter and receiver have the same oscillator physically, environmental differences such as temperature, dynamics and vibration create differences in noise profiles. Additionally, as has been demonstrated in the previous chapters, oscillators are fundamentally stochastic processes, so even if all external circumstances were equal, the stochastic nature of these systems does not allow for the guarantee that all clocks across the network will produce identical phase measurements, or time-stamps. We analyze a single receiver antenna. There are two clocks in this scenario, the transmit clock, and the receive clock. The transmit clock will be referred to as "Clock A" and the receive clock will be referred to as "Clock B". The procedure in algorithm 6 describes how to achieve realistic time-series datum which represents timing offsets between clocks.

5.2 Appending Colored Noise States to an HPPC Like Time-Keeping Adaptive Kalman Filter

In this section, Flicker Frequency Noise (FFN) states are added to the HPPC timing protocol shown in figure 5.2, with time of arrival states (τ,$\dot{\tau}$,$\ddot{\tau}$) omitted. This allows for a more straight forward investigation of how colored states improve estimation accuracy, if at all, when embedded into a non-linear navigation filter. The non-linear measurement becomes 5.13

$$\begin{bmatrix} t_{B,Rx}^{(n-1)} \\ t_{A,Rx}^{(n)} \end{bmatrix} = \begin{bmatrix} t_{A,Tx}^{(n-1)} \\ t_{B,Tx}^{(n)} \end{bmatrix} + \begin{bmatrix} -T^{(n-1)} \\ T^{(n-1)} + l_A \dot{T}^{(n-1)} + \frac{1}{2} l_A^2 \ddot{T}^{(n-1)} + \\ [F_{4,7}, F_{4,8}, F_{4,9}, F_{4,10}, F_{4,11}, F_{4,12}][\dot{T}_{FFN,i}, \dot{T}_{FFN,6}]^T \end{bmatrix}$$
$$= \begin{bmatrix} t_{A,Tx}^{(n-1)} \\ t_{B,Tx}^{(n)} \end{bmatrix} + \mathbf{h_A} \left(T^{(n-1)}, \dot{T}^{(n-1)}, \ddot{T}^{(n-1)}, \dot{T}_{FFN,i}, \dot{T}_{FFN,6} \right) \tag{5.13}$$

Note that the source of the non-linearity is l_A, which is found by solving the following equation

$$(\frac{1}{2}\ddot{T}^{(n-1)})(l_A^{(n-1)})^2 + (\dot{T}^{(n-1)} - 1) l_A^{(n-1)} + (T^{(n-1)} + t_{B,Tx}^n - t_{A,Tx}^{(n-1)}) + [F_{4,7}, F_{4,8}, F_{4,9}, F_{4,10}, F_{4,11}, F_{4,12}][\dot{T}_{FFN,i}, \dot{T}_{FFN,6}]^T = 0 \tag{5.14}$$

$$l_A = f(T, \dot{T}, \ddot{T}, \dot{T}_{FFN,i}, \dot{T}_{FFN,6}) \tag{5.15}$$

Where $i = 1, .., 5$. The linear state dynamics are now:

$$\begin{bmatrix} T \\ \dot{T} \\ \ddot{T} \\ \dot{T}_{FFN,1} \\ \dot{T}_{FFN,2} \\ \dot{T}_{FFN,3} \\ \dot{T}_{FFN,4} \\ \dot{T}_{FFN,5} \\ \dot{T}_{FFN,6} \end{bmatrix}^{(n-1)} = \begin{bmatrix} 1 & L_A & \frac{1}{2}L_A^2 & F_{4,7} & F_{4,8} & F_{4,9} & F_{4,10} & F_{4,11} & F_{4,12} \\ 0 & 1 & L_A & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & F_{7,7} & F_{7,8} & F_{7,9} & F_{7,10} & F_{7,11} & F_{7,12} \\ 0 & 0 & 0 & 0 & F_{8,8} & F_{8,9} & F_{8,10} & F_{8,11} & F_{8,12} \\ 0 & 0 & 0 & 0 & 0 & F_{9,9} & F_{9,10} & F_{9,11} & F_{9,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & F_{10,10} & F_{10,11} & F_{10,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & F_{11,11} & F_{11,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{(n-1)} \begin{bmatrix} T \\ \dot{T} \\ \ddot{T} \\ \dot{T}_{FFN,1} \\ \dot{T}_{FFN,2} \\ \dot{T}_{FFN,3} \\ \dot{T}_{FFN,4} \\ \dot{T}_{FFN,5} \\ \dot{T}_{FFN,6} \end{bmatrix}^{(n-3)} \tag{5.16}$$

Since closed form solutions to 5.20 are not available, numerical perturbation methods are used to compute the partial derivatives of l_A to populate the Jacobian measurement matrix. In order to gain insight into the effect of both adaptive filtering, and the inclusion of colored noise states in the model, a Monte Carlo simulation was conducted. The results can be seen in Appendix B, sections 1 and 2. Four scenarios are investigated

- Case (I): Clock States only - Only the standard 3 clock states (T,$\dot{T}$,$\ddot{T}$) are included in the Kalman Filter Model, no adaptive filtering
- Case (II): Clock States with FFN -The standard 3 clock states as well as the FFN states are included in the Kalman Filter Model

  $$(T, \dot{T}, \ddot{T}, \dot{T}_{FFN,1}, \dot{T}_{FFN,2}, \dot{T}_{FFN,3}, \dot{T}_{FFN,4}, \dot{T}_{FFN,5}, \dot{T}_{FFN,6})$$

  with no adaptive filtering
- Case (III): Clock States only with IAE - Only the standard 3 clock states (T,$\dot{T}$,$\ddot{T}$) are included in the Kalman Filter Model, and the IAE algorithm in algorithm 3 is used
- Case (IV): Clock States with FFN and IAE -The standard 3 clock states as well as the FFN states are included in the Kalman Filter Model

  $$(T, \dot{T}, \ddot{T}, \dot{T}_{FFN,1}, \dot{T}_{FFN,2}, \dot{T}_{FFN,3}, \dot{T}_{FFN,4}, \dot{T}_{FFN,5}, \dot{T}_{FFN,6})$$

  and the IAE algorithm in algorithm 3 is used

The parameters in table 5.3 were used as well as the initial covariance matrices shown in equation 5.19. Exactly 83 simulations were iterated with the parameters defined in tables 5.3 and 5.4. The results are the following

- Result (I): In order of increasing estimation accuracy, Case I < Case II < Case III < Case IV

- Result (II): Case III is almost always an order of magnitude increase over Case I

- Result (III): An unusual result is that is that Case III is always more accurate than Case II, results I and II lead to the conclusion that adaptively estimating the process noise covariance matrix using IAE, even if colored noise states are included, is the most influential design artifact in this framework. This might occur because $Q(1,1)$ includes a significant FFN term - an observation which comes from the Gauss Markov approach process noise covariance matrix (equation 3.16). While the process noise covariance matrix used to simulate noise comes from equation 3.22, closed form expressions for this matrix are not readily available because of computational complexity.

- Result (IV): Case IV is nearly always 5 times more accurate than Cases I and II, frequently an order of magnitude more accurate, aside from 2 examples, out of 83.

- Result (V): Case II is only a 100 picoseconds, or at best a 500 picoseconds more accurate than Case I

- Result (VI): Case IV is frequently 2 times more accurate than Case III, always 5 times more accurate when the initial fractional frequency offset ($\dot{T}$) is greater than 1e-8 s/s.

Table 5.3: Simulated Noise Parameters.

Constituent	**Value**
WFN	2.6153e-04 Hz
WPN	0.0048 Cycle
FFN	0.0012 Hz/s
RWFN	4.7735e-07 Hz/s
RRFN	2.6153e-09 Hz/s/s

Table 5.4: Simulated Noise Parameters.

Parameter	**Value**
measurement seed A	random integer $\in [0, 100]$
measurement seed B	random integer $\in [0, 100]$
process seed A	random integer $\in [0, 100]$
process seed B	random integer $\in [0, 100]$
fractional frequency A	random integer $\in [1, 50]$ times 1e-12
fractional frequency B	random integer $\in [0, 50]$ times 1e-12
time offset	random integer $\in [0, 1000]$ times 1e-11 s
fractional frequency offset	random value in [1e-8 1e-11]s/s

$$Q_o = diag([1e-9LA, 1e-3LA, 1e-30LA1_{1\times6}]) \tag{5.17}$$

$$P_o = Q_o \tag{5.18}$$

$$R_o = diag([1e-11, 1e-11]) \tag{5.19}$$

Algorithm 6: HPPC System Colored Noise Simulation Procedure

(I) Select Initial Time and Frequency Offsets for Clock B and Clock B ;

(II) Select different measurement and process noise seeding for Clock A and Clock B;

(III) **for** *For each Clock A, and B, Do* **do**

Propogate the clock states ;

for *For all time* **do**

$$\begin{bmatrix} x^1_{k+1} \\ x^2_{k+1} \\ x^3_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T_s & \frac{T_s^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^1_k \\ x^2_k \\ x^3_k \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{WFN} + u_{FFN} \\ u_{RWFN} \\ u_{RRFN} \end{bmatrix} \tag{5.6}$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x^1_{k+1} \\ x^2_{k+1} \\ x^3_{k+1} \end{bmatrix} + u_{WPN} \tag{5.7}$$

Create the noise time series with the process noise covariance Q_k (sigma parameters selected by optimizer) ;

$$Q_k(T_s) = \begin{bmatrix} \sigma^2_{WFN} T_s + \frac{1}{3}\sigma^2_{RWFN} T_s^3 + \frac{1}{20}\sigma^2_{RRFN} T_s^5 & \frac{1}{2}T_s^2\sigma^2_{RWFN} + \frac{1}{8}\sigma^2_{RRFN} T_s^4 & \frac{1}{6}\sigma^2_{RRFN} T_s^3 \\ \frac{1}{2}T_s^2\sigma^2_{RWFN} + \frac{1}{8}\sigma^2_{RRFN} T_s^4 & \sigma^2_{RWFN} T_s + \frac{1}{3}\sigma^2_{RRFN} T_s^3 & \frac{1}{2}\sigma^2_{RRFN} T_s^2 \\ \frac{1}{6}\sigma^2_{RRFN} T_s^3 & \frac{1}{2}\sigma^2_{RRFN} T_s^2 & \sigma^2_{RRFN} T_s^2 \end{bmatrix} \tag{5.8}$$

And the measurement noise covariance $R = \sigma^2_{WPN}$;

Add colored noise states if required using a Kasdin, Gauss-Markov, or Oustaloup approach ;

end

;

(IV) Take the simulated phase x^1, for each clock, and convert to fractional frequency drift ($\dot{T}$): ;

$$\dot{T}_{clockA} = \frac{f^o_{clockA}}{F_c} + \frac{[x^{1,clockA}_{2:end} - x^{1,clockA}_{1:end-1}]}{T_s 2\pi F_c} \tag{5.9}$$

$$\dot{T}_{clockB} = \frac{f^o_{clockB}}{F_c} + \frac{[x^{1,clockB}_{2:end} - x^{1,clockB}_{1:end-1}]}{T_s 2\pi F_c} \tag{5.10}$$

Then, the total time-offset derivative between Clocks A and B is ;

$$\dot{T} = \dot{T}_{clockA} - \dot{T}_{clockB} \tag{5.11}$$

(V) Then, the net time-offset is propagated in time by ;

$$T^{n+1} = T^n + \dot{T}^n L^n_A \tag{5.12}$$

end

5.3 A Bank of Adaptive Extended Kalman Filters Applied to the HPPC Framework

In this section algorithm 5 is applied to the non-linear timing protocol model. A simple sinusoid is use to model τ. The process described in algorithm 6 is used to model T and $\dot{T}$. A bank of 50 Extended Kalman Filters is used. The initial triplet (P_o, Q_o, R_o) are selected as random positive definite diagonal matrices. State estimates are all initialized to 0. A Monte Carlo simulation was conducted with difference time and fractional frequency offsets. The worst case timing error across all simulations is roughly 80ps over a 10 second simulation. For the initial simulation the Kasdin and Oustaloup methods were used to generate colored noise. Future work will include the Oustaloup method and Gauss-Markov, as well as including colored noise states in the Bank of Adaptive Extended Kalman Filters. This numerical experiment demonstrates the reliability of method. Figures 5.3,5.4 ,5.5 and 5.6 represent standard behavior observed in the Monte Carlo simulation. Appendix B contains a large array of different results. All results assume that the initial posterior probability distribution of each bank is equally likely.

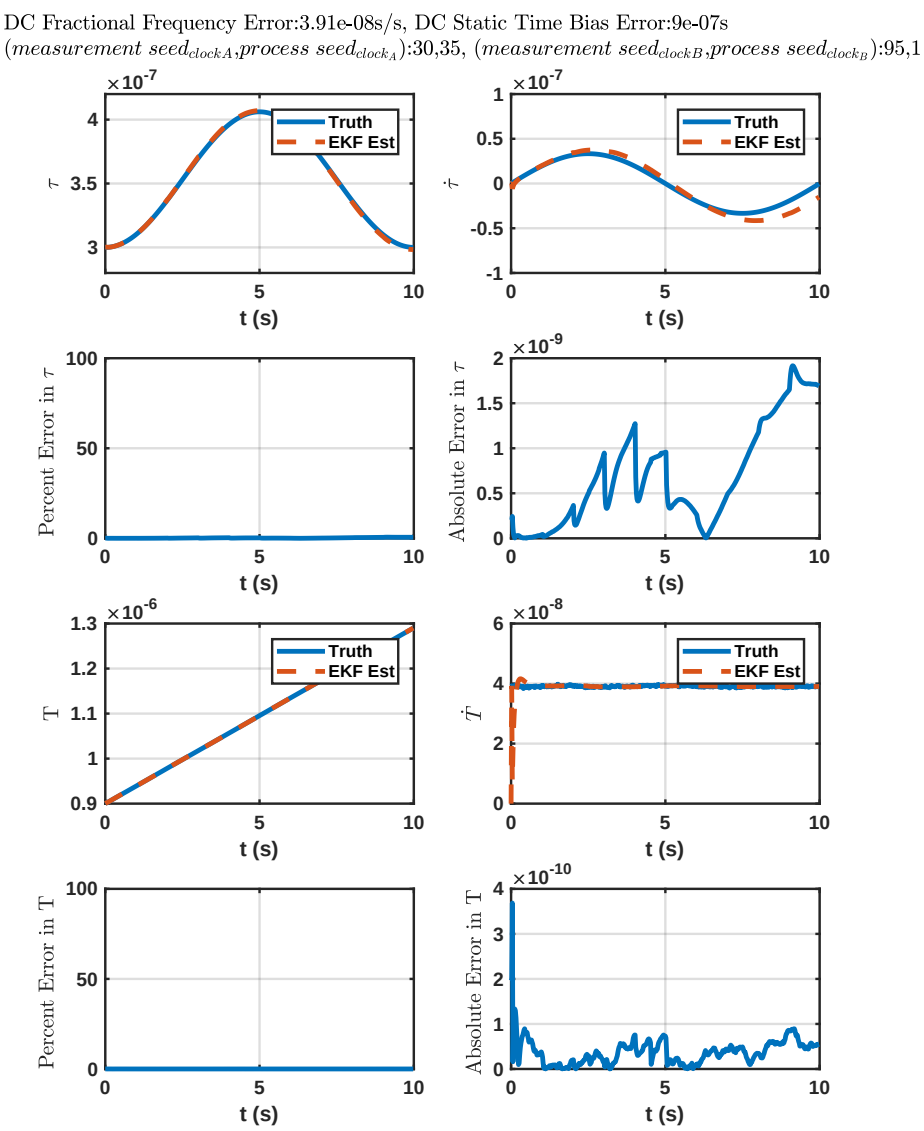

Figure 5.3: State Estimates for HPPC Timing Protocol Adaptive Extended Kalman Filter using the Kasdin Method For Noise Generation.

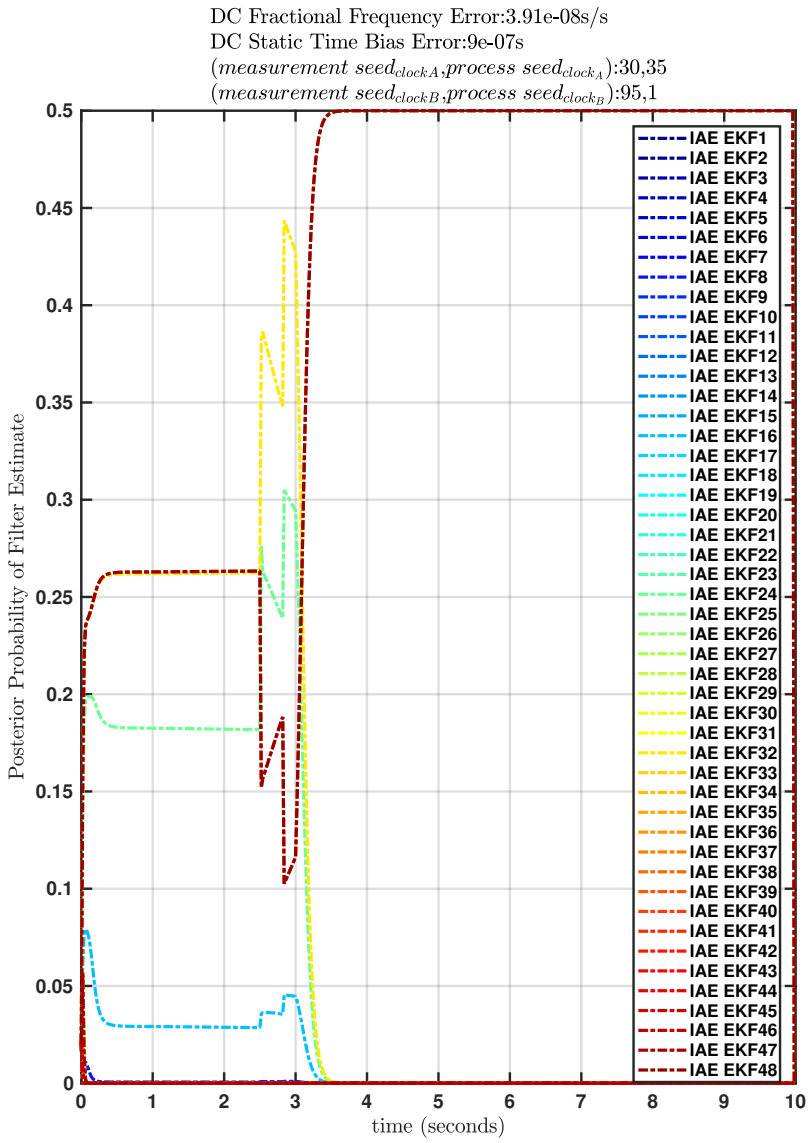

Figure 5.4: Posterior Probability of State Estimates From Each Filter Bank Constituent Using the Kasdin Method Ror Noise Generation.

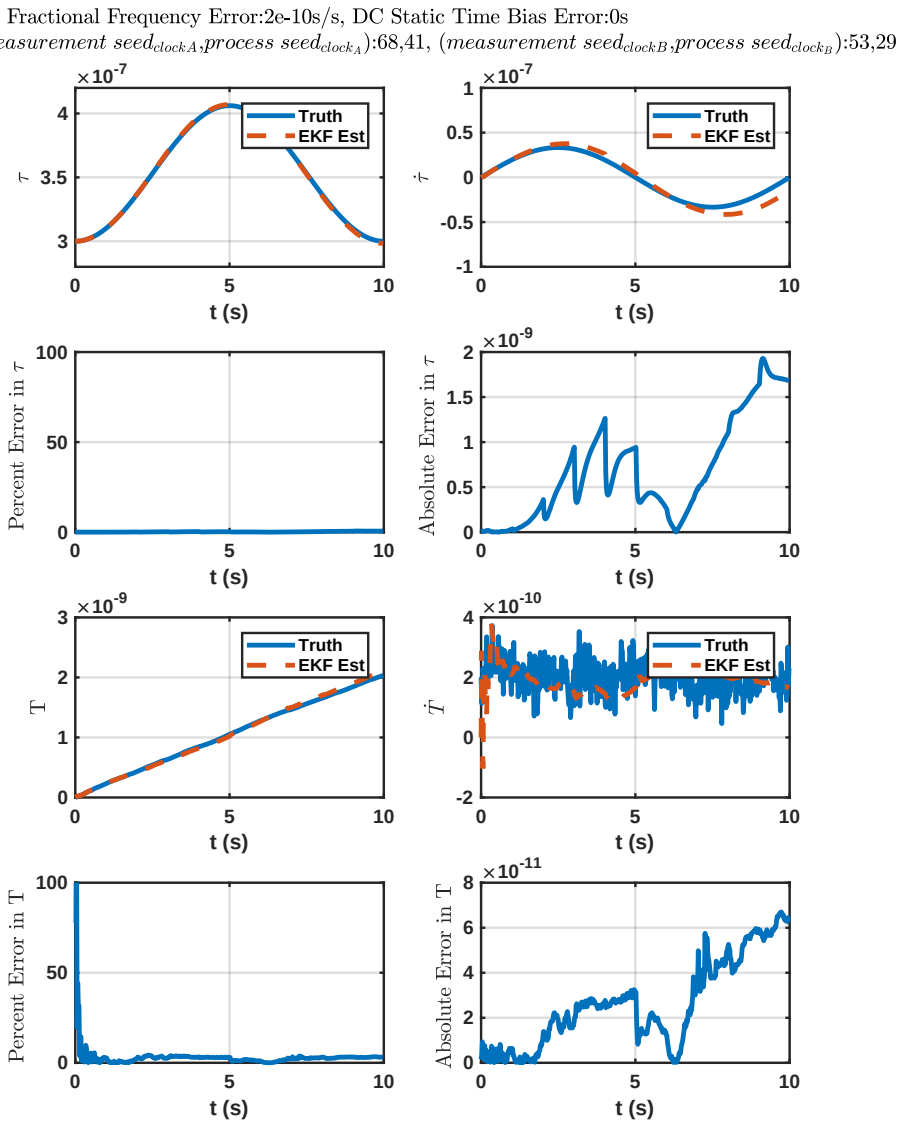

Figure 5.5: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using The Oustaloup Method For Noise Generation.

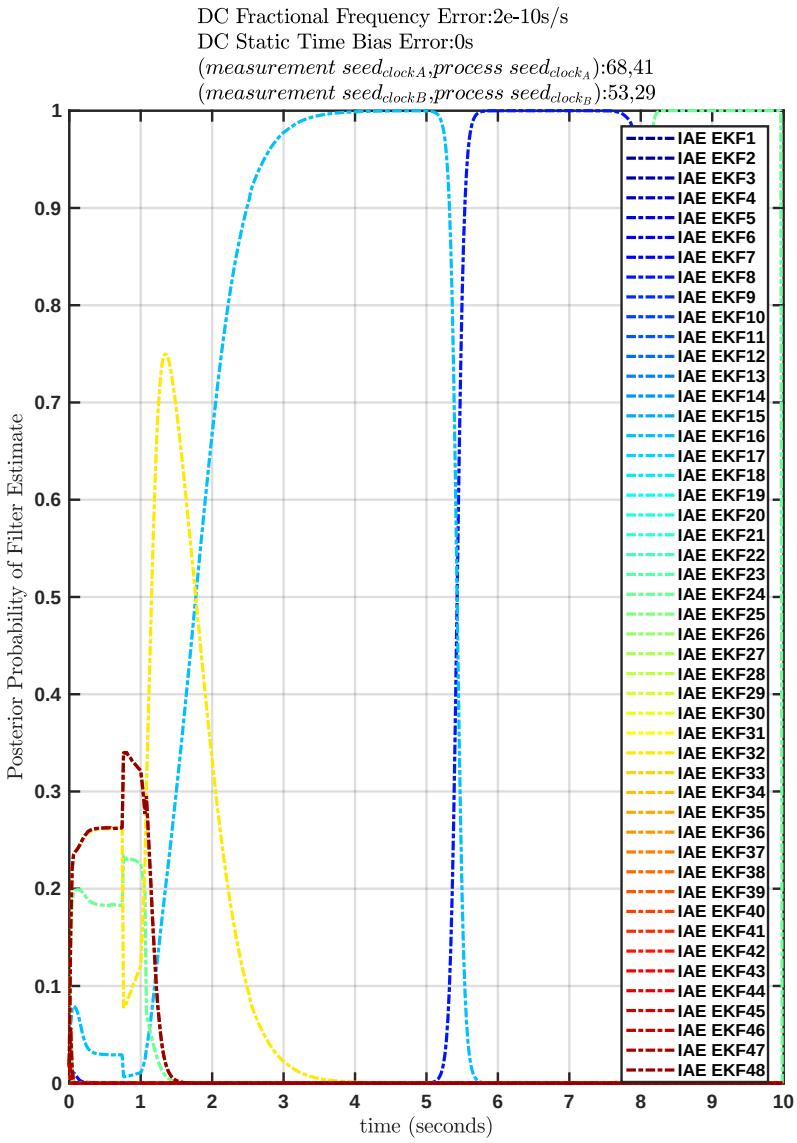

Figure 5.6: Posterior Probability of State Estimates from each Filter Bank Constituent using the Oustaloup Method for noise generation.

5.4 Appending Colored Noise States to HPPC Time-Keeping Adaptive Kalman Filter

As it was demonstrated in figure 3.14 that only at most 6 states are needed for a notable estimation accuracy increase, 6 states are appended to the F matrix in equation 5.2, and can be seen in equation 5.49 as well as table 5.5. All other entries are 0. The addition of new states also changes equation 5.5 from a quadratic, to one that doesn't have a closed form solution, which is shown in equation 5.20. The specific entries in the F matrix are defined in equation 5.49.

$$(\frac{1}{2}\ddot{T}^{(n-1)})(l_A^{(n-1)})^2 + (\dot{T}^{(n-1)} - 1)l_A^{(n-1)} + (T^{(n-1)} + t_{B,Tx}^n - t_{A,Tx}^{(n-1)}) + [F_{4,7}, F_{4,8}, F_{4,9}, F_{4,10}, F_{4,11}, F_{4,12}][\dot{T}_{FFN,i}, \dot{T}_{FFN,6}]^T = 0 \quad (5.20)$$

the state update becomes 5.21

$$\begin{bmatrix} \tau \\ \dot{\tau} \\ \ddot{\tau} \\ T \\ \dot{T} \\ \ddot{T} \\ \dot{T}_{FFN,1} \\ \dot{T}_{FFN,2} \\ \dot{T}_{FFN,3} \\ \dot{T}_{FFN,4} \\ \dot{T}_{FFN,5} \\ \dot{T}_{FFN,6} \end{bmatrix}^{(n-1)} = \begin{bmatrix} 1 & L_A & \frac{1}{2}L_A^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & L_A & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & L_A & \frac{1}{2}L_A^2 & F_{4,7} & F_{4,8} & F_{4,9} & F_{4,10} & F_{4,11} & F_{4,12} \\ 0 & 0 & 0 & 0 & 1 & L_A & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & F_{7,7} & F_{7,8} & F_{7,9} & F_{7,10} & F_{7,11} & F_{7,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & F_{8,8} & F_{8,9} & F_{8,10} & F_{8,11} & F_{8,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & F_{9,9} & F_{9,10} & F_{9,11} & F_{9,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & F_{10,10} & F_{10,11} & F_{10,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & F_{11,11} & F_{11,12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{(n-1)} \begin{bmatrix} \tau \\ \dot{\tau} \\ \ddot{\tau} \\ T \\ \dot{T} \\ \ddot{T} \\ \dot{T}_{FFN,1} \\ \dot{T}_{FFN,2} \\ \dot{T}_{FFN,3} \\ \dot{T}_{FFN,4} \\ \dot{T}_{FFN,5} \\ \dot{T}_{FFN,6} \end{bmatrix}^{(n-3)} \quad (5.21)$$

The non-linear measurement becomes 5.23, where $t = l_A$, and

$$l_A = f(\tau, \dot{\tau}, \ddot{\tau}, T, \dot{T}, \ddot{T}, \dot{T}_{FFN,1}, \dot{T}_{FFN,2}, \dot{T}_{FFN,3}, \dot{T}_{FFN,4}, \dot{T}_{FFN,5}, \dot{T}_{FFN,6}) \quad (5.22)$$

$$\begin{bmatrix} t_{B,Rx}^{(n-1)} \\ t_{A,Rx}^{(n)} \end{bmatrix} = \begin{bmatrix} t_{A,Tx}^{(n-1)} \\ t_{B,Tx}^{(n)} \end{bmatrix} + \begin{bmatrix} \tau^{(n-1)} - T^{(n-1)} \\ \tau^{(n-1)} + l_A\dot{\tau}^{(n-1)} + \frac{1}{2}l_A^2\ddot{\tau}^{(n-1)} + \\ T^{(n-1)} + l_A\dot{T}^{(n-1)} + \frac{1}{2}l_A^2\ddot{T}^{(n-1)} + \\ [F_{4,7}, F_{4,8}, F_{4,9}, F_{4,10}, F_{4,11}, F_{4,12}][\dot{T}_{FFN,i}, \ddot{T}_{FFN,6}]^T \end{bmatrix}$$

$$= \begin{bmatrix} t_{A,Tx}^{(n-1)} \\ t_{B,Tx}^{(n)} \end{bmatrix} + \mathbf{h_A} \begin{pmatrix} \tau^{(n-1)}, \dot{\tau}^{(n-1)}, \ddot{\tau}^{(n-1)} \\ T^{(n-1)}, \dot{T}^{(n-1)}, \ddot{T}^{(n-1)}, \dot{T}_{FFN,i}, \dot{T}_{FFN,6} \end{pmatrix} \tag{5.23}$$

Since closed form solutions to 5.20 are not available, numerical perturbation methods are used to compute the partial derivatives of l_A to populate the Jacobian measurement matrix.

Table 5.5: HPPC Timing Protocol For a Single Antenna State Space Model Description.

State	**Physical Meaning**	**Unit**
τ	Time of Arrival (range)	s
$\dot{\tau}$	Time of Arrival Derivative (range rate)	s/s
$\ddot{\tau}$	Time of Arrival Acceleration	s/s^2
T	Timing Drift	s
$\dot{T}$	Time Drift Derivative	s/s
$\ddot{T}$	Time Drift Acceleration	s/s^2
$\dot{T}_{FFN,1}$	Flicker Time Drift 1	s/s
$\dot{T}_{FFN,2}$	Flicker Time Drift 2	s/s
$\dot{T}_{FFN,3}$	Flicker Time Drift 3	s/s
$\dot{T}_{FFN,4}$	Flicker Time Drift 4	s/s
$\dot{T}_{FFN,5}$	Flicker Time Drift 5	s/s
$\ddot{T}_{FFN,6}$	Flicker Time Drift Acceleration	s/s^2

$$F_{4,7} = 4466835.921 - 4466835.921\,\mathrm{e}^{-0.000002238721139\,t} \tag{5.24}$$

$$F_{4,8} = 62878.62811\,\mathrm{e}^{-0.000002238721139\,t} + 8912.509405 - 71791.13752\,\mathrm{e}^{-0.0001412537545\,t} \tag{5.25}$$

$$F_{4,9} = 109.6925132\,\mathrm{e}^{-0.000002238721139\,t} + 17.78279432 - 1138.062739\,\mathrm{e}^{-0.008912509381\,t} + 1010.587432\,\mathrm{e}^{-0.0001412537545\,t}$$

$$F_{4,10} = 0.2184295207\,\mathrm{e}^{-0.000002238721139\,t} + 0.03548133101 + 16.02024902\,\mathrm{e}^{-0.008912509381\,t} - 18.03714165\,\mathrm{e}^{-0.5623413252\,t} + 1.762981776\,\mathrm{e}^{-0.0001412537545\,t} \tag{5.26}$$

$$F_{1,11} = 0.0004358104849\,\mathrm{e}^{-0.000002238721139\,t} + 0.00007079454808 - 0.2858694459\,\mathrm{e}^{-35.48133892\,t} + 0.02794751468\,\mathrm{e}^{-0.008912509381\,t} + 0.2539047197\,\mathrm{e}^{-0.5623413252\,t} + 0.003510606433\,\mathrm{e}^{-0.0001412537545\,t} \tag{5.27}$$

$$F_{1,12} = -869.5549834\,\mathrm{e}^{-0.000002238721139\,t} + 996.5583627 - 0.2498806149\,\mathrm{e}^{-35.48133892\,t} - 13.97899013\,\mathrm{e}^{-0.008912509381\,t} - 1.762932169\,\mathrm{e}^{-0.5623413252\,t} + 0.0003162278708\,t - 111.0115763\,\mathrm{e}^{-0.0001412537545\,t} \tag{5.28}$$

$$F_{7,7} = \mathrm{e}^{-0.000002238721139\,t} \tag{5.29}$$

$$F_{7,8} = -0.02815354279\,\mathrm{e}^{-0.00007174623782\,t}\sinh\left(0.00006950751668\,t\right) \tag{5.30}$$

$$F_{7,9} = -0.00002455709483\,\mathrm{e}^{-0.000002238721139\,t} - 0.0001981554931\,\mathrm{e}^{-0.0001412537545\,t} + 0.0002227125879\,\mathrm{e}^{-0.008912509381\,t} \tag{5.31}$$

$$F_{7,10} = -0.00000004890027708\,\mathrm{e}^{-0.000002238721139\,t} - 0.0000003456846104\,\mathrm{e}^{-0.0001412537545\,t} - 0.000003135074188\,\mathrm{e}^{-0.008912509381\,t} + 0.000003529659075\,\mathrm{e}^{-0.5623413252\,t} \tag{5.32}$$

$$F_{7,11} = 0.00000005594130190\,\mathrm{e}^{-35.48133892\,t} - 9.756582145 \times 10^{-11}\,\mathrm{e}^{-0.000002238721139\,t} - 0.0000000006883579940\,\mathrm{e}^{-0.0001412537545\,t} - 0.00000005469174153\,\mathrm{e}^{-0.008912509381\,t} - 0.00000004968620393\,\mathrm{e}^{-0.5623413252\,t} \tag{5.33}$$

$$F_{7,12} = 0.00000004889870925\,\mathrm{e}^{-35.48133892\,t} + 0.0001946691675\,\mathrm{e}^{-0.000002238721139\,t} + 0.00002176709643\,\mathrm{e}^{-0.0001412537545\,t} + 0.000002735611111\,\mathrm{e}^{-0.008912509381\,t} + 0.0000003449853443\,\mathrm{e}^{-0.5623413252\,t} - 0.0002195657591 \tag{5.34}$$

$$F_{8,8} = \mathrm{e}^{-0.0001412537545\,t} \tag{5.35}$$

$$F_{8,9} = -0.02815354281\,\mathrm{e}^{-0.004526881568\,t}\sinh\left(0.004385627813\,t\right) \tag{5.36}$$

$$F_{8,10} = -0.00002455709489\,\mathrm{e}^{-0.0001412537545\,t} + 0.0002227125880\,\mathrm{e}^{-0.5623413252\,t} - 0.0001981554931\,\mathrm{e}^{-0.008912509381\,t} \tag{5.37}$$

$$F_{8,11} = -0.00000004890027648\,\mathrm{e}^{-0.0001412537545\,t} - 0.000003135074190\,\mathrm{e}^{-0.5623413252\,t} + 0.000003529659077\,\mathrm{e}^{-35.48133892\,t} - 0.0000003456846108\,\mathrm{e}^{-0.008912509381\,t} \tag{5.38}$$

$$F_{8,12} = 0.001546313233\,\mathrm{e}^{-0.0001412537545\,t} + 0.00002176770529\,\mathrm{e}^{-0.5623413252\,t} + 0.000003085301326\,\mathrm{e}^{-35.48133892\,t} + 0.0001729070366\,\mathrm{e}^{-0.008912509381\,t} - 0.001744073277 \tag{5.39}$$

$$F_{9,9} = \mathrm{e}^{-0.008912509381\,t} \tag{5.40}$$

$$F_{9,10} = -0.02815354280\,\mathrm{e}^{-0.2856269173\,t}\sinh\left(0.2767144079\,t\right) \tag{5.41}$$

$$F_{9,11} = 0.0002227125880\,\mathrm{e}^{-35.48133892\,t} - 0.00002455709485\,\mathrm{e}^{-0.008912509381\,t} - 0.0001981554931\,\mathrm{e}^{-0.5623413252\,t} \tag{5.42}$$

$$F_{9,12} = 0.0001946747343\,\mathrm{e}^{-35.48133892\,t} + 0.01228314567\,\mathrm{e}^{-0.008912509381\,t} + 0.001375849535\,\mathrm{e}^{-0.5623413252\,t} - 0.01385366994 \tag{5.43}$$

$$F_{10,10} = \mathrm{e}^{-0.5623413252\,t} \tag{5.44}$$

$$F_{10,11} = -0.02815354281\,\mathrm{e}^{-18.02184012\,t}\sinh\left(17.45949880\,t\right) \tag{5.45}$$

$$F_{10,12} = 0.01230461088\,\mathrm{e}^{-35.48133892\,t} - 0.1100436095 + 0.09773899861\,\mathrm{e}^{-0.5623413252\,t} \tag{5.46}$$

$$F_{11,11} = \mathrm{e}^{-35.48133892\,t} \tag{5.47}$$

$$F_{11,12} = -0.8741074589 + 0.8741074589\,\mathrm{e}^{-35.48133892\,t} \tag{5.48}$$

$$F_{12,12} = 1 \tag{5.49}$$

Chapter 6

MODERN CONTROL THEORY IN PHASE LOCKED LOOP DESIGN

Phase Locked Loops are a critical tool in communications systems in general, but especially in ranging systems [13, 19]. Within a ranging system, such as a ranging receiver, there are many different phase locked loops. On the analog end, there are frequency synthesizes, that typically produce a sample clock, and the downconverting wave. Such systems can be observed in figure

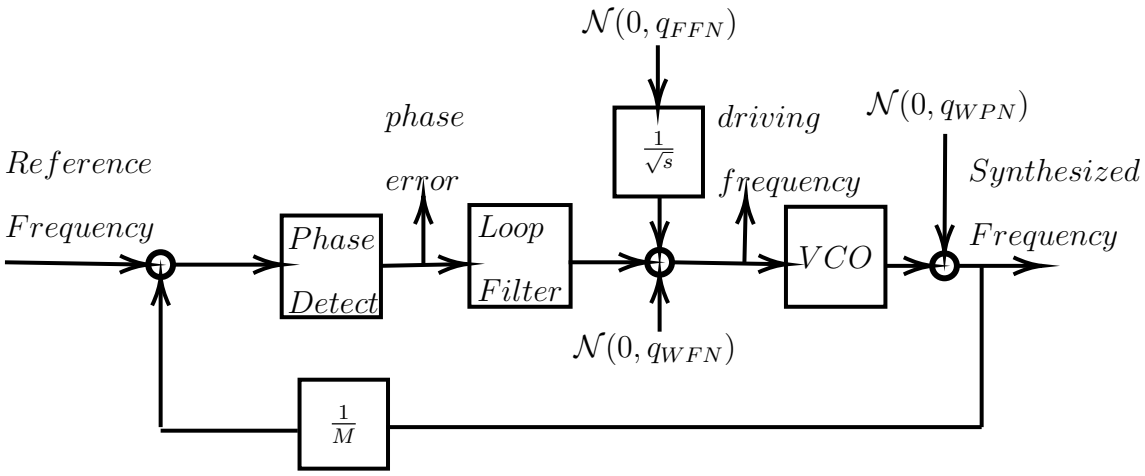

Figure 6.1: Topology of a Standard Phase Locked Loop.

Where M is the divider gain, the phase detector is a non-linear phase detection operation, the loop filter is typically a PI, or a PID controller. If spurs are a concern, a notch filter may be appended to a PI or a PID controller. The Voltage Controlled Oscillator (VCO) is represented by the expression $\frac{K_{osc}}{s}$ - an integrator and a gain. The noise sources that drive the VCO are typically a combination of a flicker and floor term. This means, that the noise within the in-band, or the loop-noise bandwidth

of the transfer function from reference frequency to synthesized frequency, is strictly worse from a combination of the VCO's noise constituents, and the divider leading to amplification of the in-band noise. For downconverter and sample clock design, designers typically strive to achieve a target phase margin and bandwidth. However, there are many different closed loop transfer function that contribute to the quality of the synthesizes frequency, such as the FFN and WFN plant input disturbances. Hence, it is desirable to shape the closed loop transfer function to minimize frequency stability degradation imposed by the VCO's inherent noise characteristics. Nothing in the literature currently addresses this. Classical control is the standard for the design of analog phase locked loops.

Additionally, tracking loops are typically used to deal with Doppler and phase dynamics in a signal demodulation process. In this case $M = 1$, as these loops strive to track the phase and or frequency of the received, downconverted and digitized signal. For the design, again classical control, or the work in [13] is used. However, for high dynamics, or loop-shaping procedures, these methods are lacking. In this section, novel $\mathcal{H}^\infty$ control methods will be used to design Phase Locked Loops for high dynamics and minimizing frequency stability degradation.

6.1 $\mathcal{H}^\infty$ Control Preliminaries

First, an $\mathcal{H}^\infty$ space is the following

$\mathcal{H}^\infty$ – ***Algebra of Stable Systems.***

$$\|G\|_{\mathcal{H}^\infty} \stackrel{\text{def}}{=} \sup_{Re\ s>0} \bar{\sigma}_{max}[G(s)] = \text{ess sup}\, \bar{\sigma}_{max}[G(j\omega)]$$

(i.e. peak value on magnitude response, or peak singular value in the multivariable case)

where $\bar{\sigma}()$ represents the maximum singular value. Similarly, the topology of $\mathcal{H}^\infty$ spaces are defined by the norm

Induced $\mathcal{L}^2$ Norm. (System Gain)

$$\|T\|_{\mathcal{L}^2\to\mathcal{L}^2} = \sup_{\substack{x\in\mathcal{L}^2\\ x\neq 0}} \frac{\|Tx\|_{\mathcal{L}^2}}{\|x\|_{\mathcal{L}^2}} = \|T\|_{\mathcal{H}^\infty} \tag{6.1}$$

and stability criterion ***Stability.***

$$\exists\gamma\in[0,\infty) \text{ s.t. } \|Tx\|_{\mathcal{L}^2} \leq \gamma \|x\|_{\mathcal{L}^2} \tag{6.2}$$

6.2 Generalized $\mathcal{H}^\infty$ Control Design Framework

The Generalized $\mathcal{H}^\infty$ problem was introduced in [29] and [28]. The novelty of the approach is that it allows for the designer to shape closed loop transfer functions at the plant input and the plant output. In addition to this, the method can be applied to linear infinite-dimensional plants. The problem can be described as the following:

Given: Possibly infinite-dimensional plant, $P\in\mathcal{H}^\infty$. Stable finite-dimensional weightings, $W_1 - W_6 \in R\mathcal{H}^\infty$.

$$\tilde{\mu_n}(\gamma) \stackrel{\text{def}}{=} \inf_{\substack{K_n \text{ stabilizing}\\ C(T_{wz}(K_n))<\gamma}} \left\{ \max \left(\left\| \begin{bmatrix} W_1 \\ W_2 K_n \\ W_3 P K_n \end{bmatrix} [I - PK_n]^{-1} \right\|_{\mathcal{H}^\infty}, \left\| \begin{bmatrix} W_4 \\ W_5 P \\ W_6 K_n P \end{bmatrix} [I - K_n P]^{-1} \right\|_{\mathcal{H}^\infty} \right) < \gamma \right\}$$

$W_1 - W_6, W_2^{-1}, \in R\mathcal{H}^\infty$

3) Compute near-optimal K_n such that

$$|\tilde{\mu_n}(\gamma) - \mu_n(\gamma)| \leq \epsilon.$$

While such an approach is typically used for multivariable plants, it is also useful when the closed loop transfer function $T_{d_i \to synthfreq} = P([I - K_n P]^{-1})$ is of interest in the design of the sample clock phase locked loop, because it allows the designer to minimize frequency stability degradation caused by the VCO's inherent noise characteristics. Since phase locked loops are fundamentally single-input-single-output, it's given that $([I - K_n P]^{-1}) = [I - PK_n]^{-1}$, which means that many of the expressions in 6.3 are the same. With this in mind, we modify the objective function to be the following:

$$\tilde{\mu_n}(\gamma) \stackrel{\text{def}}{=} \inf_{\substack{K_n \ stabilizing \\ C(T_{wz}(K_n)) < \gamma}} \left\{ \max \left(\left\| \begin{bmatrix} W_1 \\ W_2 K_n \\ W_3 P K_n \end{bmatrix} [I - PK_n]^{-1} \right\|_{\mathcal{H}^\infty}, \left\| [W_5 P] [I - K_n P]^{-1} \right\|_{\mathcal{H}^\infty} \right) < \gamma \right\}$$

$W_1 - W_3, W_4, W_2^{-1}, \in R\mathcal{H}^\infty$

Equation 6.3 can be visualized for the phase locked loop design problem by the block diagram in figure 6.2:

The closed loop transfer functions associated with 6.2

$$T_{referencefrequency \to synthfrequency} = PK[1 + PK]^{-1} \tag{6.3}$$

$$T_{referencefrequency \to phaseerror} = [1 + PK]^{-1} \tag{6.4}$$

$$T_{referencefrequency \to drivingfrequency} = K[1 + PK]^{-1} \tag{6.5}$$

$$T_{VCOinputdisturbance \to drivingfrequency} = P[1 + PK]^{-1} \tag{6.6}$$

these closed loop transfer functions should be shaped with weights W_1, W_2, W_3, W_5 such that

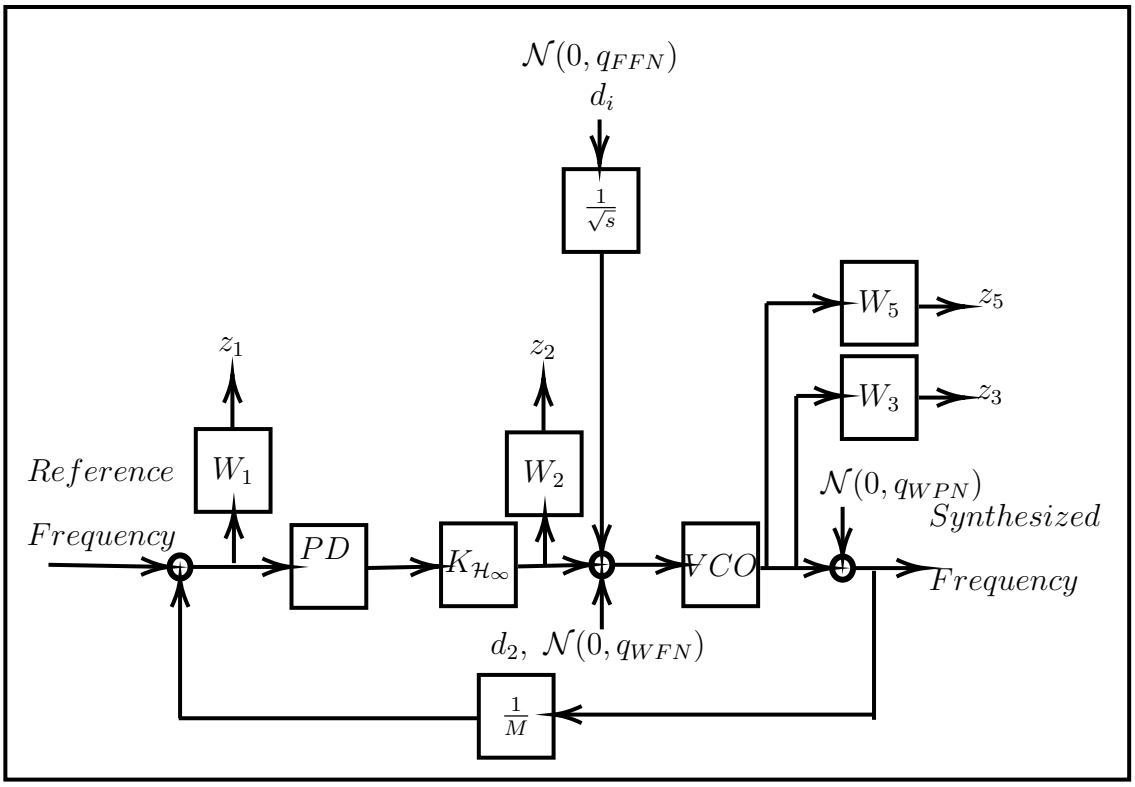

Figure 6.2: $\mathcal{H}^\infty$ Framework Applied to Phase Locked Loop Design

- $T_{referencefrequency \rightarrow phaseerror}$ should be small at low frequencies for good low frequency master oscillator command following and disturbance attenuation

- $T_{referencefrequency \rightarrow synthfrequency}$ should be small at high frequencies for good high frequency master oscillator noise attenuation

- $T_{referencefrequency \rightarrow synthfrequency}$ should be not too large at low frequencies, in order for the closed loop system to be robust multiplicative modeling errors at the plant input and output

- $T_{VCOinputdisturbance \rightarrow drivingfrequency}$ should be small for all frequency for VCO input disturbance attenuation

Moreover, the system $\mathcal{H}^\infty$ norm is an indicator of closed-loop robustness. Let $S = T_{referencefrequency \to phaseerror}$ and $T = T_{referencefrequency \to synthfrequency}$ be system sensitivity and complementary sensitivity functions, respectively. There is a straight forward relationship between gain and phase margins and closed loop $\mathcal{H}^\infty$ norms [21]

$$GM \geq 1 + \frac{1}{\|T\|_{\mathcal{H}^\infty}} \tag{6.7}$$

$$GM \geq 1 + \frac{1}{1 - \frac{1}{\|S\|_{\mathcal{H}^\infty}}} \tag{6.8}$$

$$gm \leq 1 - \frac{1}{\|S\|_{\mathcal{H}^\infty} + 1} \tag{6.9}$$

$$gm \leq 1 - \frac{1}{\|T\|_{\mathcal{H}^\infty}} \tag{6.10}$$

$$PM \geq \arcsin\left(\frac{1}{2\|T\|_{\mathcal{H}^\infty}}\right) \tag{6.11}$$

$$PM \geq \arcsin\left(\frac{1}{2\|S\|_{\mathcal{H}^\infty}}\right) \tag{6.12}$$

and hence

$$GM > \max\left\{\frac{\|S\|_{\mathcal{H}^\infty}}{\|S\|_{\mathcal{H}^\infty} - 1}, \frac{\|T\|_{\mathcal{H}^\infty} + 1}{\|T\|_{\mathcal{H}^\infty}}\right\} \tag{6.13}$$

$$gm < \min\left\{\frac{\|S\|_{\mathcal{H}^\infty}}{\|S\|_{\mathcal{H}^\infty} + 1}, \frac{\|T\|_{\mathcal{H}^\infty} - 1}{\|T\|_{\mathcal{H}^\infty}}\right\} \tag{6.14}$$

$$|PM| > 2\max\left\{\arcsin\left(\frac{1}{2\|T\|_{\mathcal{H}^\infty}}\right), \arcsin\left(\frac{1}{2\|S\|_{\mathcal{H}^\infty}}\right)\right\} \tag{6.15}$$

$$\|T_{referencefrequency \to phaseerror}\|_{\mathcal{H}^\infty} > \max\left\{\frac{GM}{GM - 1}, \frac{gm}{1 - gm}, \frac{1}{2\sin\frac{PM}{2}}\right\} \tag{6.16}$$

$$\|T_{referencefrequency \to synthfrequency}\|_{\mathcal{H}^\infty} > \max\left\{\frac{1}{GM - 1}, \frac{1}{1 - gm}, \frac{1}{2\sin\frac{PM}{2}}\right\} \tag{6.17}$$

which implies that if $\|S\|_{\mathcal{H}^\infty}$ and $\|T\|_{\mathcal{H}^\infty}$ are small, gain and phase margins will be large.

6.3 Fixed Order $\mathcal{H}^\infty$ Control

While $\mathcal{H}^\infty$ methods makes loopshaping significantly more straight forward, it has practical issues. The order of the resulting controller is, at a minimum, the size of the design plant, plus the size of the weighting function matrices. This results in a controller that is not easy to implement, and in some cases, impossible to implement. This issue motivated the development of a Matlab based tool known as HINFSTRUCT [12, 4, 3]. That is, to solve the Weighted $\mathcal{H}^\infty$ control problem subject to the constraint that the resulting controller is that of a fixed order topology. The most commonly used compensators in practice are PI and PID controllers. Both of which are easily realizes in operational amplifiers on the hardware end, provided the bandwidth is small enough. Moreover, PI and PID controllers are easy to implement on an embedded processor.

$$\tilde{\mu}(\gamma) \stackrel{\text{def}}{=} \inf_{\substack{K\ fixed order \\ poles(T_{wz})<\lambda}} \left\{ \max \left(\left\| \begin{bmatrix} W_1 \\ W_2 K \\ W_3 PK \end{bmatrix} [I - PK]^{-1} \right\|_{\mathcal{H}^\infty}, \left\| [W_5 P] [I - KP]^{-1} \right\|_{\mathcal{H}^\infty} \right) < \gamma \right\}$$

Note that this is a non-differentiable cost function, which does not allow for the guarantee of convergence to a global minima [12, 4, 3]. However, many papers have demonstrated that the tool practical solutions with the desired loopshape [12, 4, 3]. Similarly, this problem can be visualized by the block diagram shown in figure 6.3.

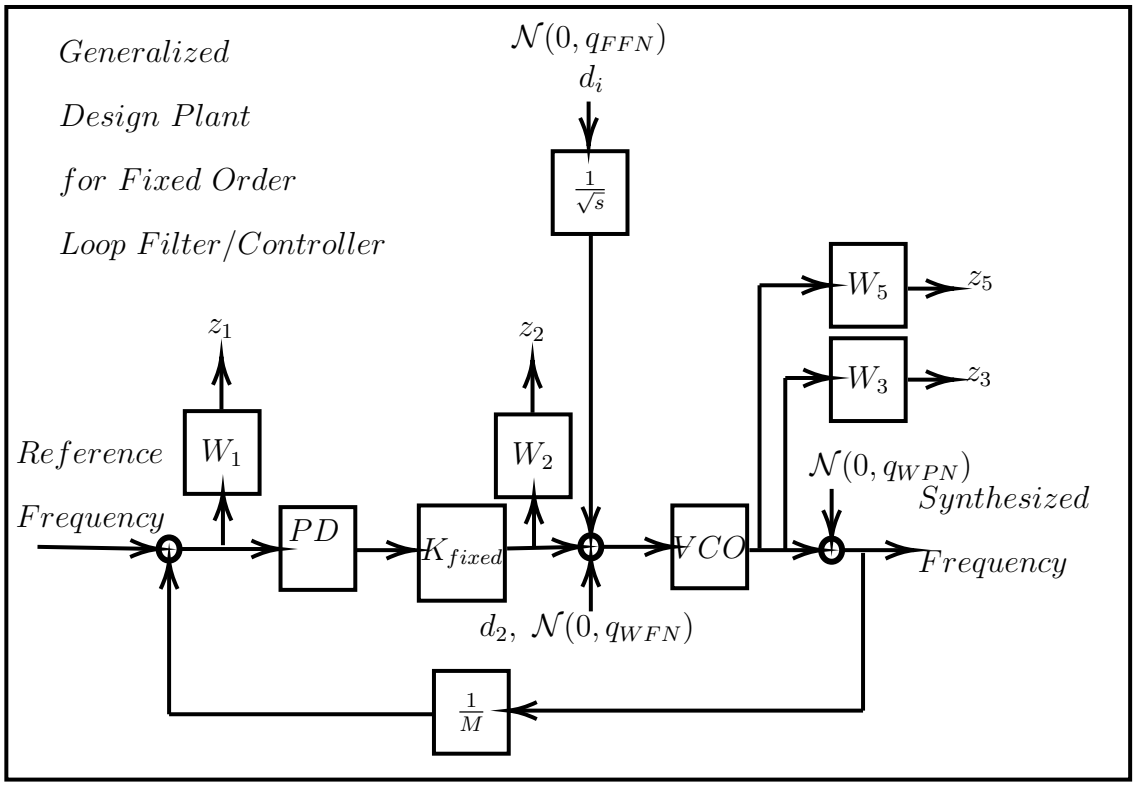

Figure 6.3: Fixed Order $\mathcal{H}^\infty$ Design Framework Applied to Phase Locked Loops

6.4 Designing Fixed Order Compensators for Phase and Frequency Dynamics

Since the plant, a VCO contains an integrator, only standard PI controller, with approximation roll-off terms needs to be used to track phase and frequency dynamics.

$$K_{PI}(s) = \frac{k_p(s + \frac{k_i}{k_p})}{s} \left(\frac{a_1}{s + a_2}\right) \left(\frac{b_1}{s + b_2}\right) \left(\frac{c_1}{s + c_2}\right) \tag{6.18}$$

This is a fixed order $\mathcal{H}^\infty$ problem as seen in equation 6.19

$$\tilde{\mu}(\gamma) \stackrel{\text{def}}{=} \inf_{\substack{k_p,k_i,a_1,a_2,b_1,b_2,c_1,c_2 \\ poles(T_{wz})<\lambda}} \left\{ \max \left(\left\| \begin{bmatrix} W_1 \\ W_2 K \\ W_3 PK \end{bmatrix} [I - PK]^{-1} \right\|_{\mathcal{H}^\infty}, \left\| [W_5 P] [I - KP]^{-1} \right\|_{\mathcal{H}^\infty} \right) < \gamma \right\}$$

such a controller, can track phase steps and frequency impulses with zero steady state error. Moreover, since the plant, a voltage or numerically controlled oscillator contains an integrator as well, the feedback system can additionally track frequency steps, which result in a phase ramp. In order to compensate even higher dynamics, such as frequency ramps, or frequency accelerations, the following topology is also explored

$$K_{PI2}(s) = \frac{k_p(s^2 + \frac{k_i}{k_p}s + \frac{k_{i2}}{k_p})}{s^2} \left(\frac{a_1}{s + a_2}\right) \left(\frac{b_1}{s + b_2}\right) \left(\frac{c_1}{s + c_2}\right) \tag{6.19}$$

This can be viewed as a fixed-order $\mathcal{H}^\infty$ problem as seen in equation 6.20

$$\tilde{\mu}(\gamma) \stackrel{\text{def}}{=} \inf_{\substack{k_p,k_i,k_i2,a_1,a_2,b_1,b_2,c_1,c_2 \\ poles(T_{wz})<\lambda}} \left\{ \max \left(\left\| \begin{bmatrix} W_1 \\ W_2 K \\ W_3 PK \end{bmatrix} [I - PK]^{-1} \right\|_{\mathcal{H}^\infty}, \left\| [W_5 P] [I - KP]^{-1} \right\|_{\mathcal{H}^\infty} \right) < \gamma \right\}$$

Table 6.1: HINFSTRUCT Optimization Options.

Parameter	**Value**
Max Iterations	100000
Target μ	1
$\max\{poles(T_{wz})\}$	-0.1
physical+logical CPU cores	4+4
random starts	5
objective function tolerance	1e-20

The parameters used in the Matlab solver, HINFSTRUCT, can be seen in table 6.1. Note that the closed loop pole specification dictates that the poles of the weighting functions should be smaller than -0.1, or the optimizer can run into issues, as the weights appear to be unstable The weights used are the following

$$W_1(s) = \frac{0.56234(s+17.78)}{(s+0.5)} \tag{6.20}$$

$$W_2(s) = \frac{31623(s+10)}{(s+3.162e6)} \tag{6.21}$$

$$W_3(s) = \frac{31623(s+5.623)}{(s+3.162e05)} \tag{6.22}$$

$$W_5(s) = \frac{1}{db2mag(5)} \tag{6.23}$$

The achieved closed loop properties can be seen in 6.2. , with gains as seen in table 6.3. The double integrator design results are summarized in tables 6.4 and 6.5. The simulated clock model stochastic parameters are shown in table 6.6 with the dynamic (deterministic) scenarios shown in table 6.7. The closed loop frequency responses can be seen figures 6.4 and the time responses in figures 6.10 to 6.13. The frequency response, time responses and closed-loop margins combine to show that both closed-loop systems achieve approximately the same performance and robustness margins.

The key difference is the double integrator PI controller can track higher dynamics with 0 steady state error than the single PI controller. To summarize, this section has provided a systematic strategy, motivated by modern control theory, which allows for the design and management of many different closed loop properties, as well as the ability to follow commands that are highly dynamic.

Table 6.2: Single Integrator PI Controller Design Specs.

Parameter	**Value**
GM	2.65e5
gm	0.005
PM	86.43 degrees
DM	0.151 seconds
ω_{GM}	3.56 kHz
ω_{gm}	0.002 Hz
ω_{PM}	1.6 Hz
$\max\{poles(T_{wz})\}$	-0.1

Table 6.3: Single Integrator PI Controller Gains From HINFSTRUCT.

Parameter	**Value**
k_p	4.266306068432225e+06
k_i	4.223284559870226e+05
a_1	6.201564333818265e+04
a_2	3.608695937441208e+07
b_1	7.325286467647968e+03
b_2	1.891909378446152e+02
c_1	45.640877186868032
c_2	2.053431677428632e+07

Table 6.4: Single Integrator PI Controller Design Specs.

Parameter	**Value**
GM	2.65e+05
gm	0.005
PM	85.88 degrees
DM	0.151 seconds
ω_{GM}	3.56 kHz
ω_{gm}	0.02 Hz
ω_{PM}	1.58 Hz
$\max\{poles(T_{wz})\}$	-0.1

Table 6.5: Double Integrator PI Controller Gains From HINFSTRUCT.

Parameter	**Value**
k_p	1.654894611166640e+05
k_i	3.259891031845763e+04
k_{i2}	1.621633308664711e+03
a_1	5.405913891053951e+05
a_2	1.904123754732969e+02
b_1	7.280182570597557e+02
b_2	3.533631082601804e+07
c_1	1.168092551085963e+03
c_2	1.786456241031897e+07

Table 6.6: Simulated Clock Noise Parameters.

Constituent	**Value**
WFN	2.6153e-04 Hz
WPN	0.0048 Cycle
FFN	0.0012 Hz/s
RWFN	0 Hz/s
RRFN	0 Hz/s/s

Table 6.7: Simulated Clock Dynamics.

Moderate Dynamics		
Case	**Magnitude**	**Event Time**
Frequency Step	1e-7 Hz	100 seconds
Phase Step	2.513274122871835 Cycle	0 seconds
High Dynamics		
Case	**Magnitude**	**Event Time**
Frequency Step	1e-7 Hz	100 seconds
Phase Step	2.513274122871835 Cycle	0 seconds
Frequency Velocity Step	1e-8 Hz/s	500 seconds

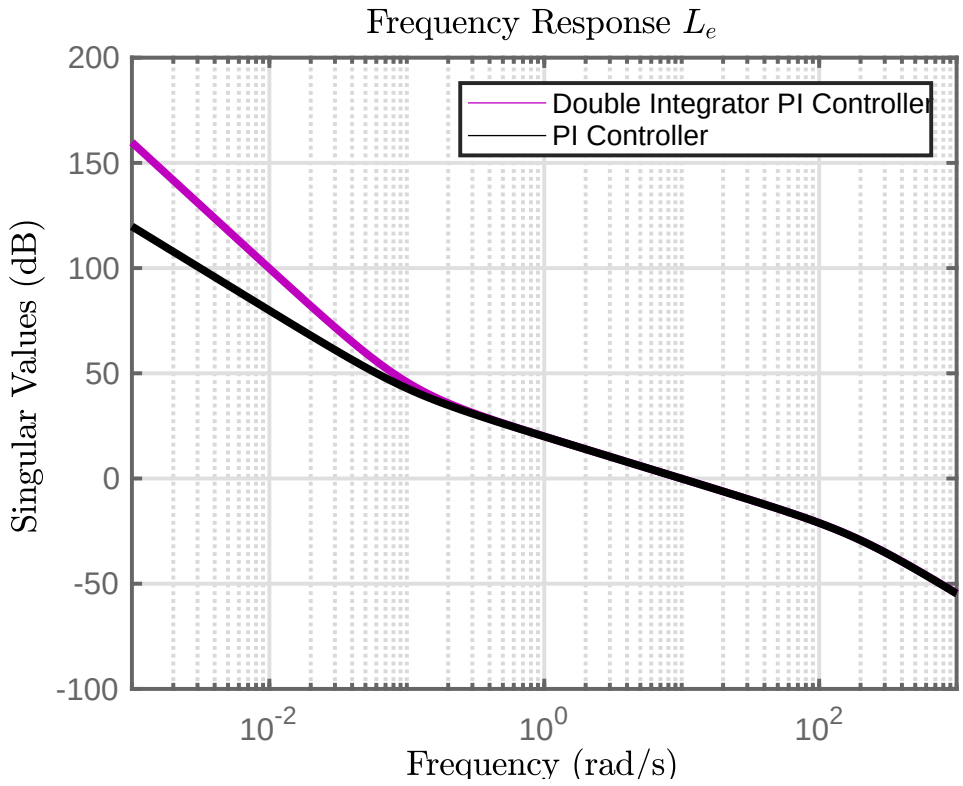

Figure 6.4: Open Loop Transfer Function Frequency Response $L = PK$. Both controllers crossover at roughly 1.5Hz. As expected, the PI controller with 2 integrators has more integral action which is observed at low frequencies. For $\omega > 0.01$, open-loop responses are identical.

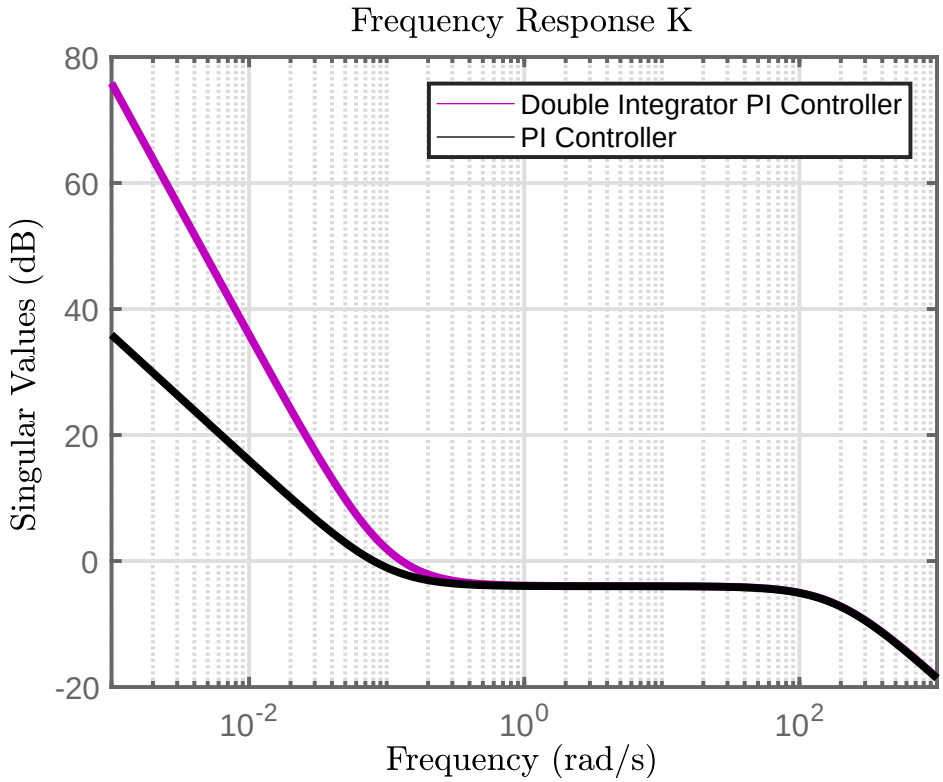

Figure 6.5: Controller Frequency Response. Similar behavior as observed in the open-loop gain $L = PK$, the double integrator PI has more integral action, and controllers are identical for $\omega > 0.01$.

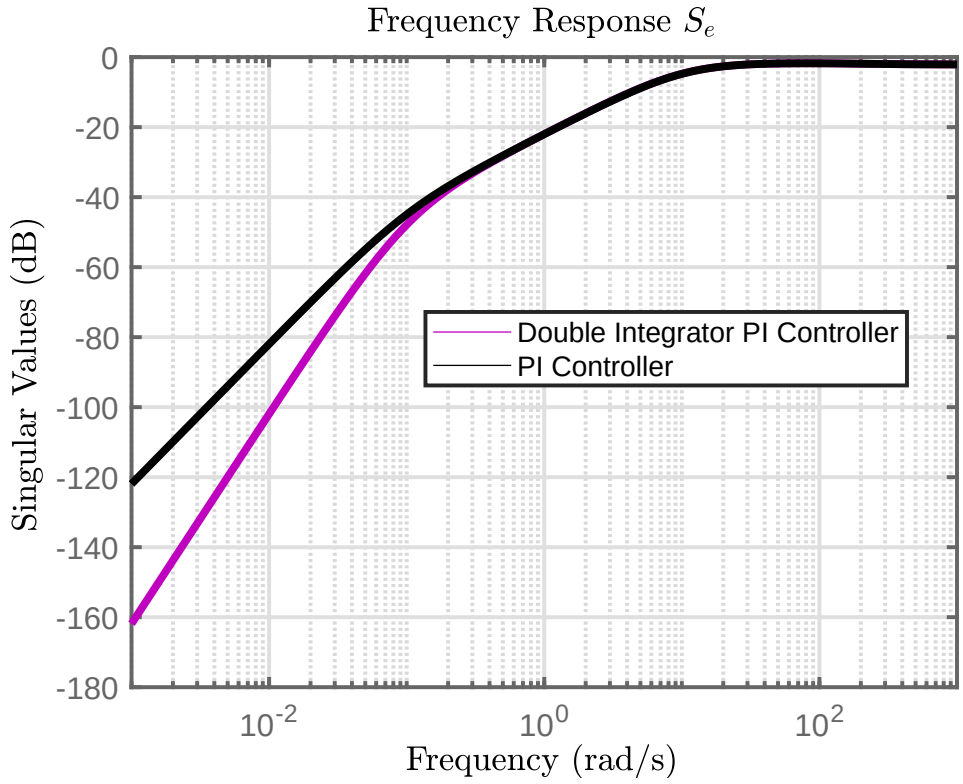

Figure 6.6: Sensitivity Frequency Response given by $T_{referencefrequency \to phaseerror} = [1 + PK]^{-1} = S_e$. A reasonable way to quantify the sensitivity function bandwidth is when it reaches -20dB. By this metric and about increasing frequency, the closed loop performance is the same for both compensators. Since the double integrator PI controller attenuates greater at low frequencies, the closed loop system will have better command following, dynamic response and smaller phase error variance. Moreover, the peak of this response being less than 0dB implies that gain and phase margins will be desirable.

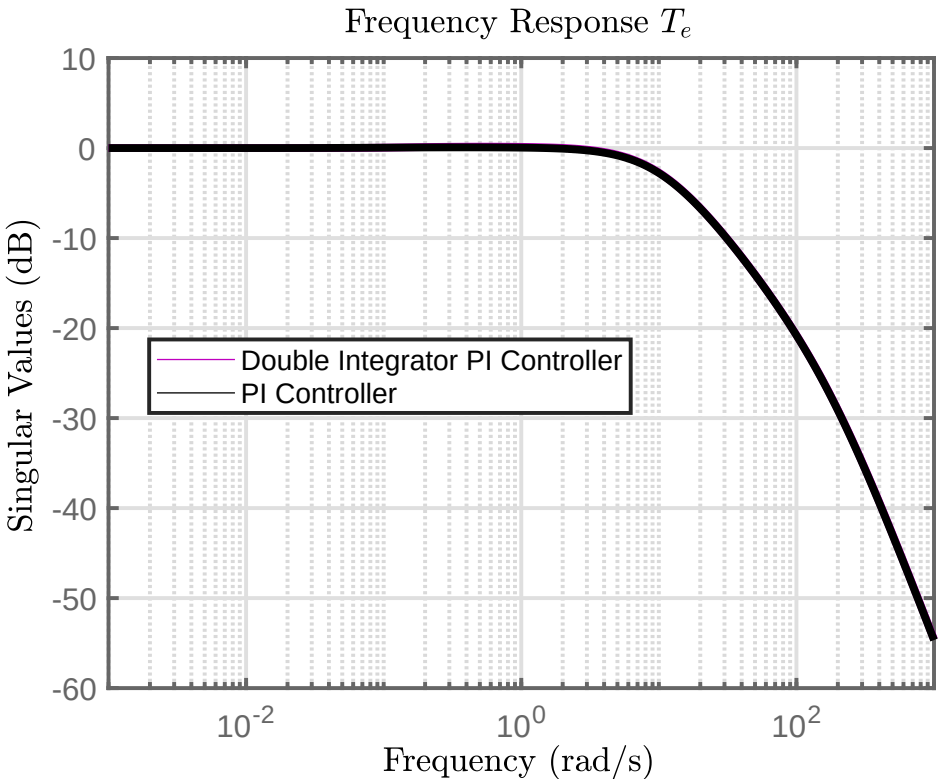

Figure 6.7: Complementary Sensitivity Frequency Response given by $T_{referencefrequency \to synthfrequency} = PK[1+PK]^{-1} = T_e$. Both controllers achieve the same frequency response. The -20 dB frequency of roughly 100 rad/s implies that reference frequency noise above 100rad/s will be attenuated. The peak of 0 dB implies desirable gain and phase margins.

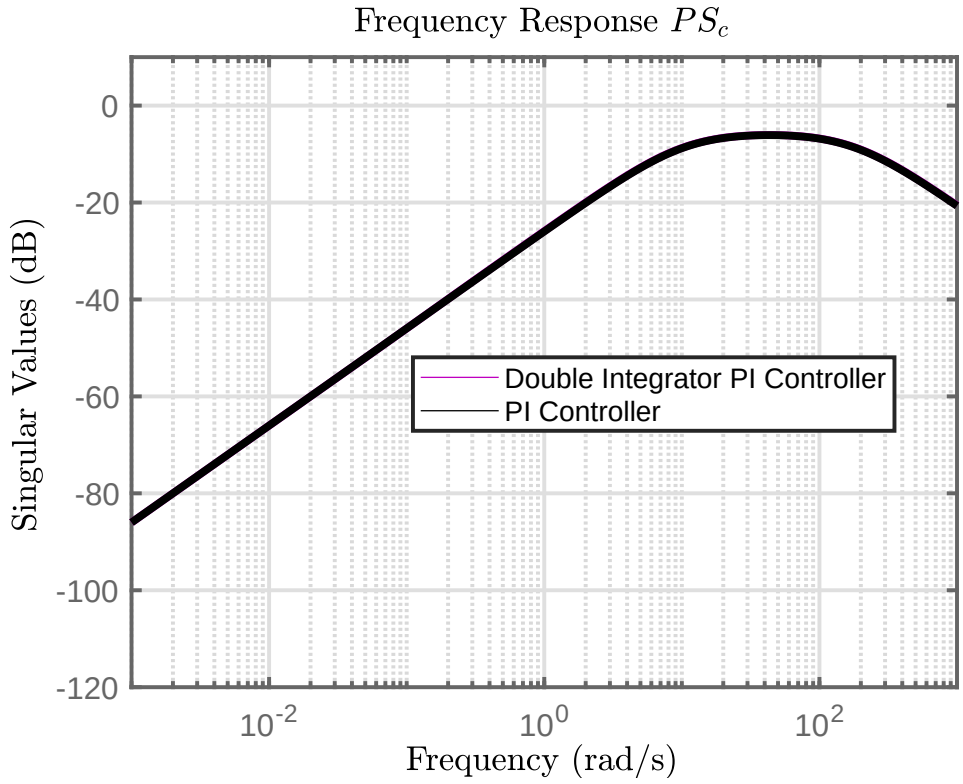

Figure 6.8: Input Disturbance Sensitivity Frequency Response given by $T_{VCOinputdisturbance \to drivingfrequency} = P[1+PK]^{-1} = PS_e$. It's desired that this closed loop filter attenuates for all frequency, and that is achieved, with a peak of -6dB. Additionally, this filter attenuates more than -20 dB everywhere, aside from a band in $[3, 400]$rad/s. It should be noted that putting a weight on this transfer function is the entire reason the standard $\mathcal{H}^\infty$ mixed sensitivity objective couldn't be used. The additional weight of W_5 allows to shape this closed loop transfer function so that it attenuates for all frequency.

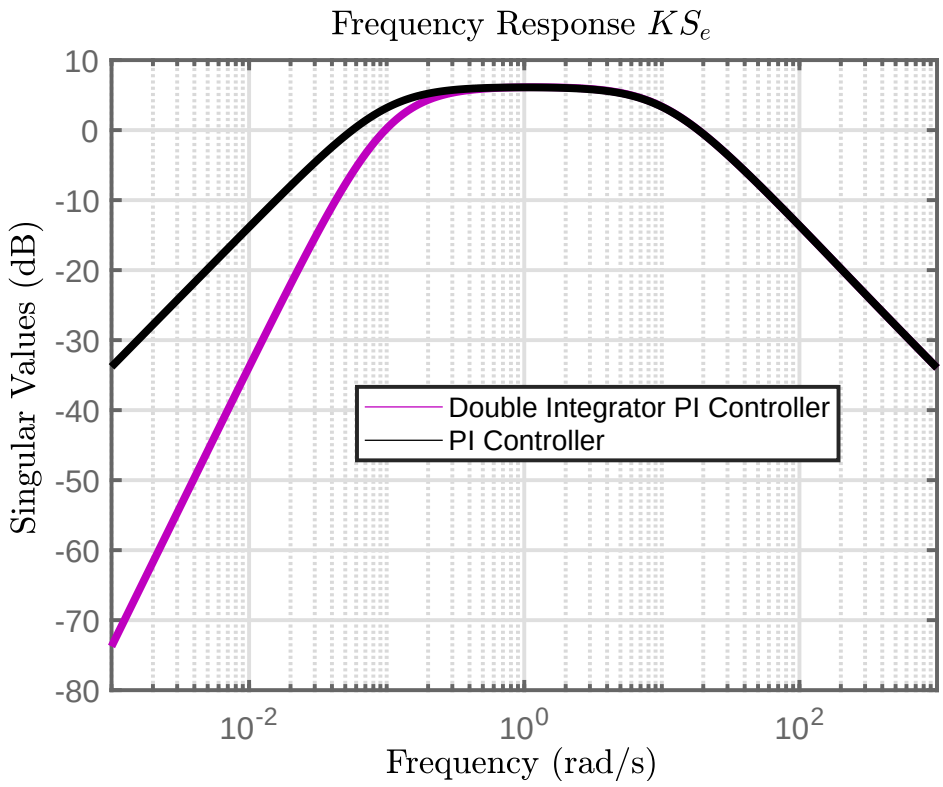

Figure 6.9: Control Action Frequency Response given by $T_{reference frequency \to driving frequency} = K[1 + PK]^{-1} = KS_e$. This response is typically called the control action. It often has a band-pass shape, and having a small peak is desirable. A large peak can result in pass-band amplification of the reference signal that results in unacceptably large control signals, which can cause saturation, and/or hurt linearity assumptions, knocking the controller and system away from the linear operating point.

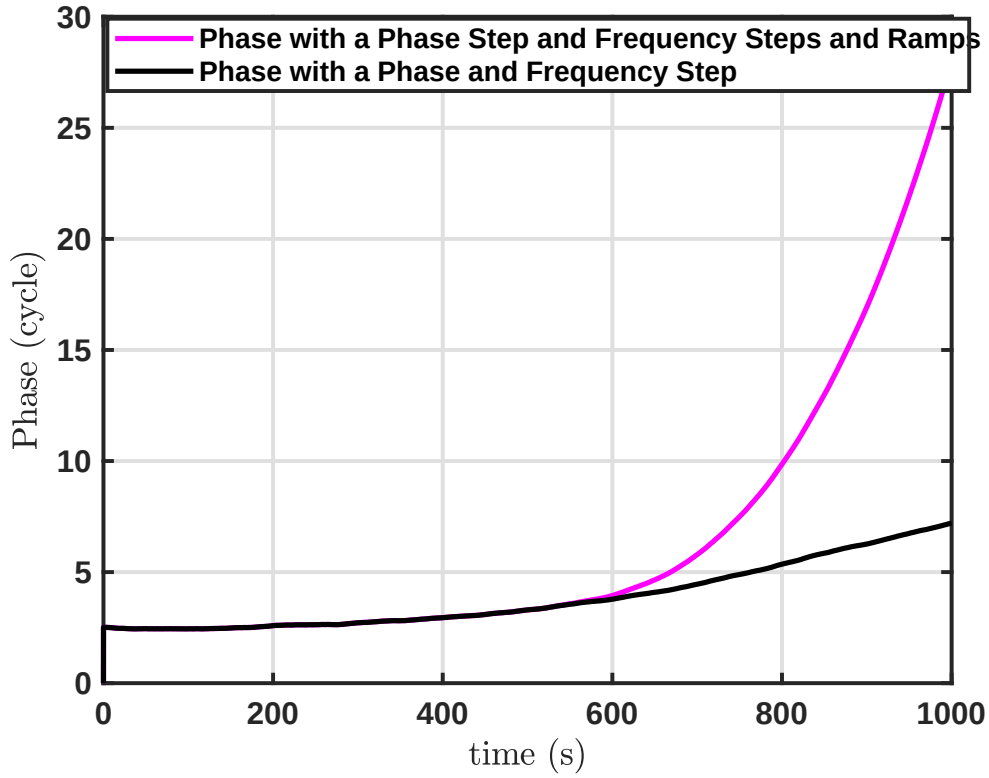

Figure 6.10: Received Phase Signal Entering Tracking Loop. Reference phases were used rather than reference frequencies to allow for linear simulations. Reference frequencies require a non-linear operation to convert frequency to phase. It is commonplace to use linear thinking to design controllers for phase locked loops and tracking loops. Each scenario, both moderate and high dynamics were designed to use the highest dynamics possible with respect to the controller, such that the closed loop system could achieve steady state error, with only residual tracking variance remaining. Thus, phase and frequency steps (phase ramps) were used for the single integrator PI controller scenario. Since another integrator is present in the VCO (the plant), the closed loop system can track phase and frequency steps. Similarly, a frequency ramp (phase quadratic) was also injected into the double integrator PI controller driven closed loop system.

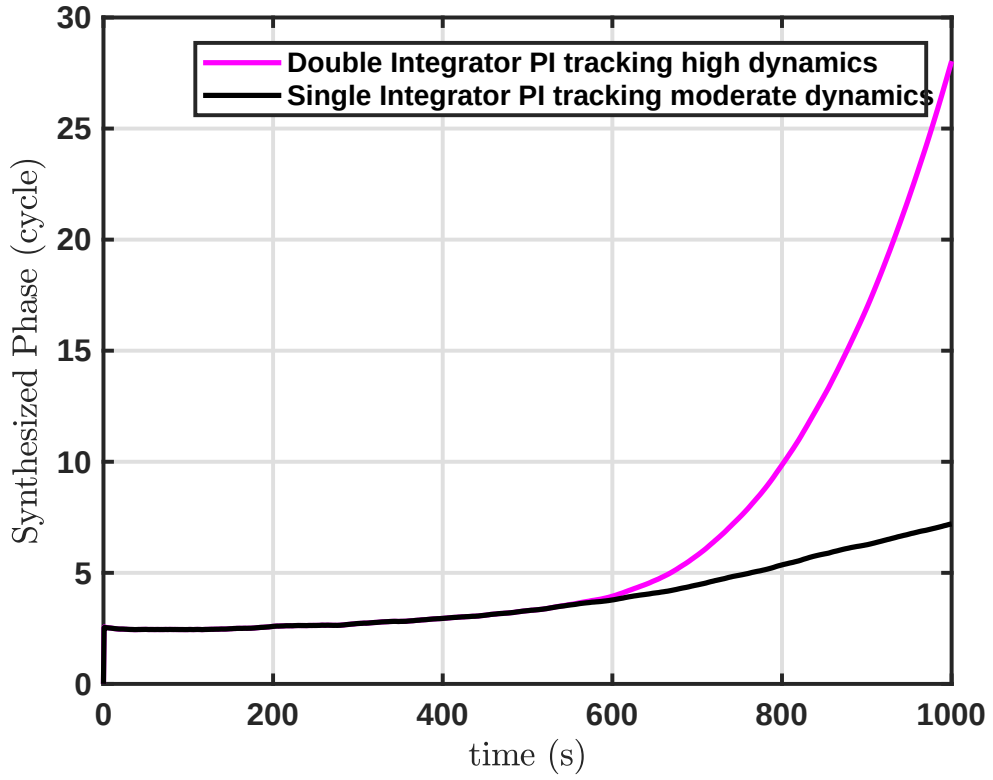

Figure 6.11: Synthesized Phase Time Response. It's observed that each controller is able to track the associated dynamics with 0 steady state error.

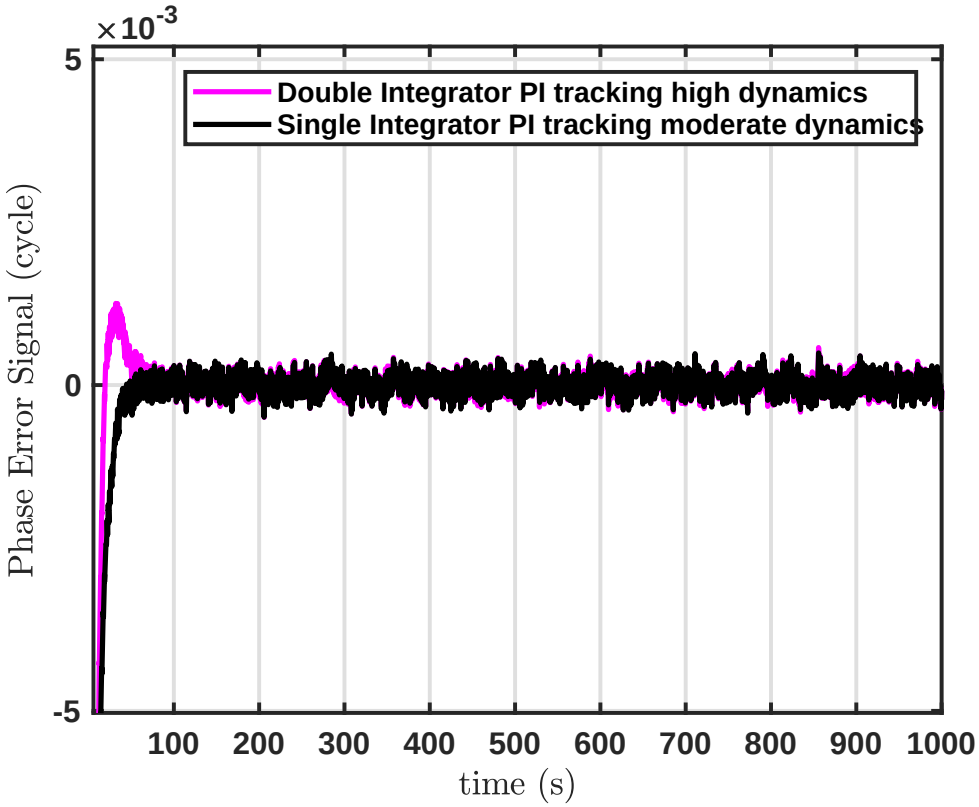

Figure 6.12: Phase Error Time Response. It's observed that each controller is able to track the associated dynamics with 0 steady state error and absolute tracking error on the order of 0.5e-3 cycles (2 picoseconds when converted to time).

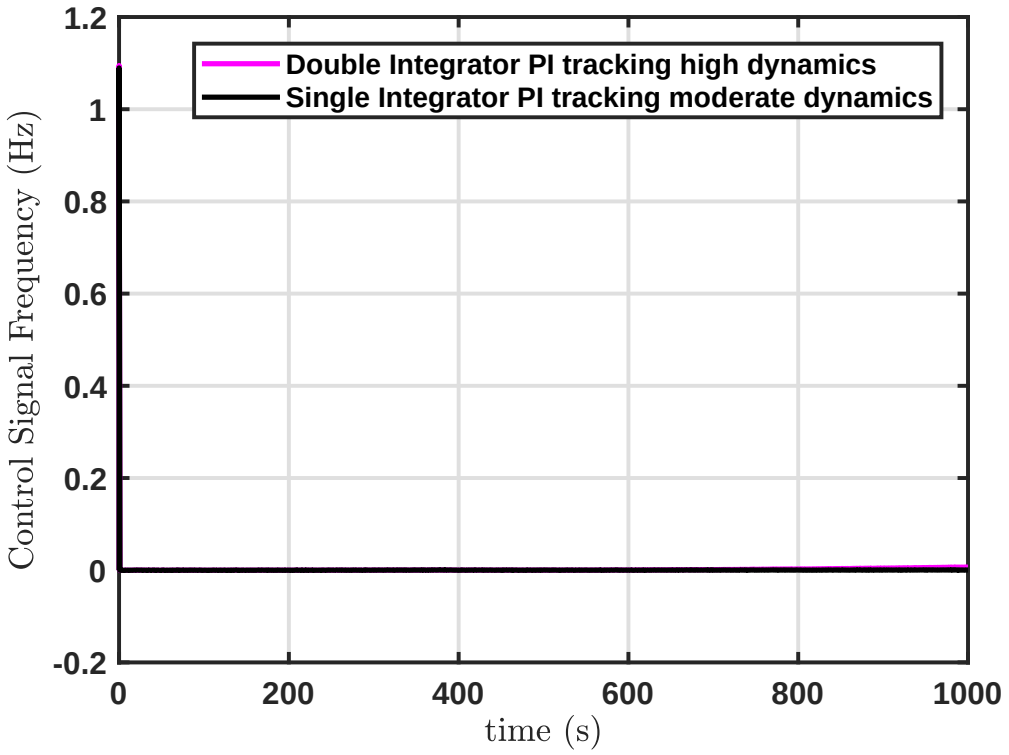

Figure 6.13: Control Signal Time Response. The control signal quickly hits 1 Hz to compensate for dynamics, then decays to a small number, indicative of the tracking variance.

6.5 Higher Order Problems: $\mathcal{H}^\infty$ Model Matching Problem

Another approach to solve the fixed order $\mathcal{H}^\infty$ problem is to solve a model matching problem. A strategy for this approach would work is the following

- 1.) Select weights $W_1 - W_3$, W_5 and solve problem 6.3, which is the Generalized $\mathcal{H}^\infty$ problem, resulting in a high order controller, $K_{\mathcal{H}^\infty}$
- 2.) Select a Fixed Order Controller that is suitable for implementation in your application, K_fixed
- 3.) Minimize the difference between the target closed loop transfer functions obtained in problem 6.3 and the fixed order closed loop transfer functions, subject to K_{fixed}

Here are some scenarios where such an approach will be valuable

- Case I: Notch filters must be present in K to deal with spurious oscillator modes
- Case II: FFN dynamics are significant enough to be included in the design process
- Case III: Very demanding attenuation specs in $T_{VCOinputdisturbance \to drivingfrequency} = P[1 + PK]^{-1} = PS_e$ must be achieved, perhaps where a notch filter is also present in K

In Case II, the problem is an infinite dimensional control problem. The approximate-then-design approach is a natural way to solve this problem, as it is consistent with standard engineering practice. The key difficulty in this problem is adequately modeling $\frac{1}{\sqrt{s}}$. This approach is used by solving a sequence of control problems such as that of figure 6.3, about increasing order of approximation (n). For each problem in

the sequence, you will get a controller K_n. Once the order of approximation (n) necessary for the application is determined, then the resulting controller can be plugged into the problem defined in figure 6.14, then a resulting fixed order controller K_{fixed} can be found. This can be visualized in block diagram form in figure 6.14.

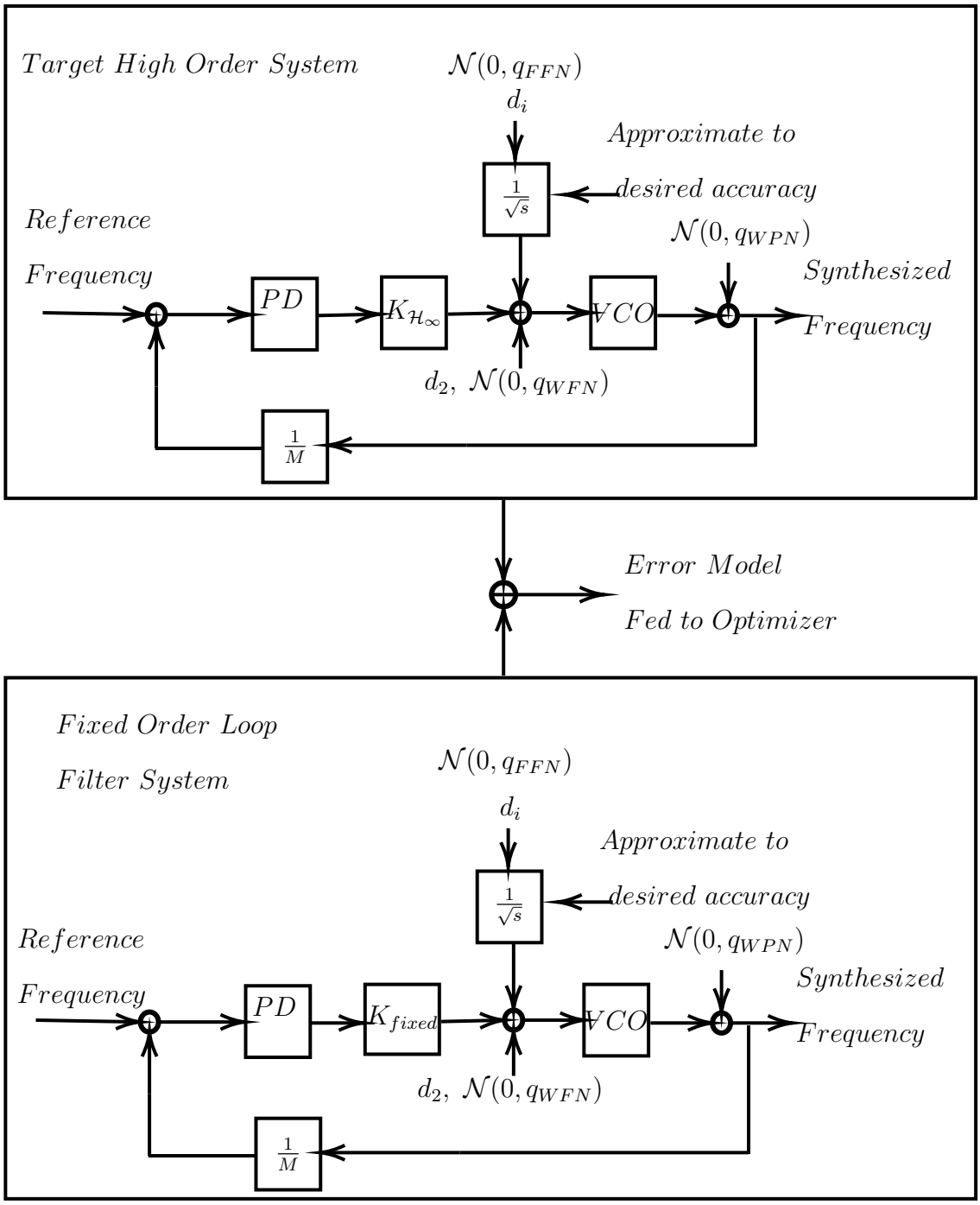

Figure 6.14: $\mathcal{H}^\infty$ Model Matching Problem

Chapter 7

SUMMARY AND FUTURE WORK

7.1 Summary

In conclusion, the work in this writeup has presented the standard methods for modeling oscillators and atomic clocks. An overview of the standard methods for modeling and filtering colored noise was presented, as well as the development of a new method with takes the Oustaloup fractional calculus approximation technique and converts it into a stochastic differential equation. It's shown that 6 states are needed when using this model to estimate colored noise associated with oscillators. The question of when and if colored noise needs to be modeled is addressed. It's shown that when an oscillators phase is measured directly, and if the FFN constituent is significant, order of magnitude increases in timing accuracy can achieved after Kalman Filter corrections are applied. Additionally, it is shown how to integrate the Oustaloup model into a realistic, non-linear navigation filter in the HPPC framework. It's observed that adaptive filtering techniques (IAE), can be used to compensate timing degradation caused by FFN, even if colored noise states are not embedded in the EKF. However, if the FFN states, and the IAE method is used, it's possible to see an order of magnitude increase in estimation accuracy. Thus, for high precision, high reliability navigation systems with oscillators that exhibit a FFN constituent, it may be necessary to both model the FFN states, and use the IAE method for adaptive process and measurement noise covariance estimation. A systematic approach for generating realistic time-offsets is proposed and applied the HPPC navigation system. In order to achieve even more robustness to the stochastic inherent to oscillator timing

degradation, novel banks of adaptive Extended Kalman Filters are used to estimate time-offsets in the HPPC navigation system non-linear timing protocol. Last, novel $\mathcal{H}^{\infty}$ methods are proposed to shape the closed loop transfer functions of phase locked loops in sample clocks, and tracking loops in demodulators, to combat high dynamic scenarios (such as ranging or clock dynamics) not currently addressed in the literature.

7.2 Future Work

The HPPC system continues to be an ongoing research project. The work herein serves as a theoretically and numerically sound starting point for design of control and estimation algorithms, that can later be further developed and implemented on hardware.

- Numerical Results for appending colored noise states to adaptive filter bank framework in the HPPC system.
- Extending the filter results to all four antennas, current work focusing on the architecture for a single antenna.
- Extending the filter results to a network of HPPC users.
- Exploring other filter methods such as UKF, MHE and Particle Filters for the HPPC system. The last 20 years has seen a renaissance in alternatives to Kalman Filters and Extended Kalman Filters. These alternative optimal estimators posses theoretical benefits for non-linear systems with non-Gaussian noise. As all navigation filtering problems are inherently non-linear, such estimators may provide better performance and robustness to the EKF.
- Using a rigorous formulation to show in what sense the Oustaloup approximates converge to $\frac{1}{\sqrt{s}}$ in the frequency domain.

- Implementing the proposed filters on an embedded processor and testing it on a rotor aircraft.

REFERENCES

[1] Akhlaghi, S., N. Zhou and Z. Huang, "Adaptive adjustment of noise covariance in kalman filter for dynamic state estimation", in "2017 IEEE Power Energy Society General Meeting", vol. 2018-, pp. 1–5 (IEEE, 2017).

[2] Allan, D. W., "Statistics of atomic frequency standards", Proceedings of the IEEE **54**, 2, 221–230 (1966).

[3] Apkarian, P., M. N. Dao and D. Noll, "Parametric robust structured control design", IEEE Transactions on Automatic Control **60**, 7, 1857–1869 (2015).

[4] Apkarian, P. and D. Noll, "Nonsmooth h∞ synthesis", IEEE Transactions on Automatic Control **51**, 1, 71–86 (2006).

[5] Athans, M. and C. B. Chang, "Adaptive estimation and parameter identification using multiple model estimation algorithm", Tech. rep., Massachusetts Institute of Technology, Lincoln Laboratory, US, Discussion paper (1976).

[6] Baugh, R. A., "Frequency modulation analysis with the hadamard variance", in "25th Annual Symposium on Frequency Control", pp. 222–225 (IEEE, 1971).

[7] Cao, H., S. Gangakhedkar, A. R. Ali, M. Gharba and J. Eichinger, "A 5g v2x testbed for cooperative automated driving", in "2016 IEEE Vehicular Networking Conference (VNC)", pp. 1–4 (IEEE, 2016).

[8] Chaffee, J. W., "Relating the allan variance to the diffusion coefficients of a linear stochastic differential equation model for precision oscillators", IEEE Transactions on Ultrasonics Ferroelectrics and Frequency Control **34**, 655–658 (1987).

[9] Davis, J., C. Greenhall and P. Stacey, "A kalman filter clock algorithm for use in the presence of flicker frequency modulation noise", Metrologia pp. 1–10, URL `http://search.proquest.com/docview/230092156/` (2005).

[10] El-Sheimy, N., H. Hou and X. Niu, "Analysis and modeling of inertial sensors using allan variance", IEEE Transactions on instrumentation and measurement **57**, 1, 140–149 (2007).

[11] Ford, J. J. and M. E. Evans, "Online estimation of allan variance parameters", Journal of guidance, control, and dynamics **23**, 6, 980–987 (2000).

[12] Gahinet, P. and P. Apkarian, "Decentralized and fixed-structure h∞ control in matlab", in "2011 50th IEEE Conference on Decision and Control and European Control Conference", pp. 8205–8210 (IEEE, 2011).

[13] Garner, F., "Phaselock techniques, hoboken", (2005).

[14] Grover Brown, R. B., "Kalman filter modeling", Tech. rep., Massachusetts Institute of Technology, Lincoln Laboratory, US, Discussion paper (1984).

[15] Groves, P. D., *Principles of GNSS, inertial, and multisensor integrated navigation systems* (Artech house, 2013).

[16] Herschfelt, A., H. Yu, S. Wu, H. Lee and D. W. Bliss, "Joint positioning-communications system design: Leveraging phase-accurate time-of-flight estimation and distributed coherence", in "2018 52nd Asilomar Conference on Signals, Systems, and Computers", pp. 433–437 (IEEE, 2018).

[17] Julier, S., "The scaled unscented transformation", vol. 6, pp. 4555–4559 (2002).

[18] Kalman, R. E. and R. S. Bucy, "New results in linear filtering and prediction theory", Journal of Basic Engineering **83**, 1, 95 (1961).

[19] Kaplan, E. and C. Hegarty, *Understanding GPS: principles and applications* (Artech house, 2005).

[20] Kasdin, N., "Discrete simulation of colored noise and stochastic processes and $1/f^{\alpha}$ power law noise generation", Proceedings of the IEEE **83**, 5, 802–827 (1995).

[21] Lehtomaki, N., N. Sandell and M. Athans, "Robustness results in linear-quadratic gaussian based multivariable control designs", IEEE Transactions on Automatic Control **26**, 1, 75–93 (1981).

[22] Lesage, P. and T. Ayi, "Characterization of frequency stability: analysis of the modified allan variance and properties of its estimate", IEEE transactions on instrumentation and measurement **33**, 4, 332–336 (1984).

[23] Makdissi, A., F. Vernotte and E. D. Clercq, "Stability variances: a filter approach", IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control **57**, 5, 1011–1028 (2010).

[24] Malti, R., X. Moreau, F. Khemane and A. Oustaloup, "Stability and resonance conditions of elementary fractional transfer functions", Automatica **47**, 11, 2462–2467 (2011).

[25] Mehra, R., "On the identification of variances and adaptive kalman filtering", IEEE Transactions on Automatic Control **15**, 2, 175–184 (1970).

[26] Mehra, R., "Approaches to adaptive filtering", IEEE Transactions on Automatic Control **17**, 5, 693–698 (1972).

[27] Mohamed, A. H. and K. P. Schwarz, "Adaptive kalman filtering for ins/gps", Journal of Geodesy **73**, 4, 193–203 (1999).

[28] Puttannaiah, K., J. A. Echols, K. Mondal and A. A. Rodriguez, "Analysis and use of several generalized h-infinity mixed sensitivity frameworks for stable multivariable plants subject to simultaneous output and input loop breaking specifications", in "2015 54th IEEE Conference on Decision and Control (CDC)", pp. 6617–6622 (IEEE, 2015).

[29] Puttannaiah, K., J. A. Echols and A. A. Rodriguez, "A generalized h-infinity control design framework for stable multivariable plants subject to simultaneous output and input loop breaking specifications", in "2015 American Control Conference (ACC)", pp. 3310–3315 (IEEE, 2015).

[30] Radeka, V., "$1/|f|$ noise in physical measurements", IEEE Transactions on Nuclear Science **16**, 5, 17–35 (1969).

[31] Rutman, J. and F. Walls, "Characterization of frequency stability in precision frequency sources", Proceedings of the IEEE **79**, 7, 952–960 (1991).

[32] Soo Ahn, I., *State Modeling of Clock Noises and Its Application*, Ph.D. thesis, URL `http://login.ezproxy1.lib.asu.edu/login?url=https://search-proquest-com.ezproxy1.lib.asu.edu/docview/303498348?accountid=4485` (1986).

[33] Srinivas, S., "Tracking delay and offset estimates using kalman filtering", (2019).

[34] Stone, J. M., E. A. LeMaster, J. D. Powell and S. Rock, "Gps pseudolite transceivers and their applications", in "Proceedings of the 1999 Institute of Navigation National Technical Meeting, San Diego, CA", (Citeseer, 1999).

[35] Toyoizumi, H. and M. Genda, "Precise 1pps signal by gps", IEEJ Transactions on Electronics, Information and Systems **125**, 1217–1222 (2005).

[36] Weinbach, U. and S. Schön, "Gnss receiver clock modeling when using high-precision oscillators and its impact on ppp", Advances in Space Research **47**, 2, 229–238 (2011).

[37] Zucca, C. and P. Tavella, "The clock model and its relationship with the allan and related variances", IEEE transactions on ultrasonics, ferroelectrics, and frequency control **52**, 2, 289–296 (2005).

APPENDIX A

LIST OF ACRONYMS

AV Allan Variance
BPF Band Pass Filter
$\mathcal{H}^\infty$ H-Infinity Space
$R\mathcal{H}^\infty$ Real Rational in H-Infinity
IAE Innovations Based Adaptive Estimation
KF Kalman Filter
EKF Extended Kalman Filter
FFN Flicker Frequency Noise
FLL Frequency Locked Loop
LPF Low Pass Filter
MMAE Multiple Model Adaptive Estimation
NCO Numerically Controlled Oscillator
P Estimation Error Covariance Matrix
PD Phase Detector
PLL Phase Locked Loop
PI Proportional Integral Controller
PID Proportional Integral Differential Controller
Q Process Noise Covariance Matrix
R Measurement Noise Covariance Matrix
RWFN Random Walk Frequency Noise
RWFN Random Run Frequency Noise
WFN White Frequency Noise
WPN White Phase Noise
UKF Unscented Kalman Filter
VCO Voltage Controlled Oscillator

APPENDIX B

RAW DATA

B.1 Monte Carlo Results for HPPC Filter with Oscillator States only - Small Time and Frequency Offset (I)

This batch of simulations are described by tables B.1 and B.2 and equations B.3.

Table B.1: Simulated Noise Parameters.

Constituent	**Value**
WFN	2.6153e-04 Hz
WPN	0.0048 Cycle
FFN	0.0012 Hz/s
RWFN	4.7735e-07 Hz/s
RRFN	2.6153e-09 Hz/s/s

Table B.2: Simulated Noise Parameters.

Parameter	**Value**
measurement seed A	random integer $\in [0, 100]$
measurement seed B	random integer $\in [0, 100]$
process seed A	random integer $\in [0, 100]$
process seed B	random integer $\in [0, 100]$
fractional frequency A	random integer $\in [1, 50]$ times 1e-12
fractional frequency B	random integer $\in [0, 50]$ times 1e-12
time offset	random integer $\in [0, 1000]$ times 1e-11 s
fractional frequency offset	random value in [1e-8 1e-11]s/s

$$Q_o = diag([1e-9LA, 1e-3LA, 1e-30LA1_{1\times 6}]) \tag{B.1}$$
$$P_o = Q_o \tag{B.2}$$
$$R_o = diag([1e-11, 1e-11]) \tag{B.3}$$

Exactly 83 simulations were iterated with the parameters defined in tables 5.3 and 5.4. Additionally, $T_{adapt} = 0.5seconds$, $T_s = 10$ms, with convergence after roughly 10 seconds.

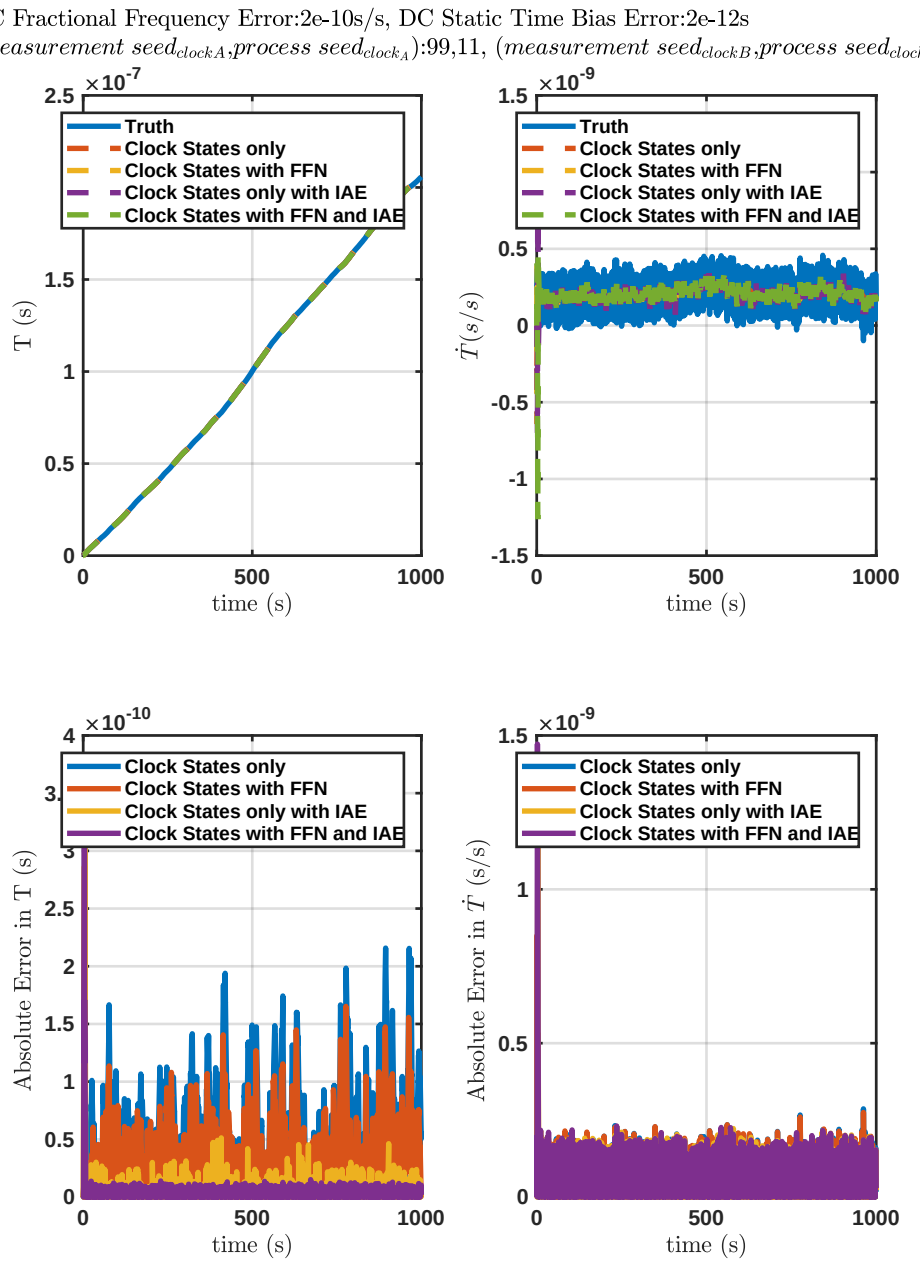

Figure B.1: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1.9e-09s/s, DC Static Time Bias Error:-1.9e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):11,74, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):96,48

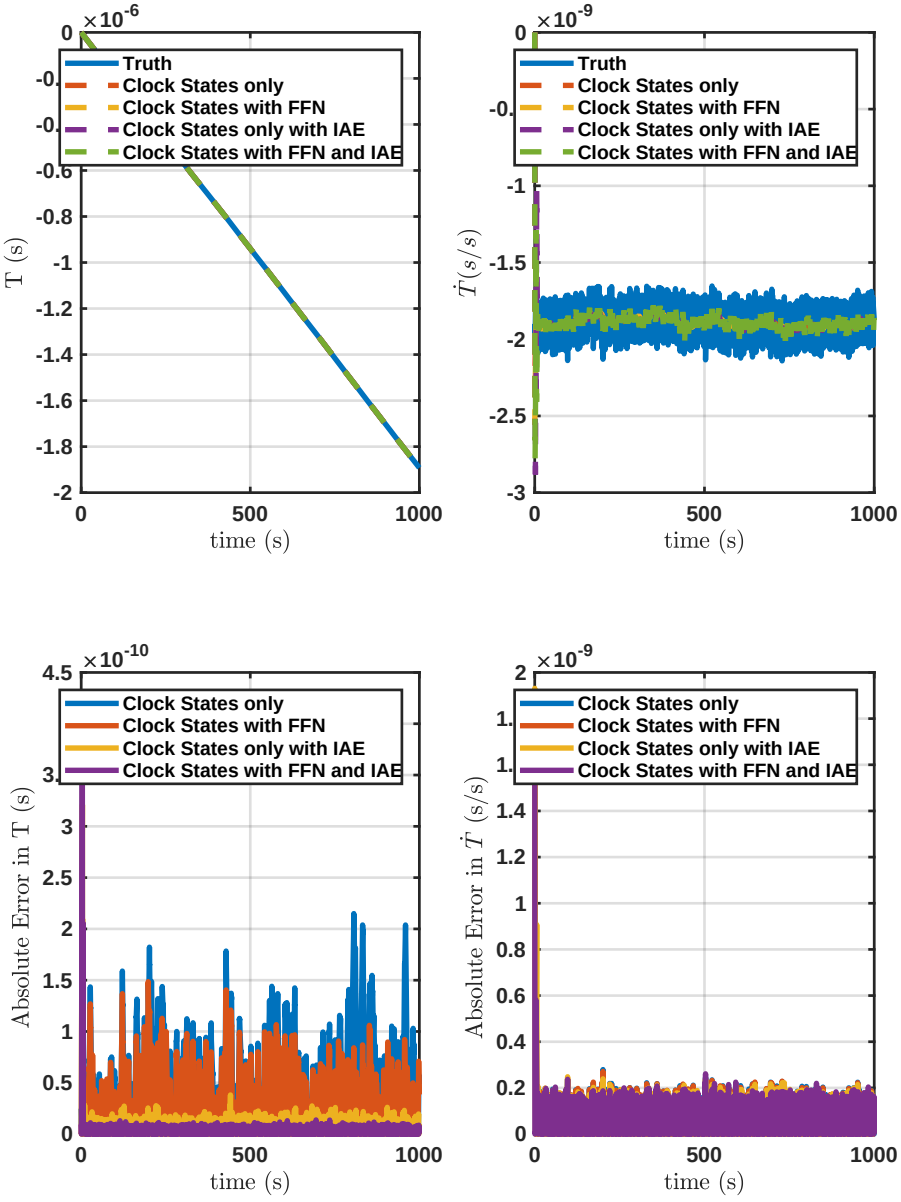

Figure B.2: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-3e-10s/s, DC Static Time Bias Error:-3e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):20,42, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):67,35

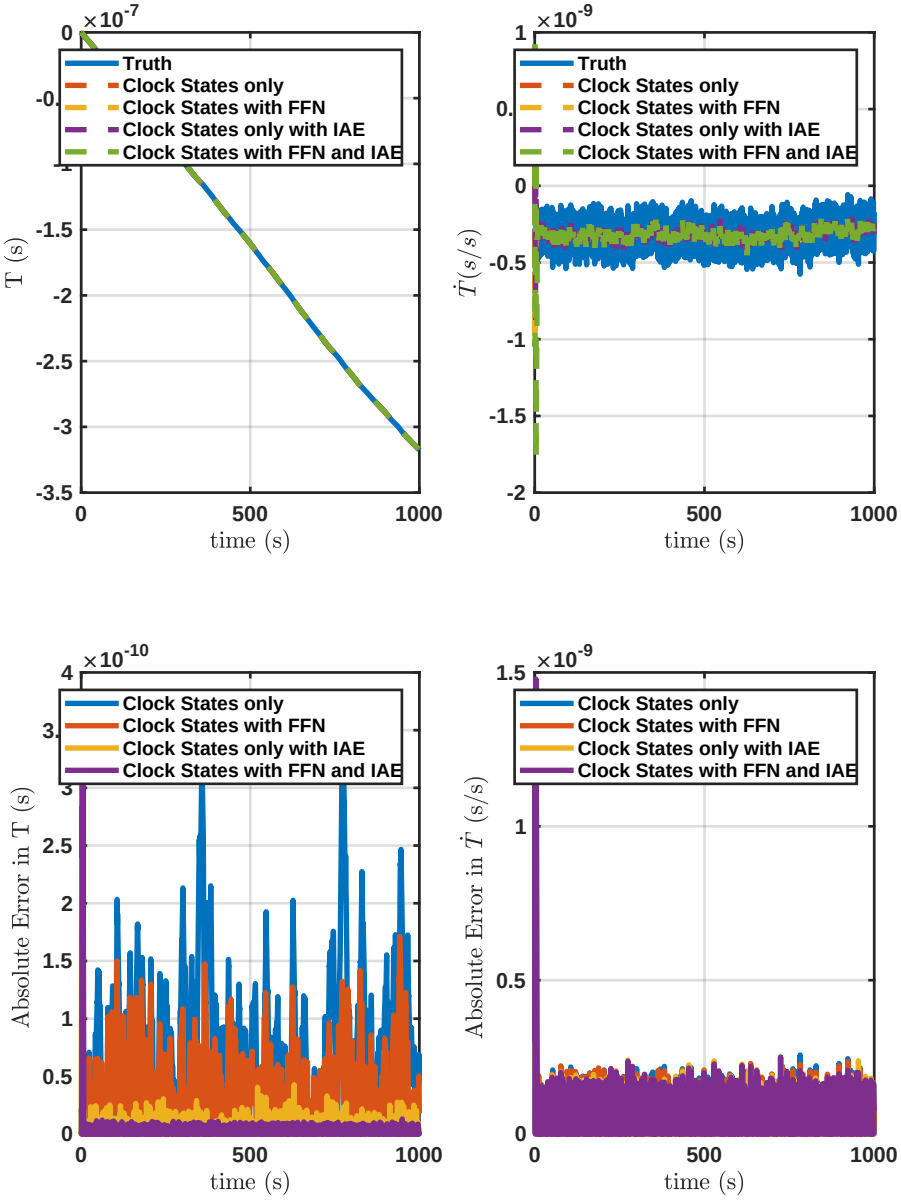

Figure B.3: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

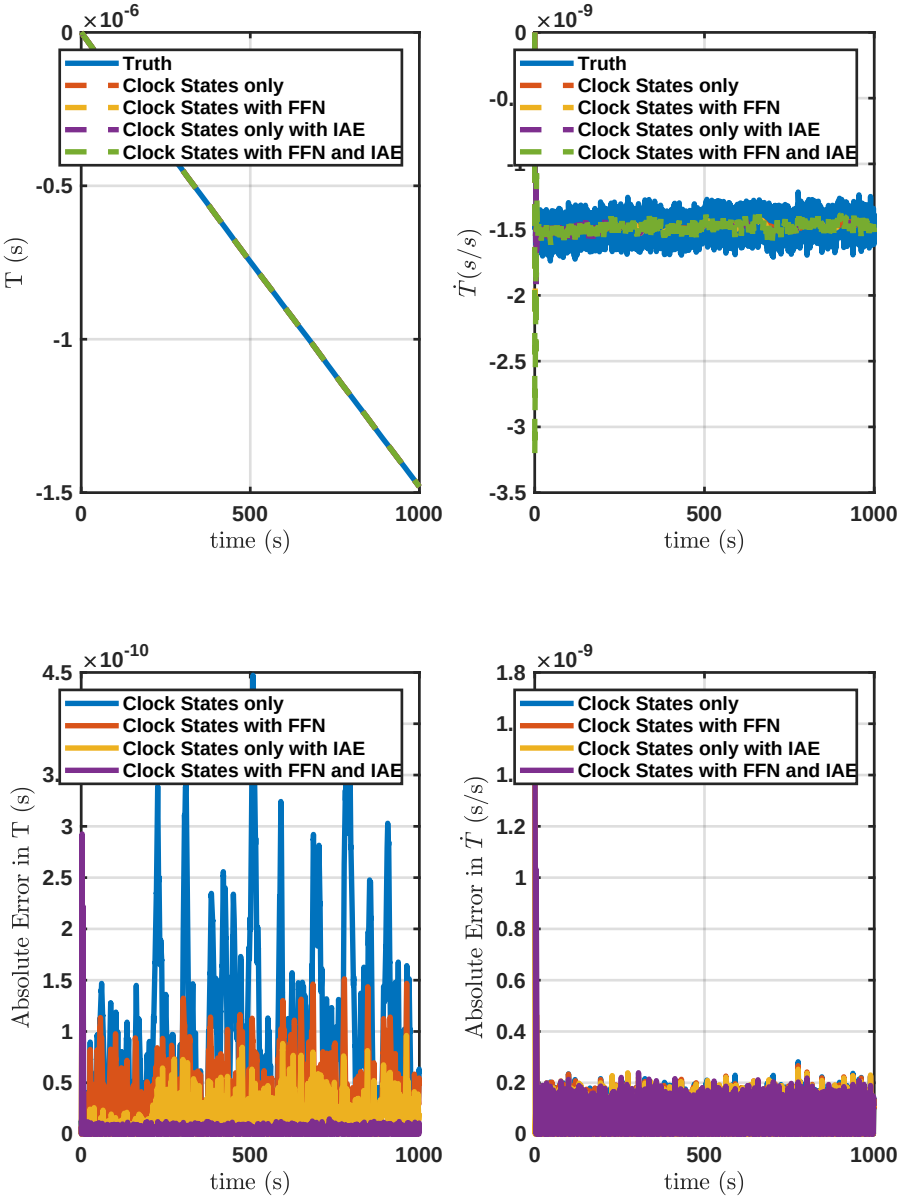

Figure B.4: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

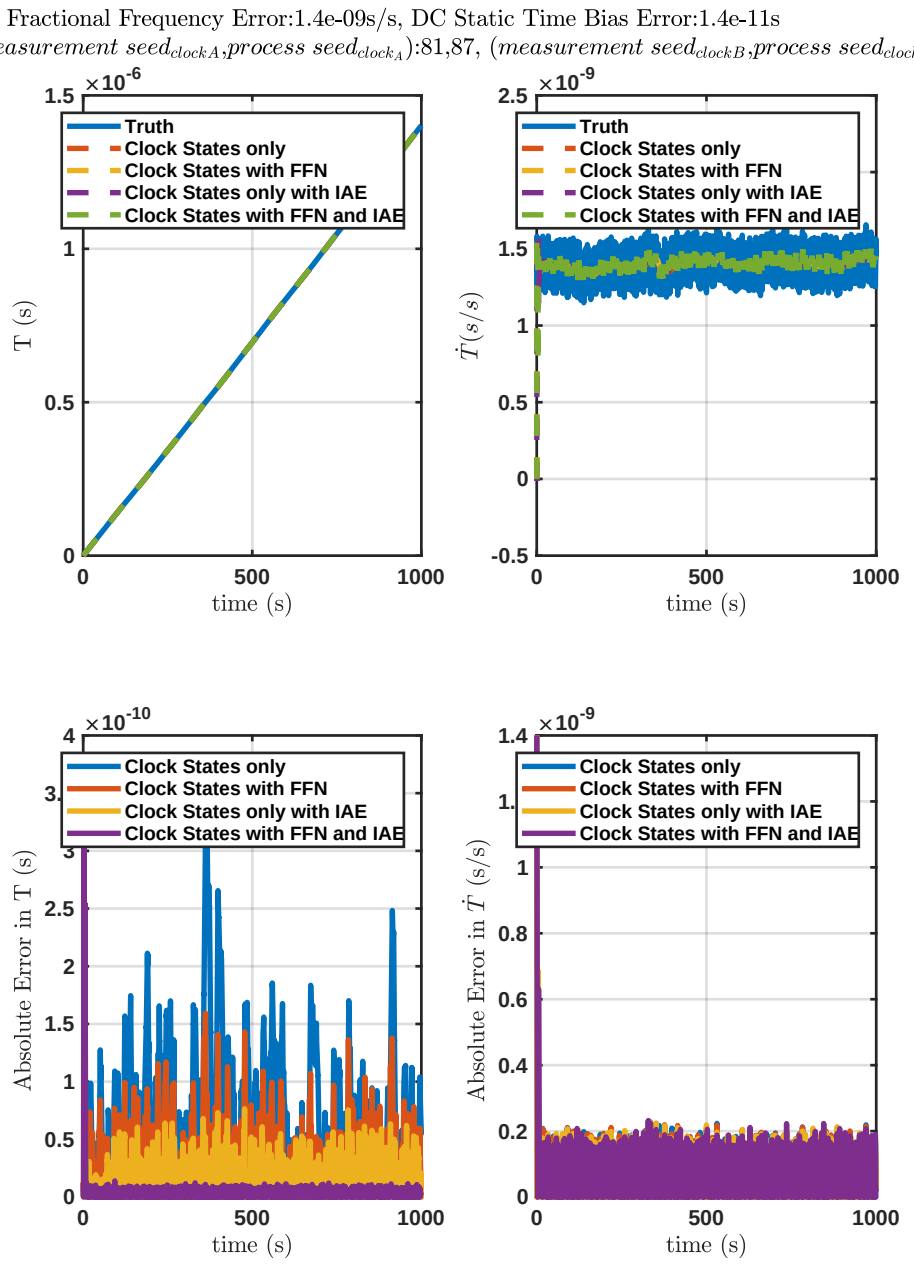

Figure B.5: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

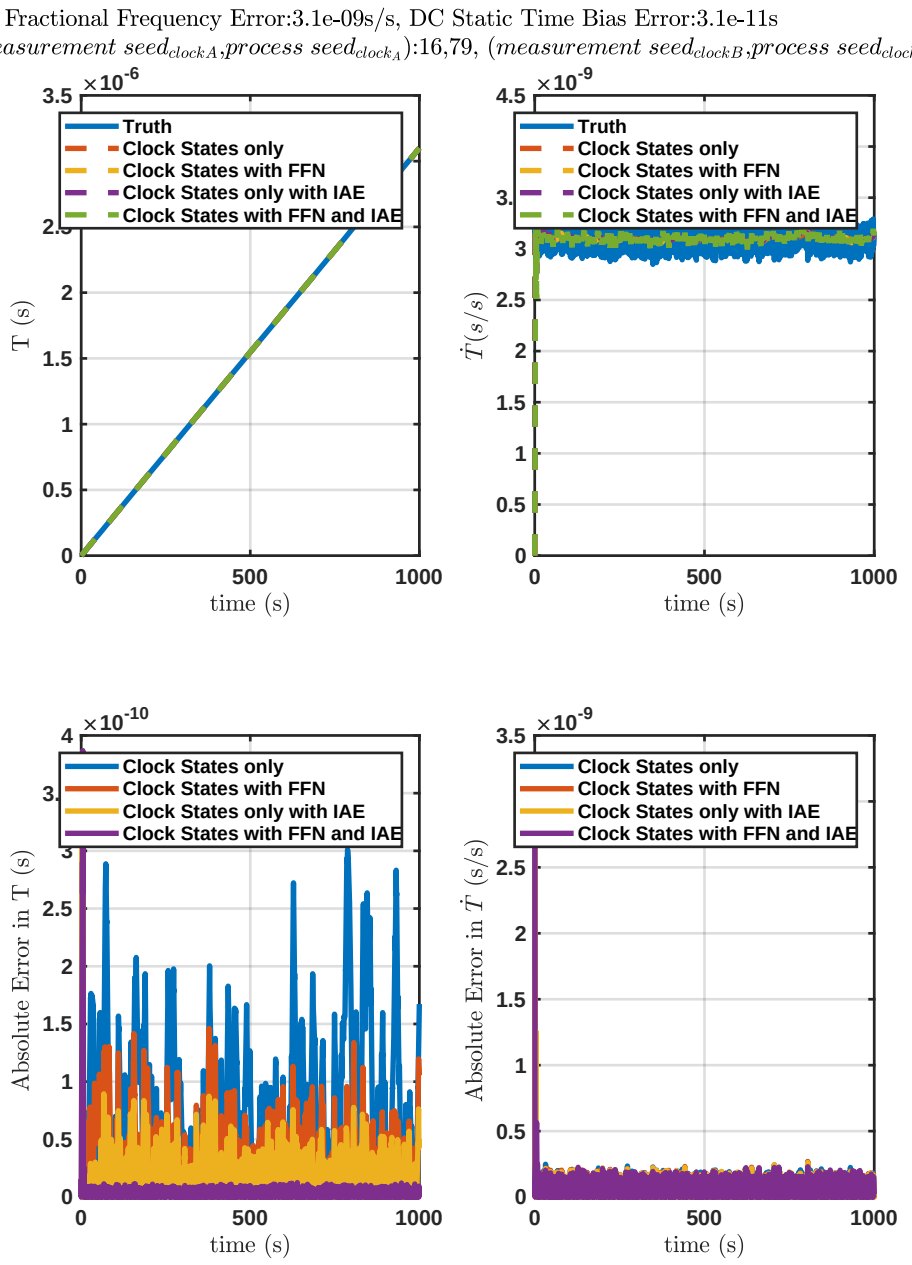

Figure B.6: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:4e-10s/s, DC Static Time Bias Error:4e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):44,61, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):67,40

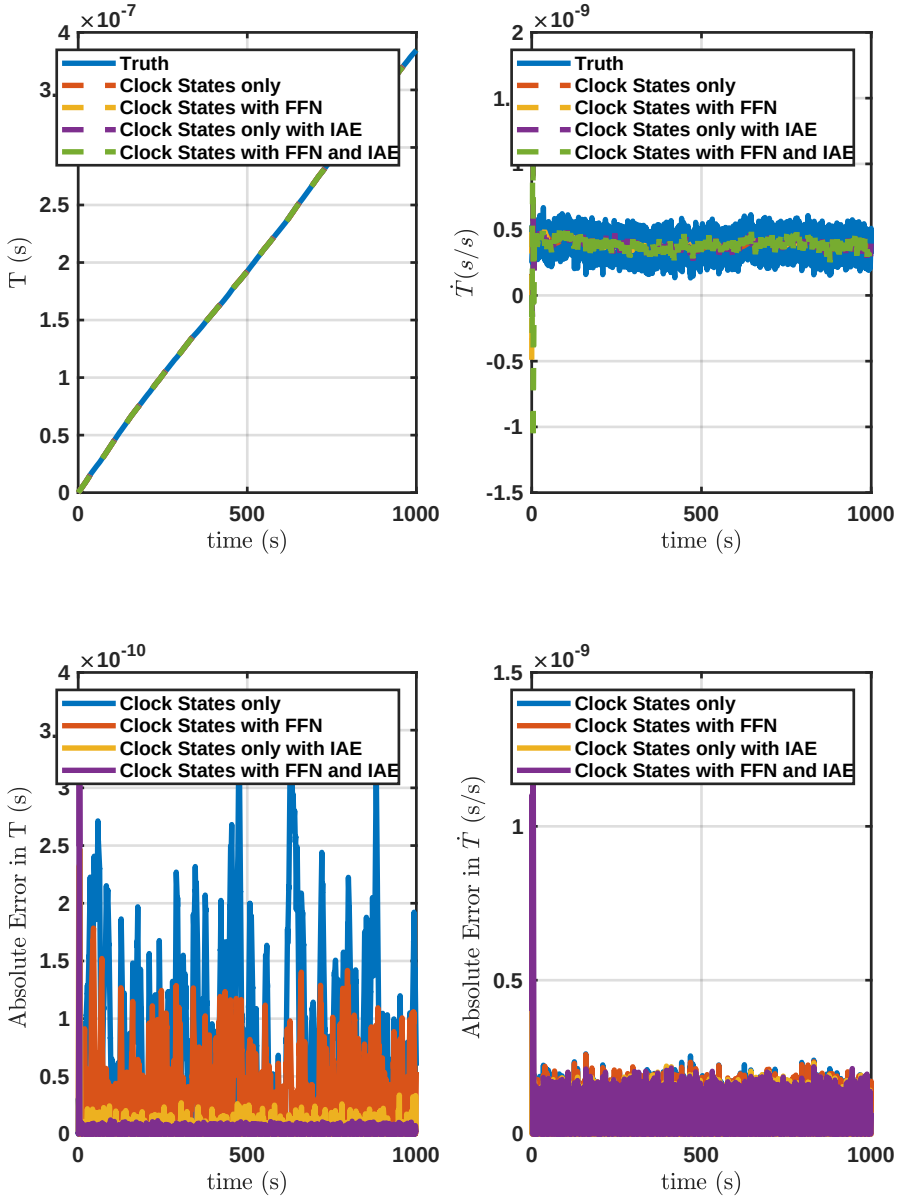

Figure B.7: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1e-09s/s, DC Static Time Bias Error:-1e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):58,61, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):14,31

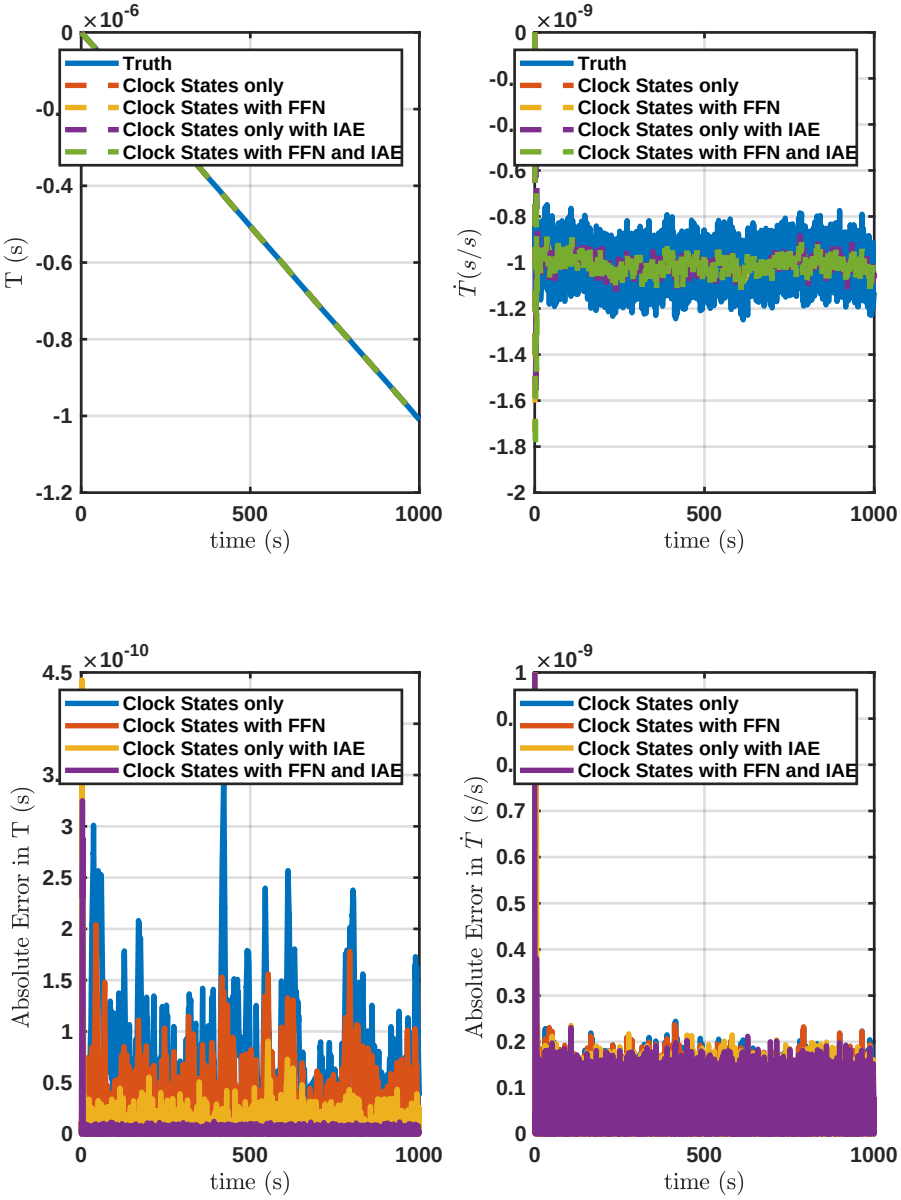

Figure B.8: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1.3e-09s/s, DC Static Time Bias Error:-1.3e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):76,37, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):41,67

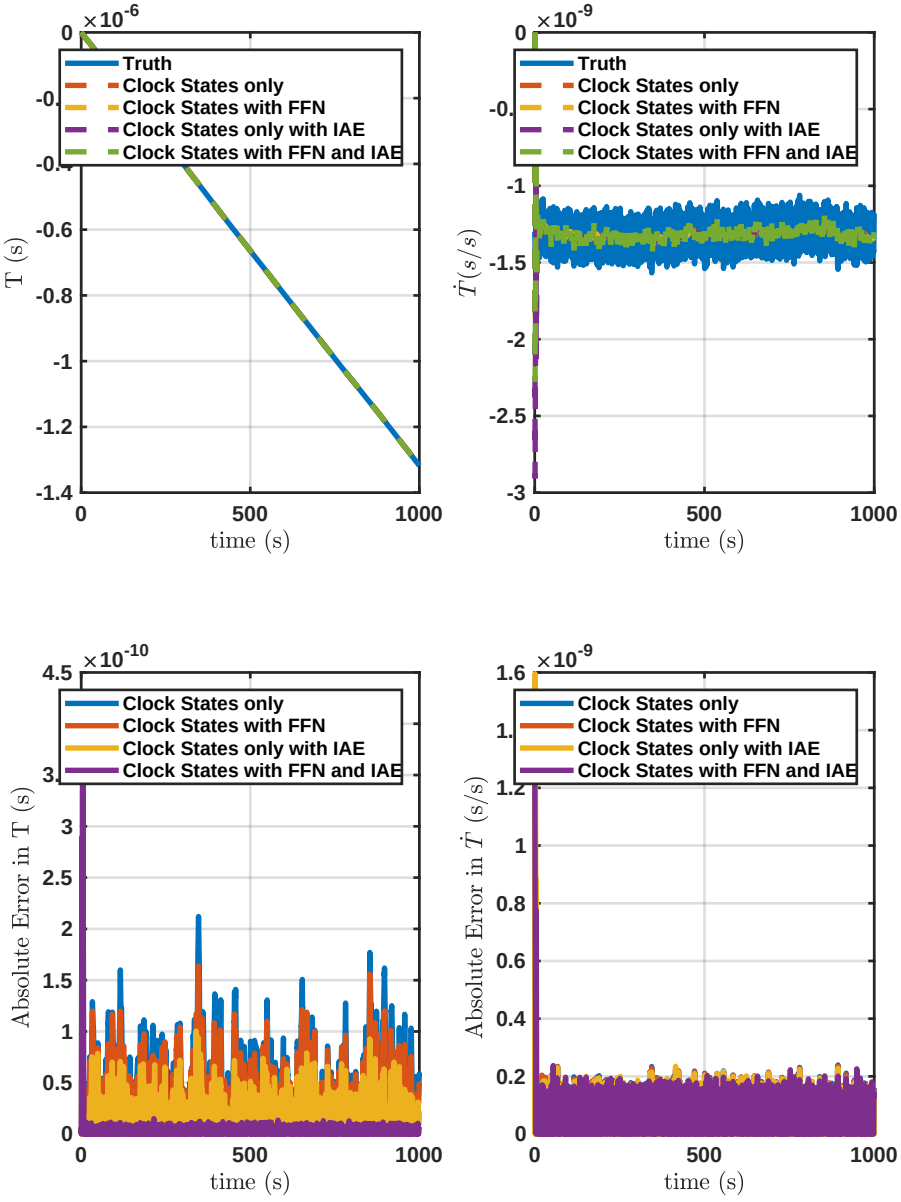

Figure B.9: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

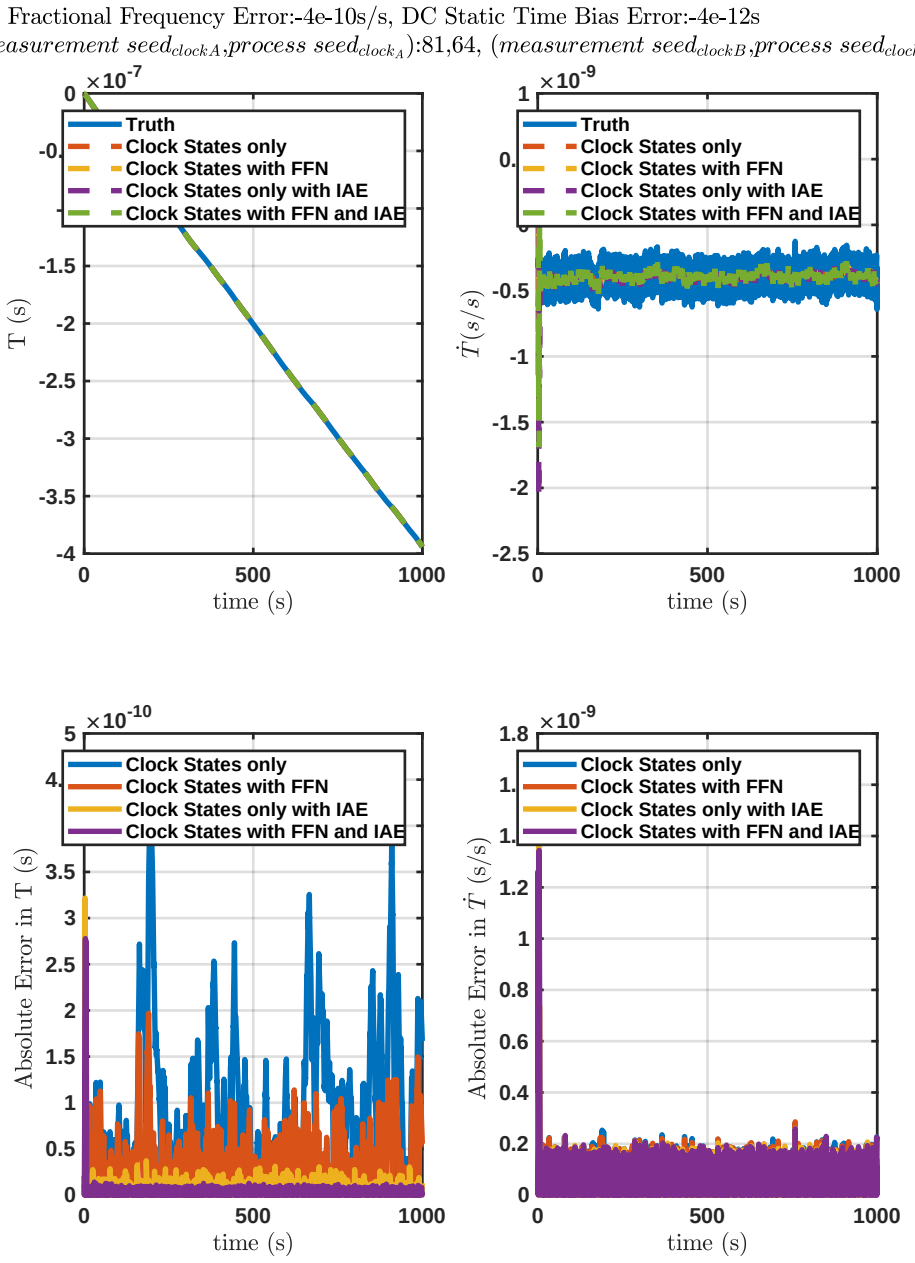

Figure B.10: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-3e-10s/s, DC Static Time Bias Error:-3e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):67,46, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):99,56

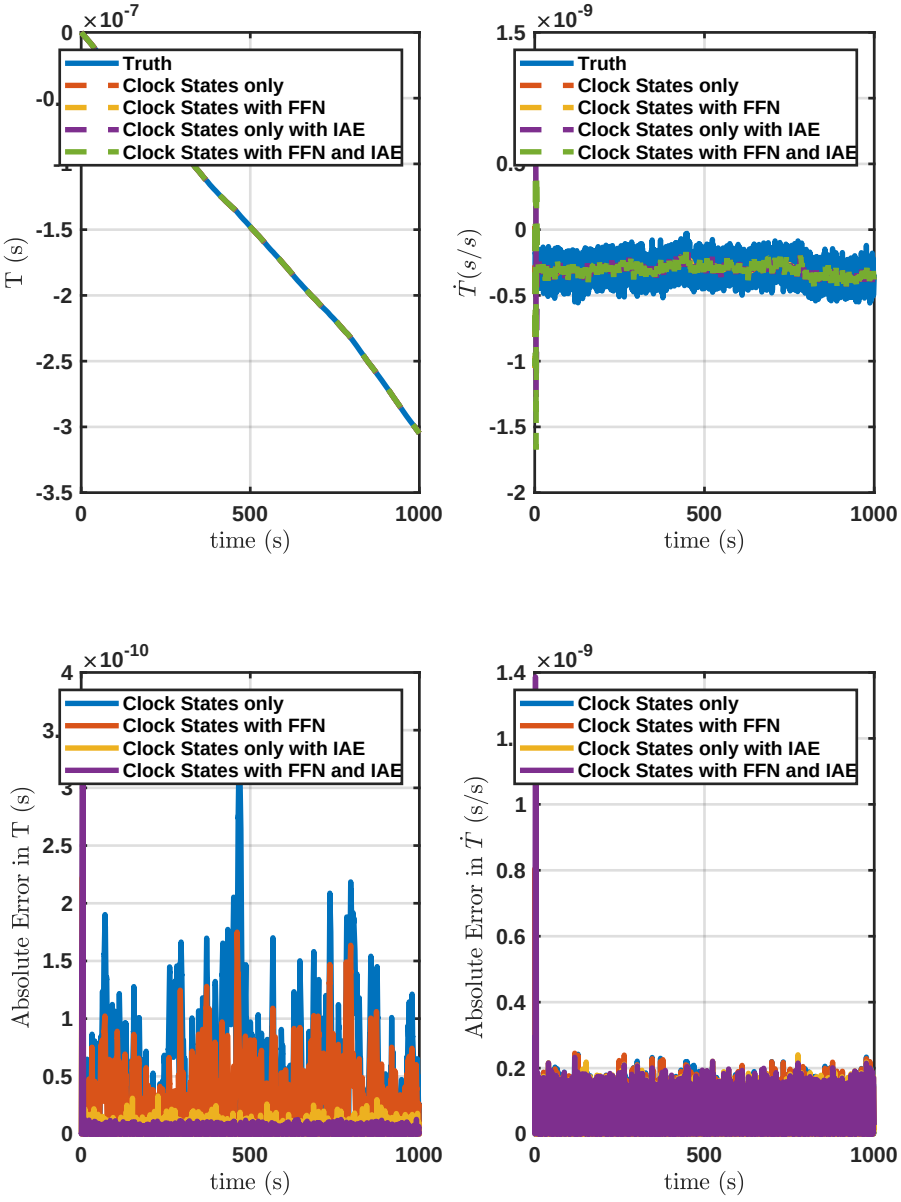

Figure B.11: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

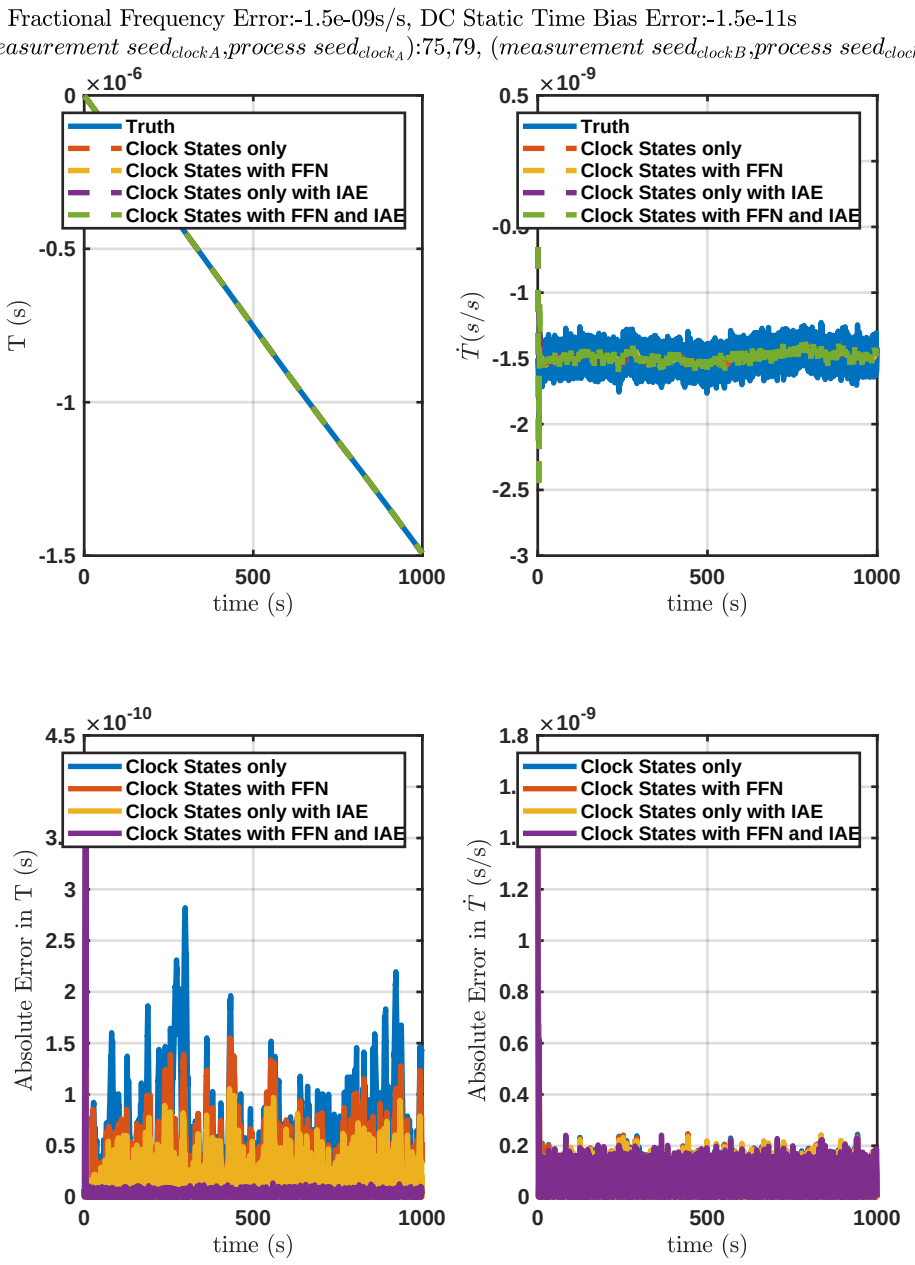

Figure B.12: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-3e-10s/s, DC Static Time Bias Error:-3e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):96,56, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):10,23

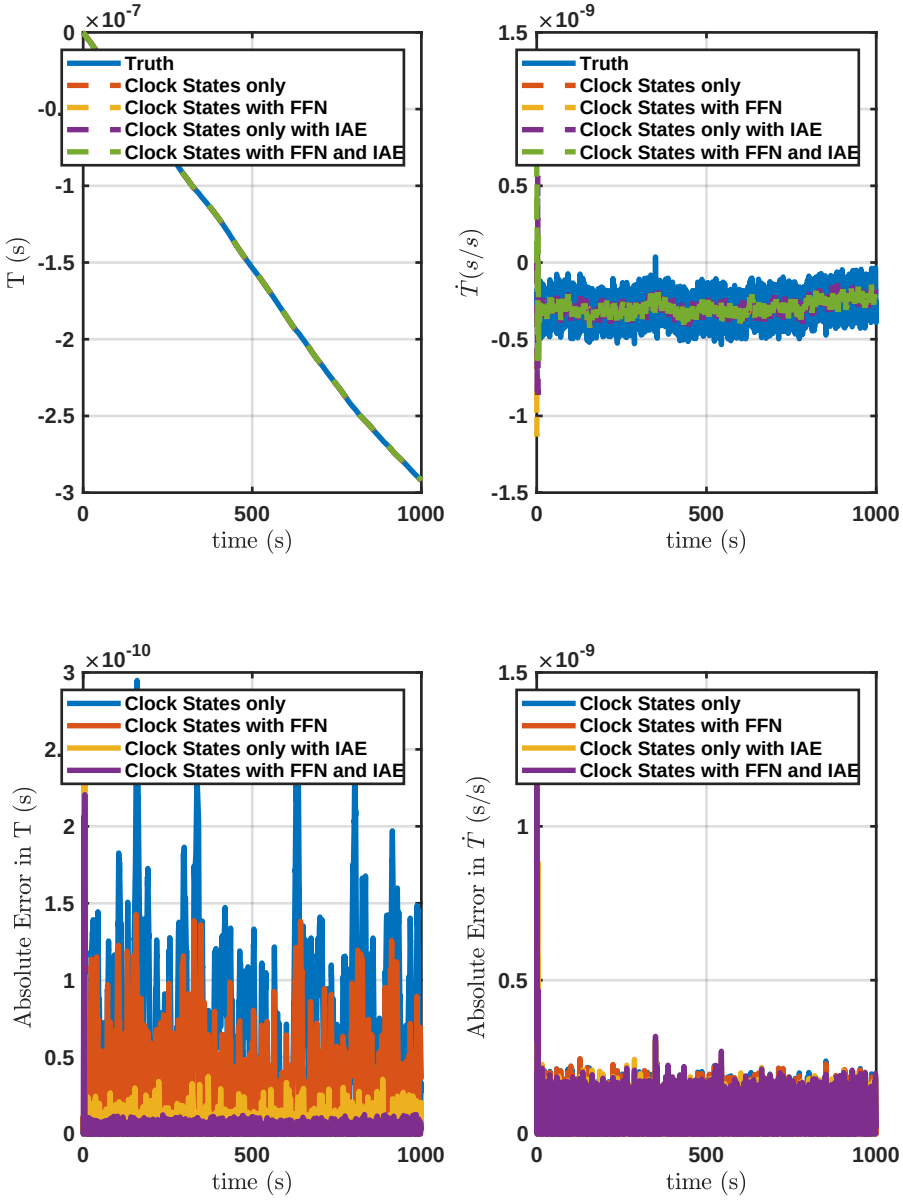

Figure B.13: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:4.4e-09s/s, DC Static Time Bias Error:4.4e-11s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):54,52, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):5,24

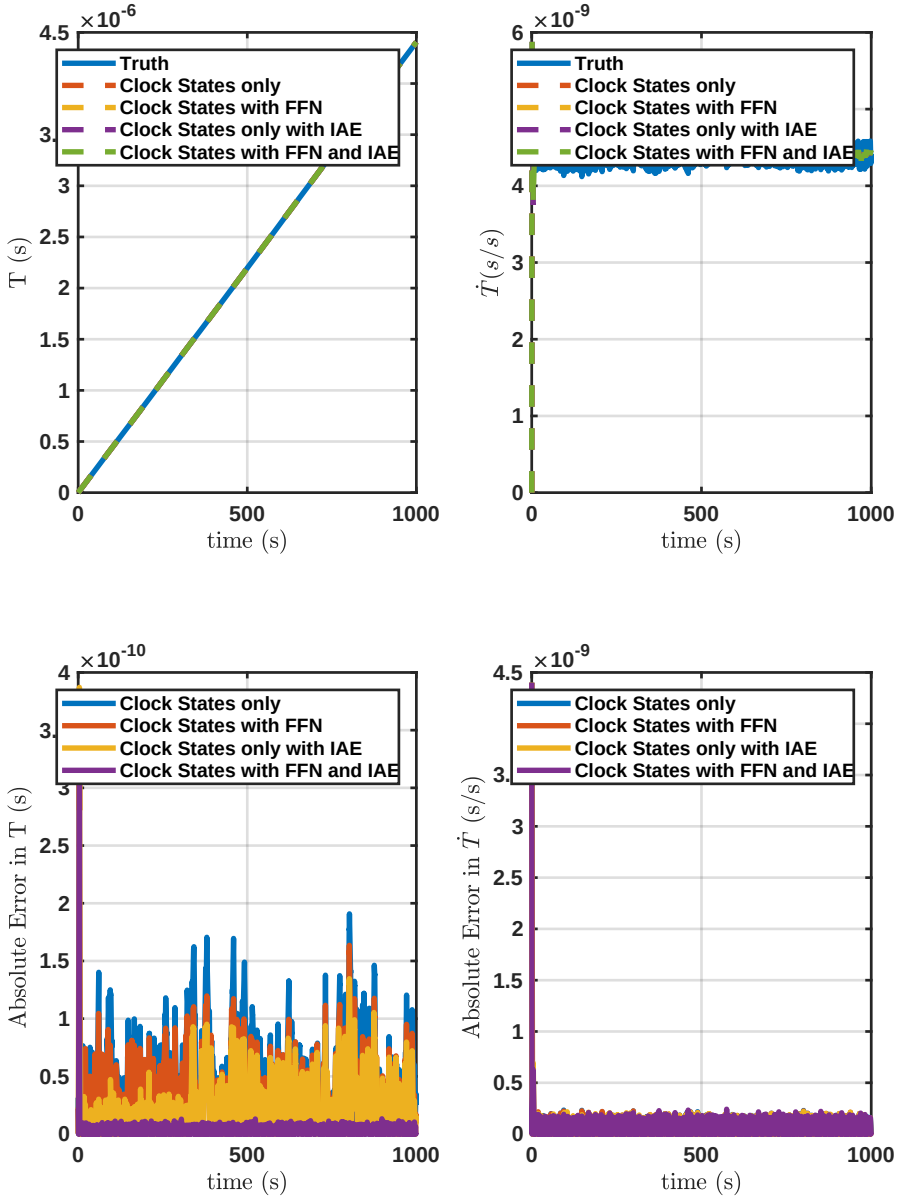

Figure B.14: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

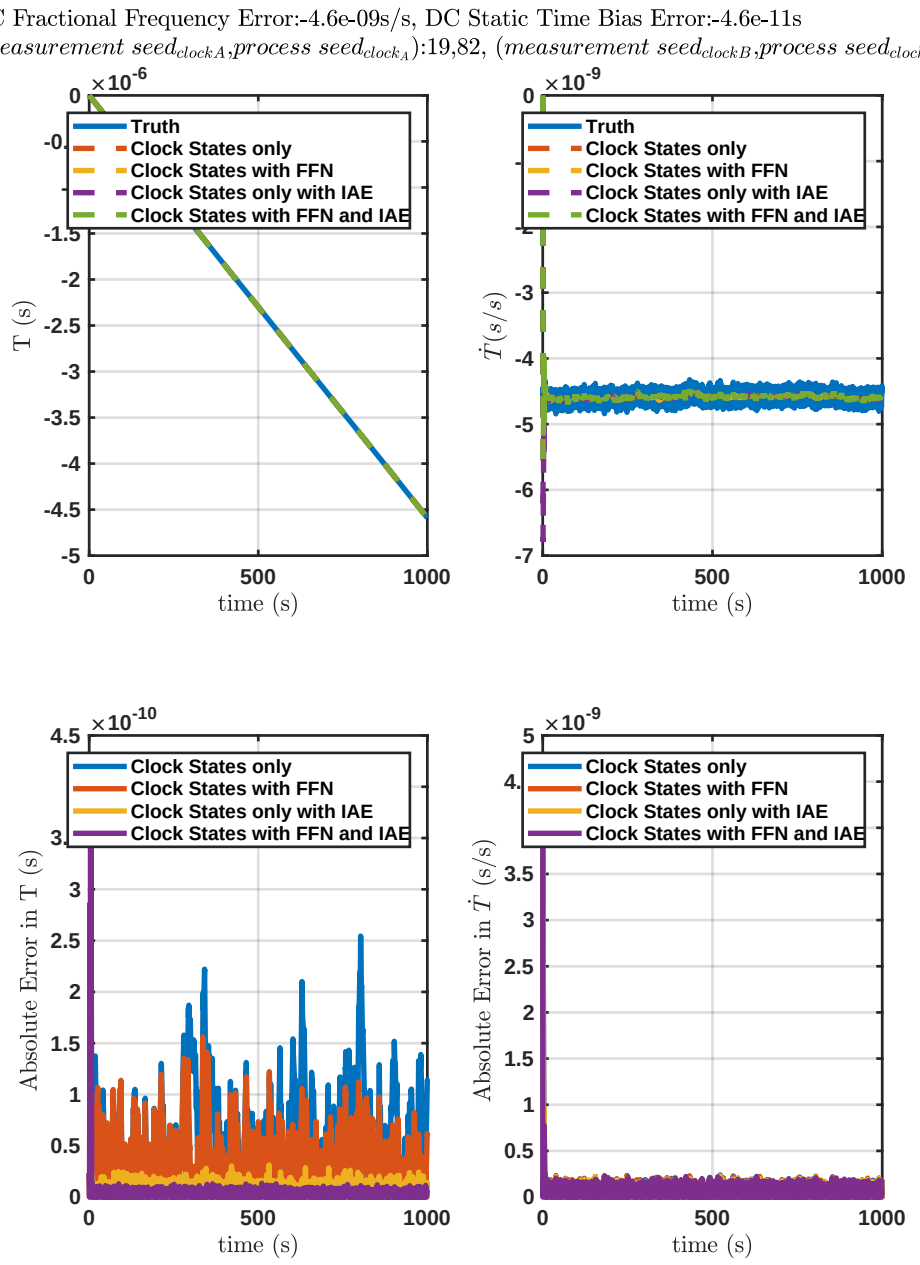

Figure B.15: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

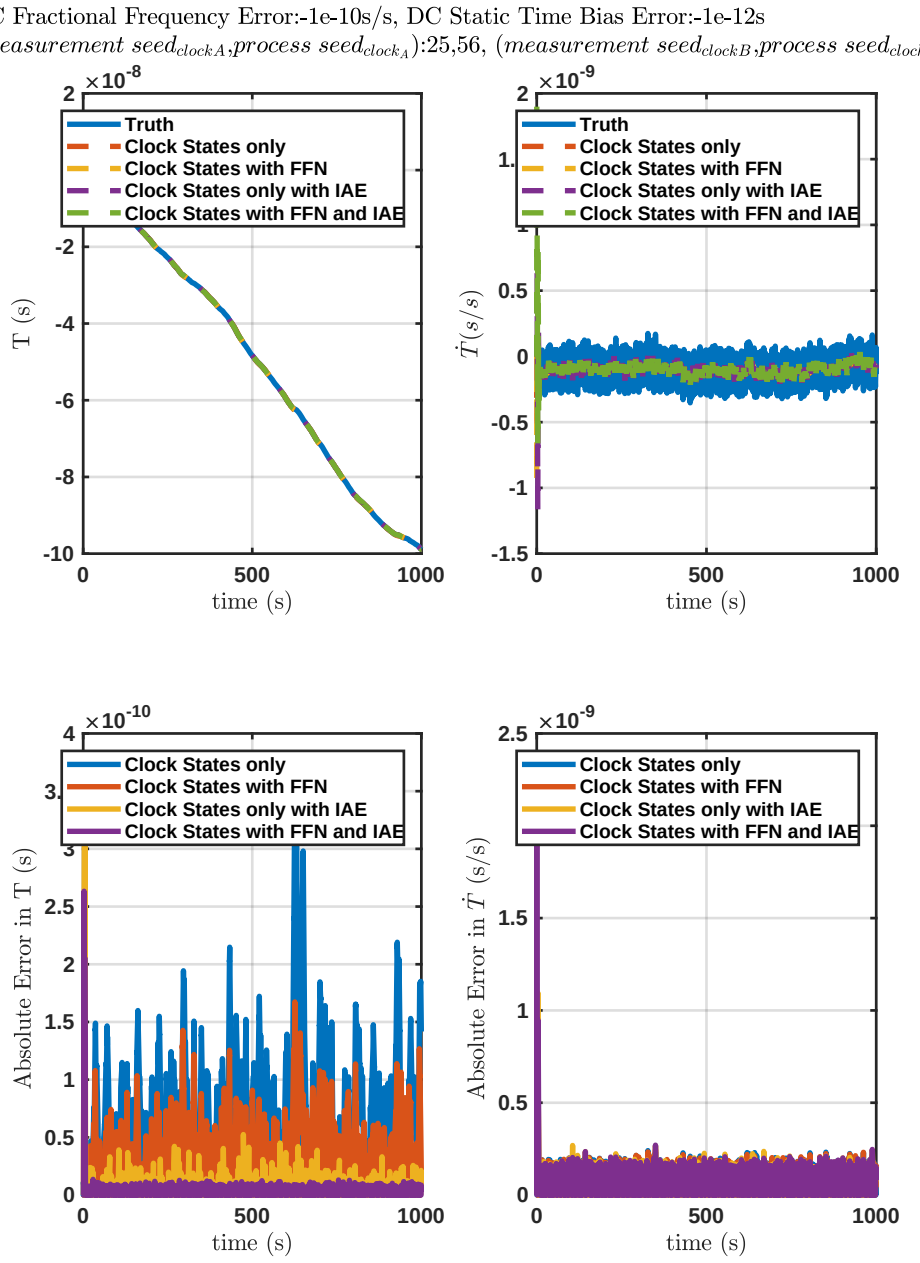

Figure B.16: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:6e-10s/s, DC Static Time Bias Error:6e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):52,65, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):81,46

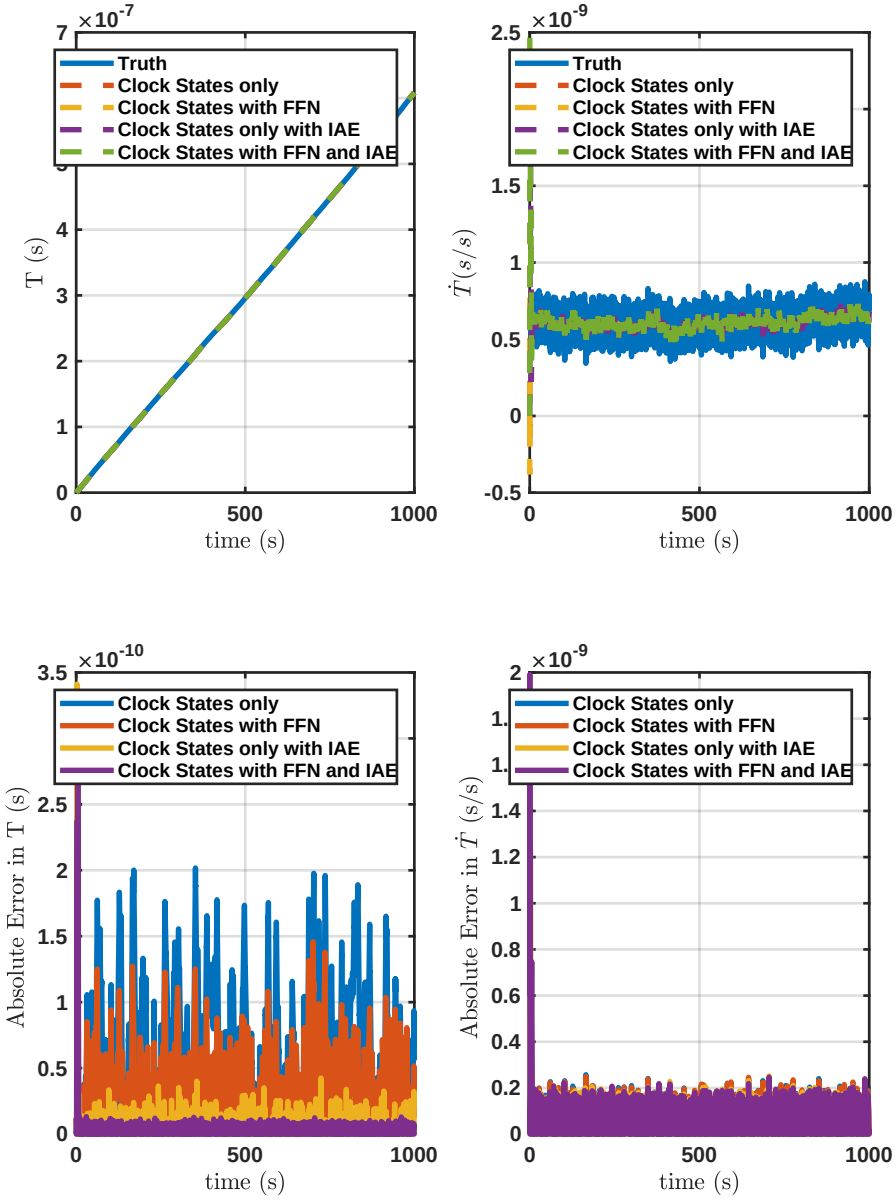

Figure B.17: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

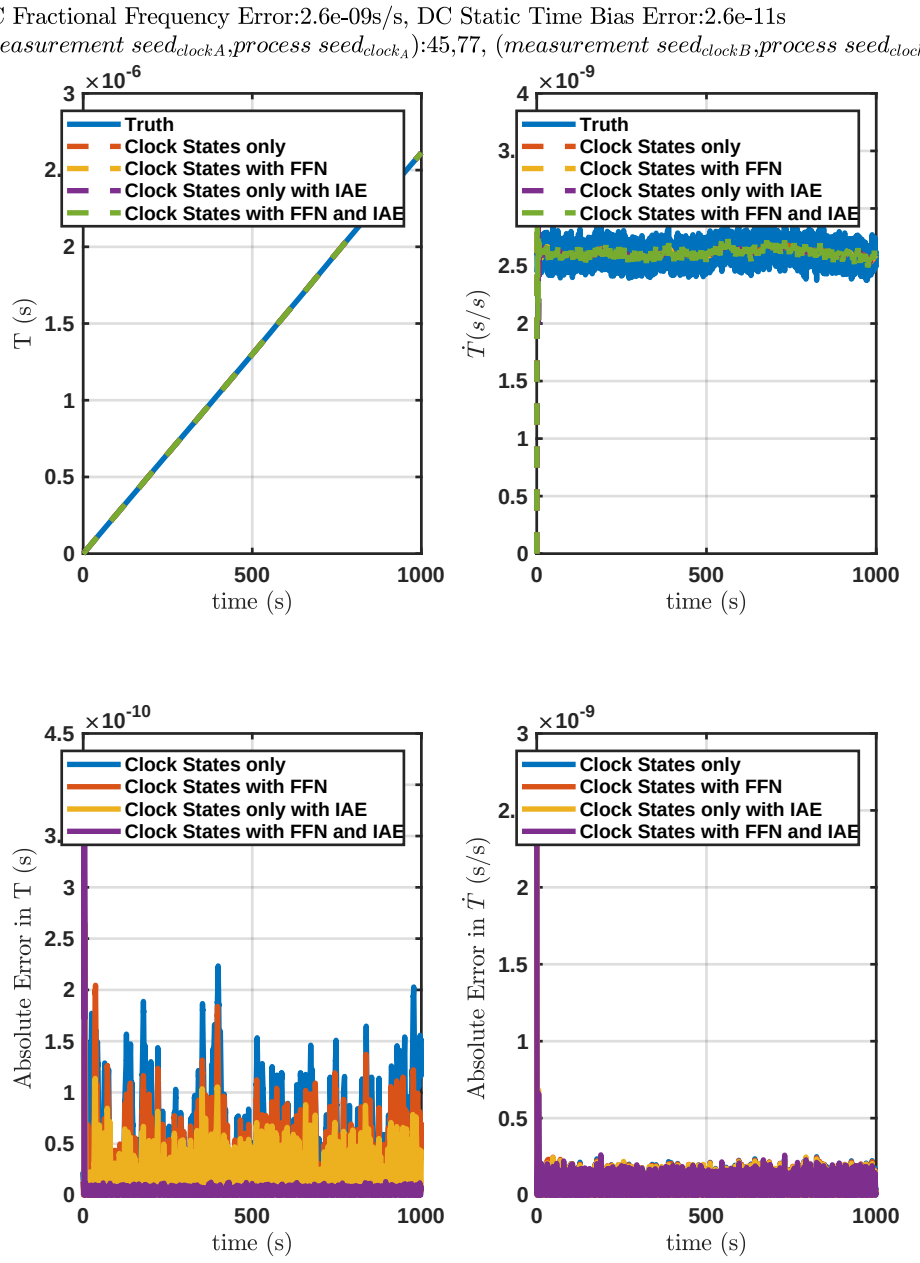

Figure B.18: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

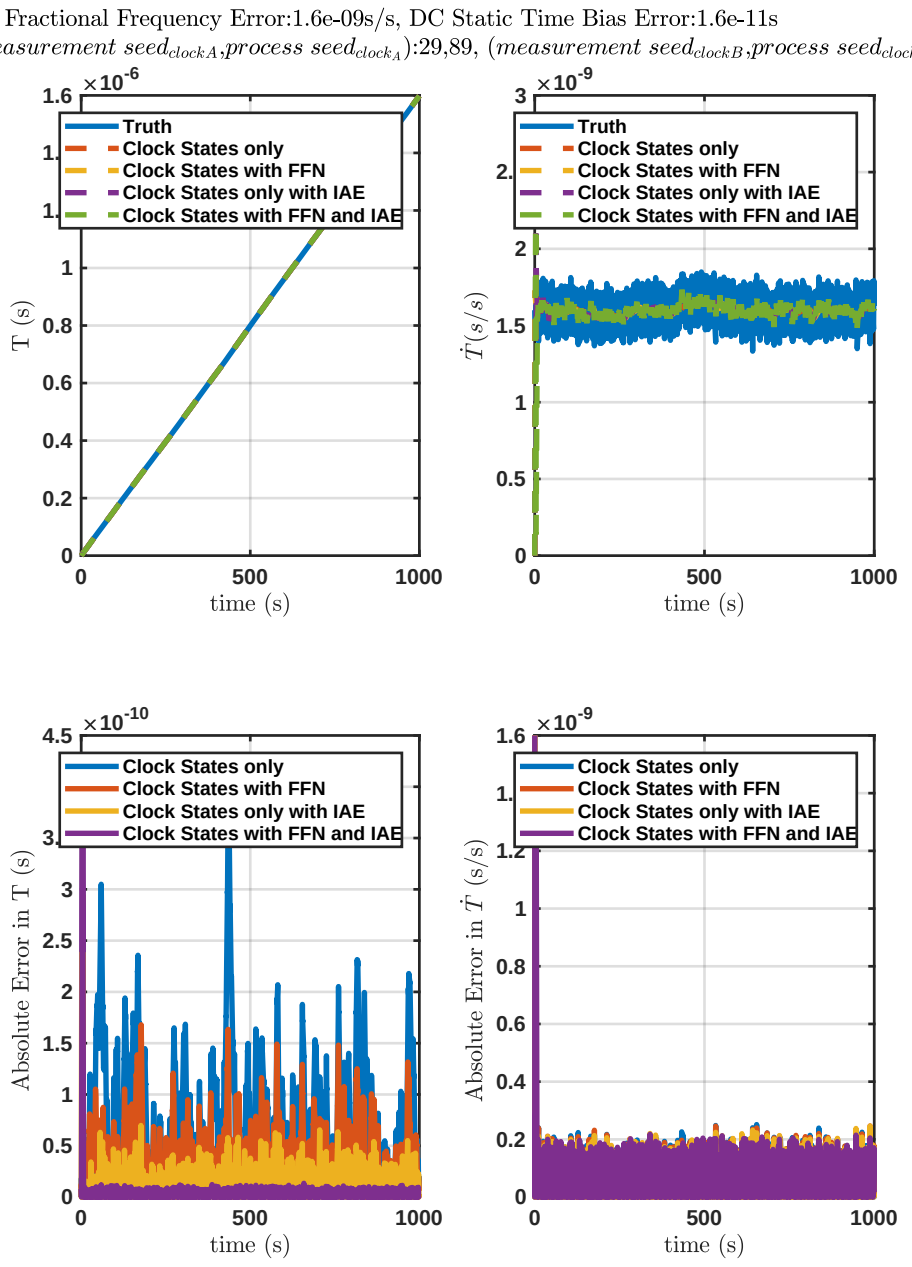

Figure B.19: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:4e-09s/s, DC Static Time Bias Error:4e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):63,27, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):37,49

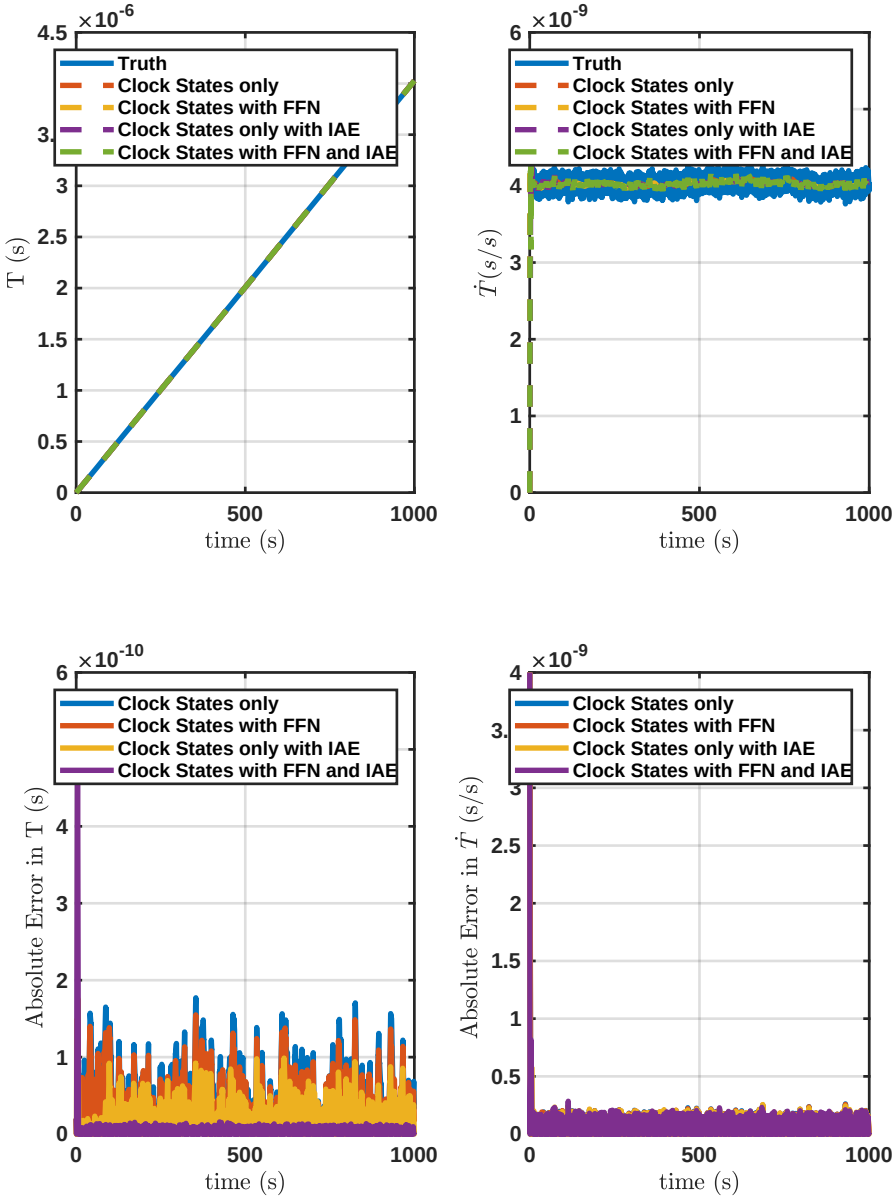

Figure B.20: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:1.9e-09s/s, DC Static Time Bias Error:1.9e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):38,15, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):59,36

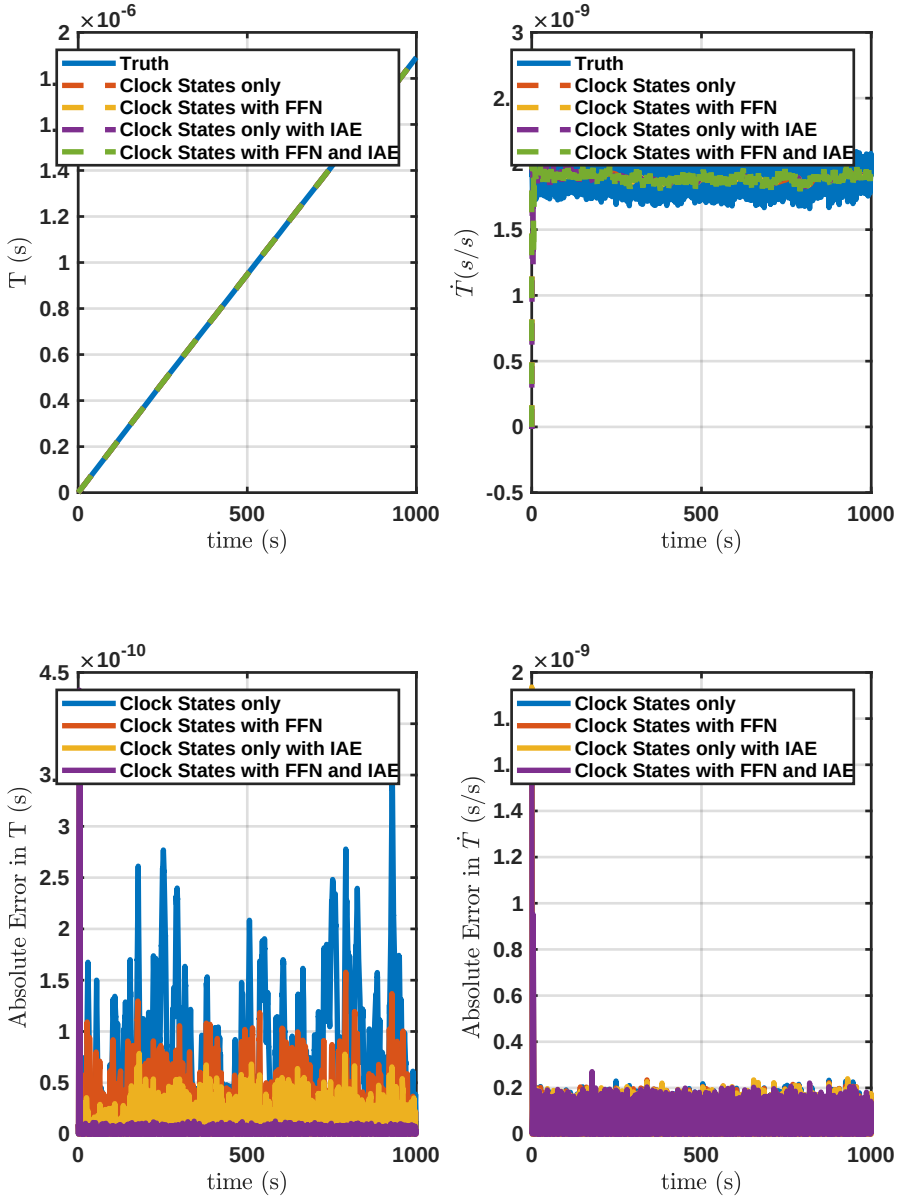

Figure B.21: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:4.7e-09s/s, DC Static Time Bias Error:4.7e-11s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):55,77, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):100,15

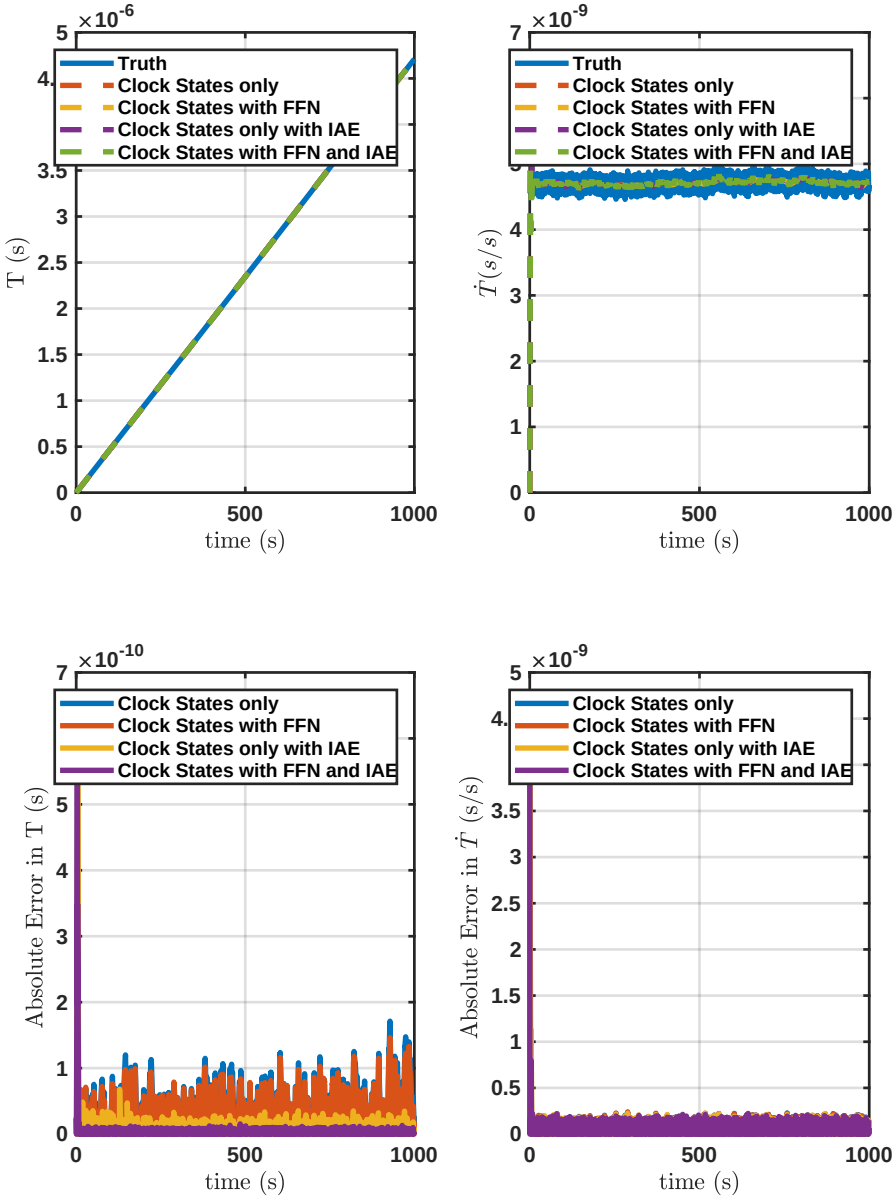

Figure B.22: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:5e-10s/s, DC Static Time Bias Error:5e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):24,30, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):76,71

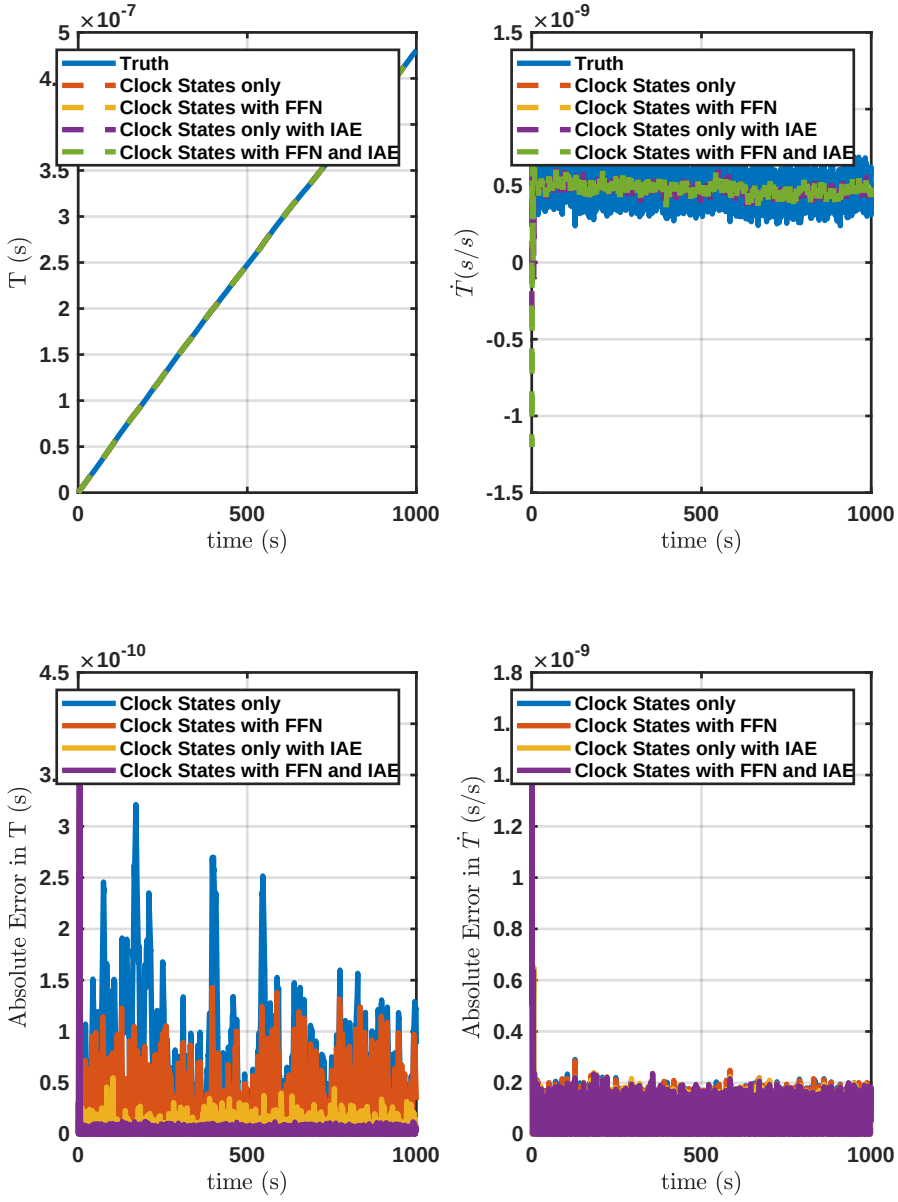

Figure B.23: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-7e-10s/s, DC Static Time Bias Error:-7e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):99,55, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):23,24

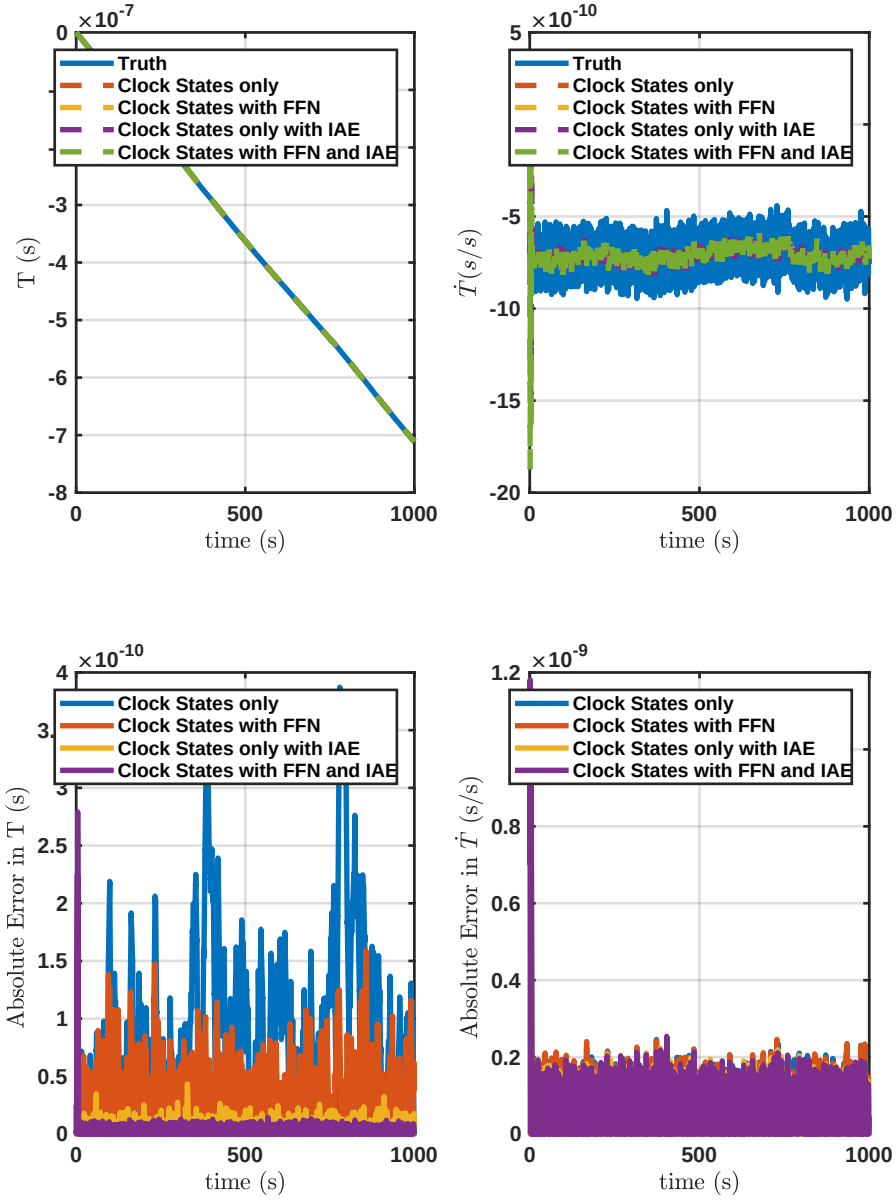

Figure B.24: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-6e-10s/s, DC Static Time Bias Error:-6e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):85,26, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):33,60

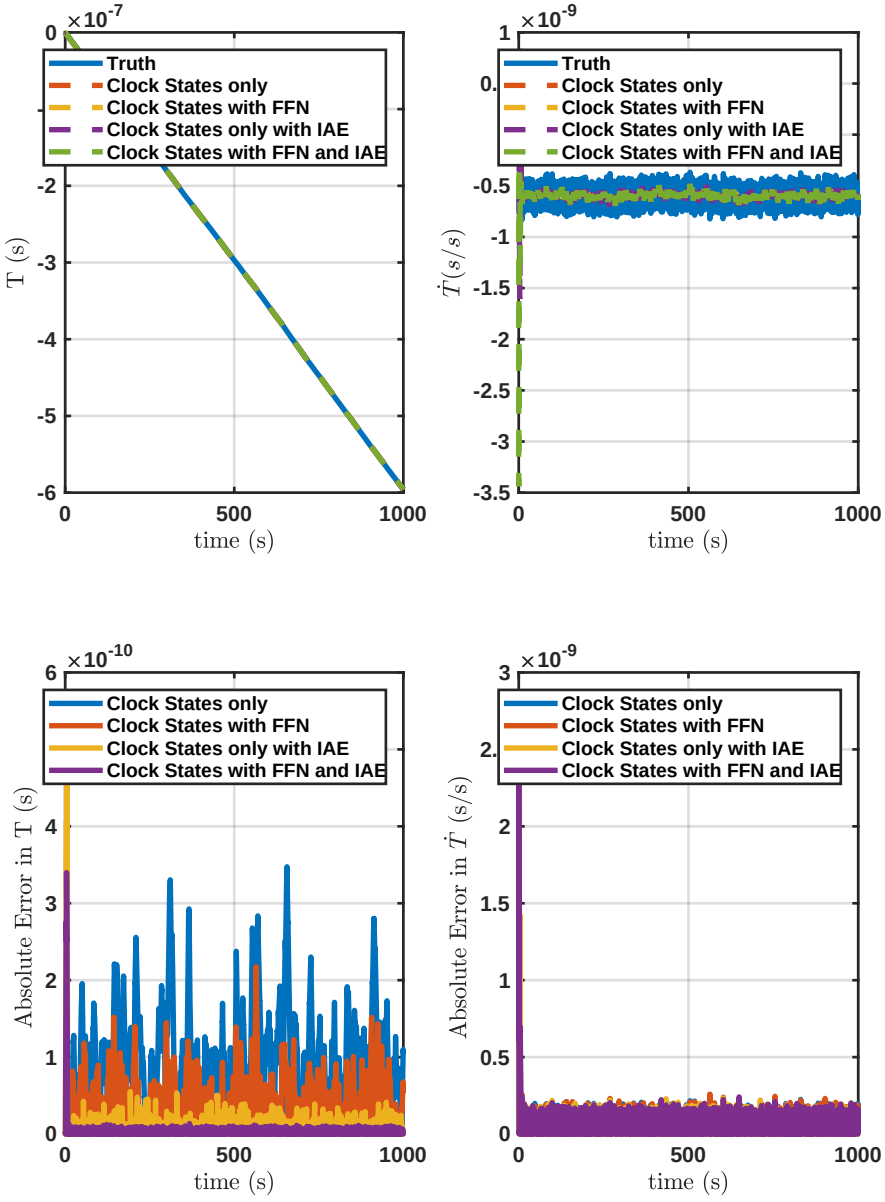

Figure B.25: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1e-10s/s, DC Static Time Bias Error:-1e-12s
($measurement\ seed_{clock\,A}$,$process\ seed_{clock_A}$):41,37, ($measurement\ seed_{clock\,B}$,$process\ seed_{clock_B}$):22,38

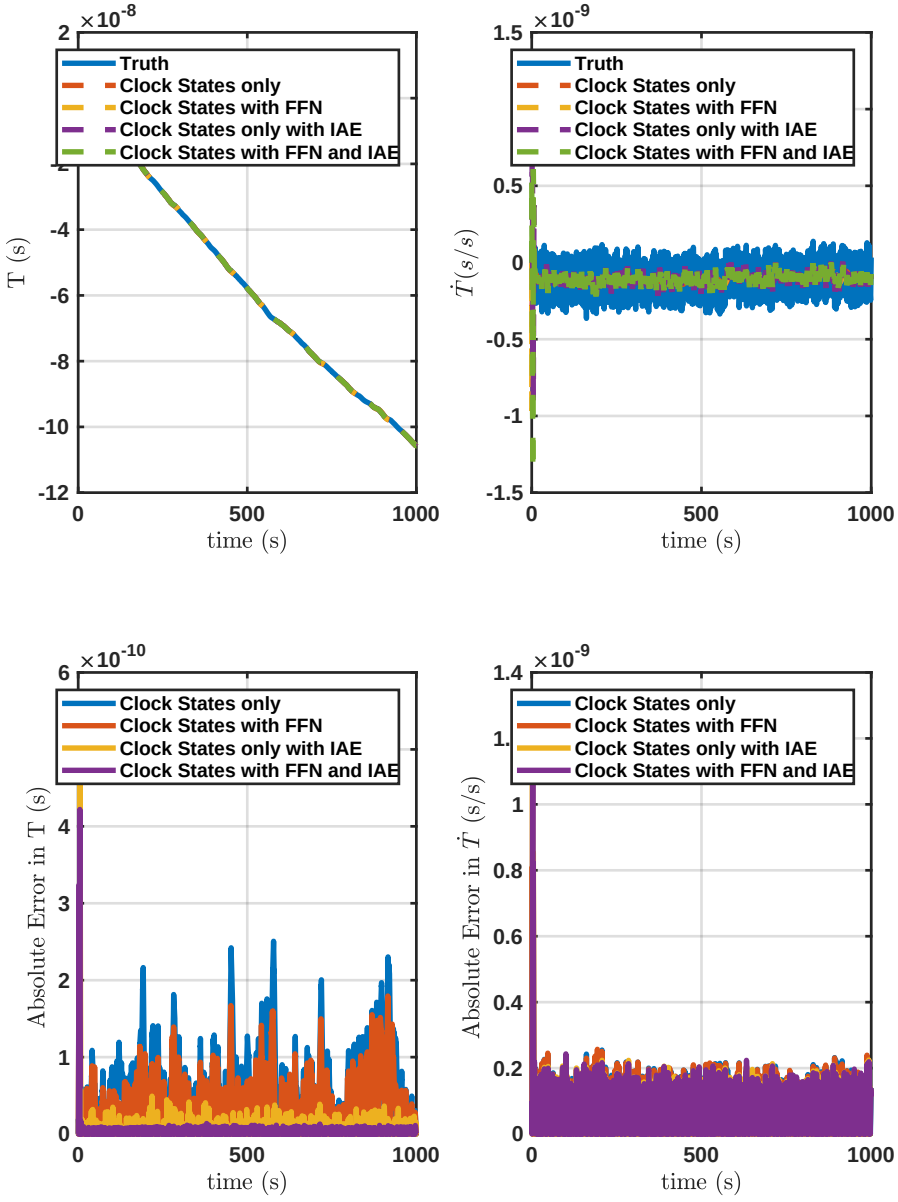

Figure B.26: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2.4e-09s/s, DC Static Time Bias Error:-2.4e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):71,77, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):25,16

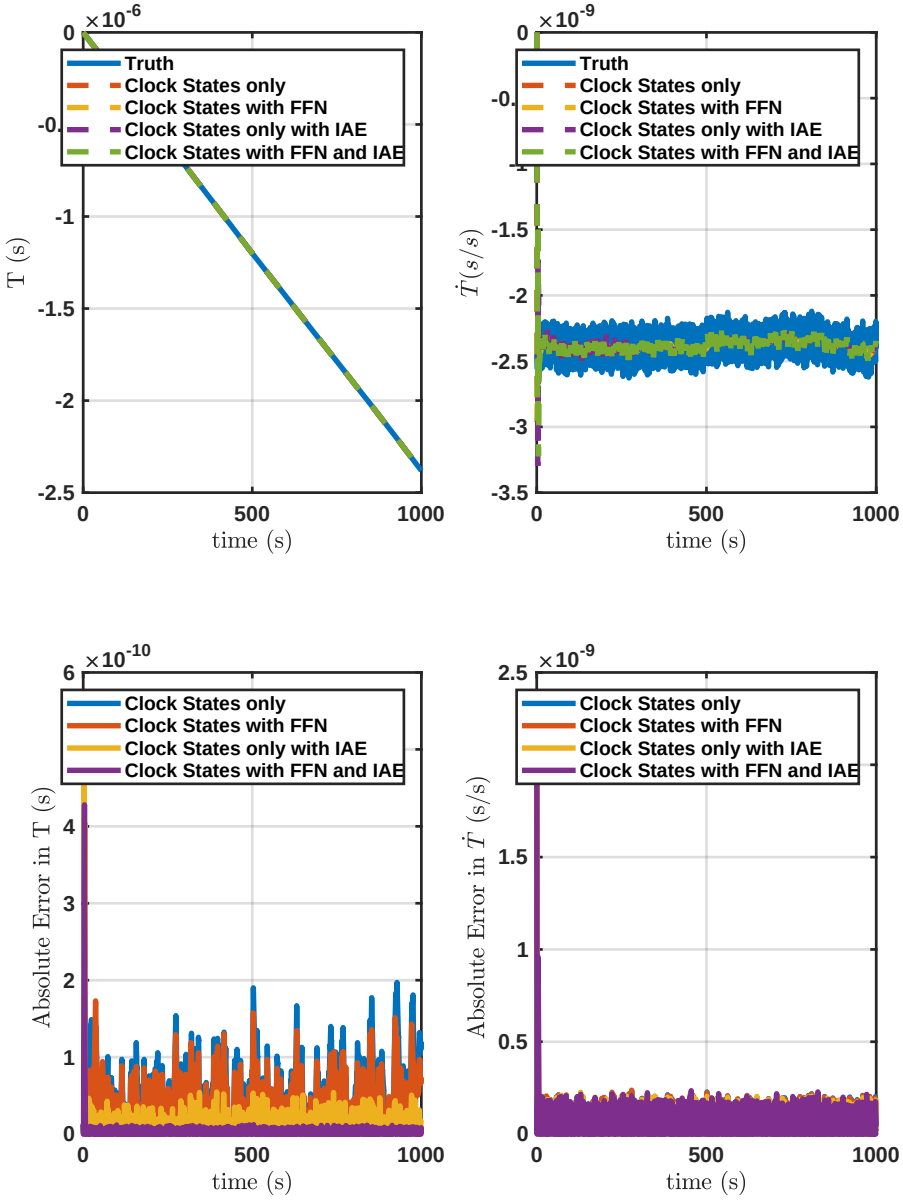

Figure B.27: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

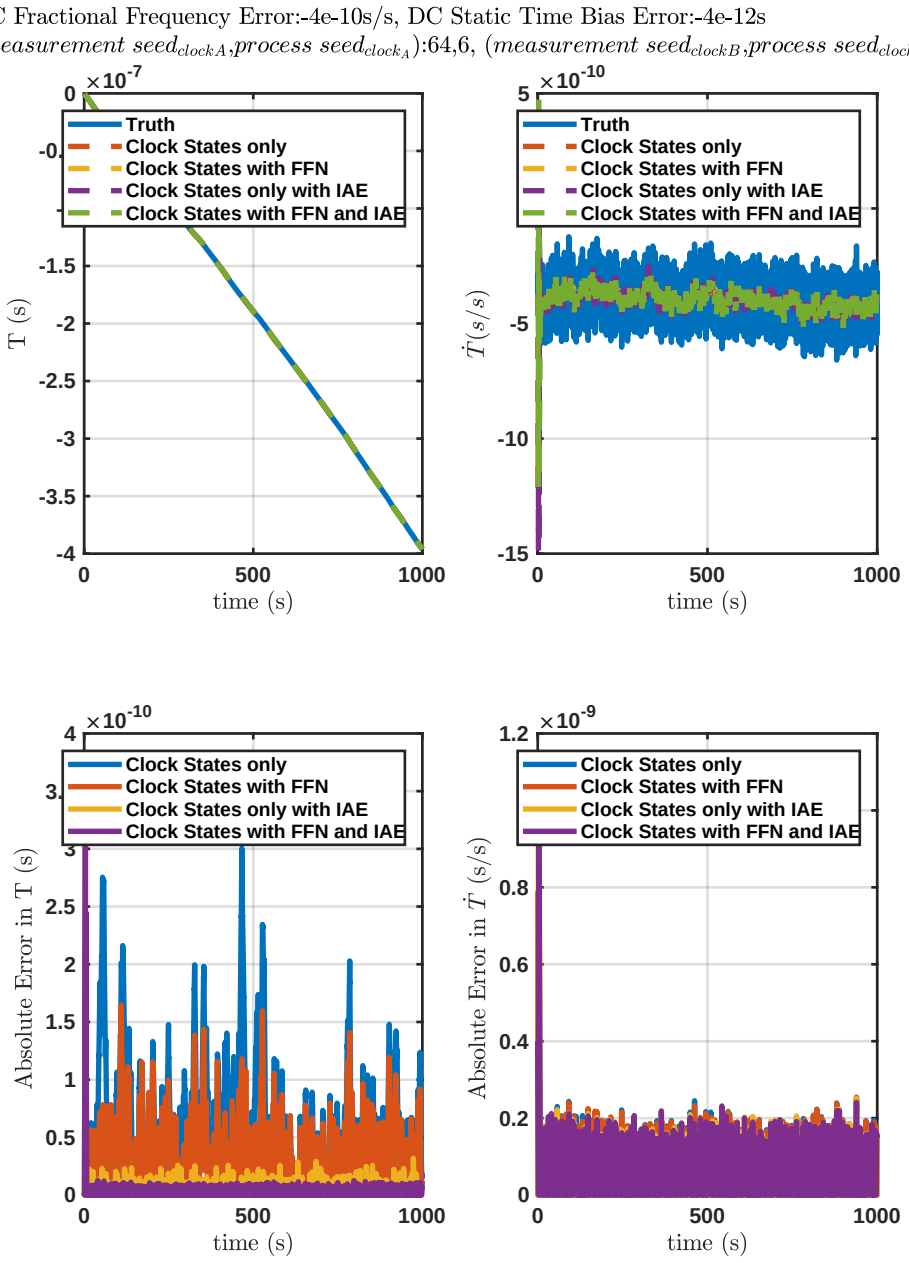

Figure B.28: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2e-09s/s, DC Static Time Bias Error:-2e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):64,100, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):92,72

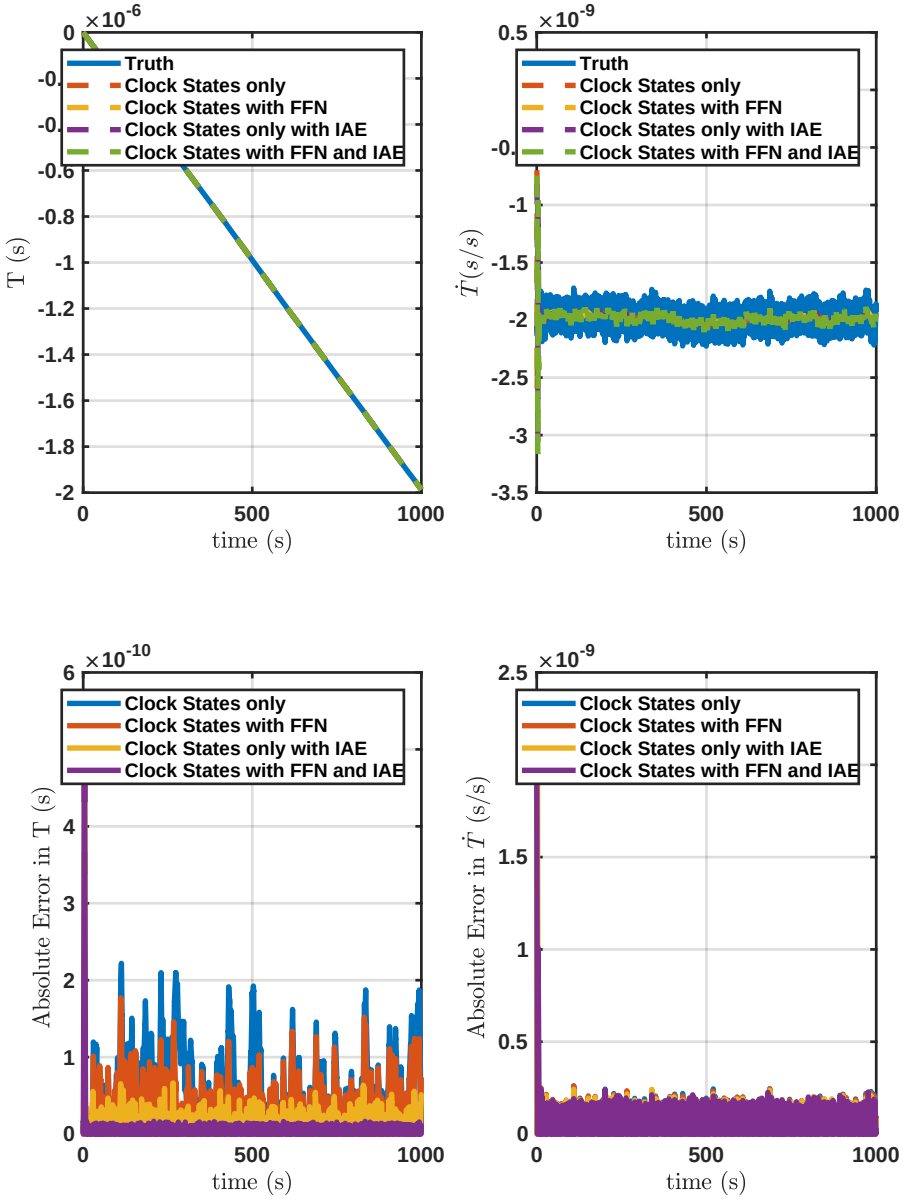

Figure B.29: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:1.1e-11s/s, DC Static Time Bias Error:2.46e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):45,27, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):84,8

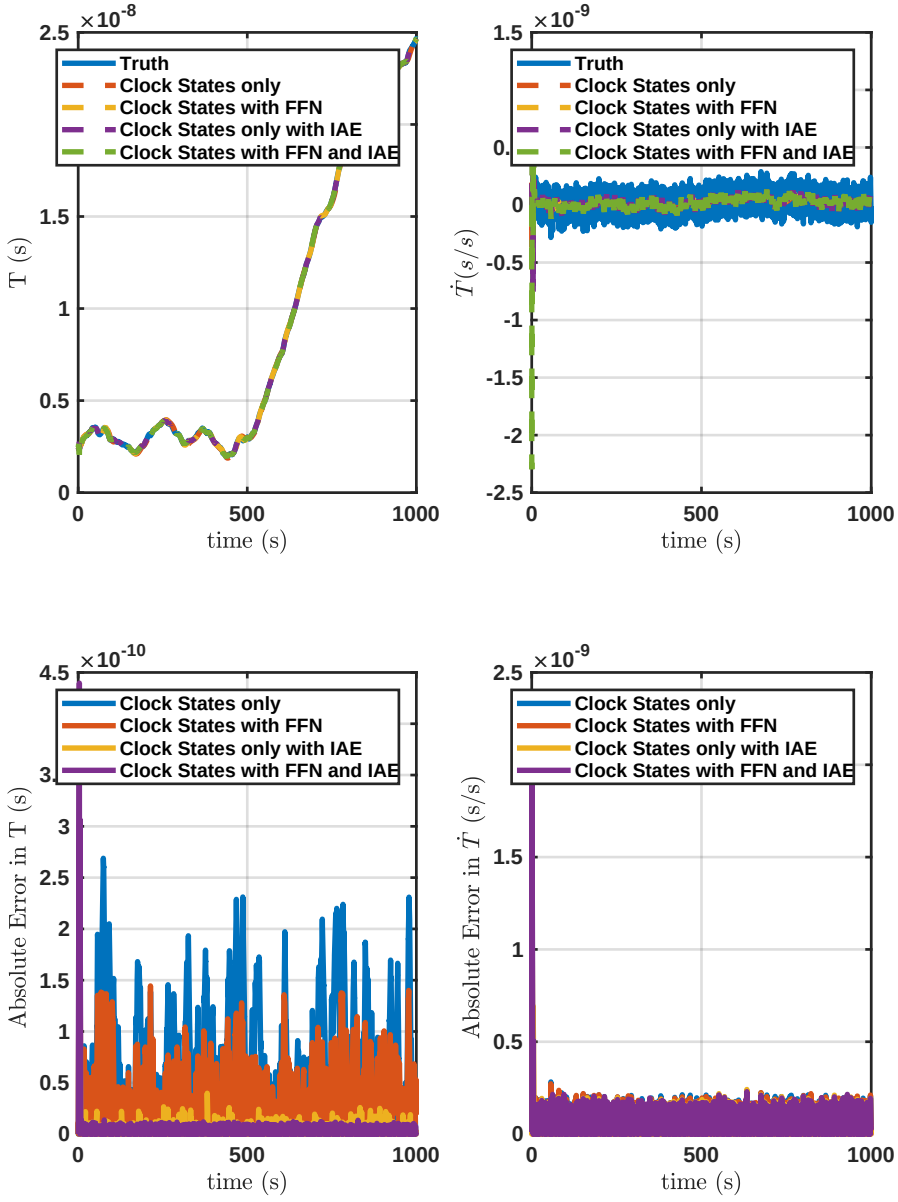

Figure B.30: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

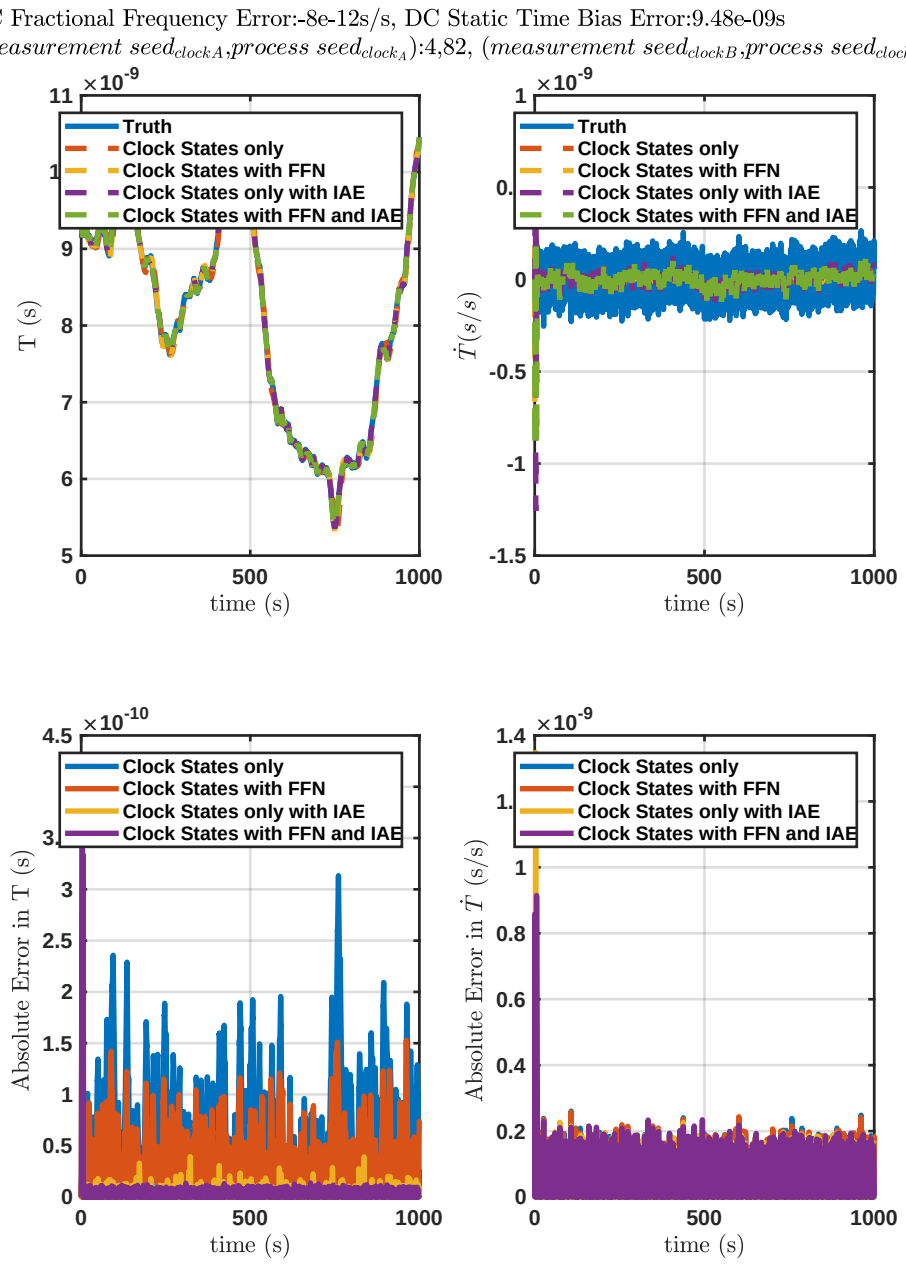

Figure B.31: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1.5e-11s/s, DC Static Time Bias Error:5.33e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):20,23, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):65,46

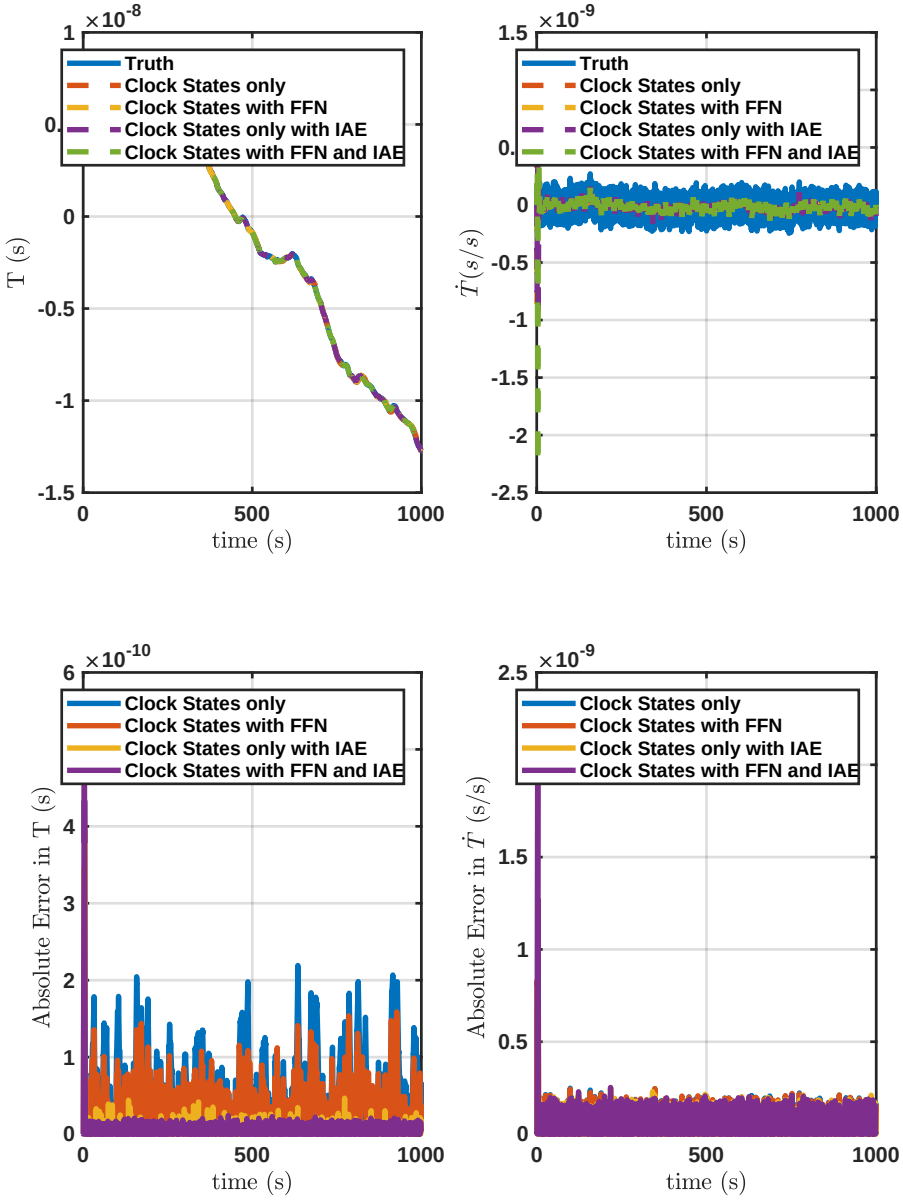

Figure B.32: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2.9e-11s/s, DC Static Time Bias Error:2e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):12,72, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):69,4

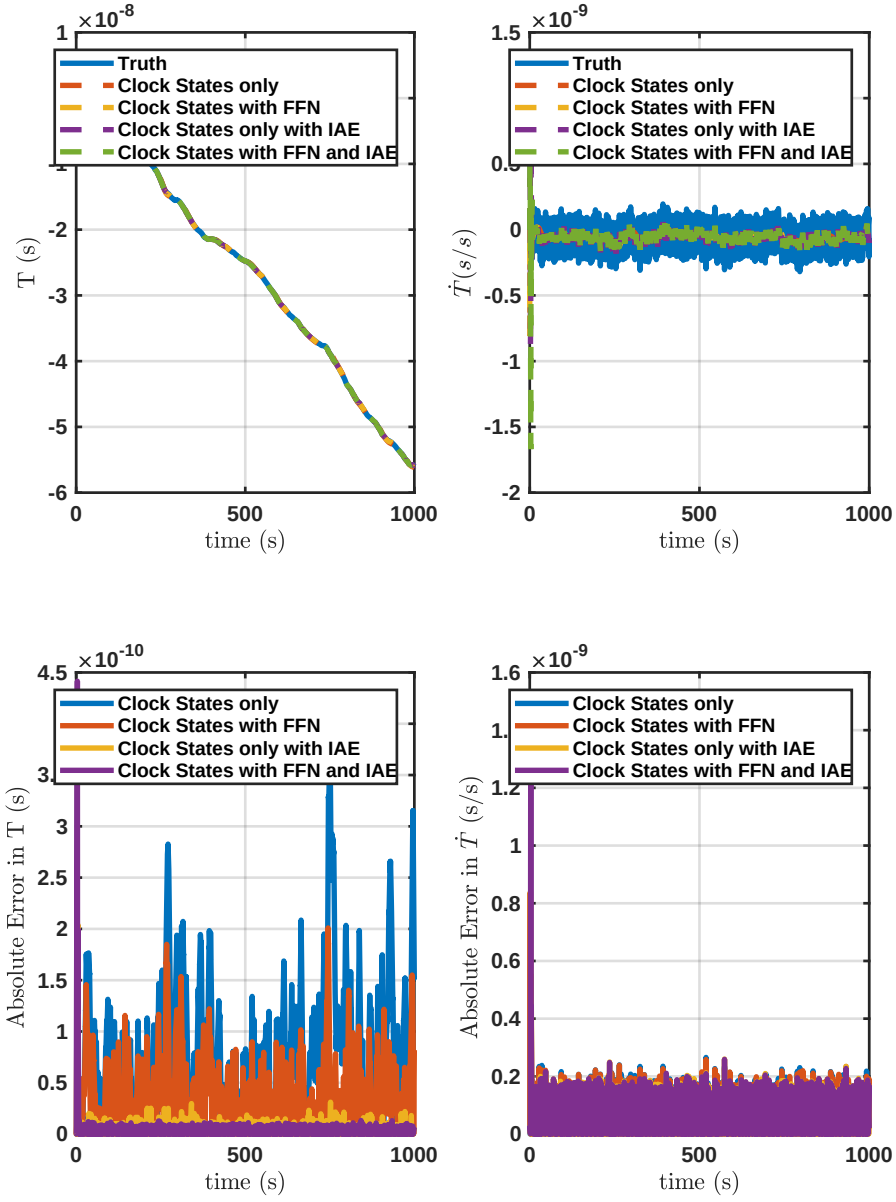

Figure B.33: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:0s/s, DC Static Time Bias Error:9.73e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):34,86, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):48,61

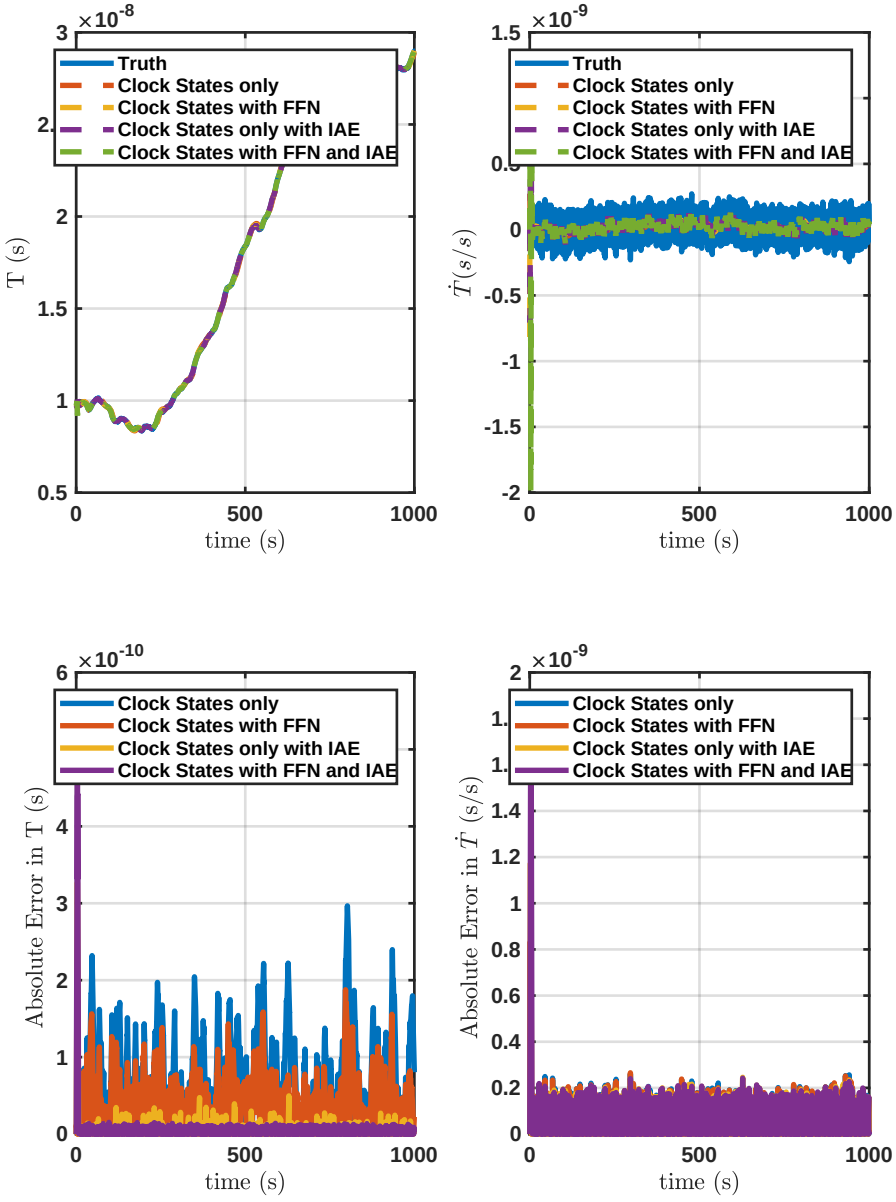

Figure B.34: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2.9e-11s/s, DC Static Time Bias Error:5.79e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):96,73, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):75,13

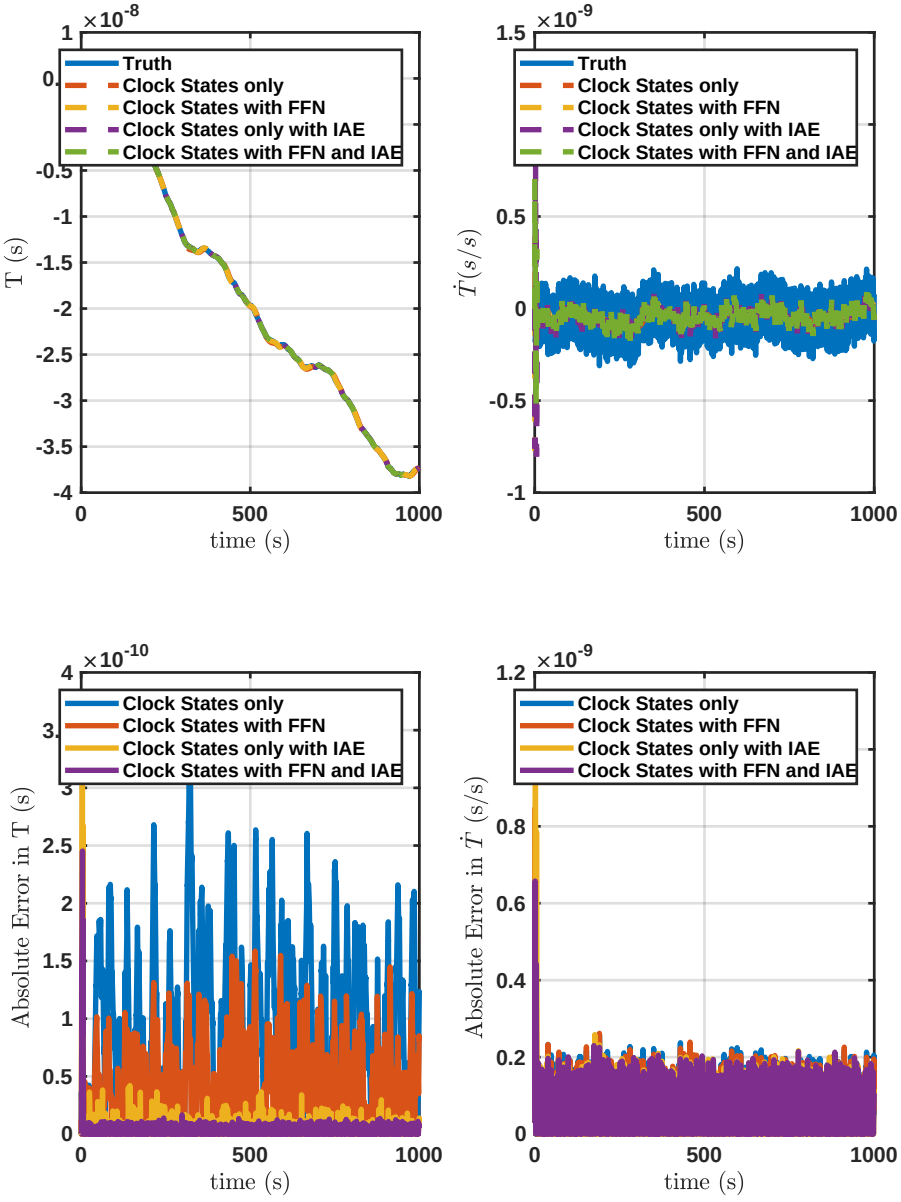

Figure B.35: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

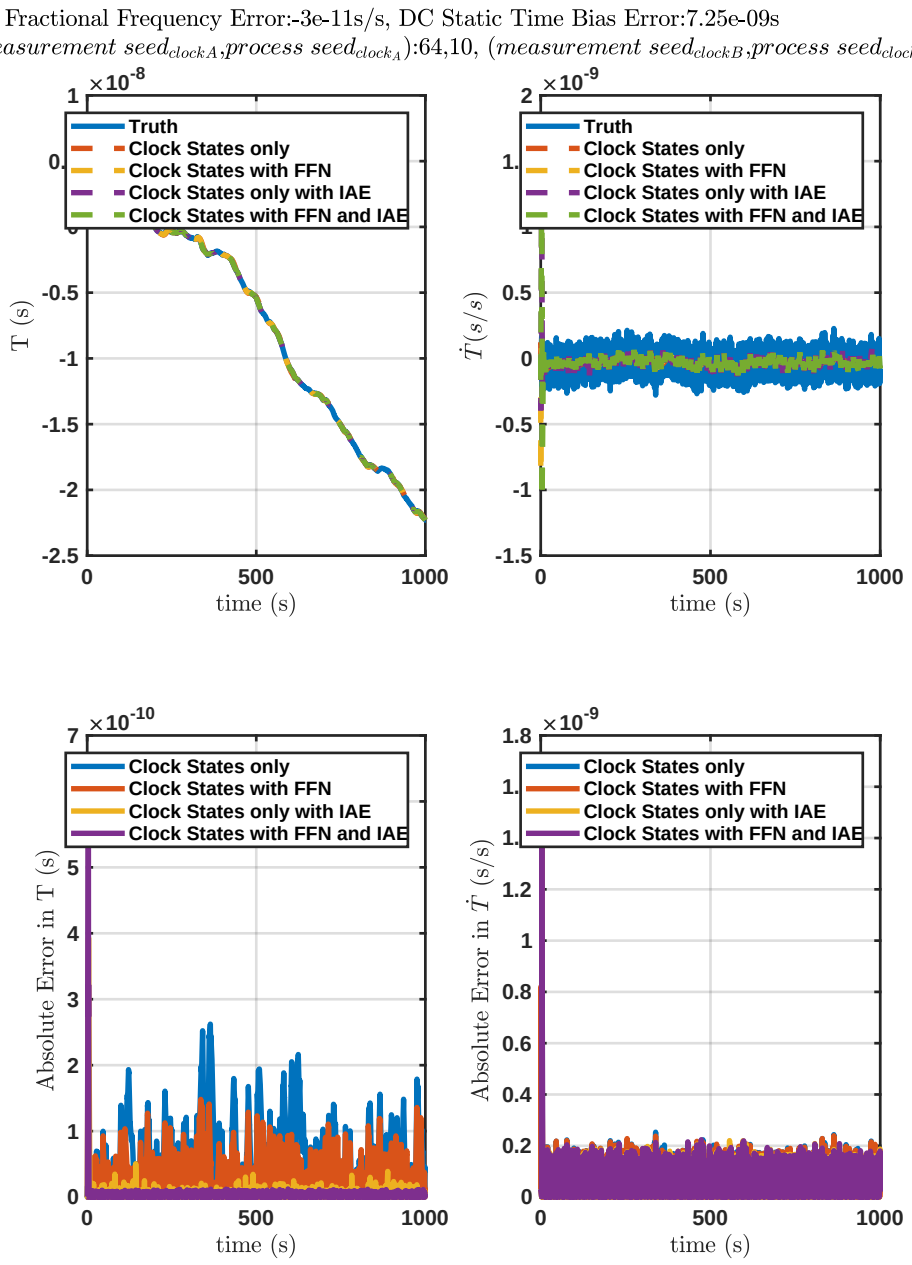

Figure B.36: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:1.2e-11s/s, DC Static Time Bias Error:7.6e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clockA}$):8,30, ($measurement\ seed_{clockB}$,$process\ seed_{clockB}$):88,41

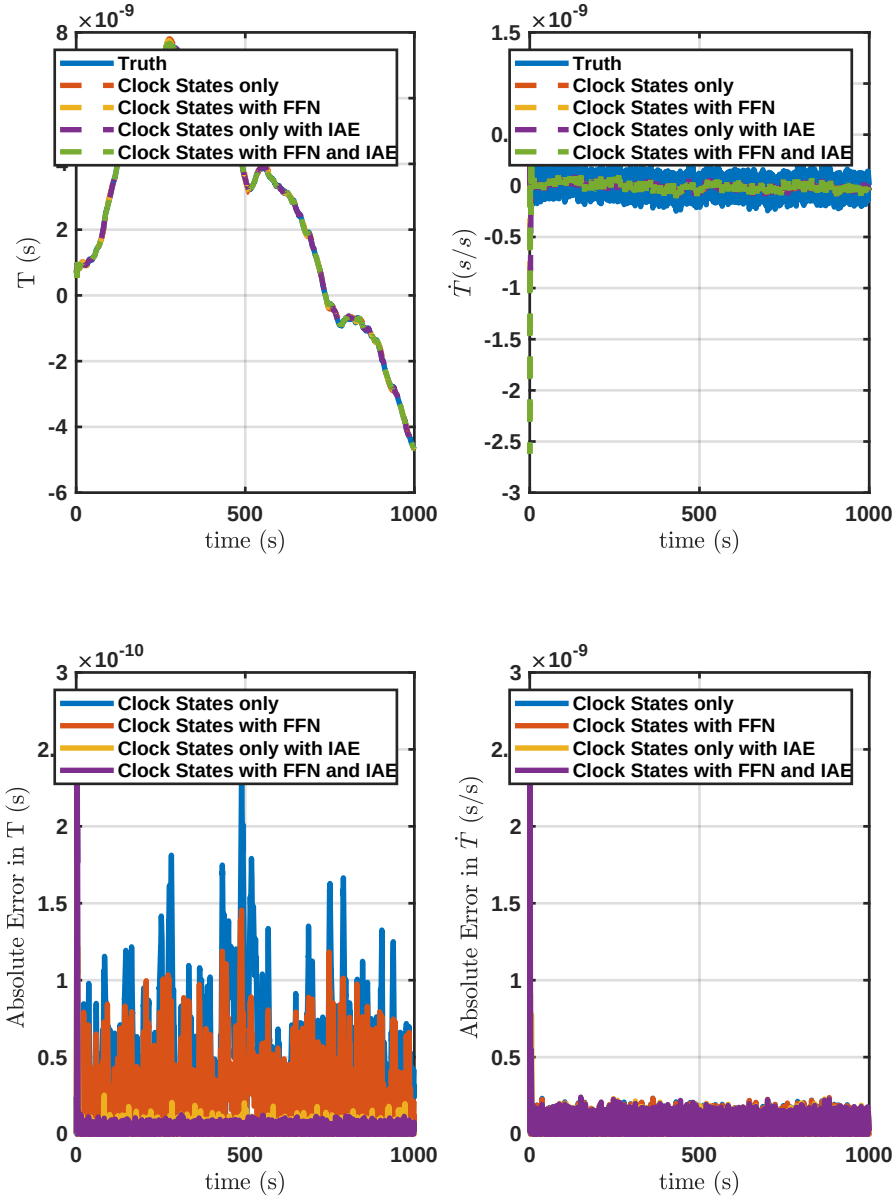

Figure B.37: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:3.1e-11s/s, DC Static Time Bias Error:9.18e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):37,14, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):44,37

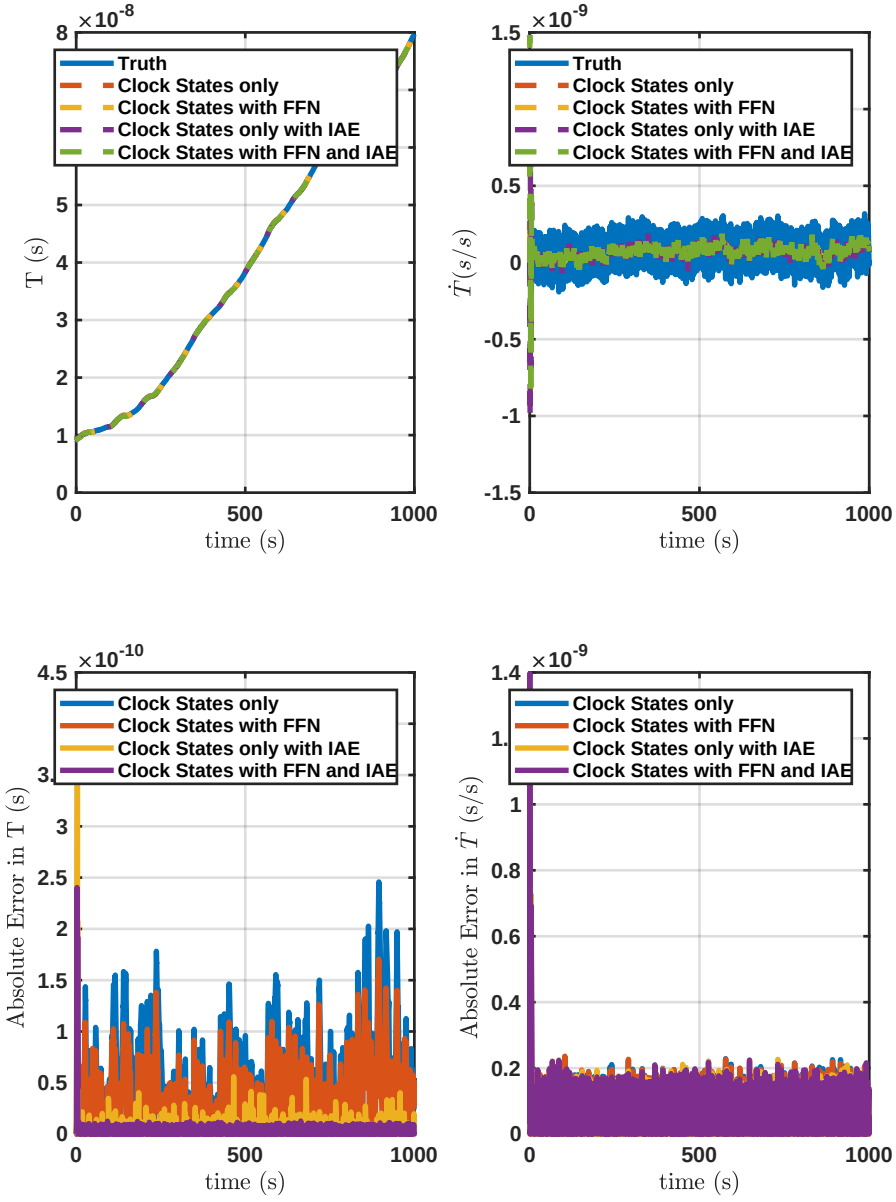

Figure B.38: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2.3e-11s/s, DC Static Time Bias Error:5.5e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):84,49, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):18,10

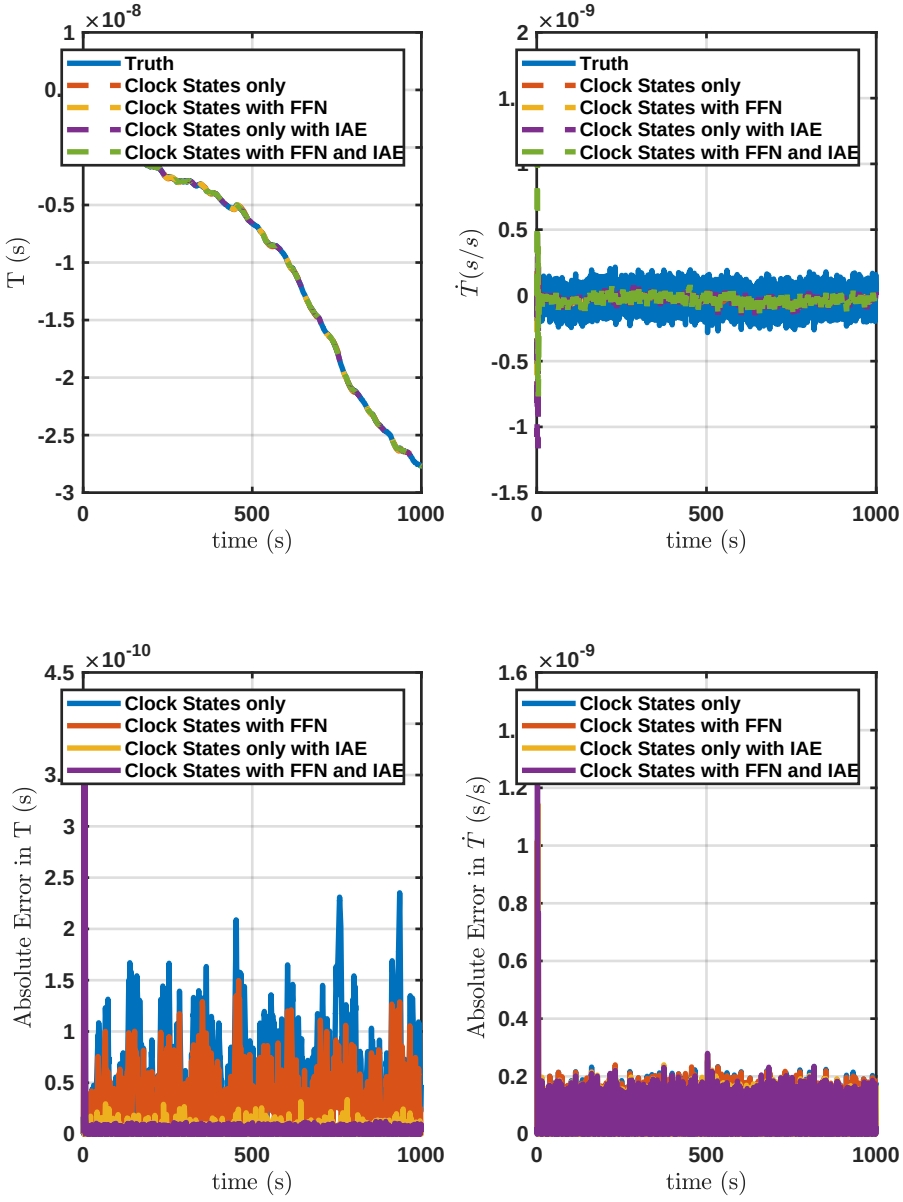

Figure B.39: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1.1e-11s/s, DC Static Time Bias Error:2.43e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):83,84, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):56,10

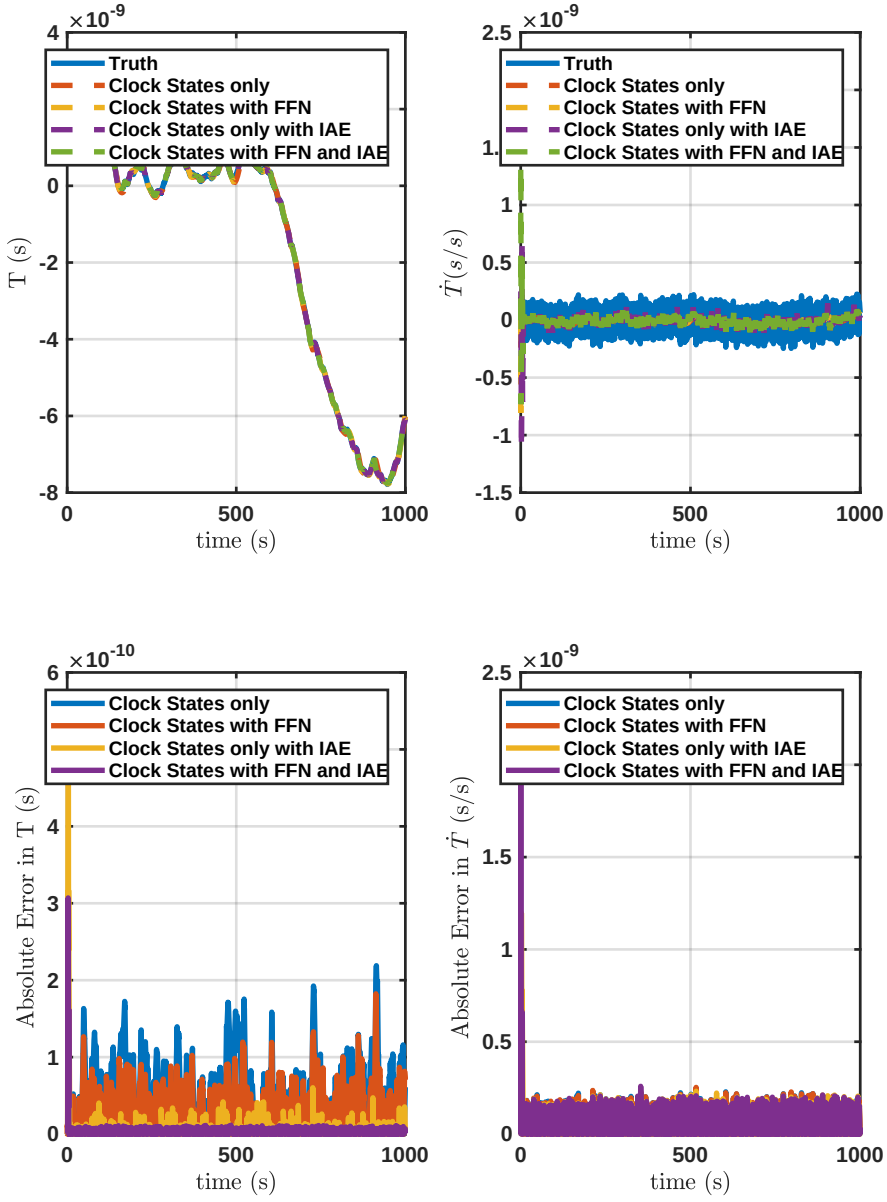

Figure B.40: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1.5e-11s/s, DC Static Time Bias Error:6.65e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):14,100, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):4,78

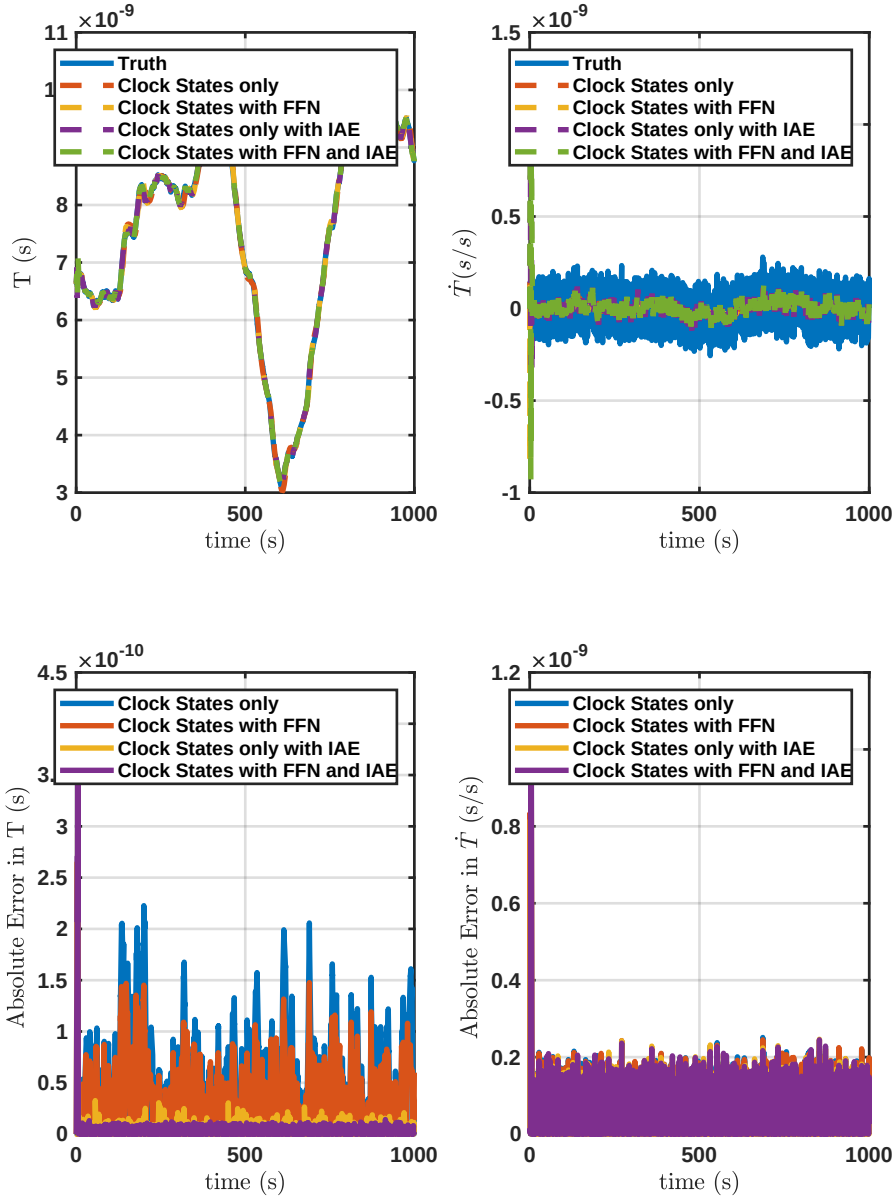

Figure B.41: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:1.8e-11s/s, DC Static Time Bias Error:3.51e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):51,11, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):99,32

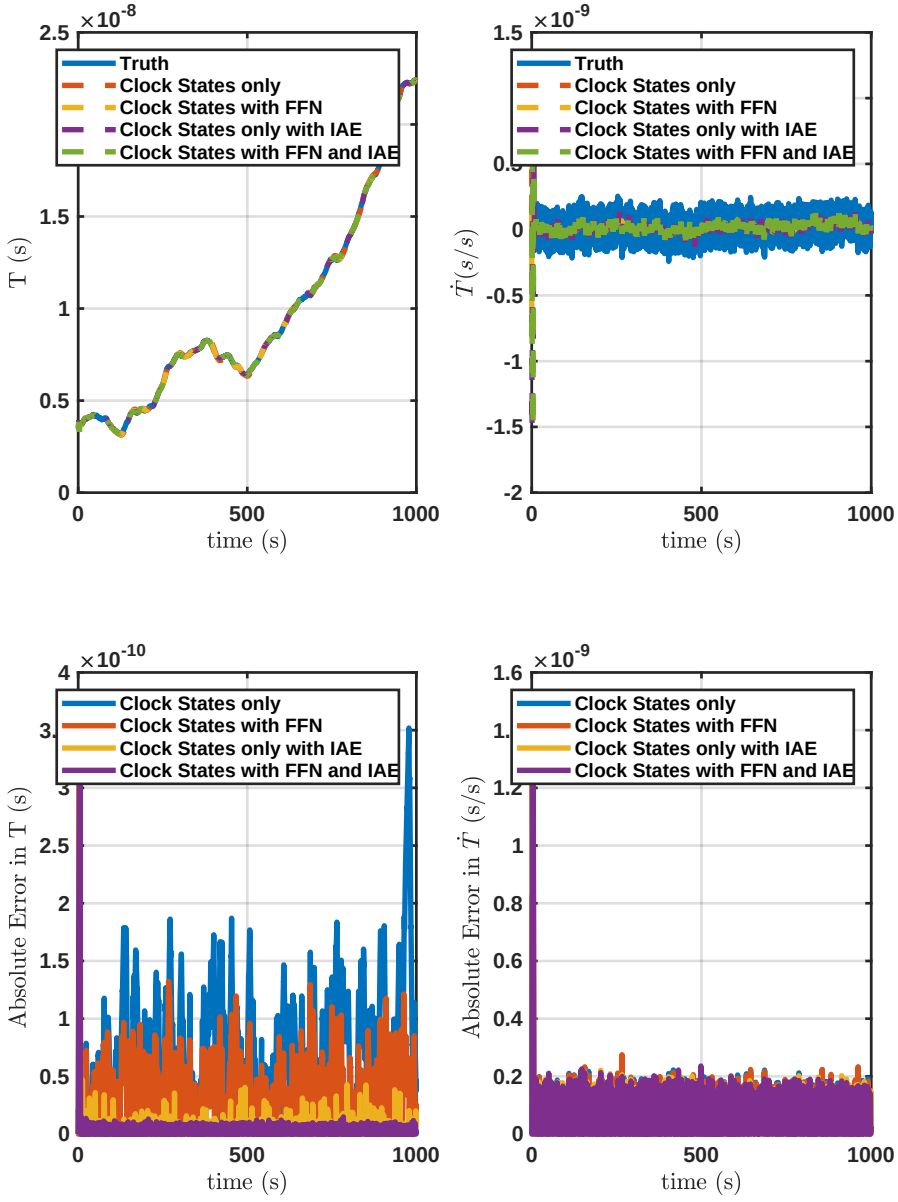

Figure B.42: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:2.4e-11s/s, DC Static Time Bias Error:9.67e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):33,58, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):17,27

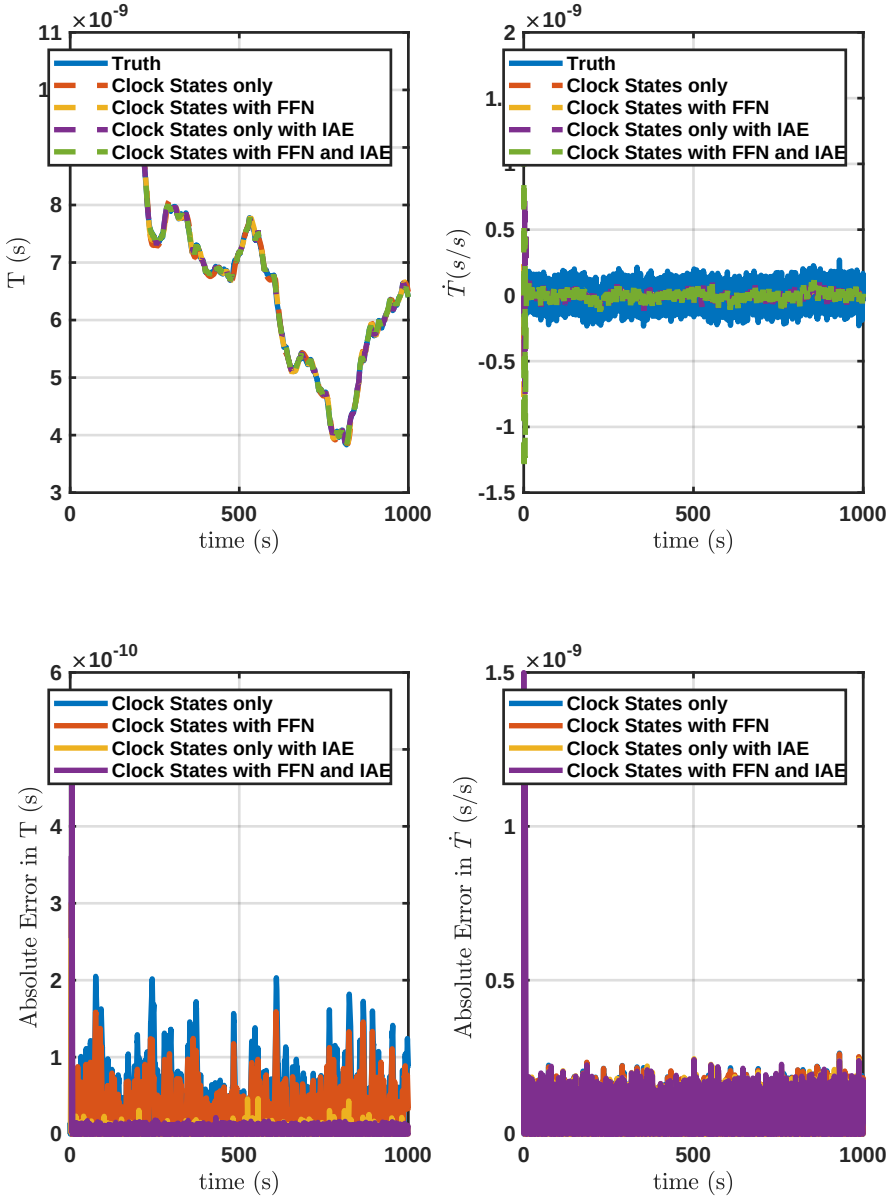

Figure B.43: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:3.6e-11s/s, DC Static Time Bias Error:9.23e-09s
($measurement\ seed_{clockA}$,$process\ seed_{clockA}$):0,73, ($measurement\ seed_{clockB}$,$process\ seed_{clockB}$):68,77

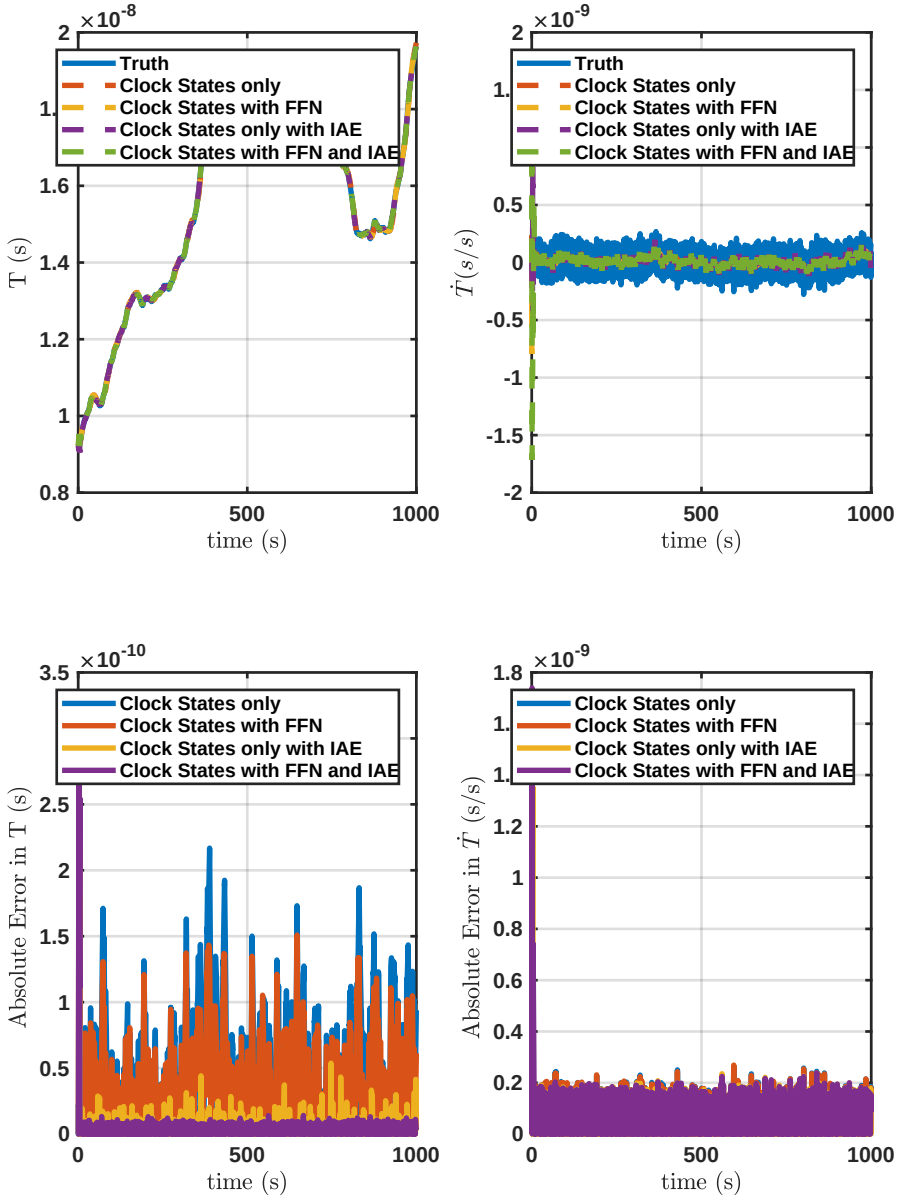

Figure B.44: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2.6e-11s/s, DC Static Time Bias Error:2.43e-09s
($measurement\ seed_{clockA}$,$process\ seed_{clockA}$):1,80, ($measurement\ seed_{clockB}$,$process\ seed_{clockB}$):56,28

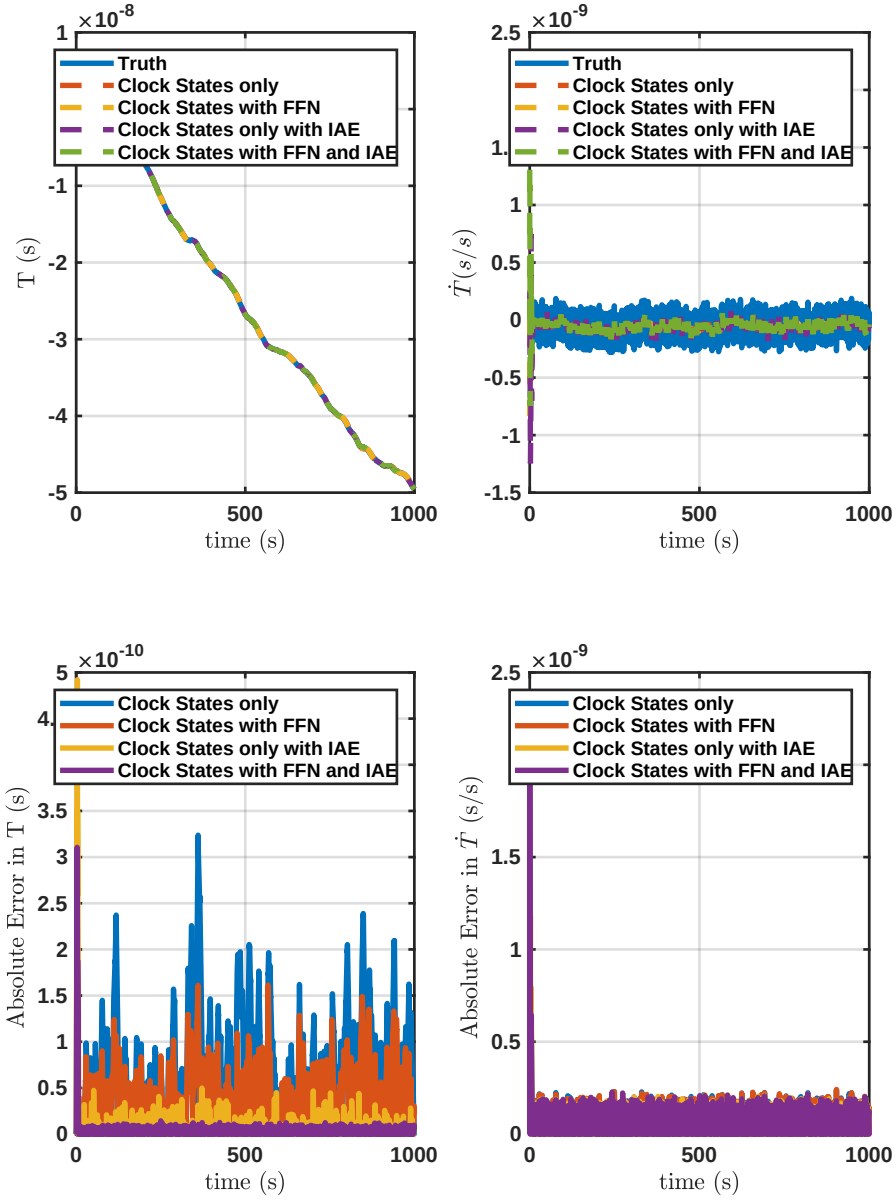

Figure B.45: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

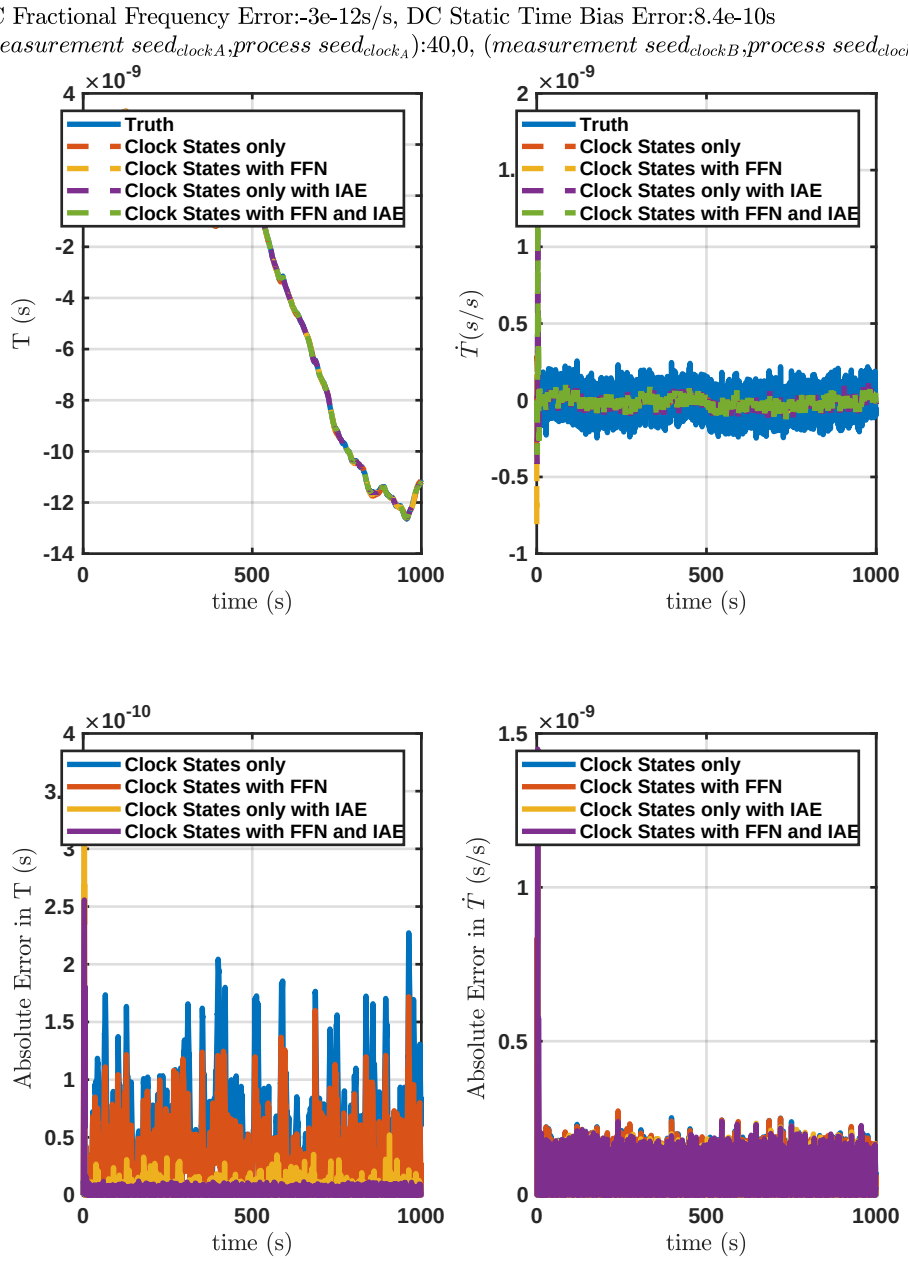

Figure B.46: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

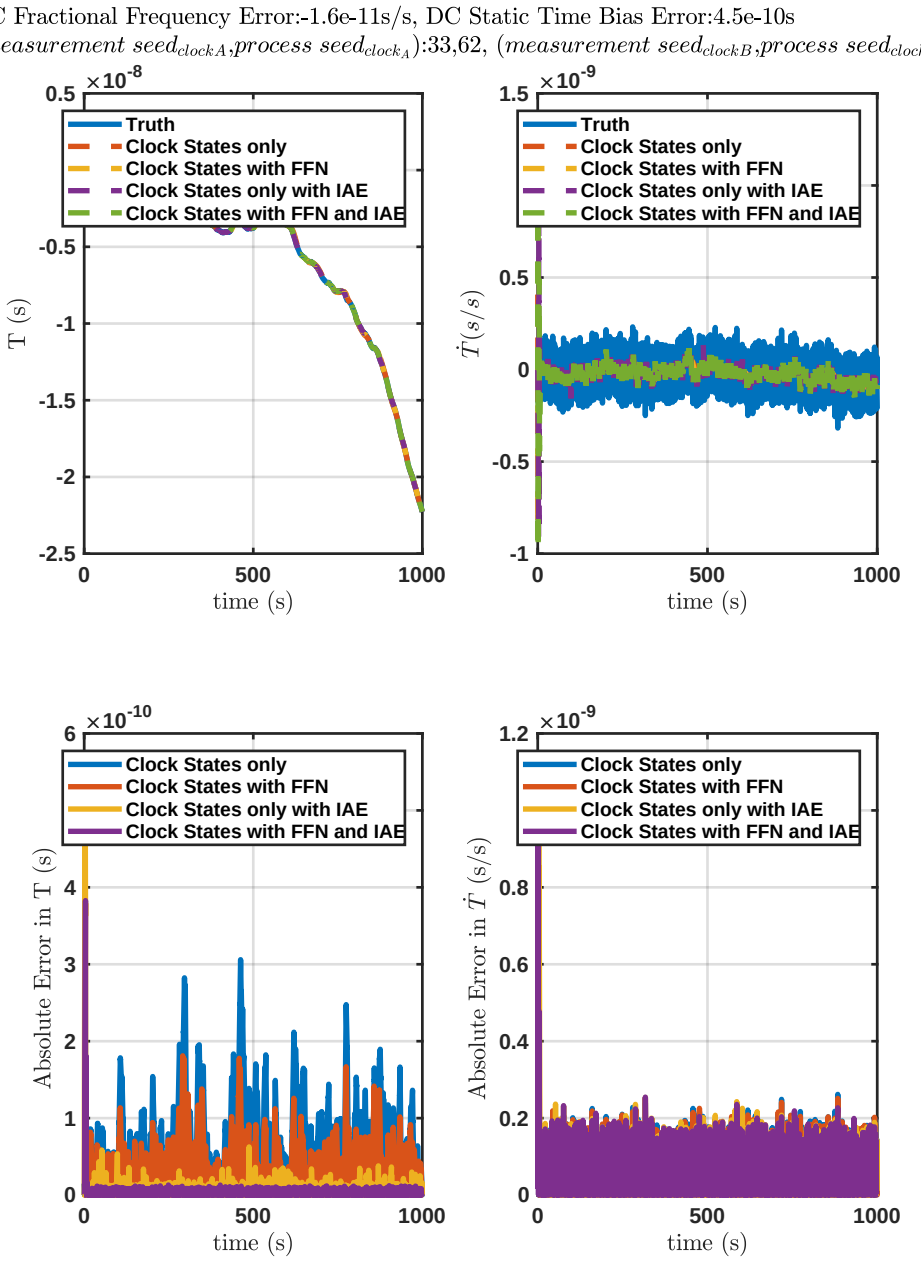

Figure B.47: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-7e-12s/s, DC Static Time Bias Error:9.48e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):35,33, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):98,19

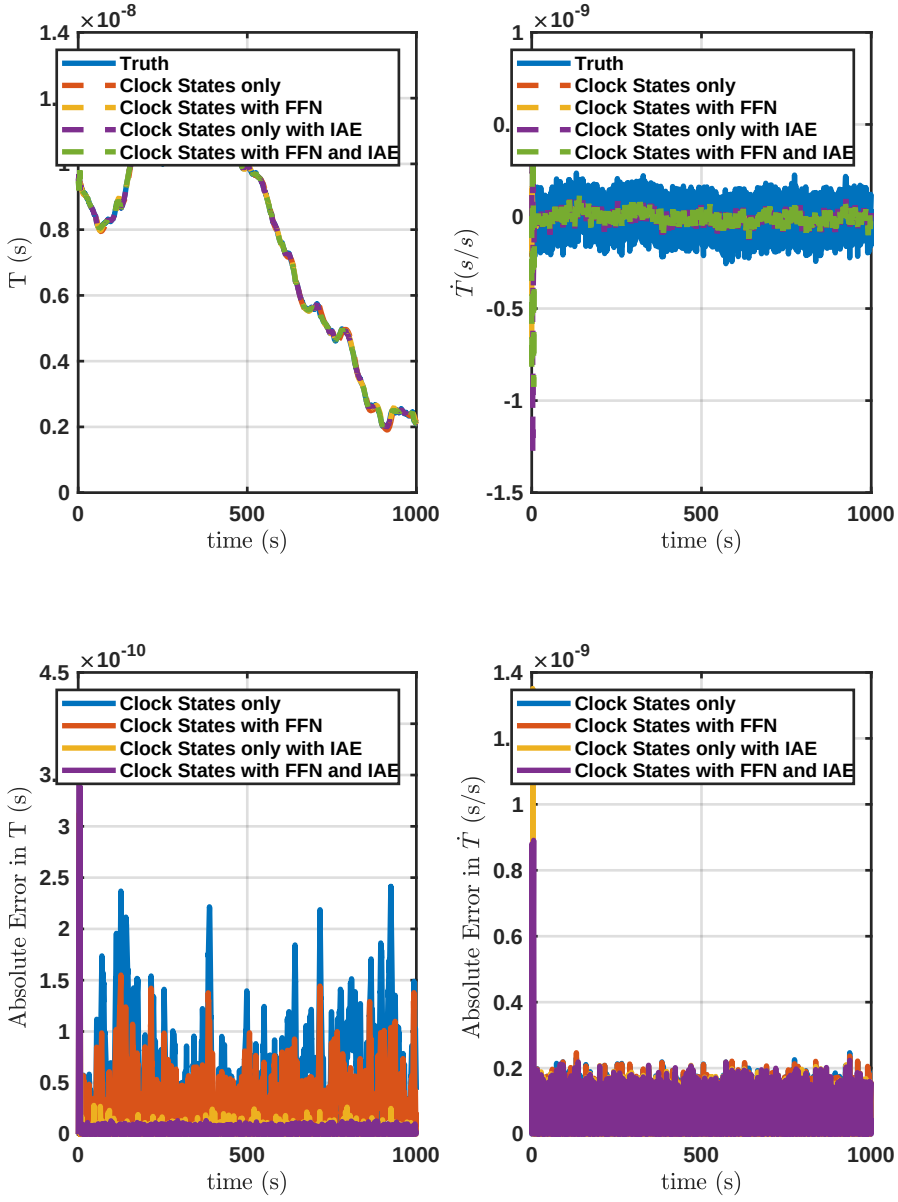

Figure B.48: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8e-12s/s, DC Static Time Bias Error:1e-08s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):10,99, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):49,1

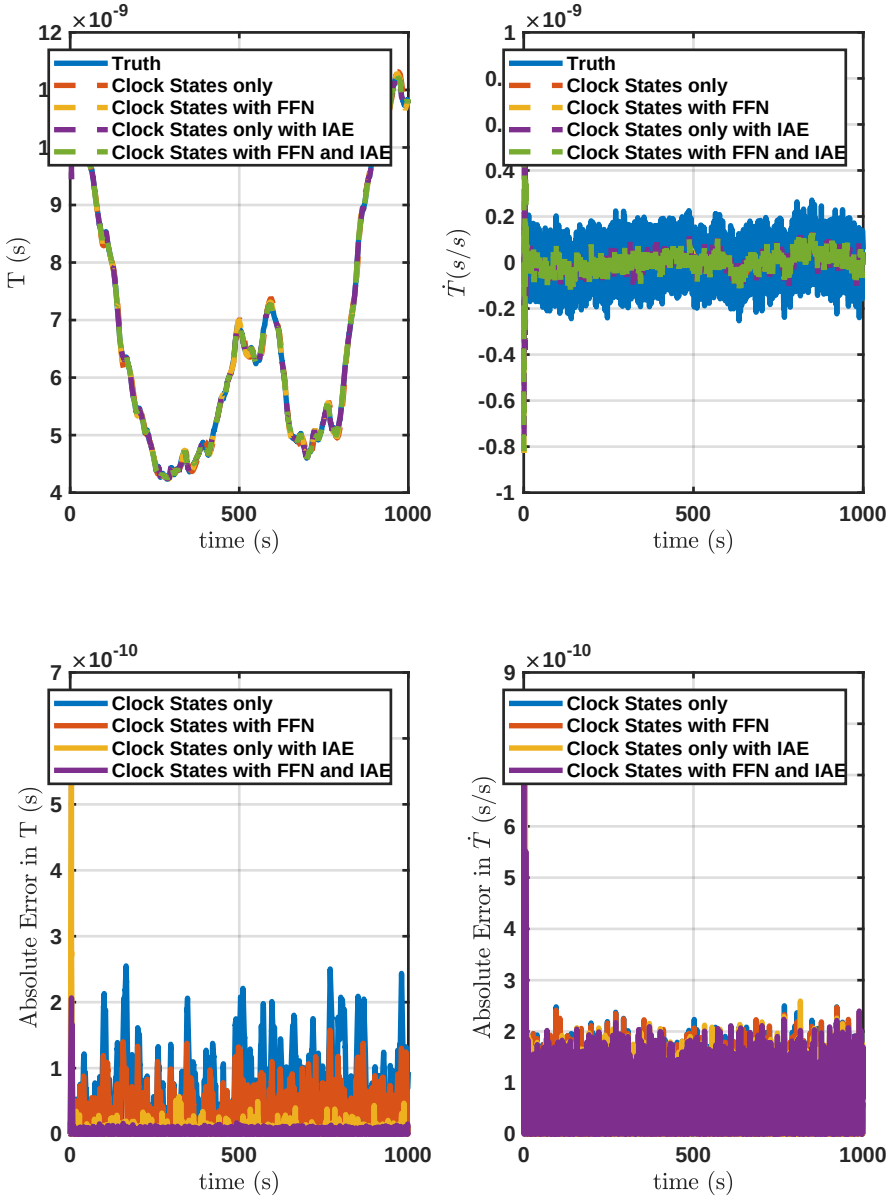

Figure B.49: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1e-11s/s, DC Static Time Bias Error:5.65e-09s
($measurement\ seed_{clockA}$,$process\ seed_{clockA}$):11,4, ($measurement\ seed_{clockB}$,$process\ seed_{clockB}$):54,95

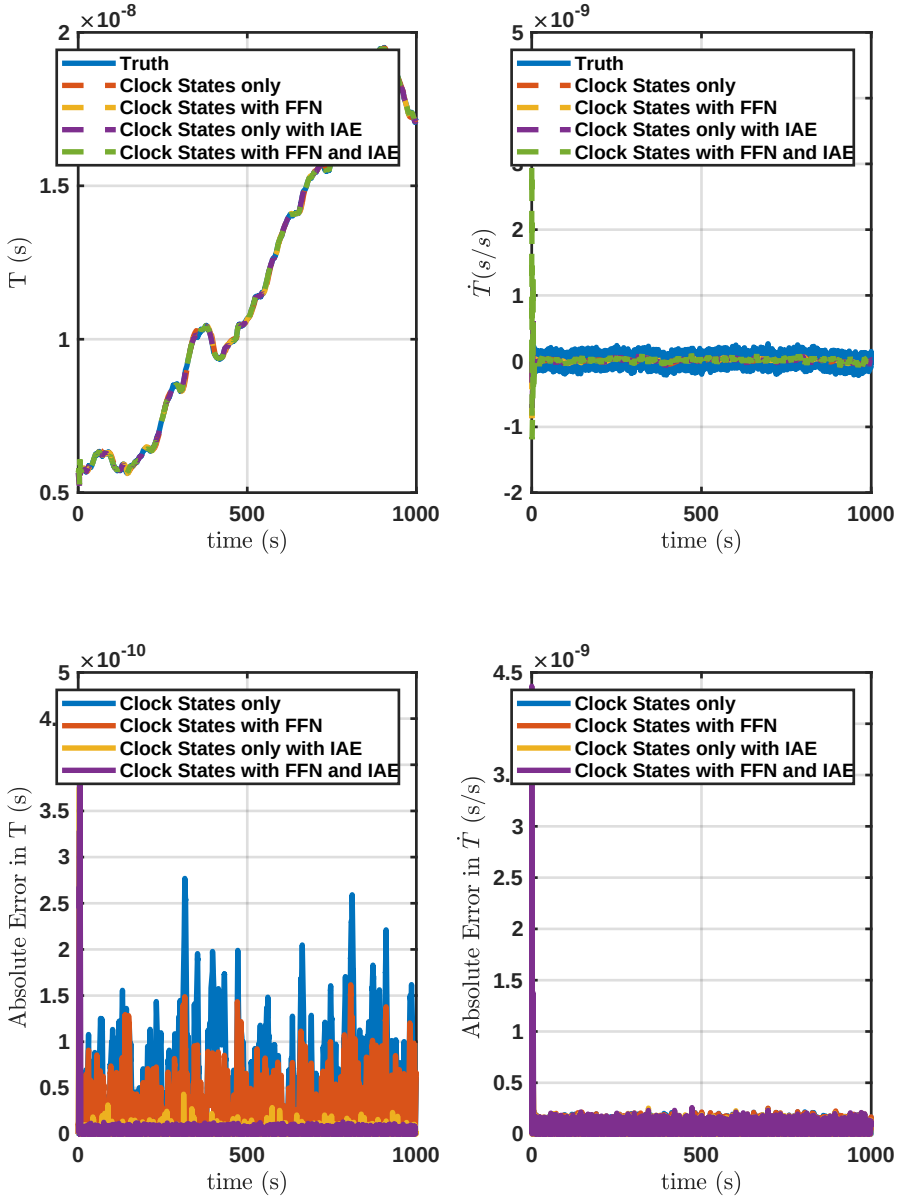

Figure B.50: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-1.7e-11s/s, DC Static Time Bias Error:5.5e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):66,47, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):18,94

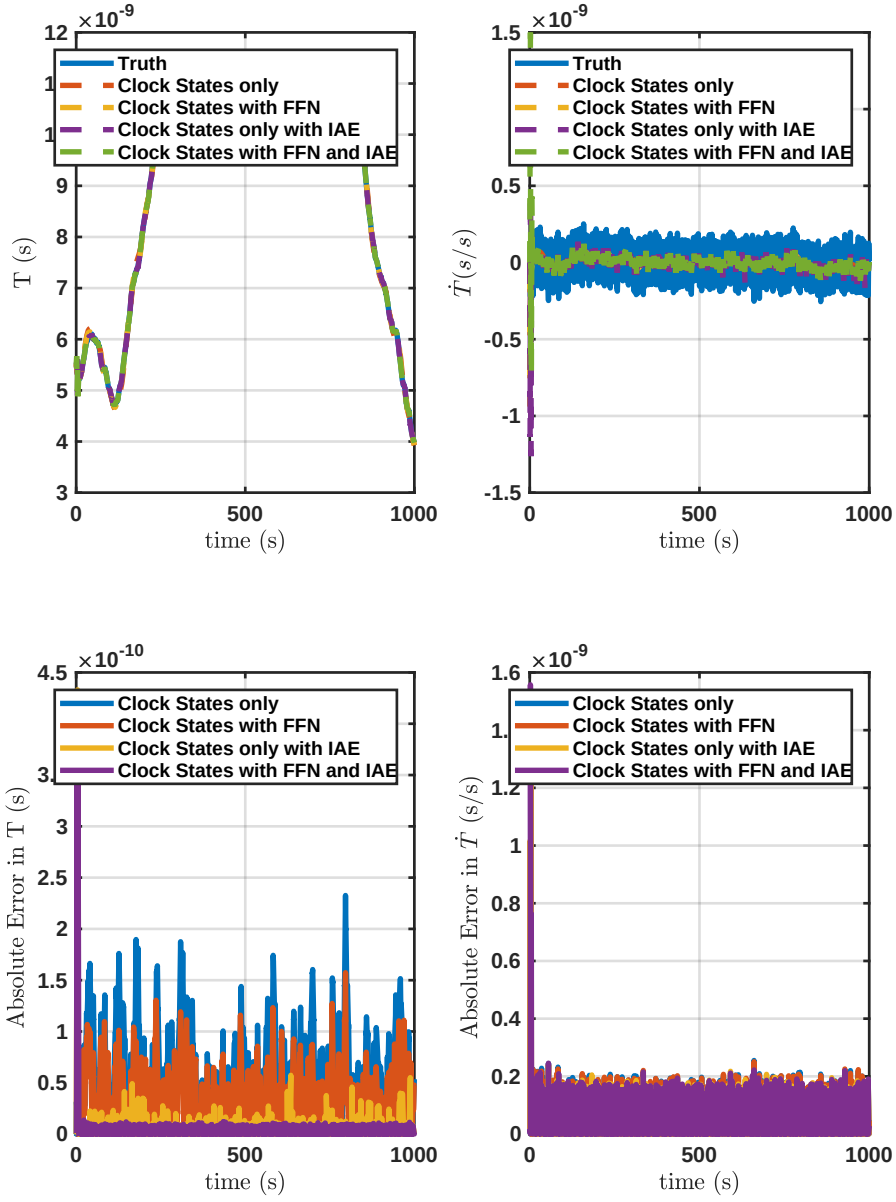

Figure B.51: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2e-12s/s, DC Static Time Bias Error:8.4e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):87,82, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):43,77

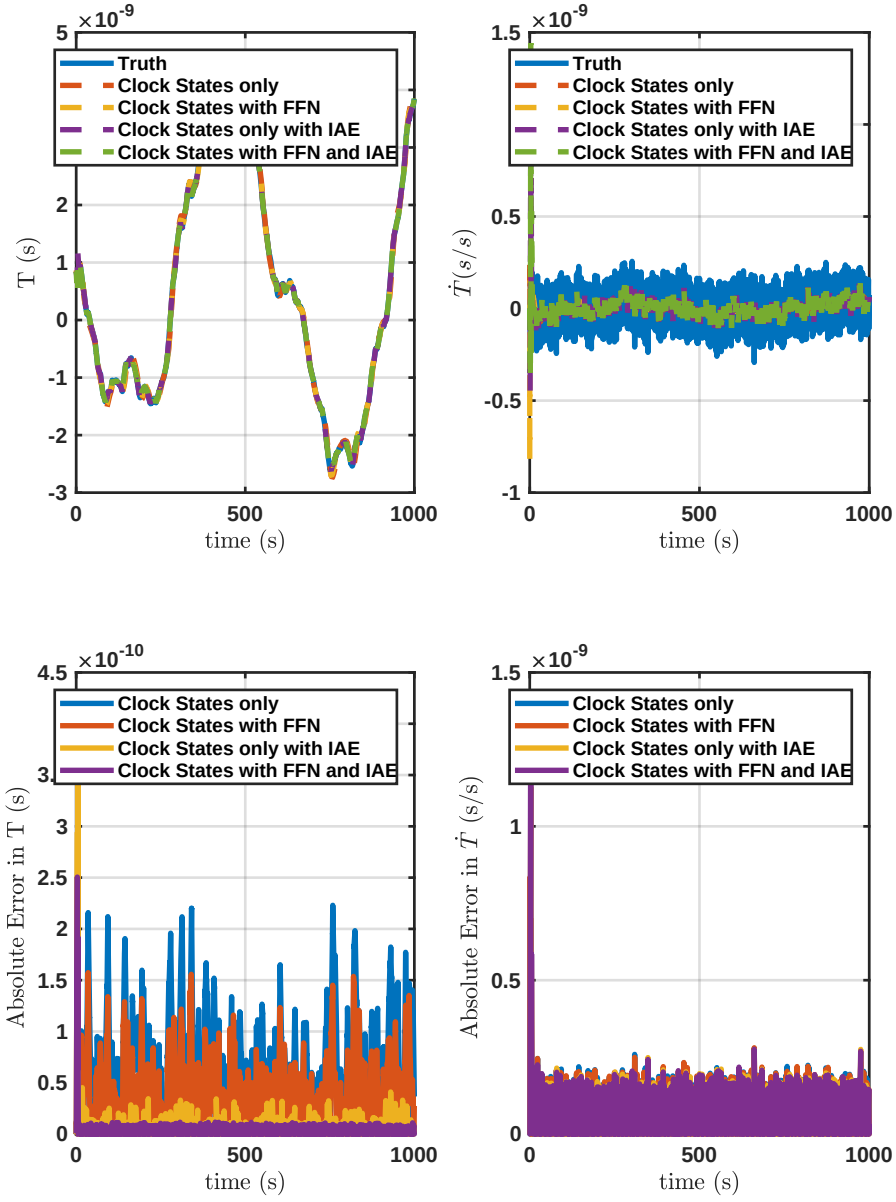

Figure B.52: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8e-12s/s, DC Static Time Bias Error:6.66e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):40,47, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):12,14

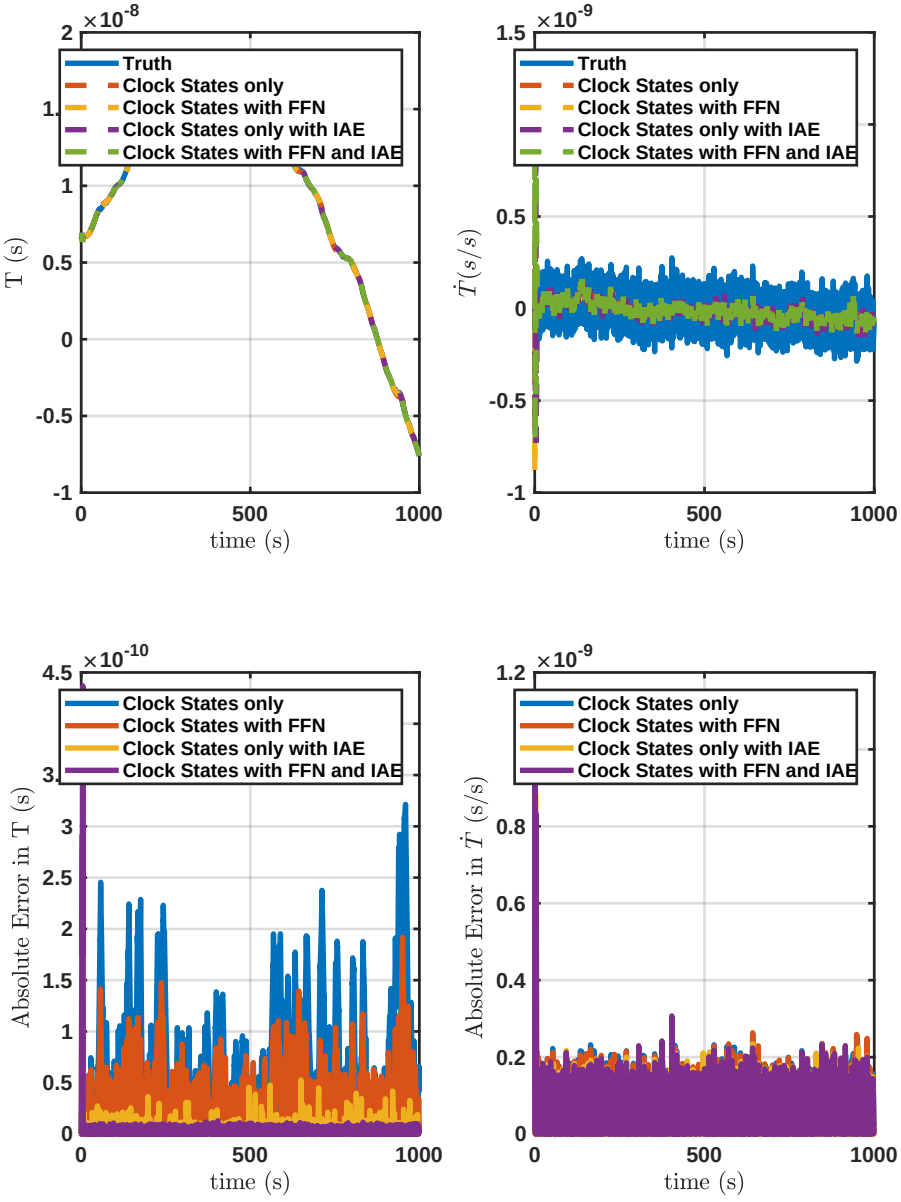

Figure B.53: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:-2.6e-11s/s, DC Static Time Bias Error:6.12e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):51,13, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):90,97

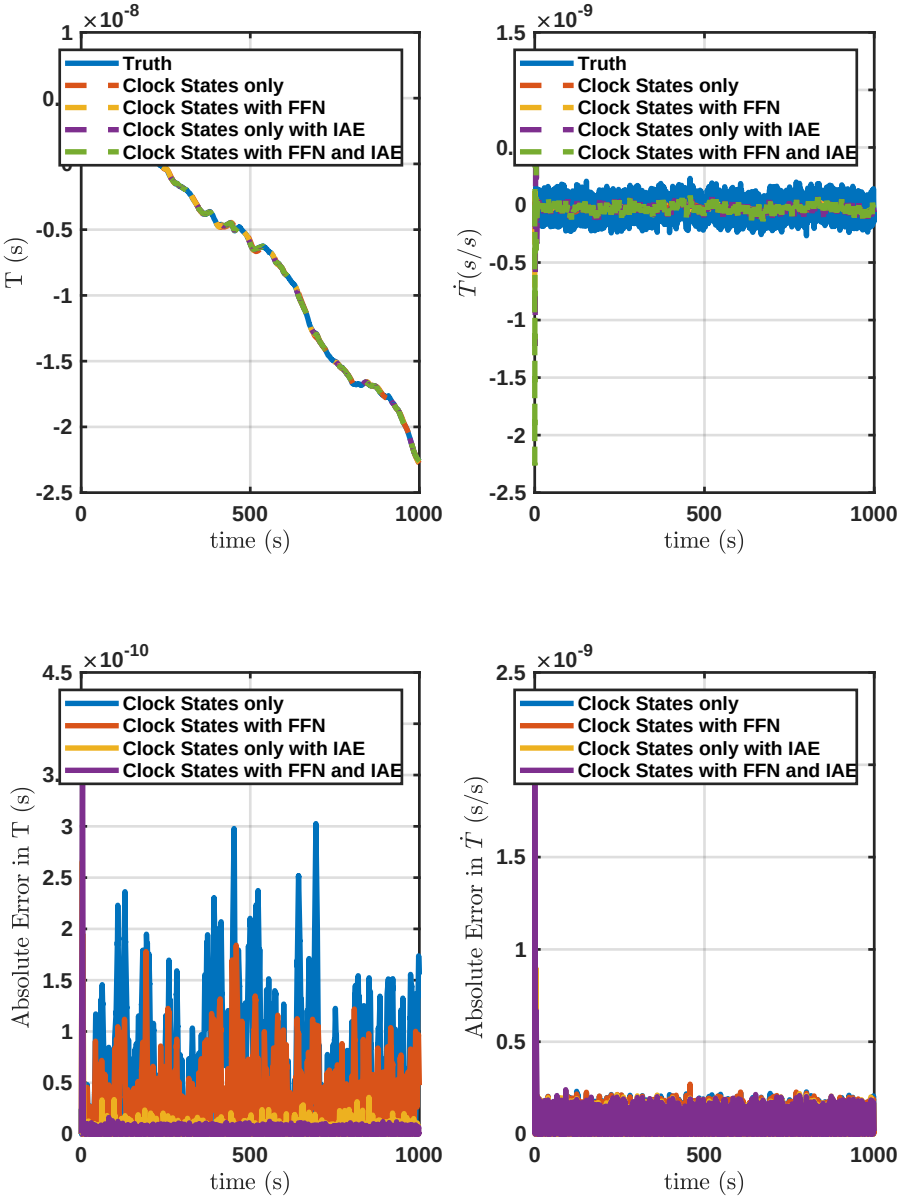

Figure B.54: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:3.9e-09s/s, DC Static Time Bias Error:3.9e-11s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):88,7, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):82,5

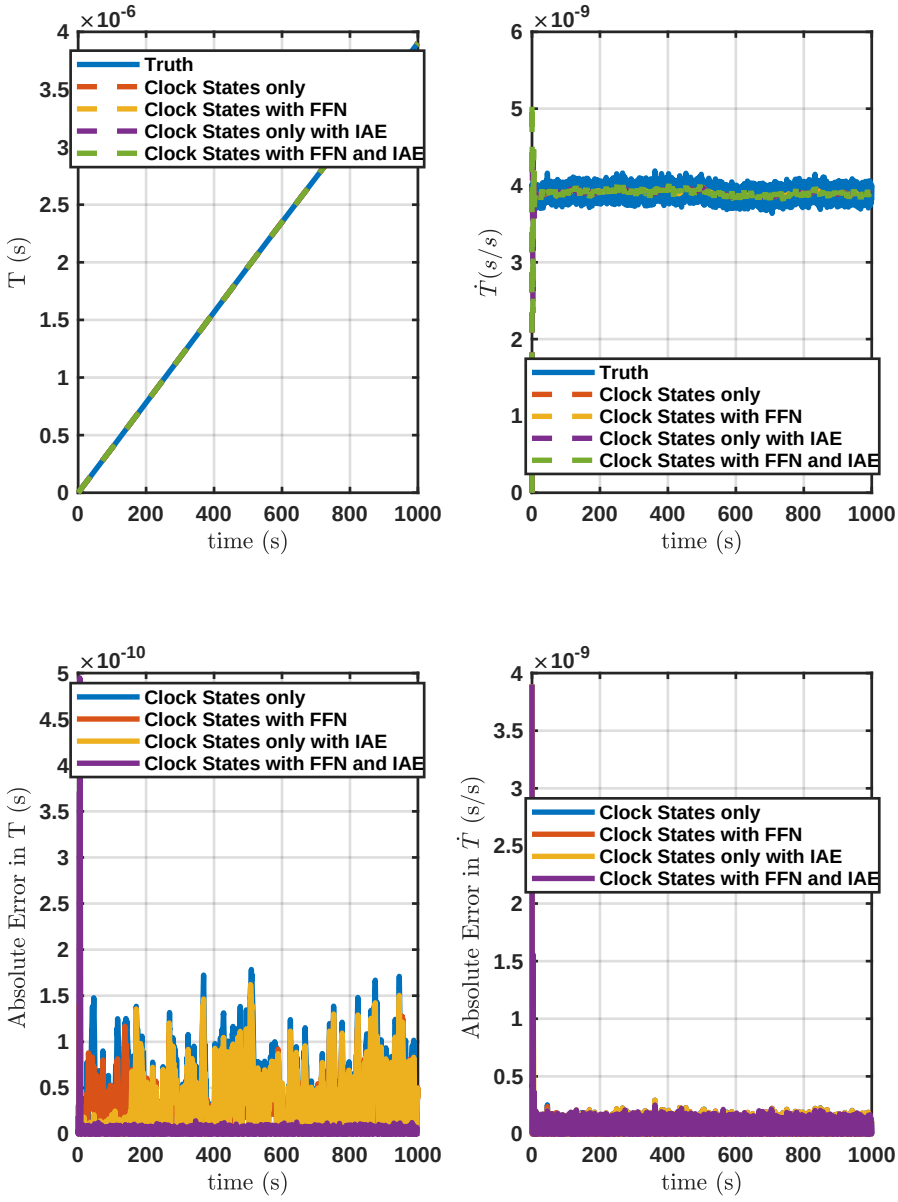

Figure B.55: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:9.77e-08s/s, DC Static Time Bias Error:9.77e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):40,44, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):19,59

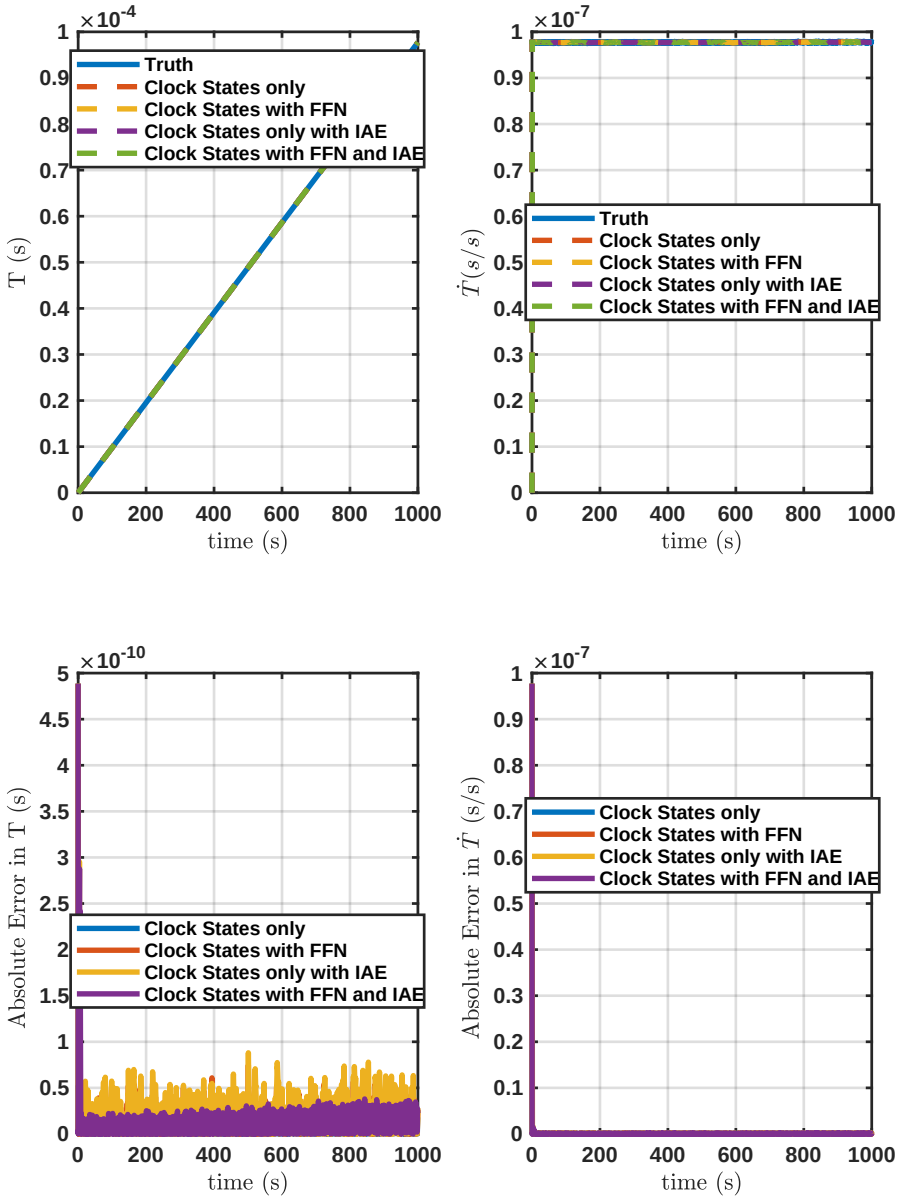

Figure B.56: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:2.8e-08s/s, DC Static Time Bias Error:2.8e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):83,47, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):11,34

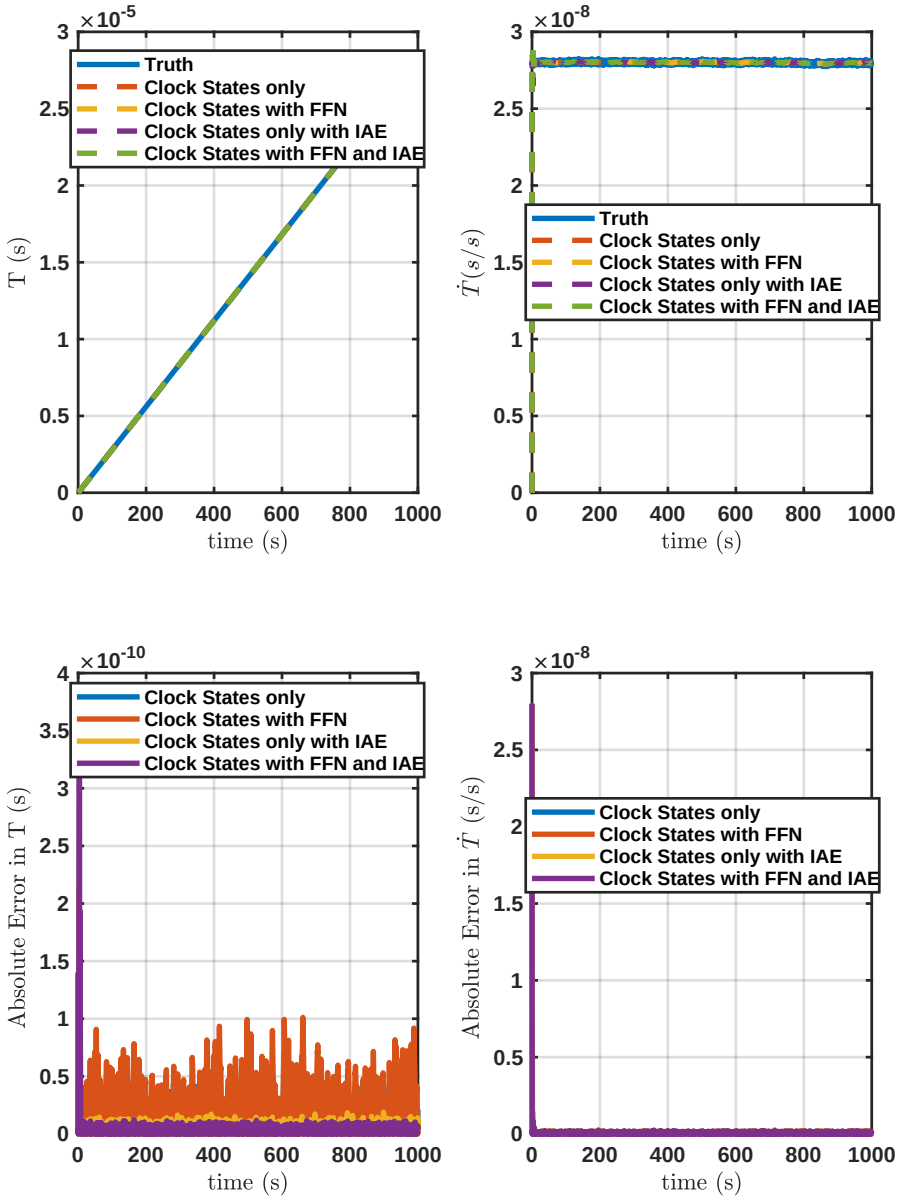

Figure B.57: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:6.21e-08s/s, DC Static Time Bias Error:6.21e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):97,46, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):0,12

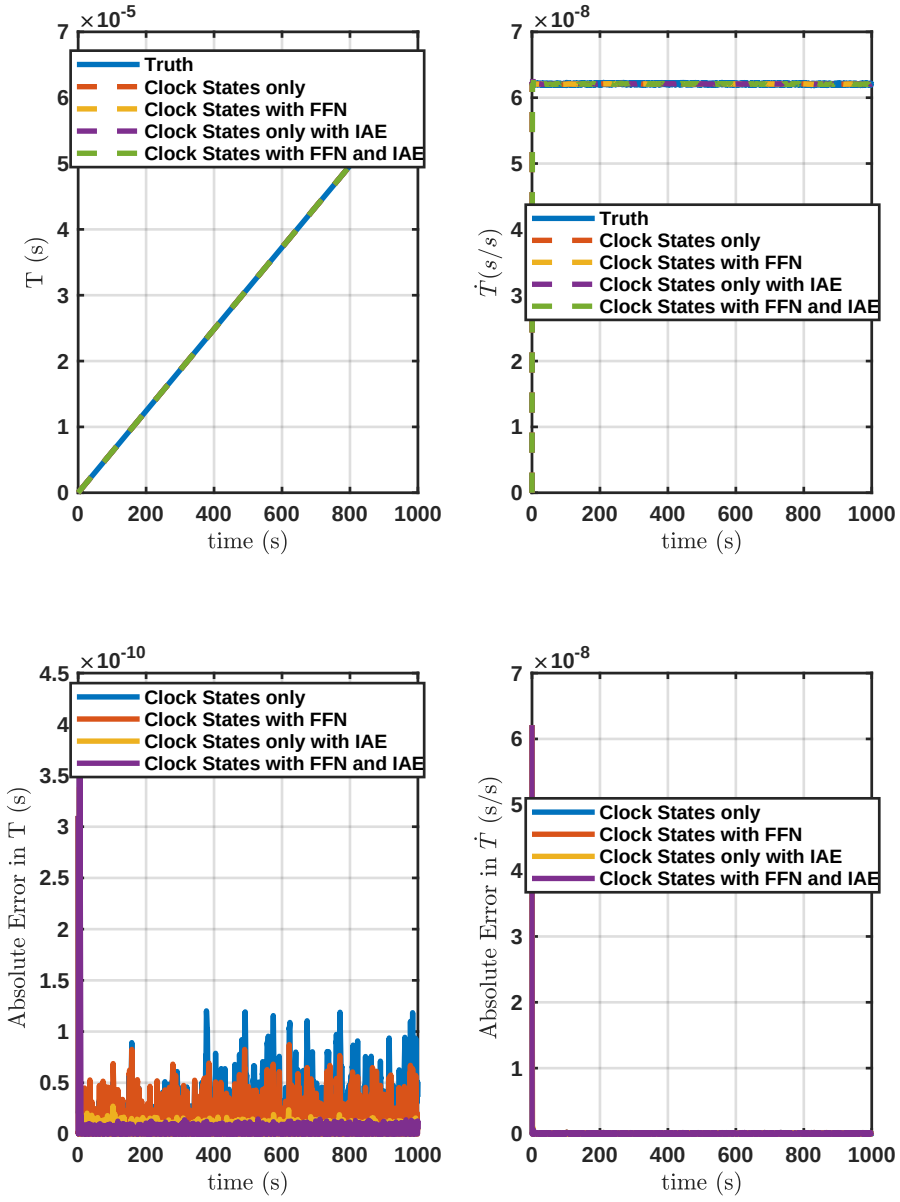

Figure B.58: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:6.89e-08s/s, DC Static Time Bias Error:6.89e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):87,14, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):38,47

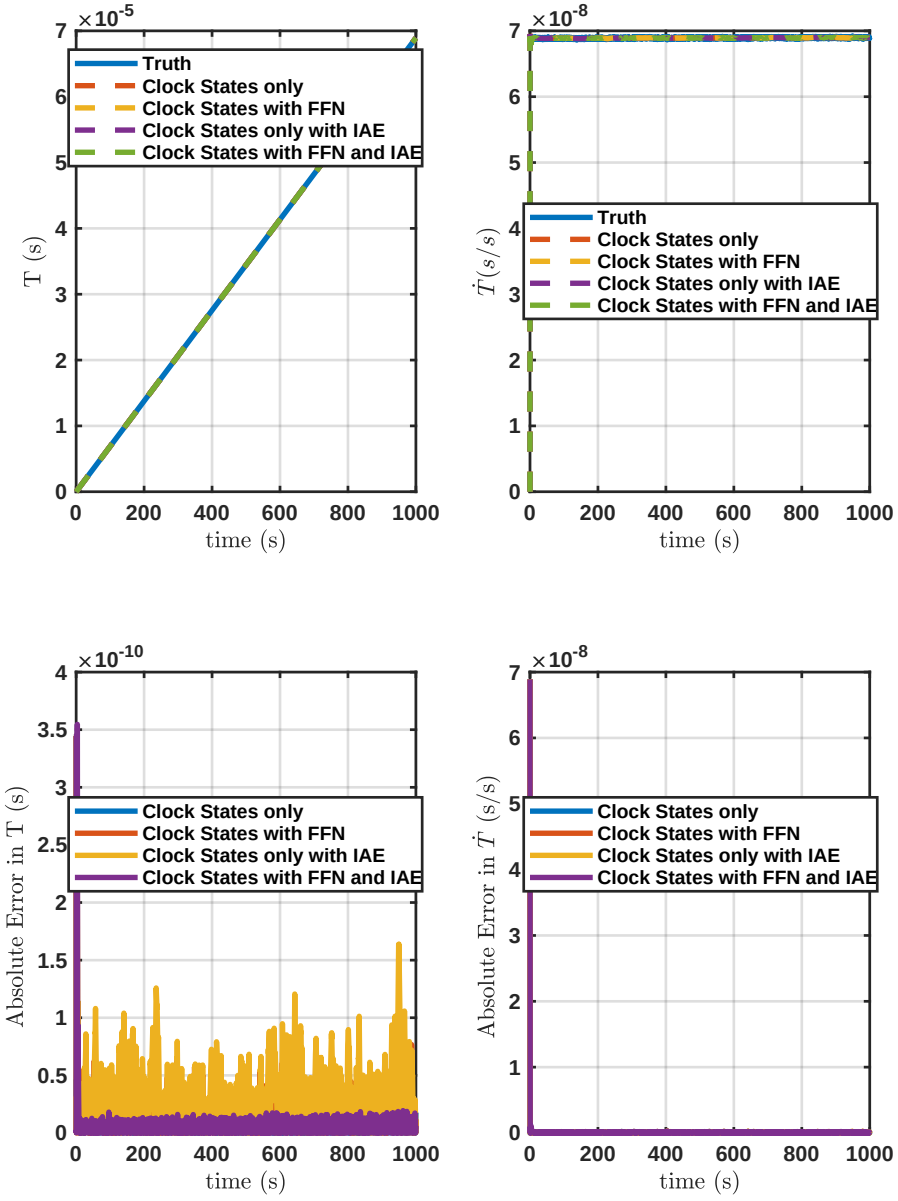

Figure B.59: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8.37e-08s/s, DC Static Time Bias Error:8.37e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):1,75, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):88,85

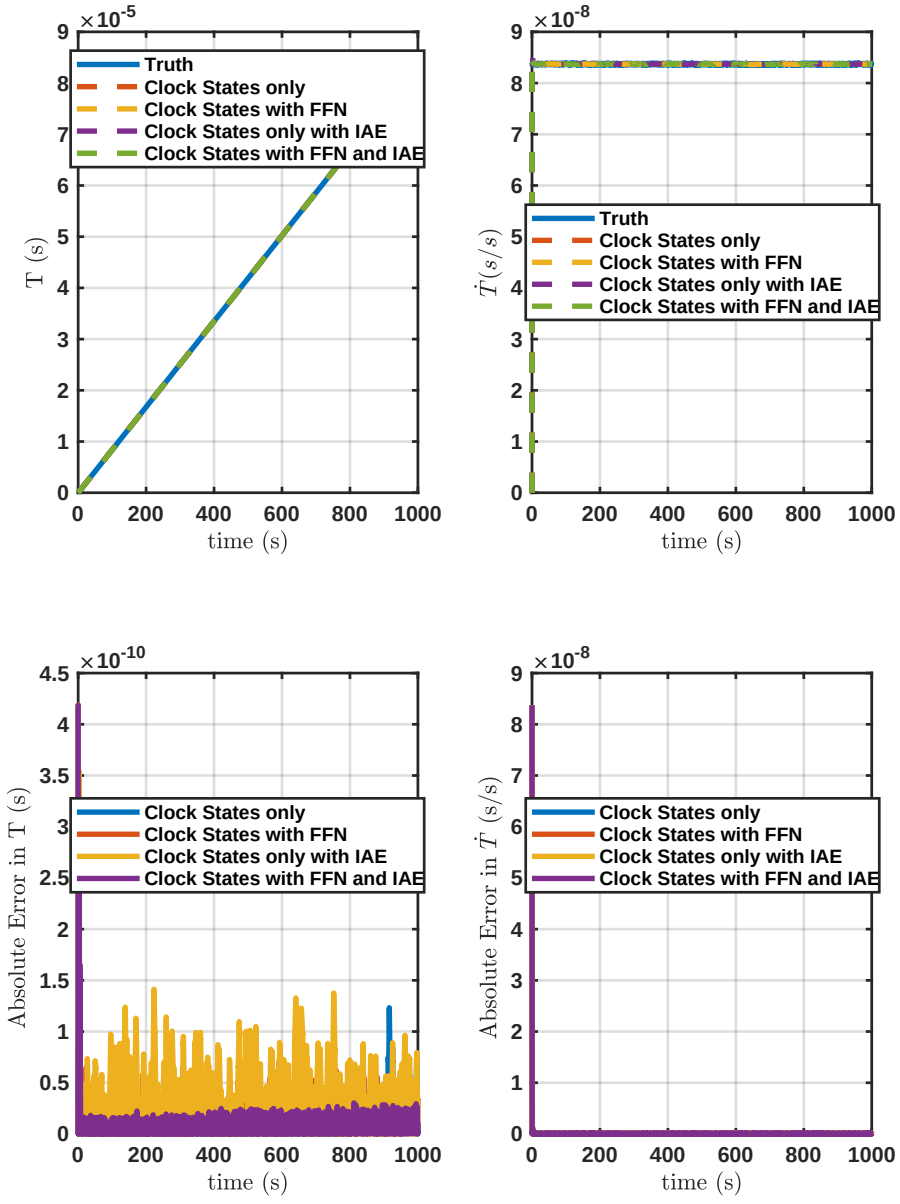

Figure B.60: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:6.34e-08s/s, DC Static Time Bias Error:6.34e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):71,22, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):71,1

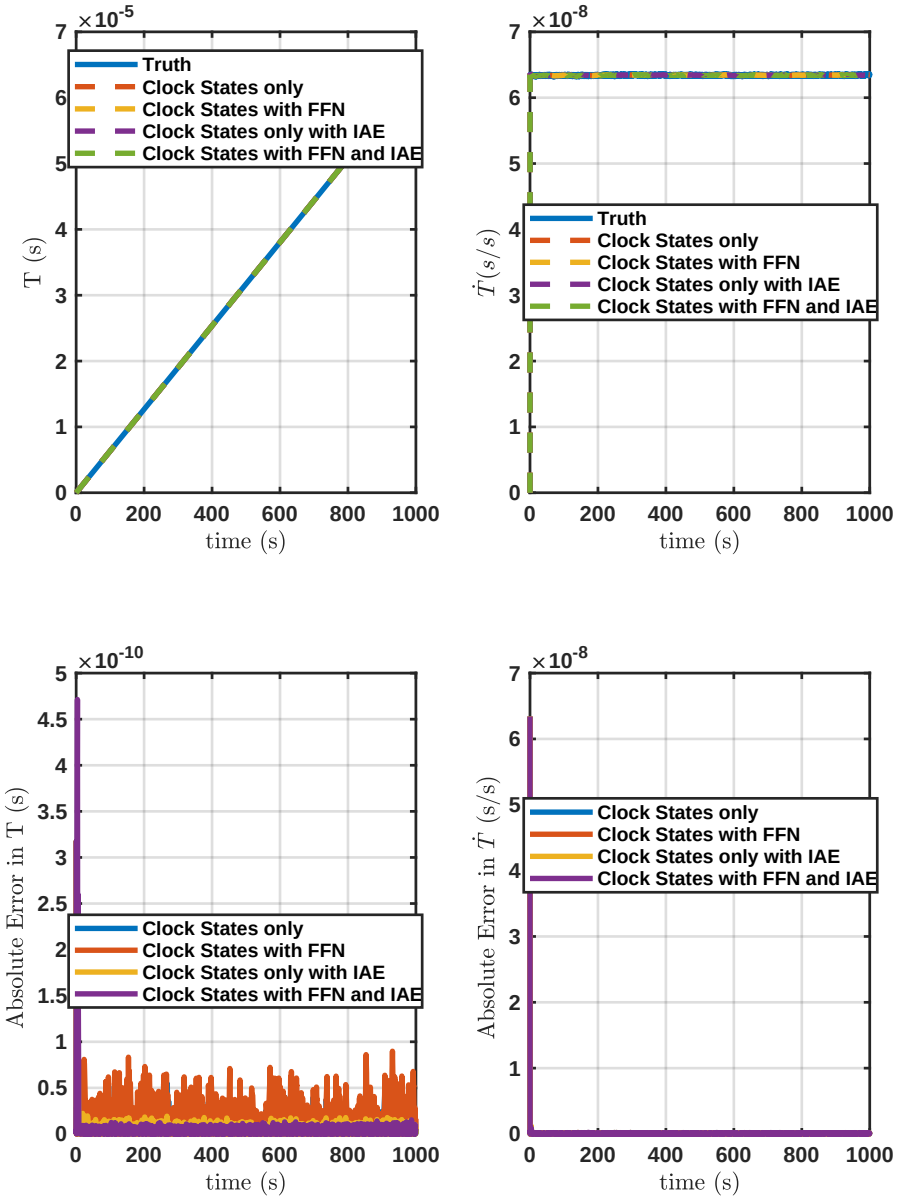

Figure B.61: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:4e-10s/s, DC Static Time Bias Error:4e-12s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):37,80, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):49,59

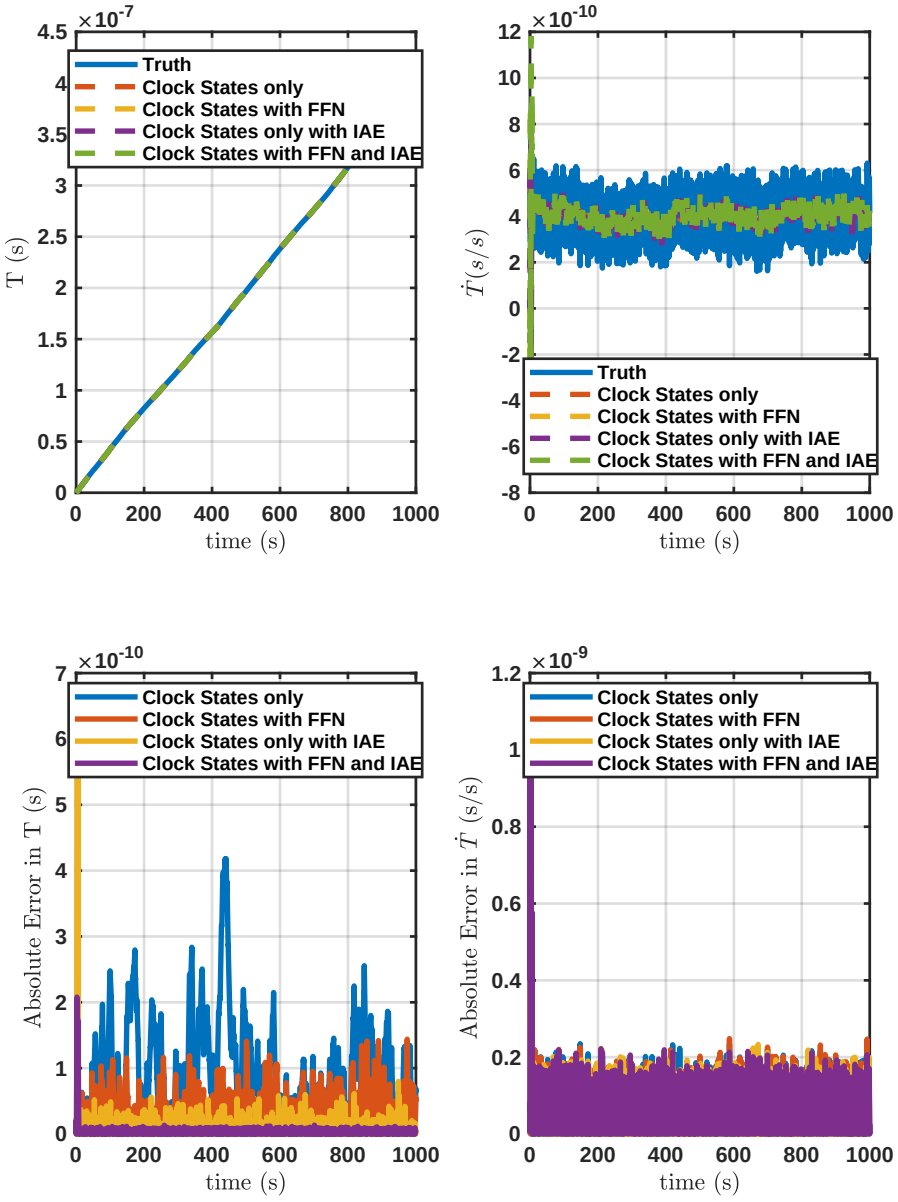

Figure B.62: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8.35e-08s/s, DC Static Time Bias Error:8.35e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):63,100, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):44,37

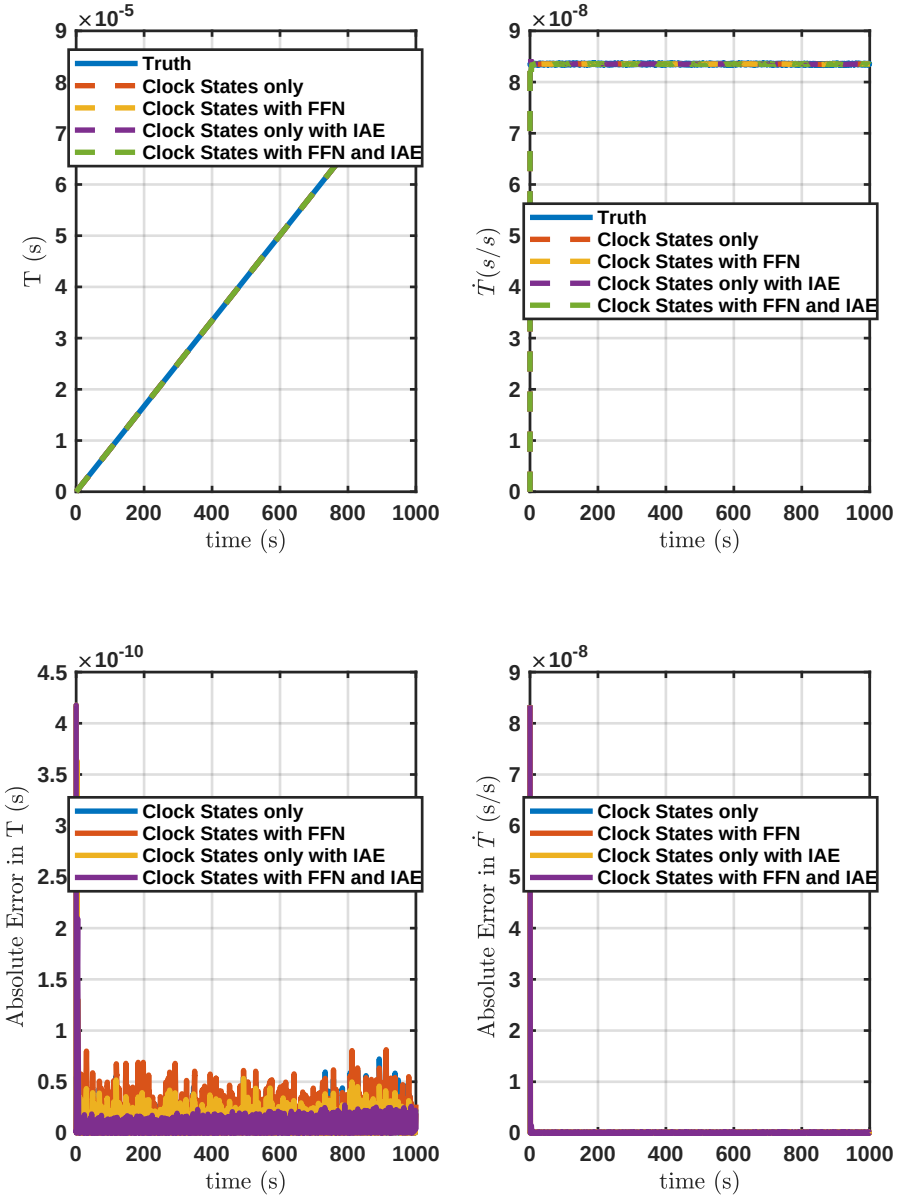

Figure B.63: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:6e-09s/s, DC Static Time Bias Error:6e-11s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):30,11, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):89,39

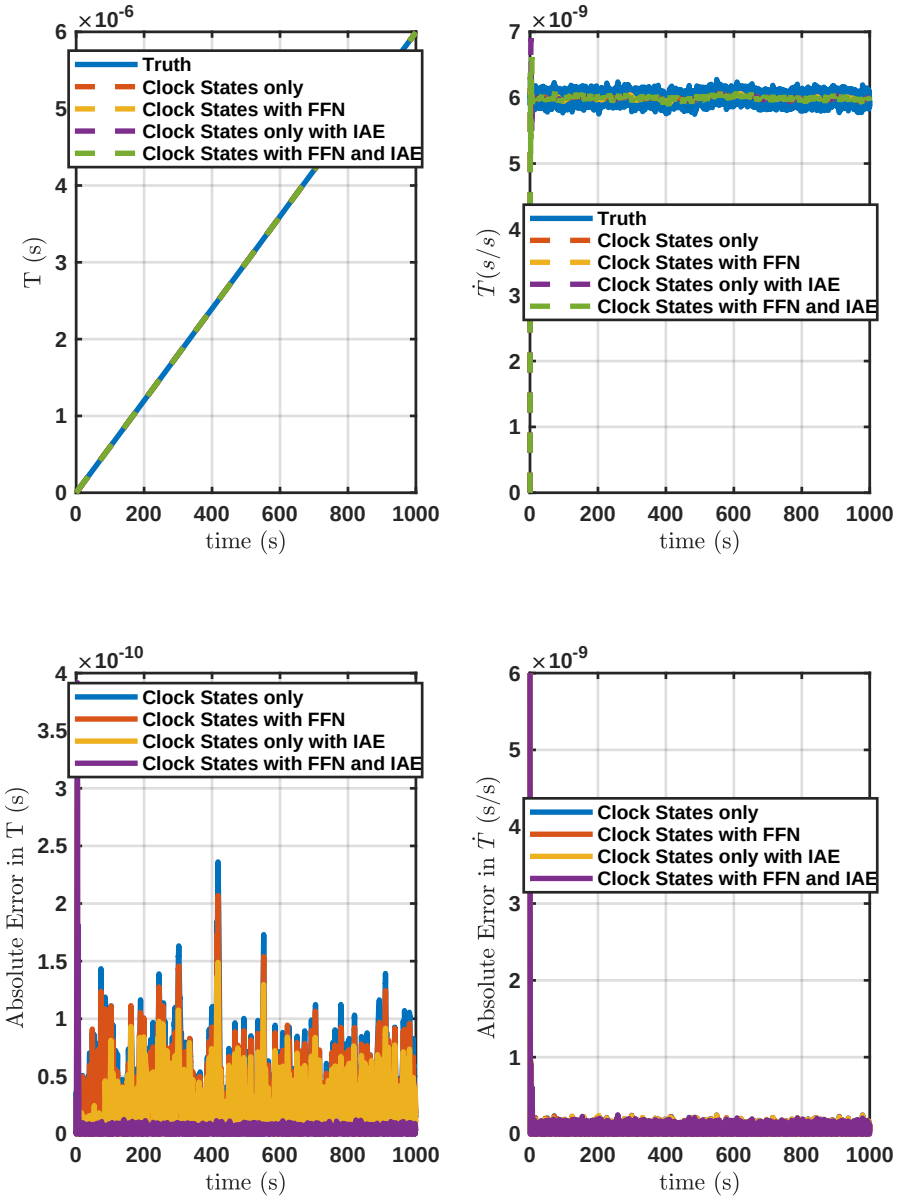

Figure B.64: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8.07e-08s/s, DC Static Time Bias Error:8.07e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):81,88, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):22,46

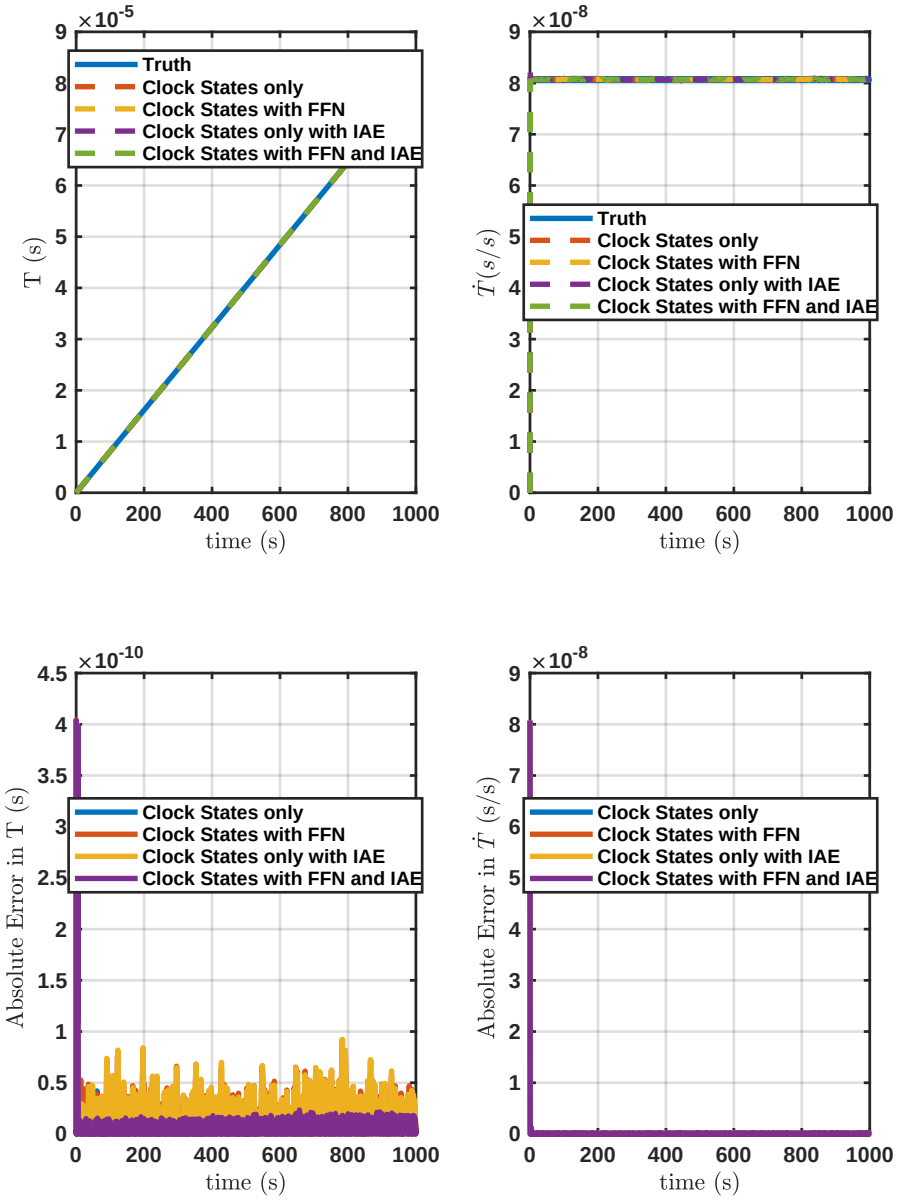

Figure B.65: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:2.11e-08s/s, DC Static Time Bias Error:2.11e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):34,39, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):57,29

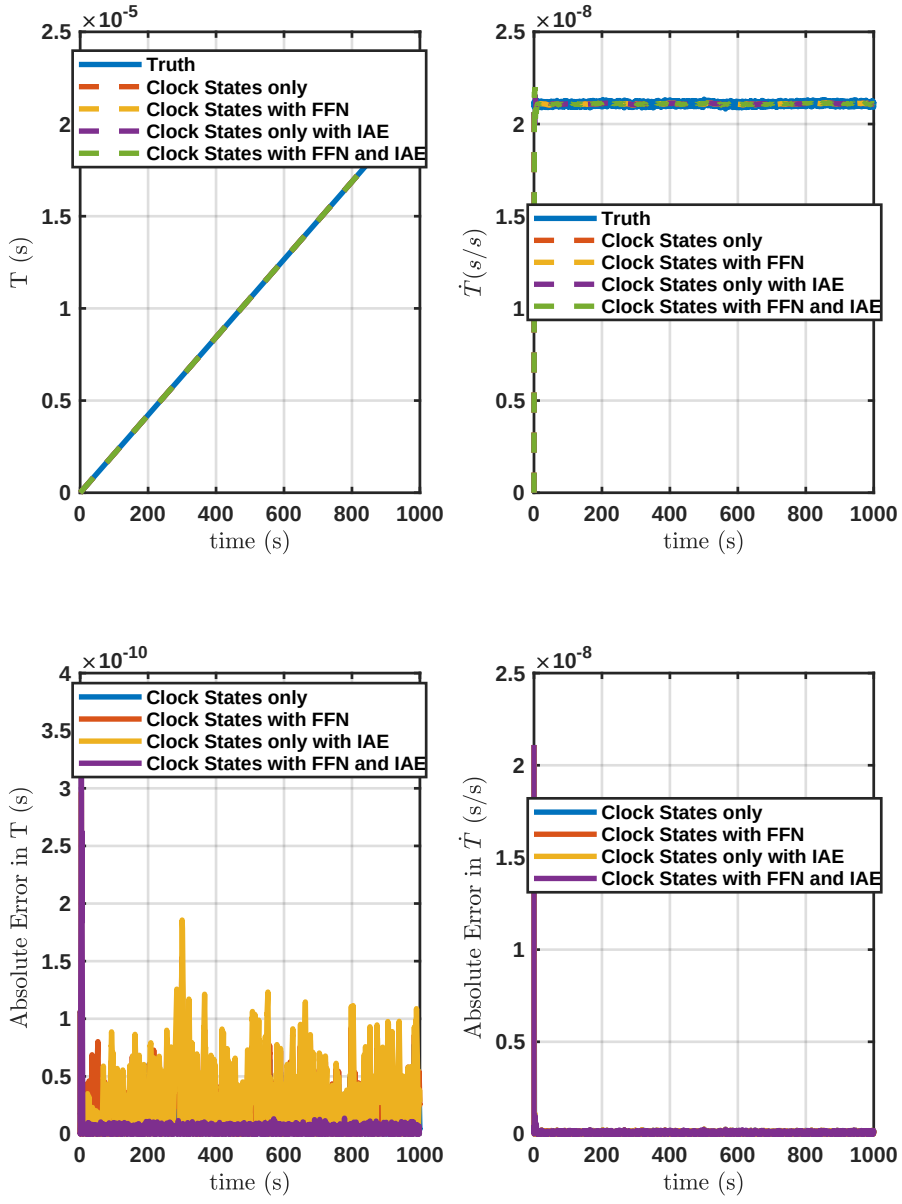

Figure B.66: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8.06e-08s/s, DC Static Time Bias Error:8.06e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):44,46, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):9,41

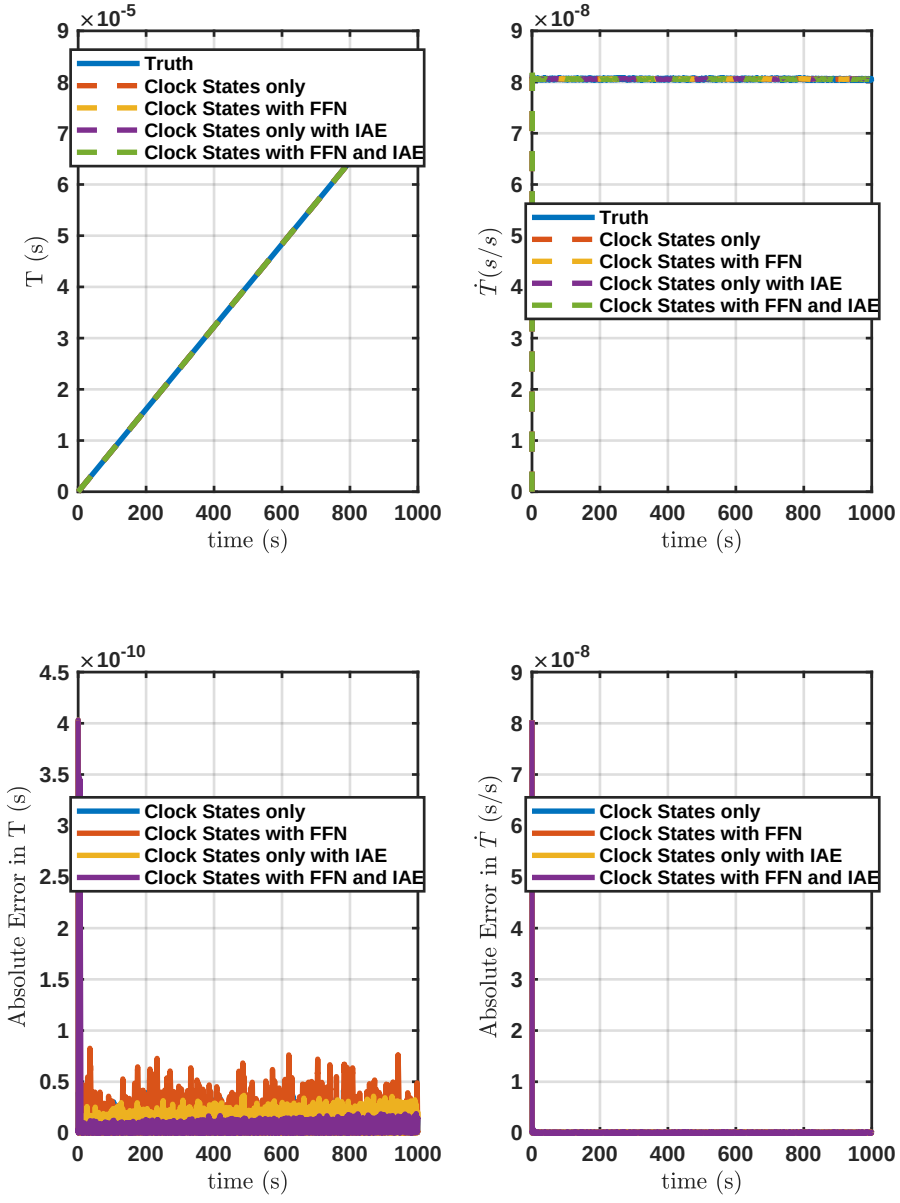

Figure B.67: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:3.37e-08s/s, DC Static Time Bias Error:3.37e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):31,88, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):54,3

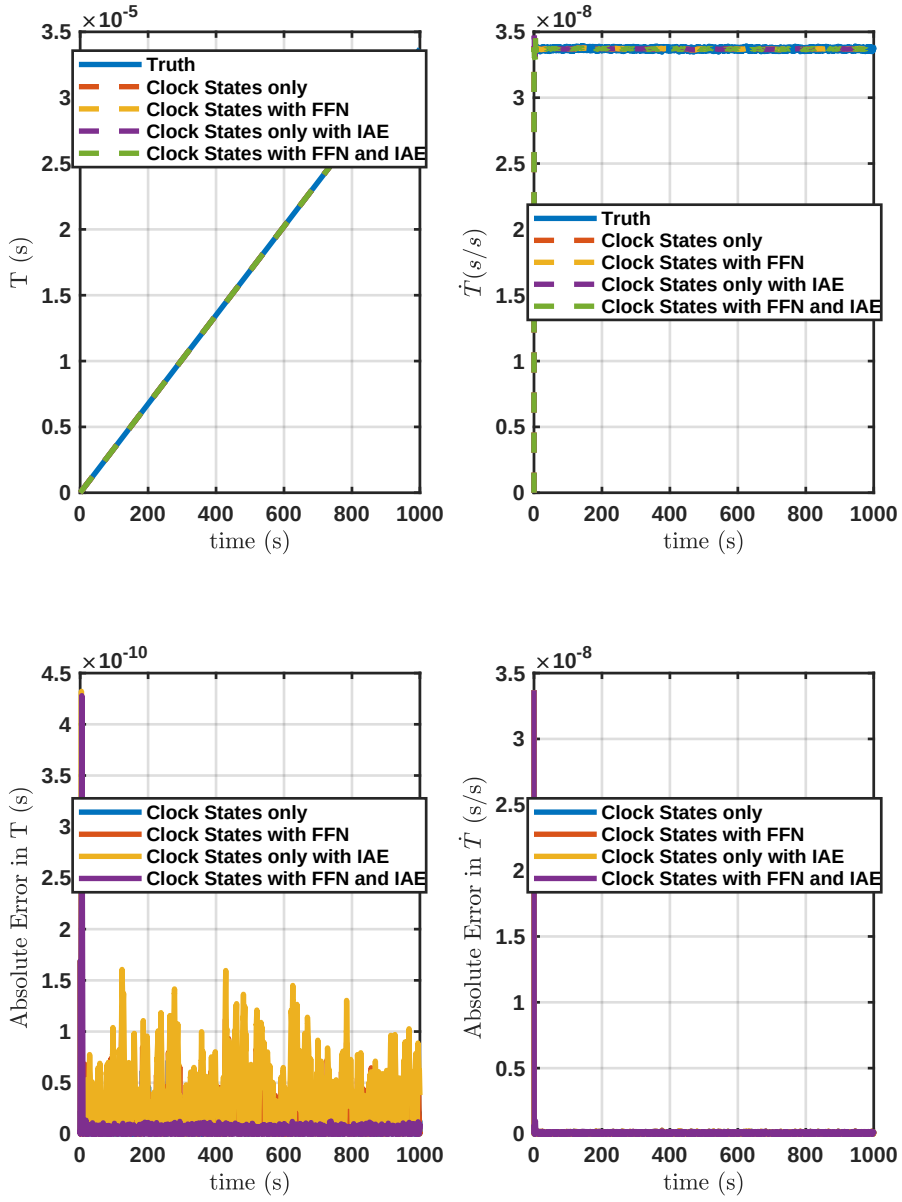

Figure B.68: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:9.19e-08s/s, DC Static Time Bias Error:9.19e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):86,20, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):92,66

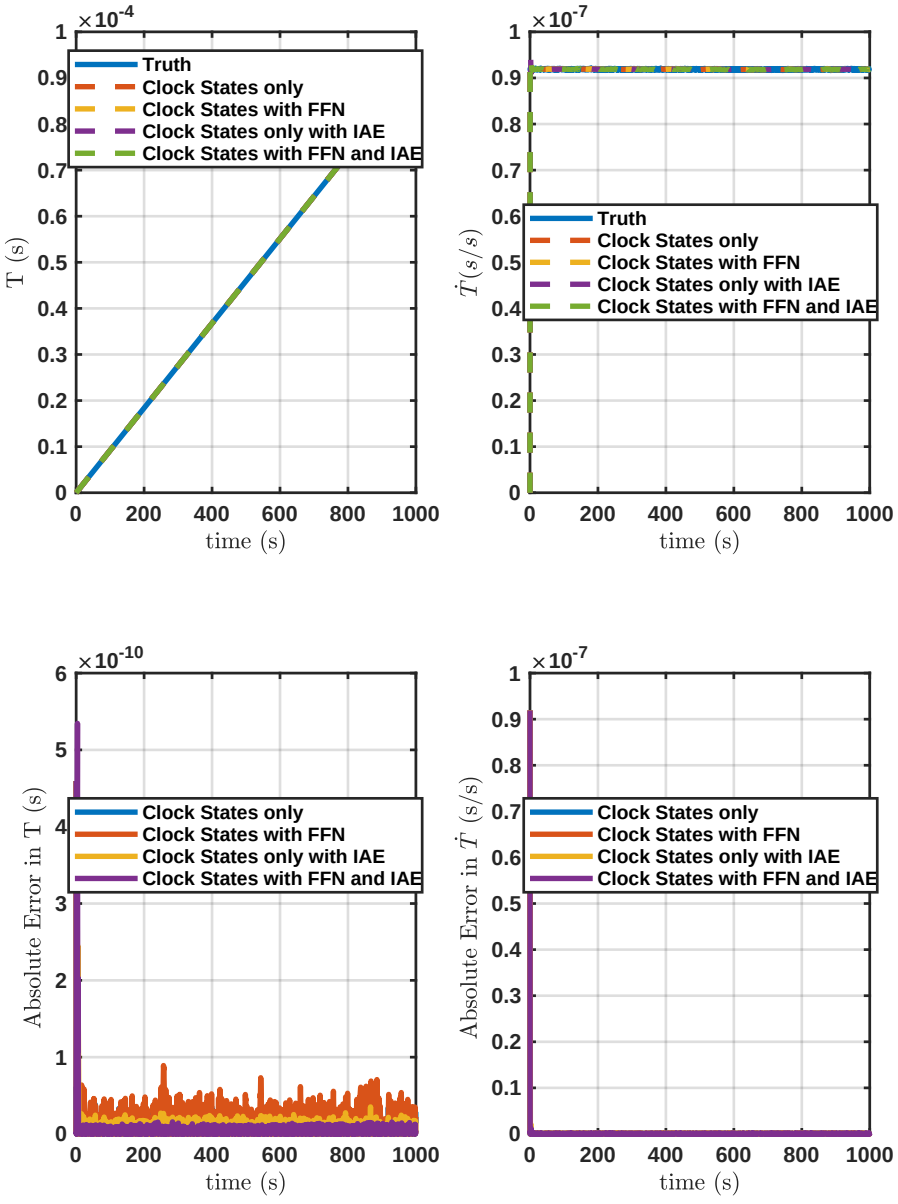

Figure B.69: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8.16e-08s/s, DC Static Time Bias Error:8.16e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):55,81, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):88,52

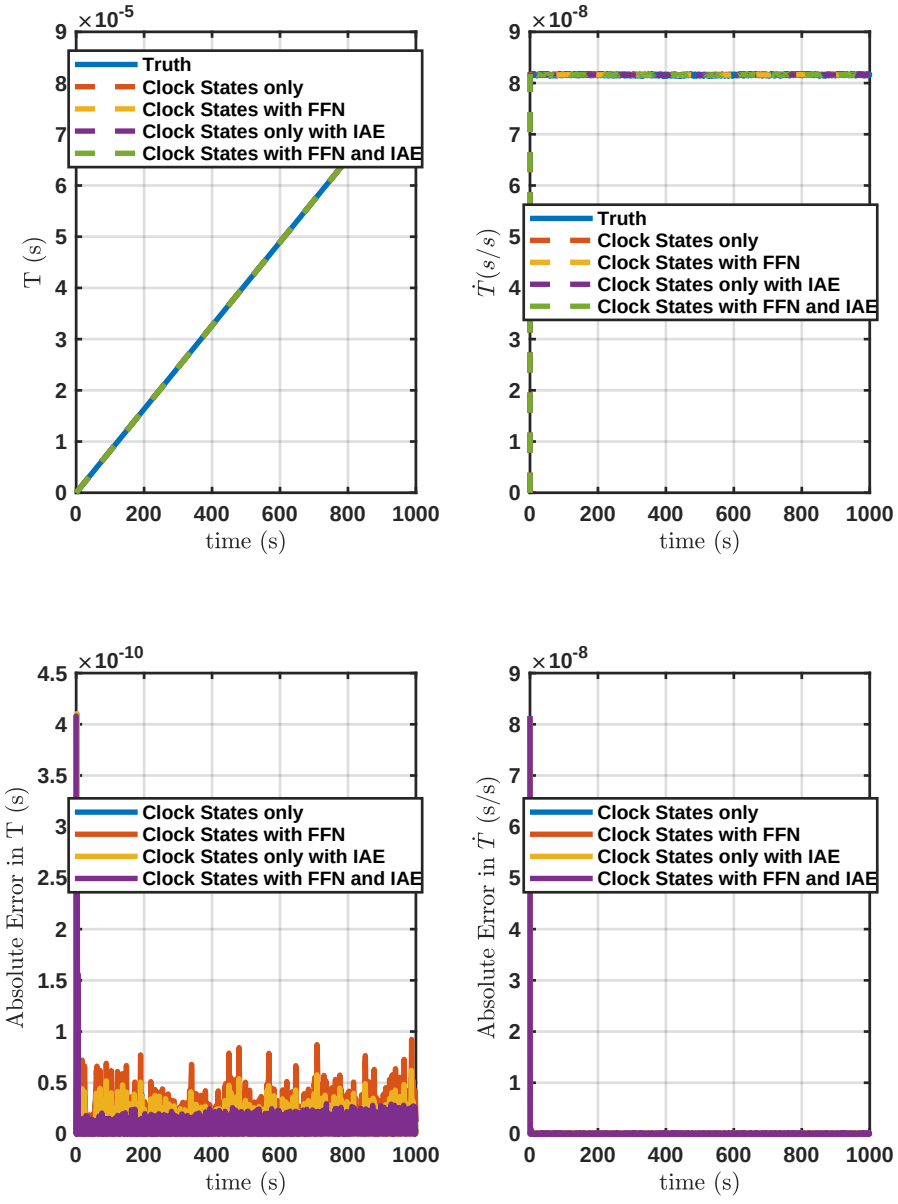

Figure B.70: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:1.47e-08s/s, DC Static Time Bias Error:1.47e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):61,20, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):73,37

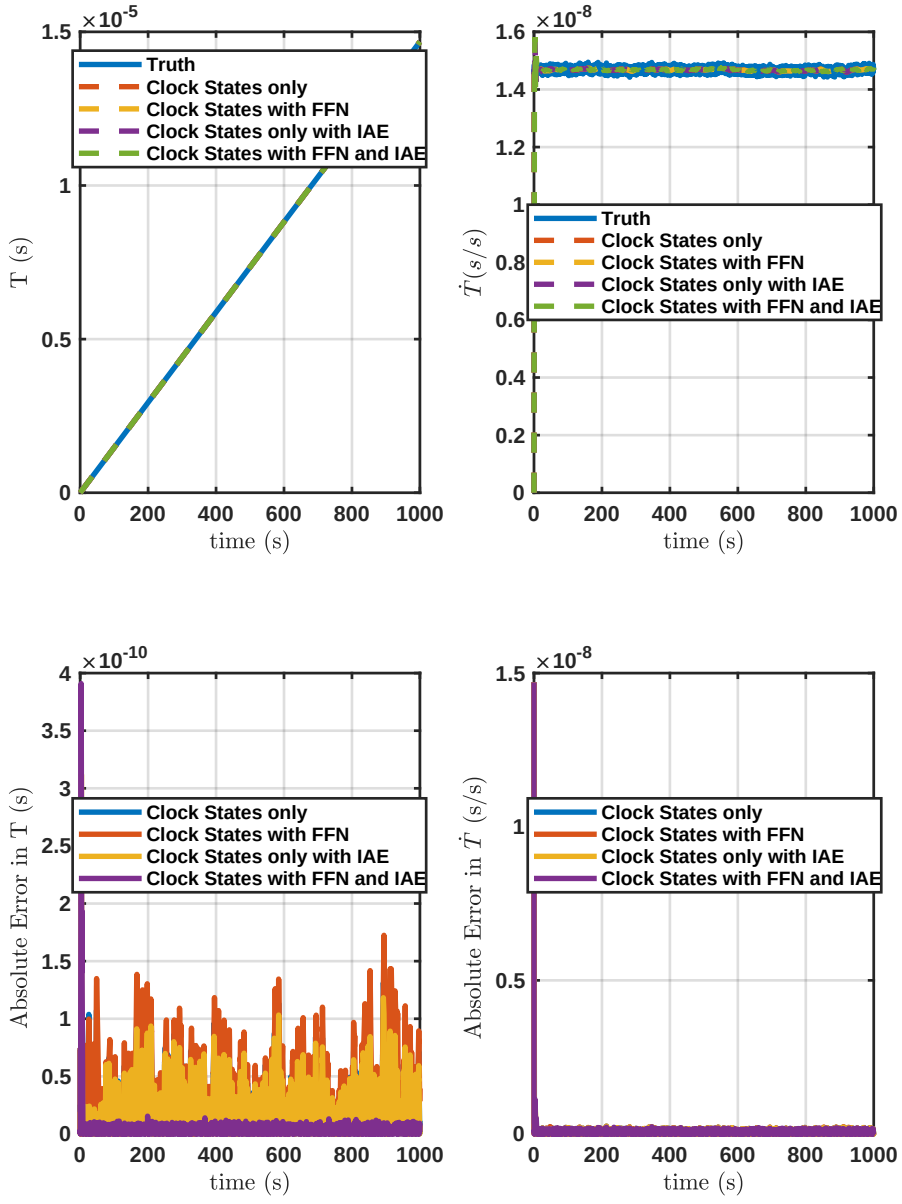

Figure B.71: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8.45e-08s/s, DC Static Time Bias Error:8.45e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):15,23, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):98,18

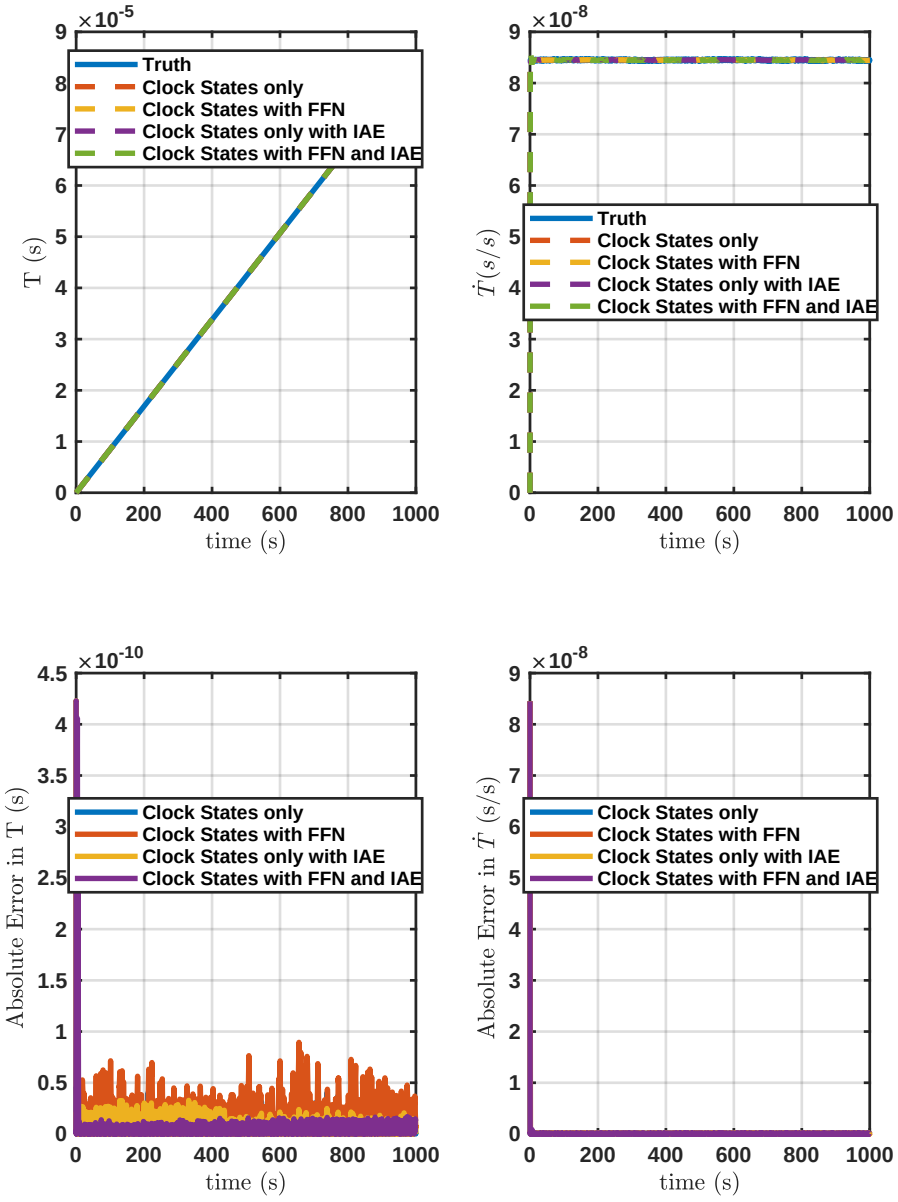

Figure B.72: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:2.05e-08s/s, DC Static Time Bias Error:2.05e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):42,36, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):97,10

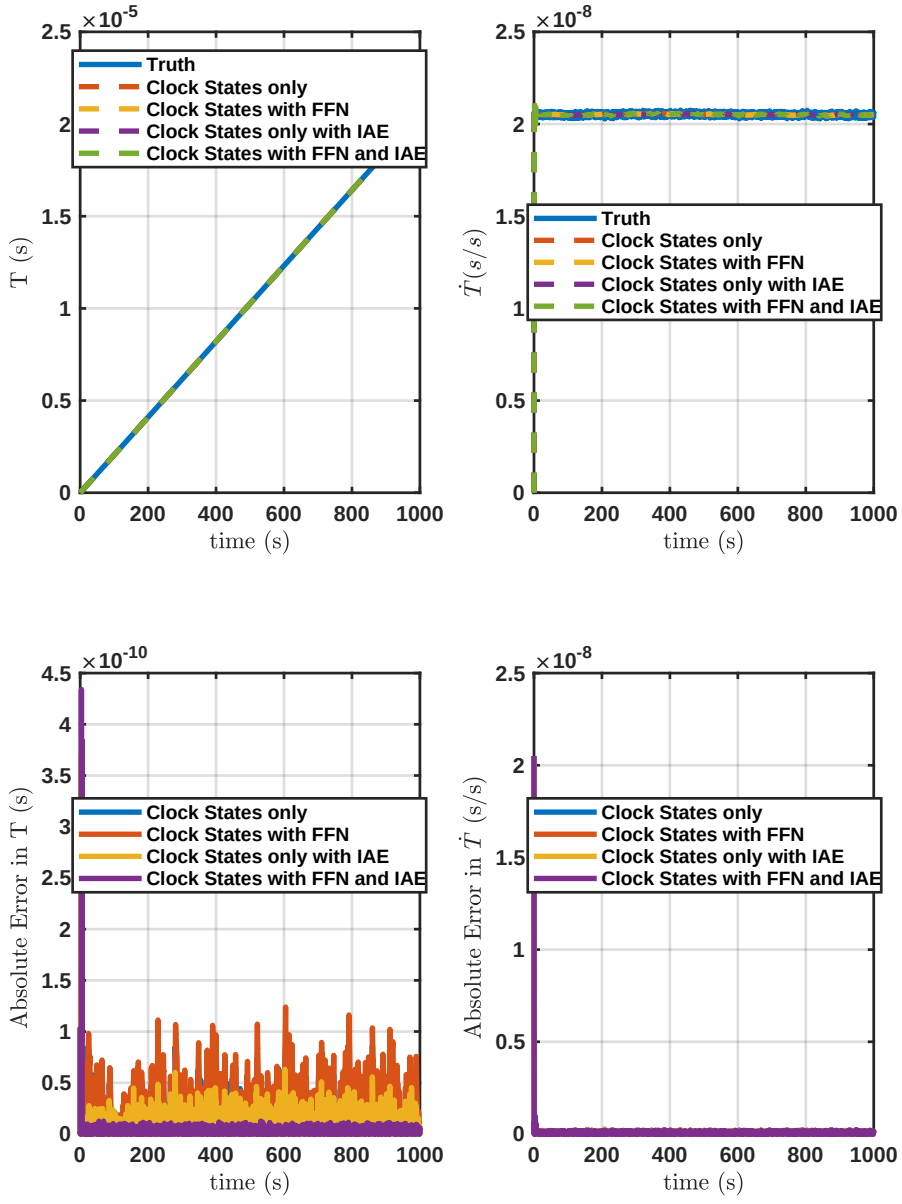

Figure B.73: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:5.14e-08s/s, DC Static Time Bias Error:5.14e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):98,79, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):80,97

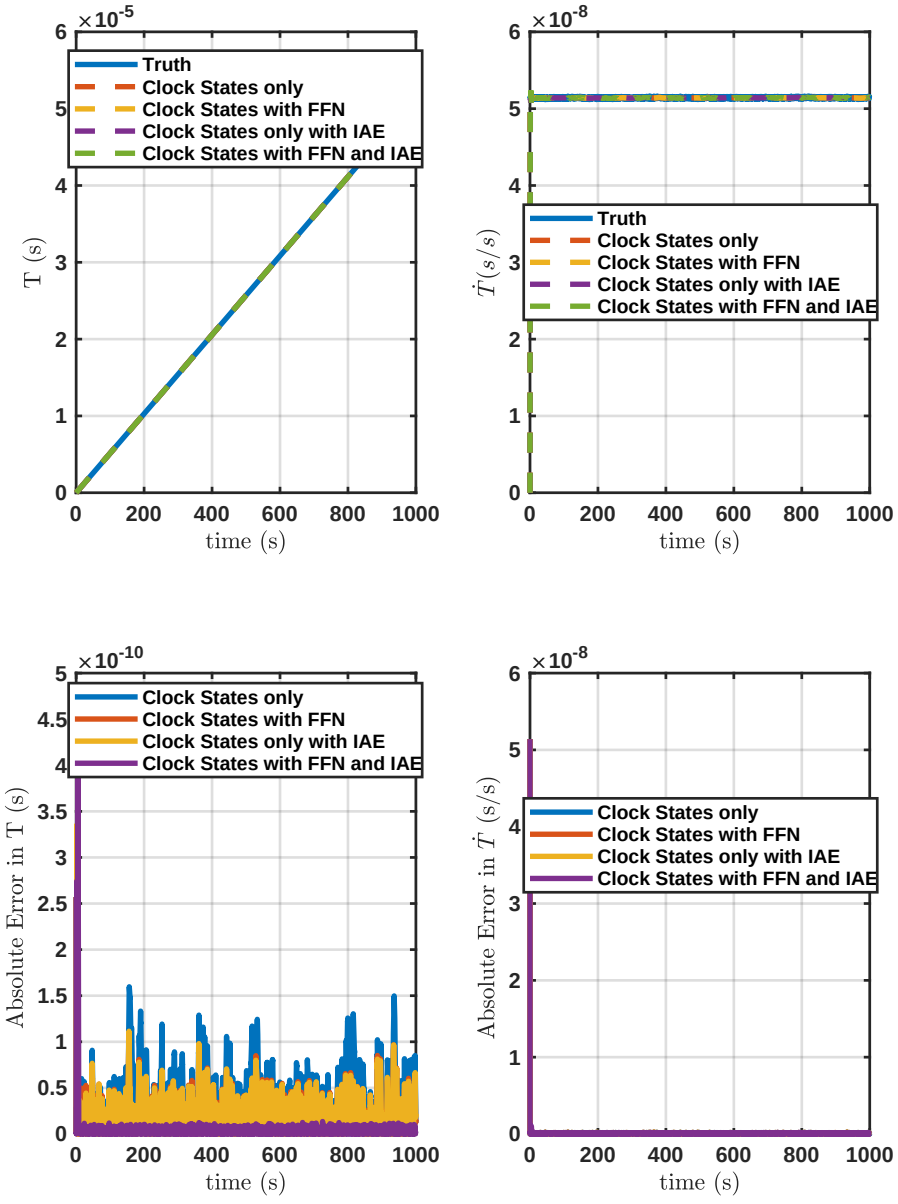

Figure B.74: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:8.38e-08s/s, DC Static Time Bias Error:8.38e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):14,20, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):46,46

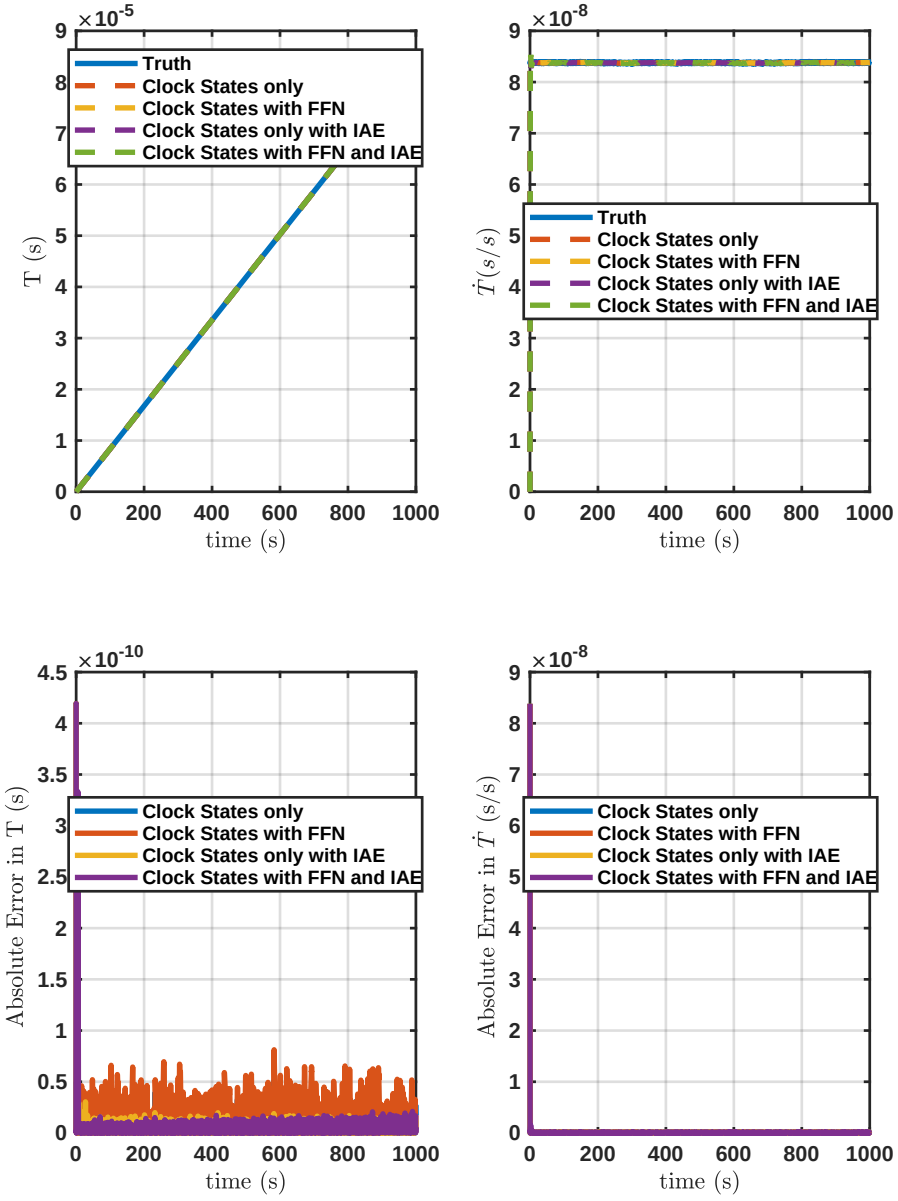

Figure B.75: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:5.18e-08s/s, DC Static Time Bias Error:5.18e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):3,22, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):11,32

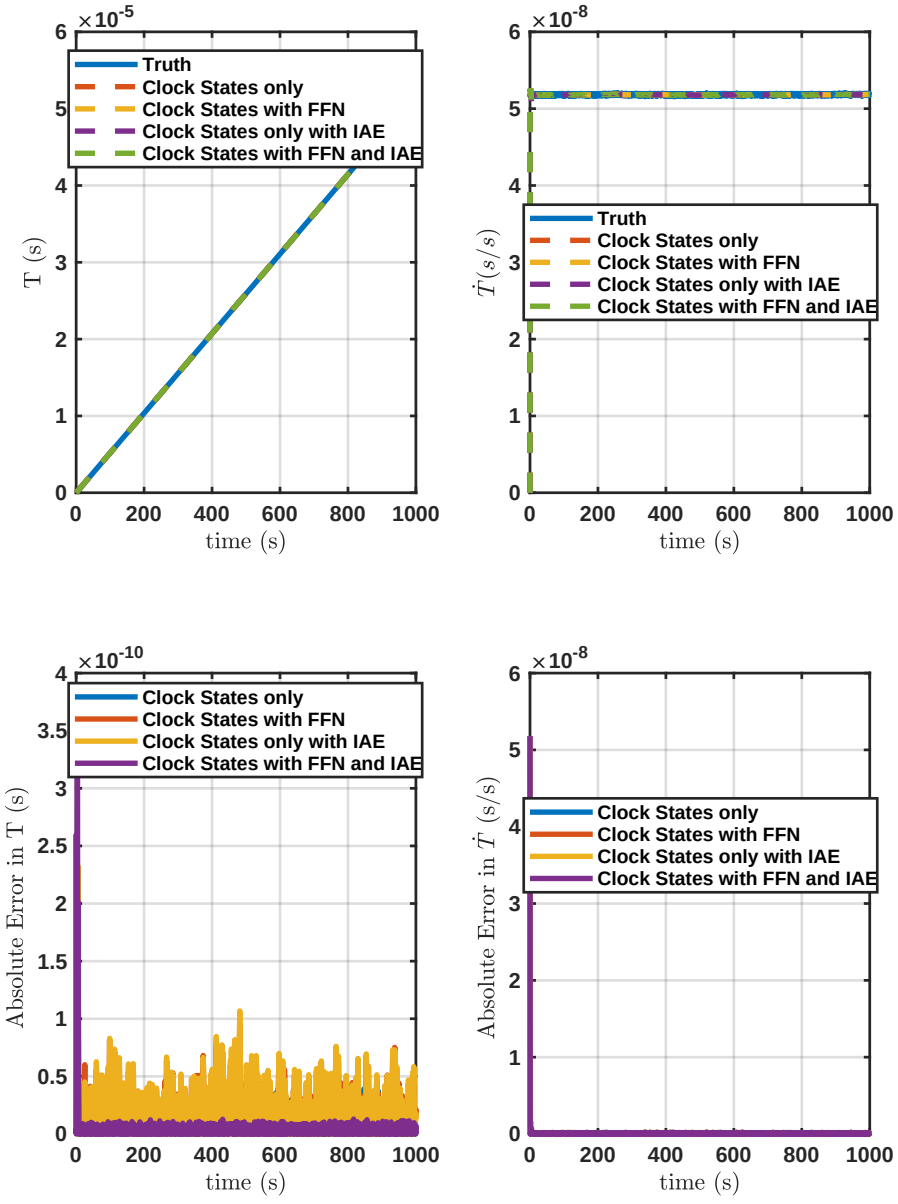

Figure B.76: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:9.01e-08s/s, DC Static Time Bias Error:9.01e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):87,91, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):82,61

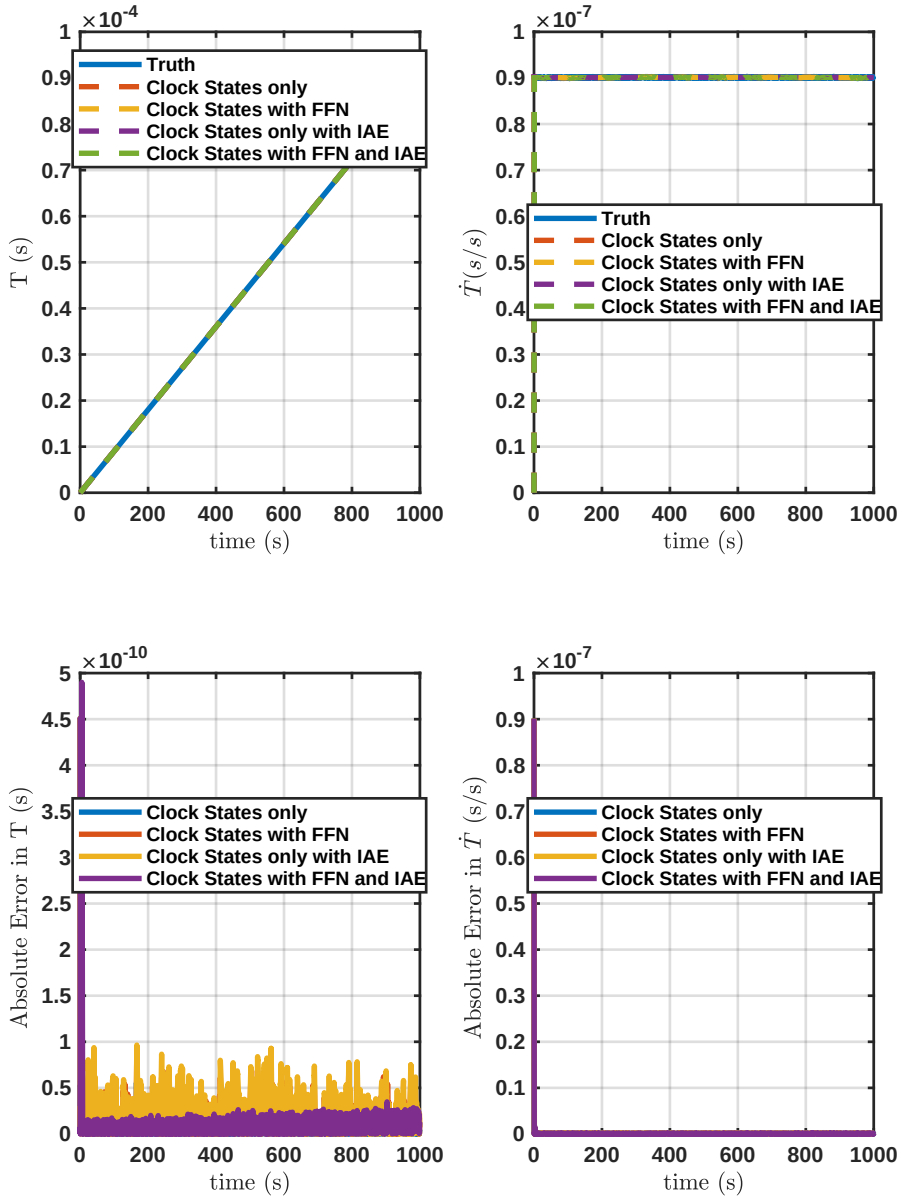

Figure B.77: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:1.25e-08s/s, DC Static Time Bias Error:1.25e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):7,59, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):31,84

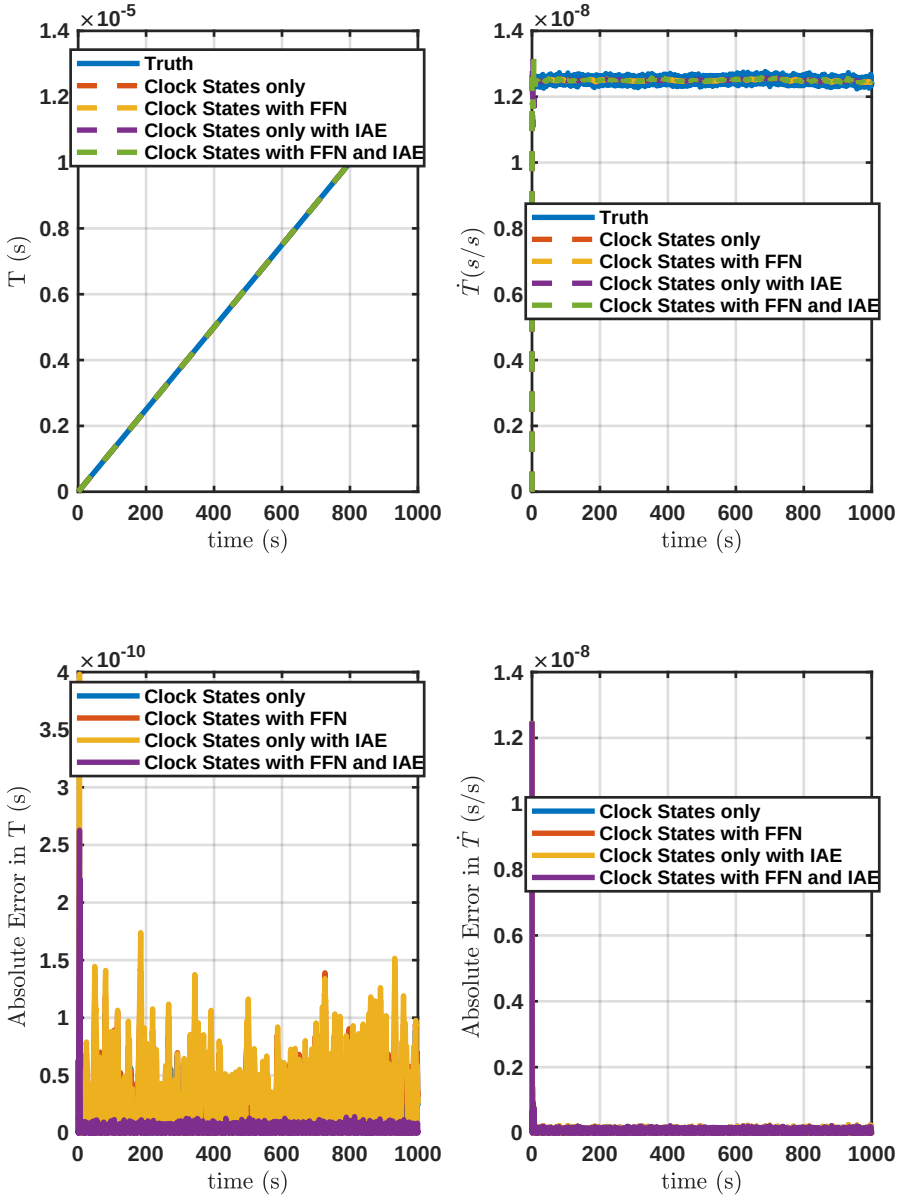

Figure B.78: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:4e-08s/s, DC Static Time Bias Error:4e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):43,1, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):88,22

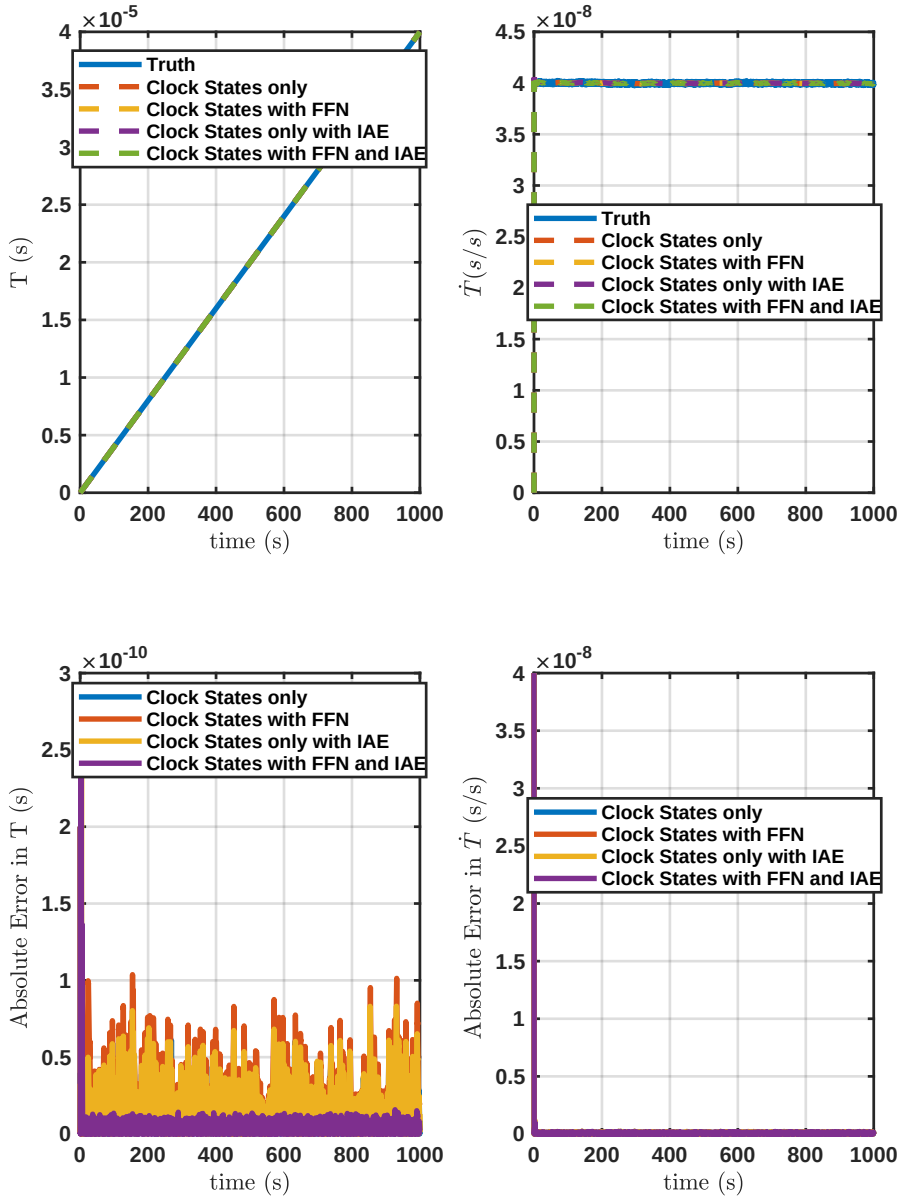

Figure B.79: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:7.83e-08s/s, DC Static Time Bias Error:7.83e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):13,69, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):80,36

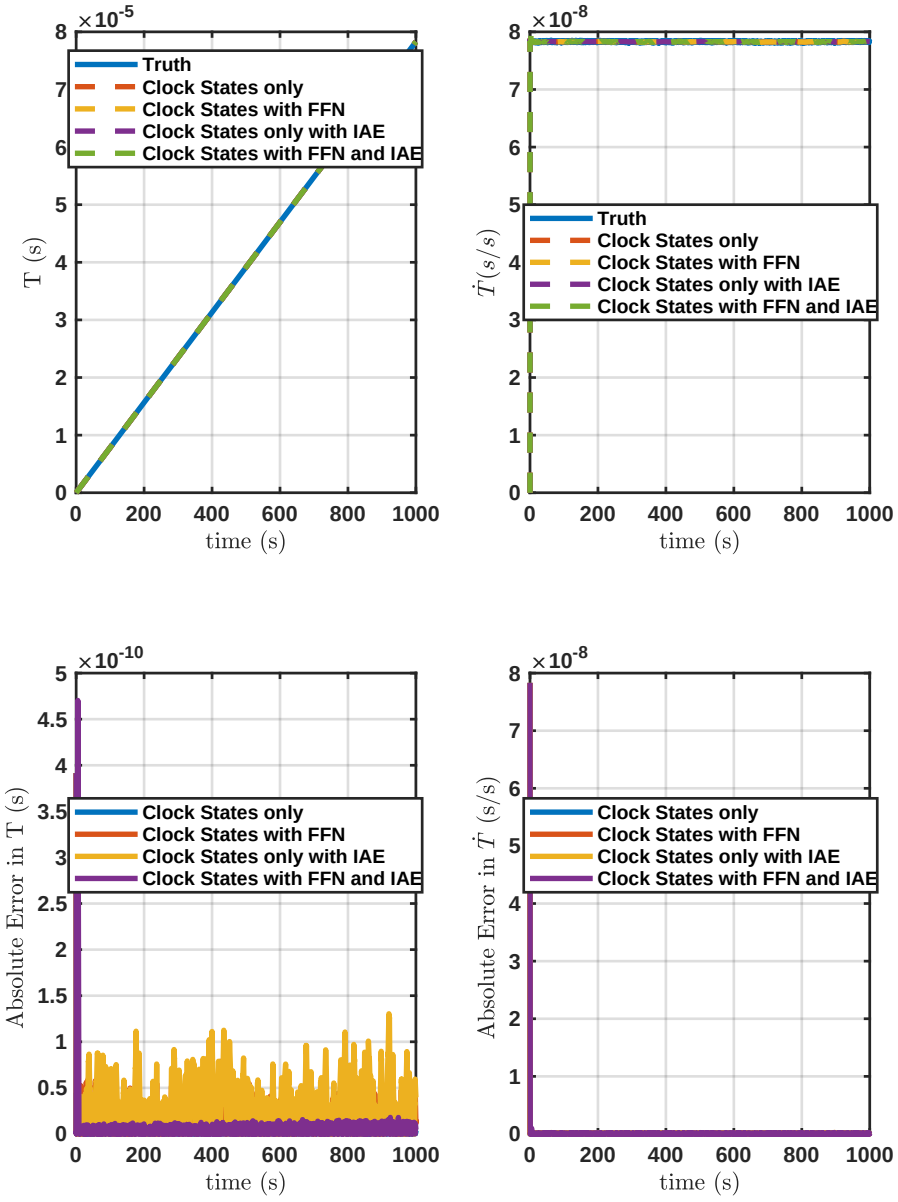

Figure B.80: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:1.45e-08s/s, DC Static Time Bias Error:1.45e-10s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):17,39, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):90,41

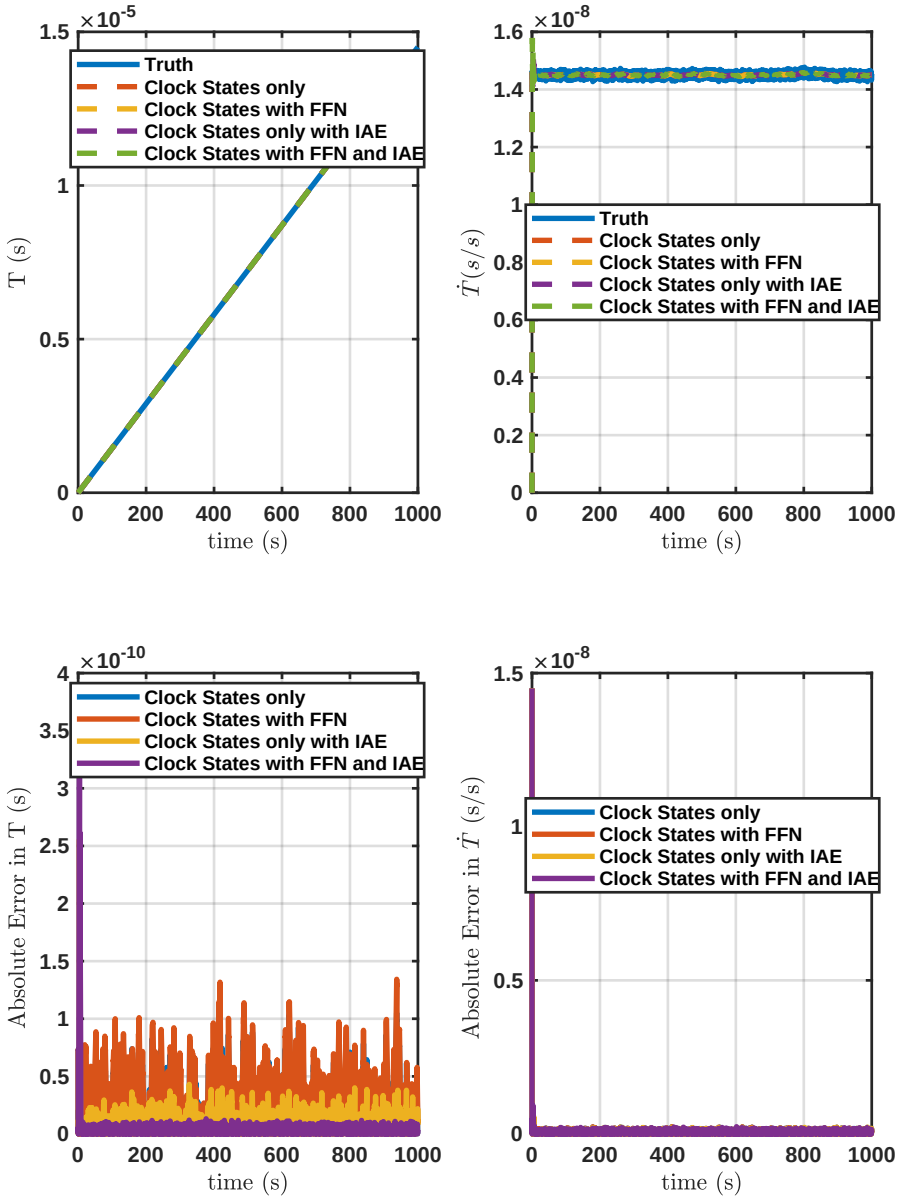

Figure B.81: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:2.36e-08s/s, DC Static Time Bias Error:2.36e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):0,12, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):92,74

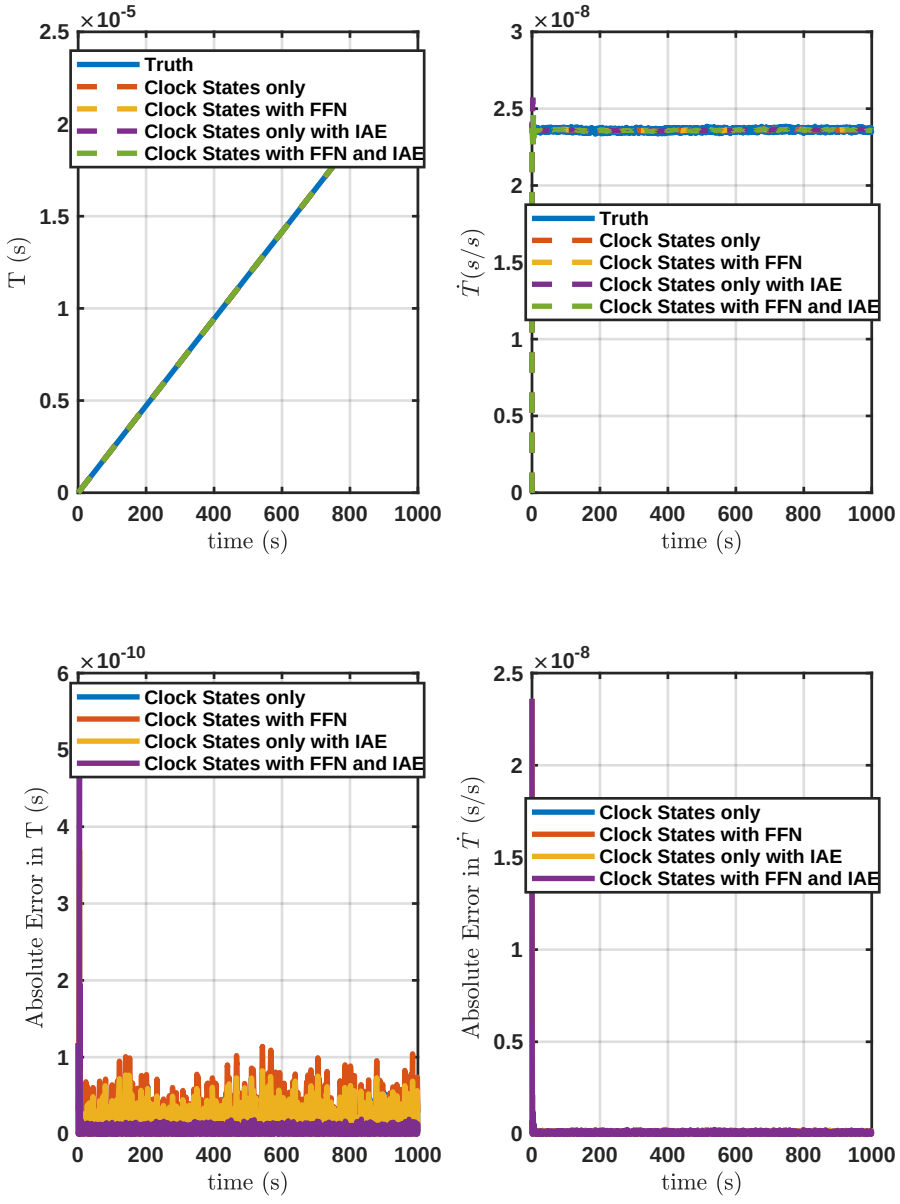

Figure B.82: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

DC Fractional Frequency Error:9.35e-08s/s, DC Static Time Bias Error:9.35e-10s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):58,3, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):81,74

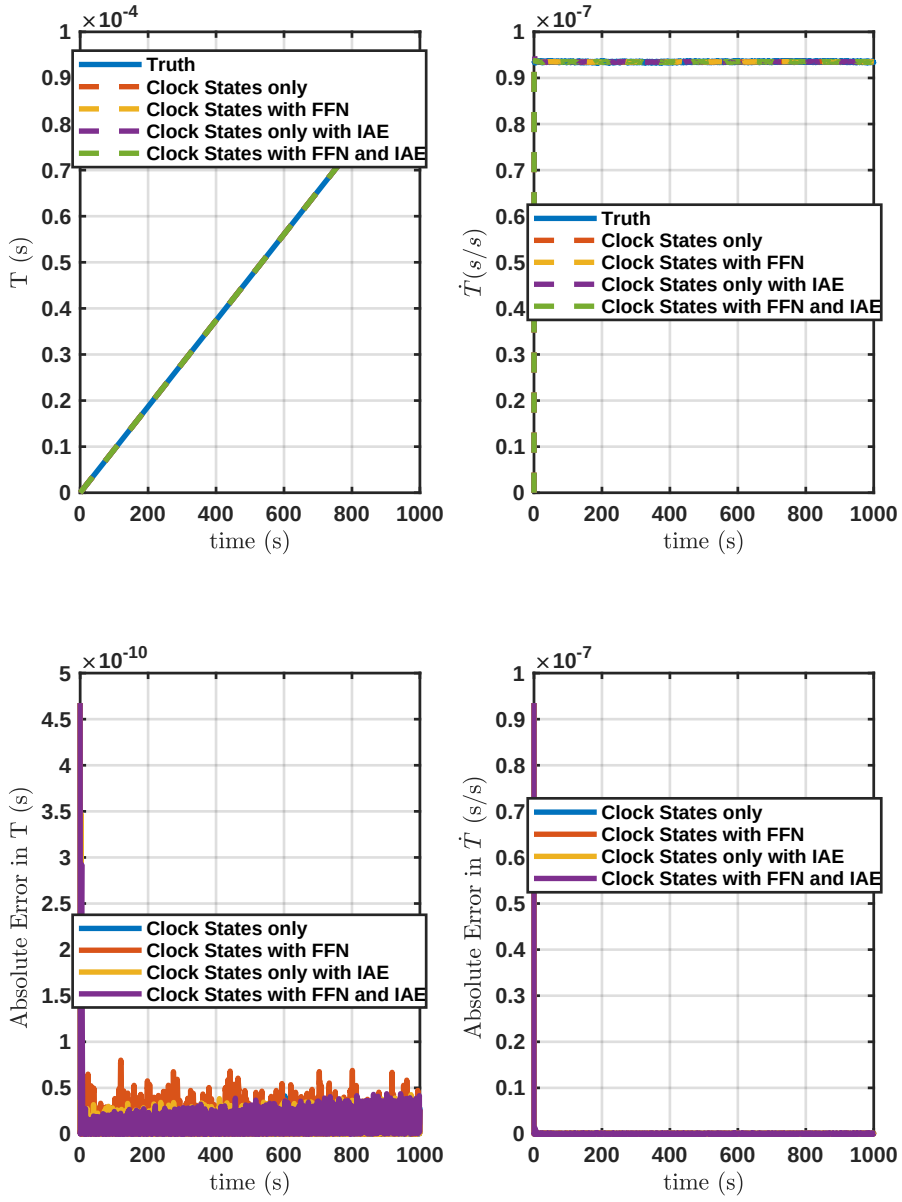

Figure B.83: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

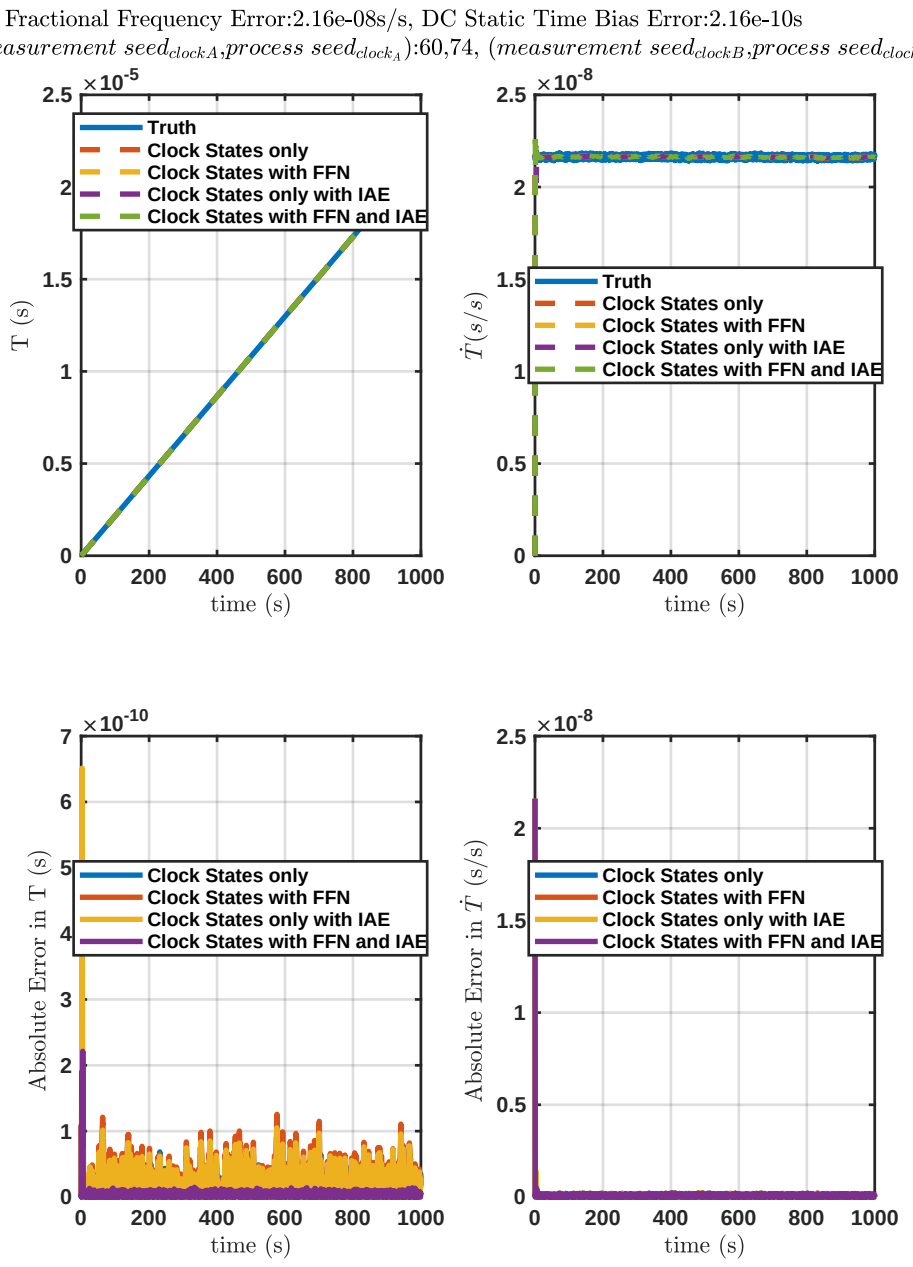

Figure B.84: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter With Only Oscillator States.

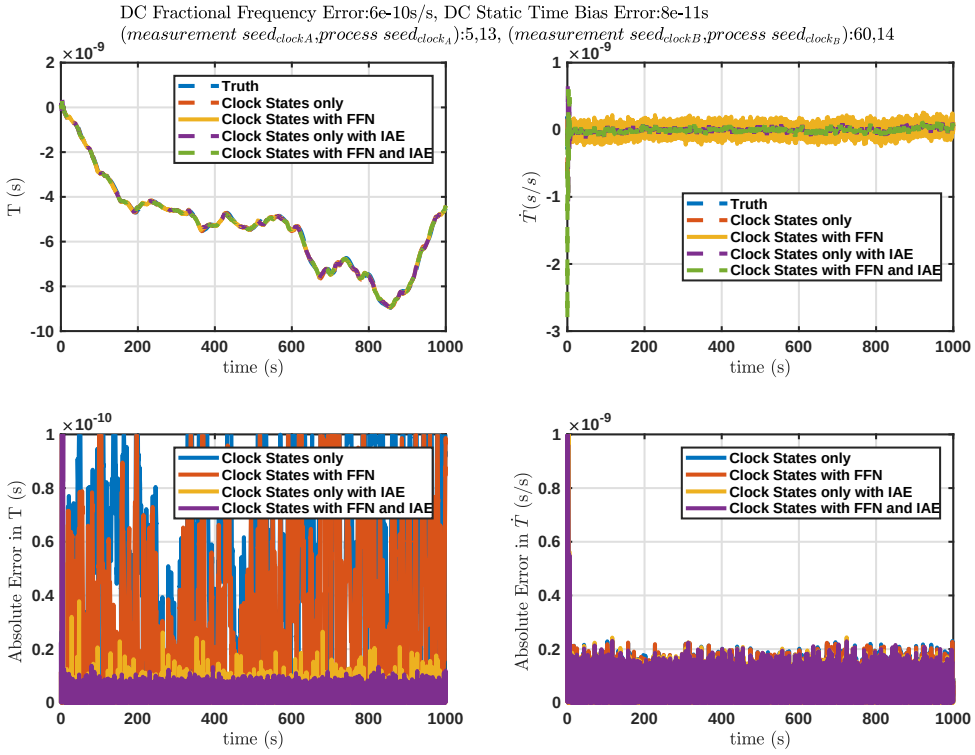

Figure B.85: Oscillator States Only In Non-Linear HPPC.

For completeness, 3 additional 10000 second simulations were ran just to see if long term noise constituents would change the results.

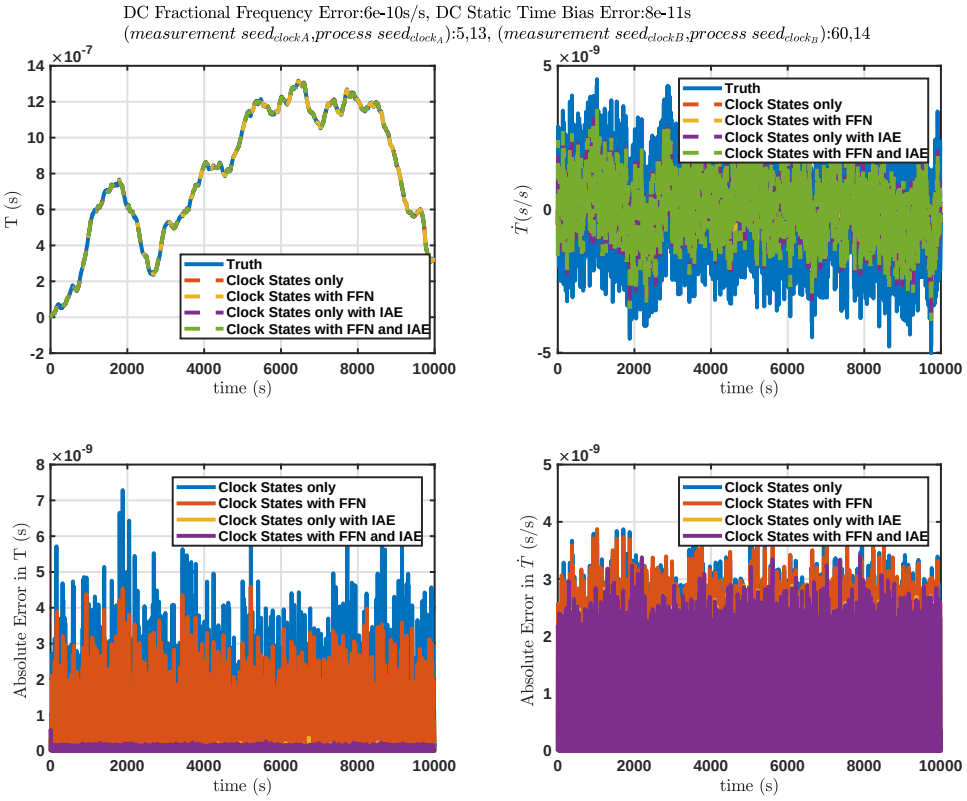

Figure B.86: Oscillator States Only In Non-Linear HPPC - Long Term Simulation.

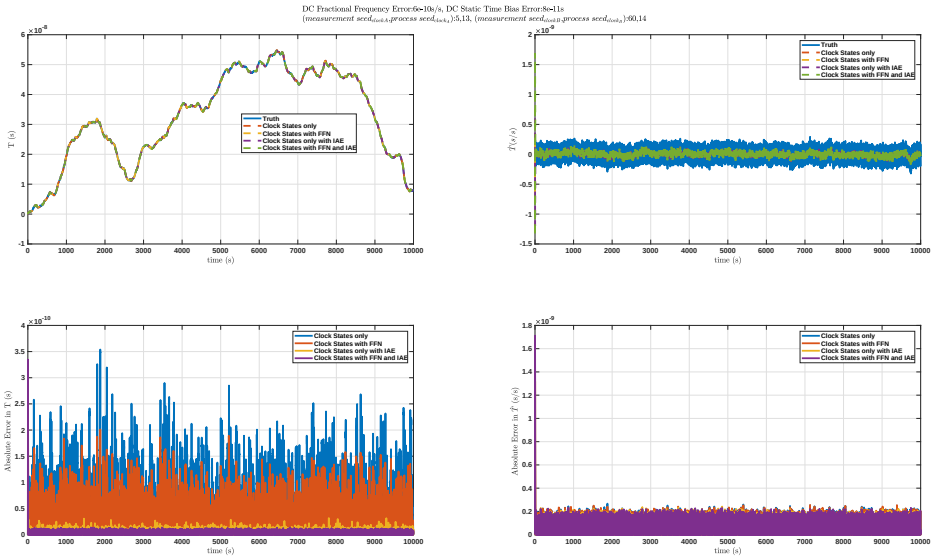

Figure B.87: Oscillator States Only In Non-Linear HPPC - Long Term Simulation.

B.2 Monte Carlo Analysis of Bank of IAE Extended Kalman Filter Applied to HPPC System Using Kasdin Noise

The first numerical experiment demonstrating the performance of the Bank of Adaptive Extended Kalman Filters. Colored noise states are used in the simulated noise measurements, but none of the Extended Kalman Filters in the filter bank contain colored noise states. The Kasdin Method is used to generate colored noise. The σ diffusion coefficients used in this experiment are shown in table B.3. Additionally $F_c = 40$ MHz and $T_s = 10$ms. All other simulations parameters are listed on the plots. Additionally, $T_{adapt} = 0.5 seconds$, with convergence after 10 seconds.

Table B.3: Simulated Noise Paramters.

Constituent	**Value**
WFN	2.6153e-04 Hz
WPN	0.0048 Cycle
FFN	0.0012 Hz/s
RWFN	0 Hz/s
RRFN	0 Hz/s/s

$$var(\tau)_k = [(1e-5)1_{1\times 8}, (1e-6)1_{1\times 8}, (1e-7)1_{1\times 8}, (1e-8)1_{1\times 8}, (1e-9)1_{1\times 8}, (1e-10)1_{1\times 10}] \quad \text{(B.4)}$$

$$var(T)_k = 1e-16[(1e-5)1_{1\times 8}, (1e-6)1_{1\times 8}, (1e-7)1_{1\times 8}, (1e-8)1_{1\times 8}, (1e-9)1_{1\times 8}, (1e-10)1_{1\times 10}] \quad \text{(B.5)}$$

$$var(t_{A,Rx})_k = [(1e-5)1_{1\times 8}, (1e-6)1_{1\times 8}, (1e-7)1_{1\times 8}, (1e-8)1_{1\times 8}, (1e-9)1_{1\times 8}, (1e-10)1_{1\times 8}, (1e-11)1_{1\times 8}, (1e-12)1_{1\times 8}] \quad \text{(B.6)}$$

$$var(t_{B,Rx})_k = var(t_{A,Rx}) \quad \text{(B.7)}$$

Where $1_{1\times n}$ is a vector of length n, composed of all 1's. Then, each initial covariance matrix is computed for each filter bank entry, for all k

$$Q_o = diag([var(\tau)_k LA, var(\tau)_k LA, var(\tau)_k LA, var(T)_k [LALALA]]) \quad \text{(B.8)}$$

$$P_o = Q_o \quad \text{(B.9)}$$

$$R_o = diag([var(t_{A,Rx})_k, var(t_{B,Rx})_k]) \quad \text{(B.10)}$$

DC Fractional Frequency Error:-1.51e-08s/s, DC Static Time Bias Error:-5.79e-06s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):25,66, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):75,46

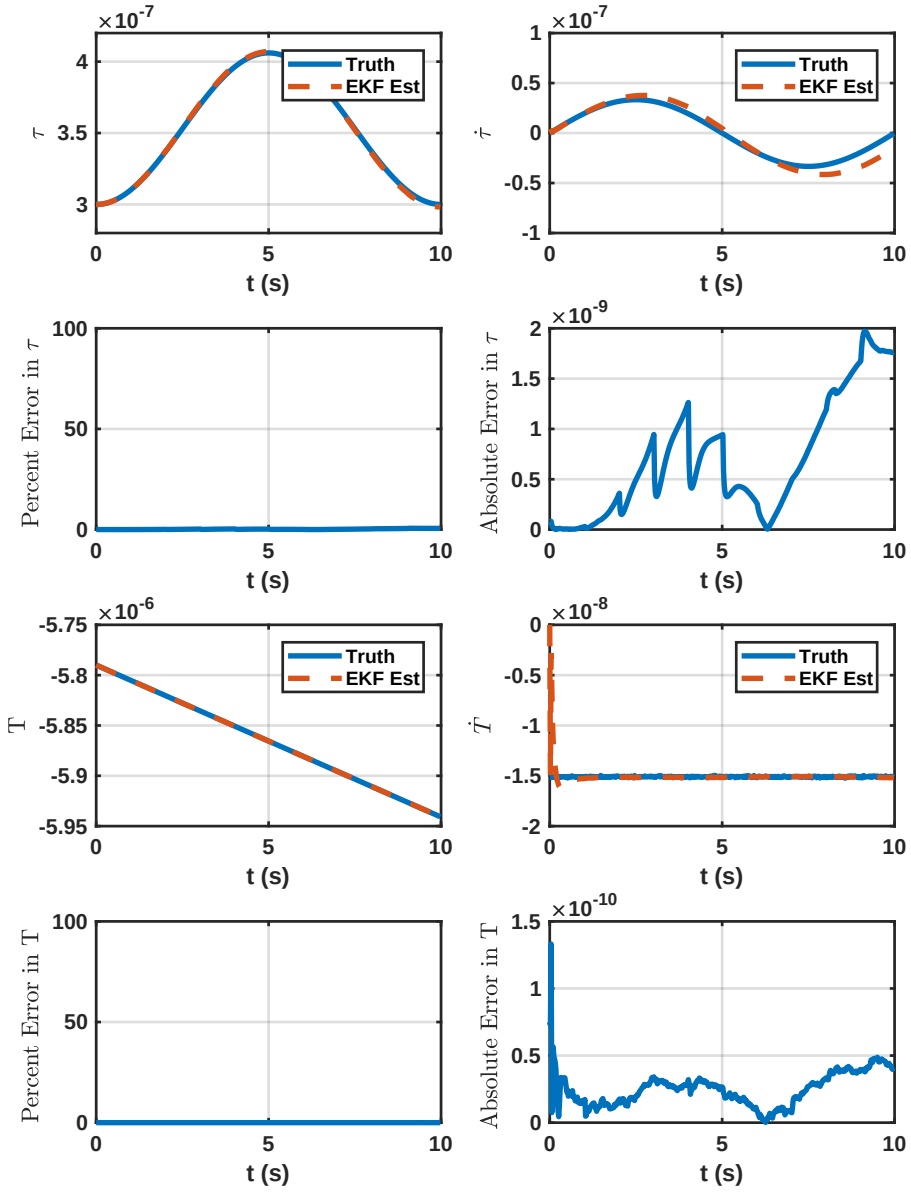

Figure B.88: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

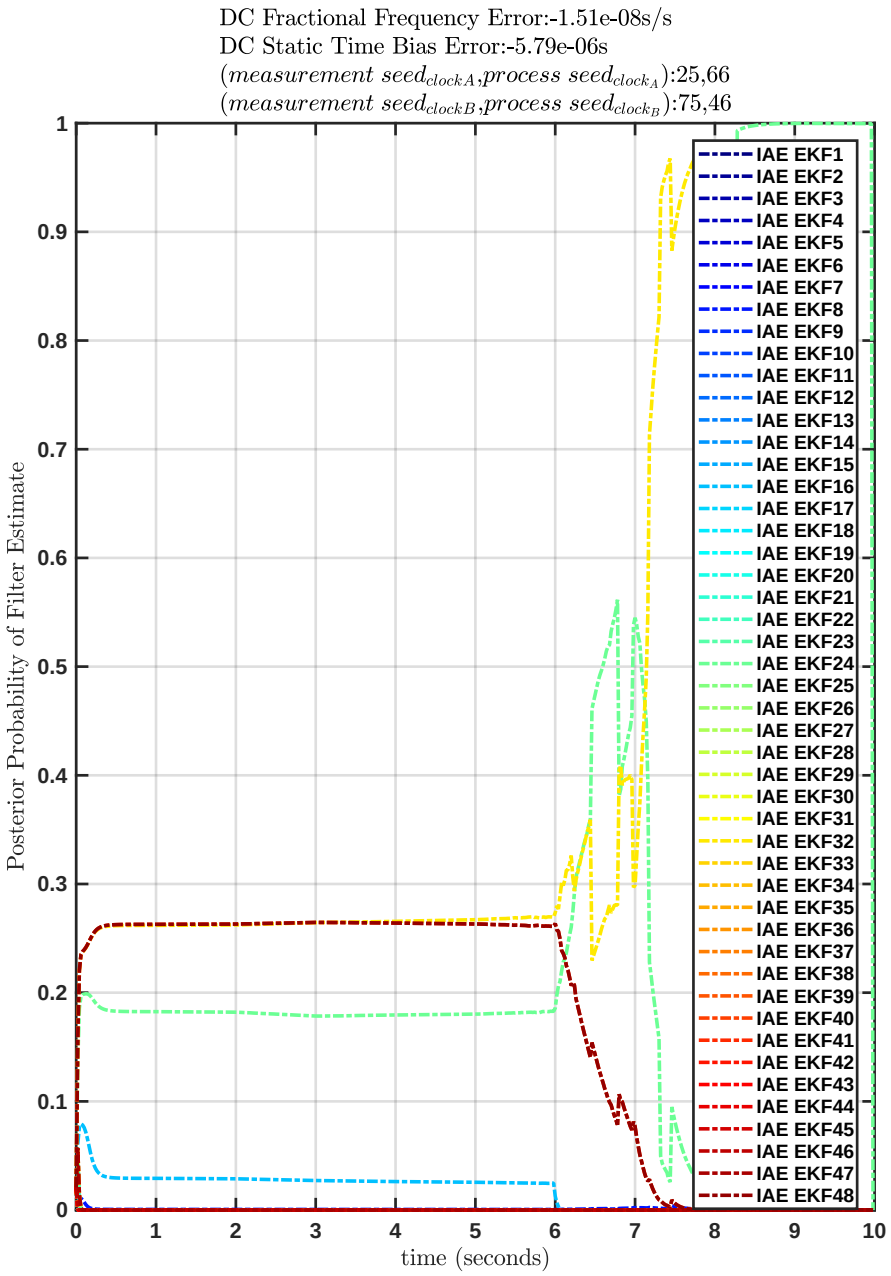

Figure B.89: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:-9.04e-08s/s, DC Static Time Bias Error:-1.03e-06s
($measurement\ seed_{clockA}$,$process\ seed_{clockA}$):83,3, ($measurement\ seed_{clockB}$,$process\ seed_{clockB}$):62,33

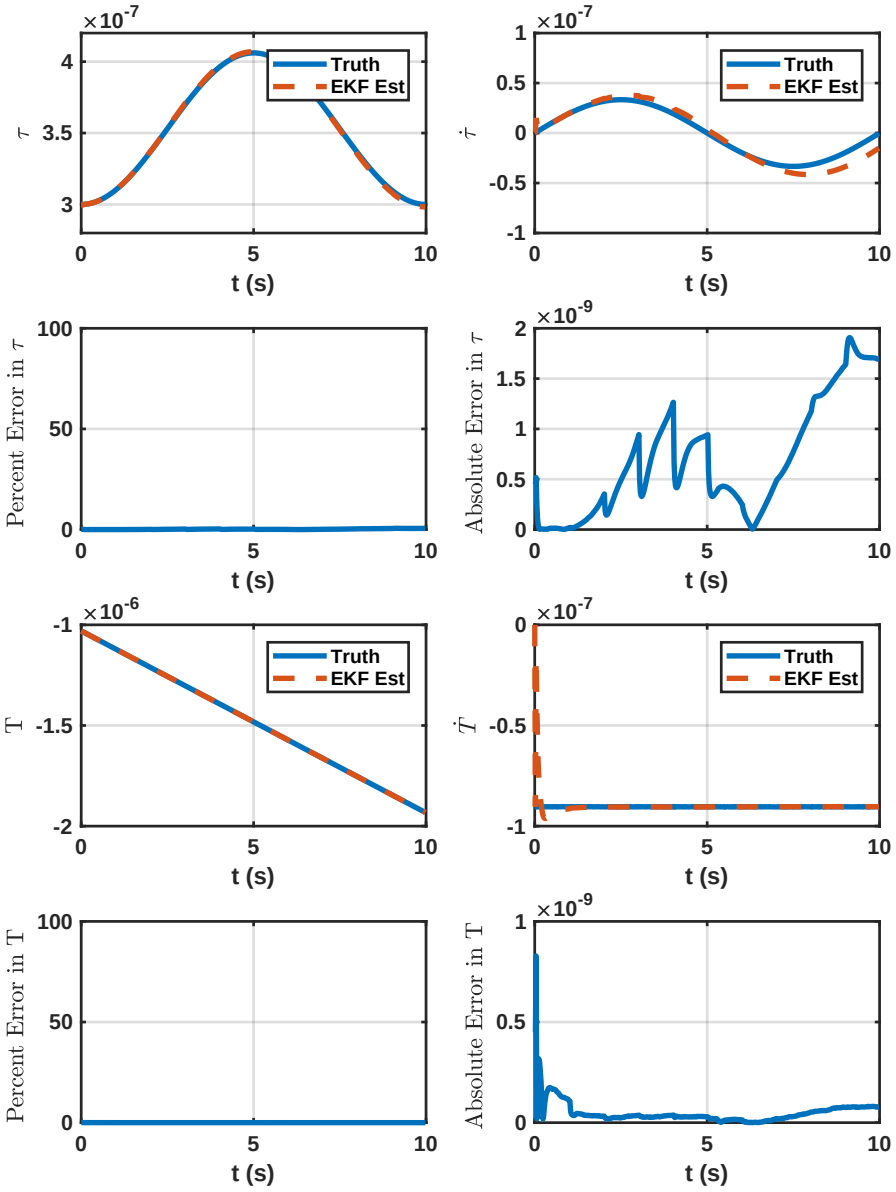

Figure B.90: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

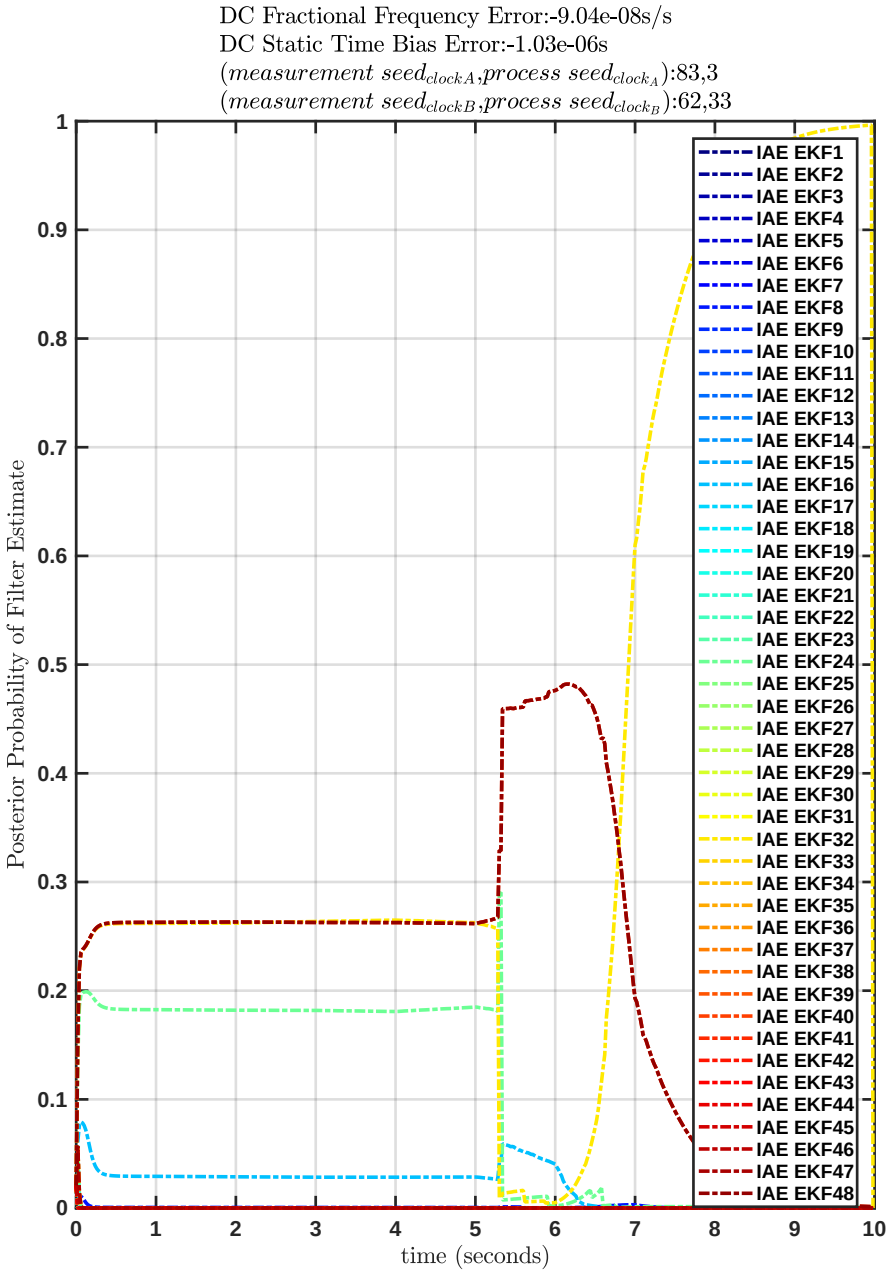

Figure B.91: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

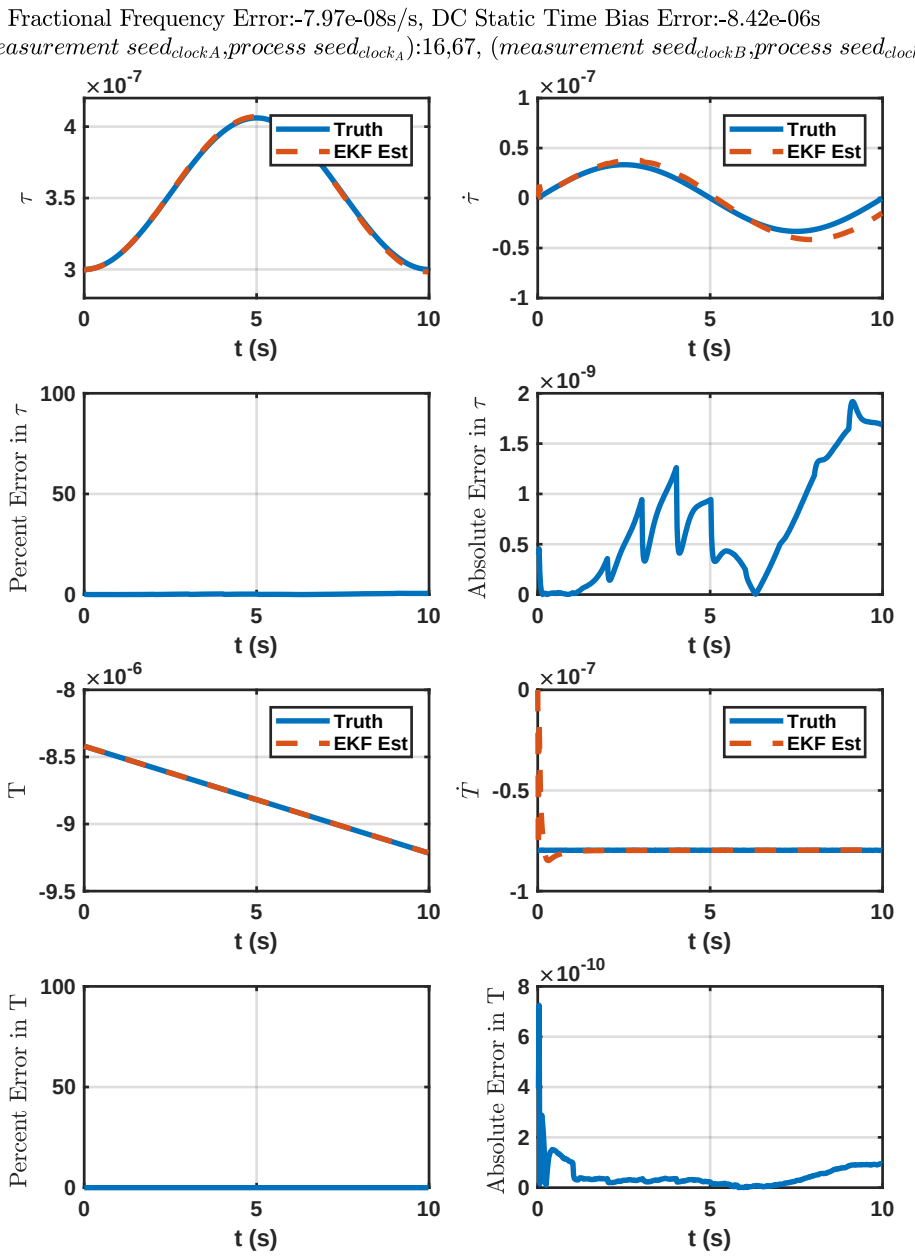

Figure B.92: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

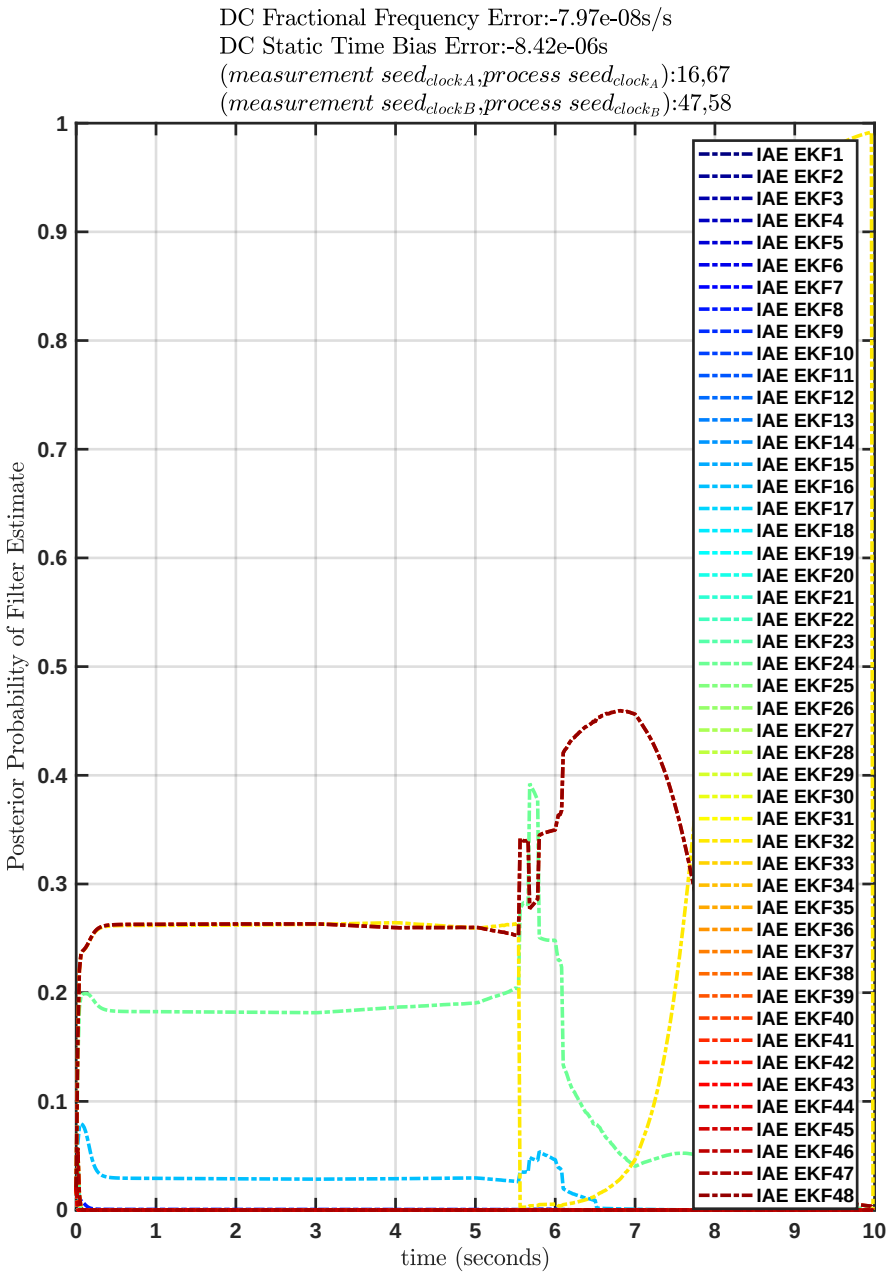

Figure B.93: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:-7.08e-08s/s, DC Static Time Bias Error:-9.6e-07s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):75,53, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):57,48

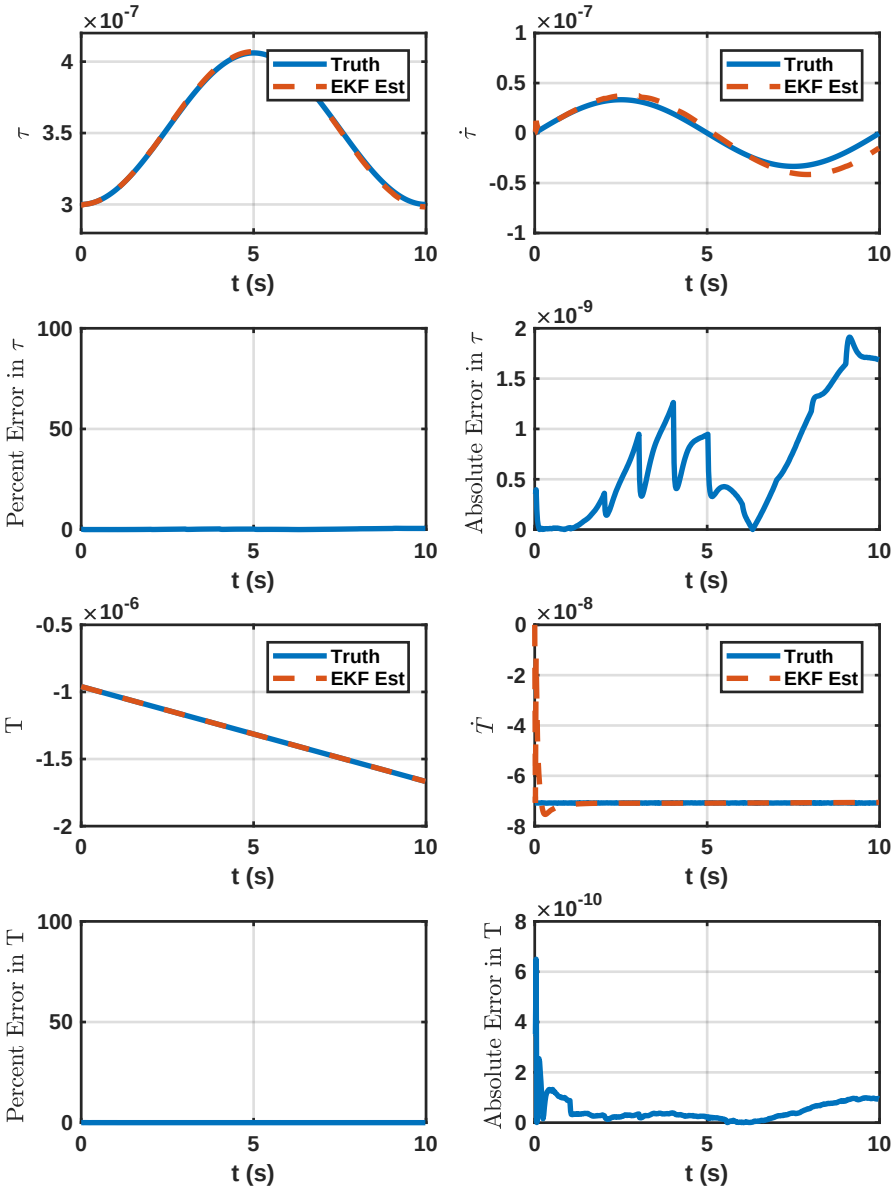

Figure B.94: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

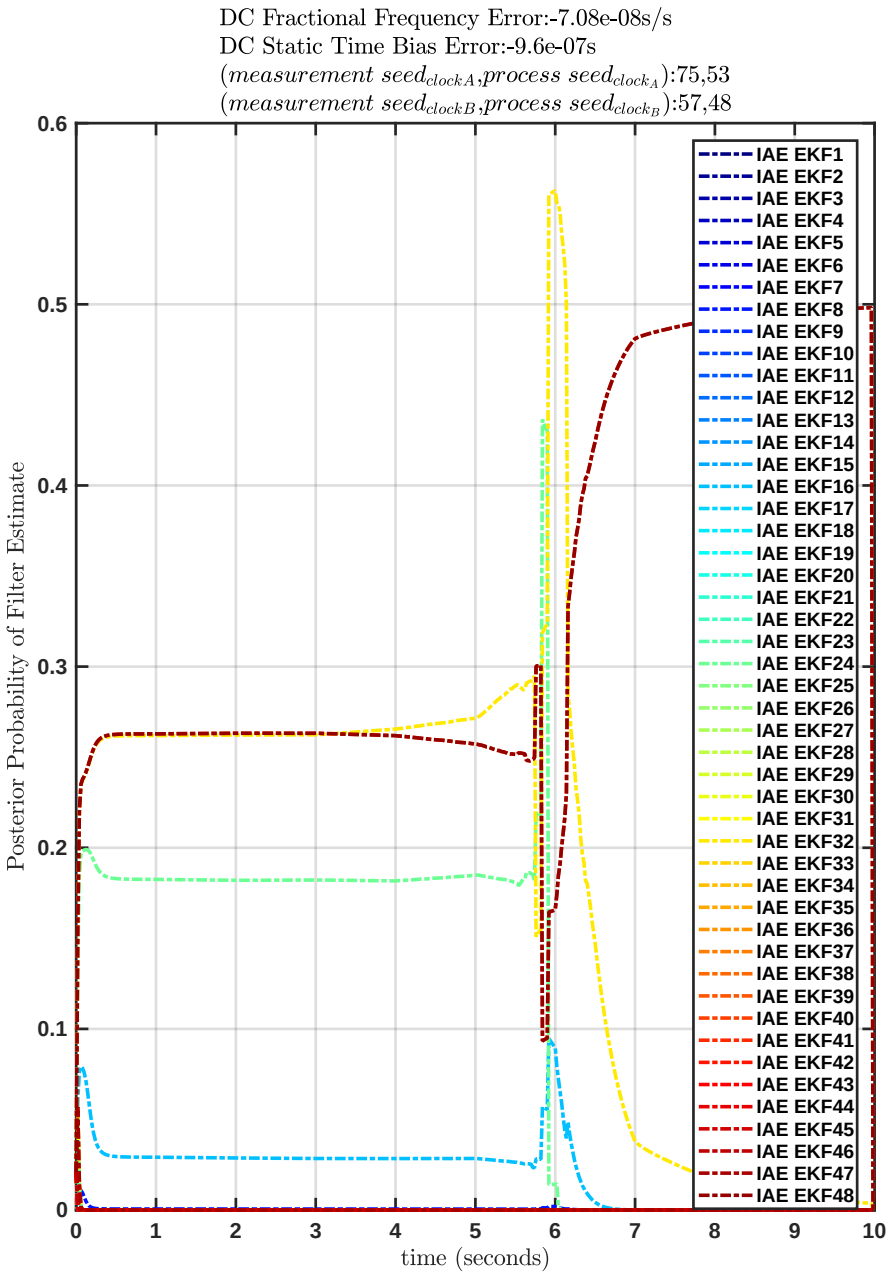

Figure B.95: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

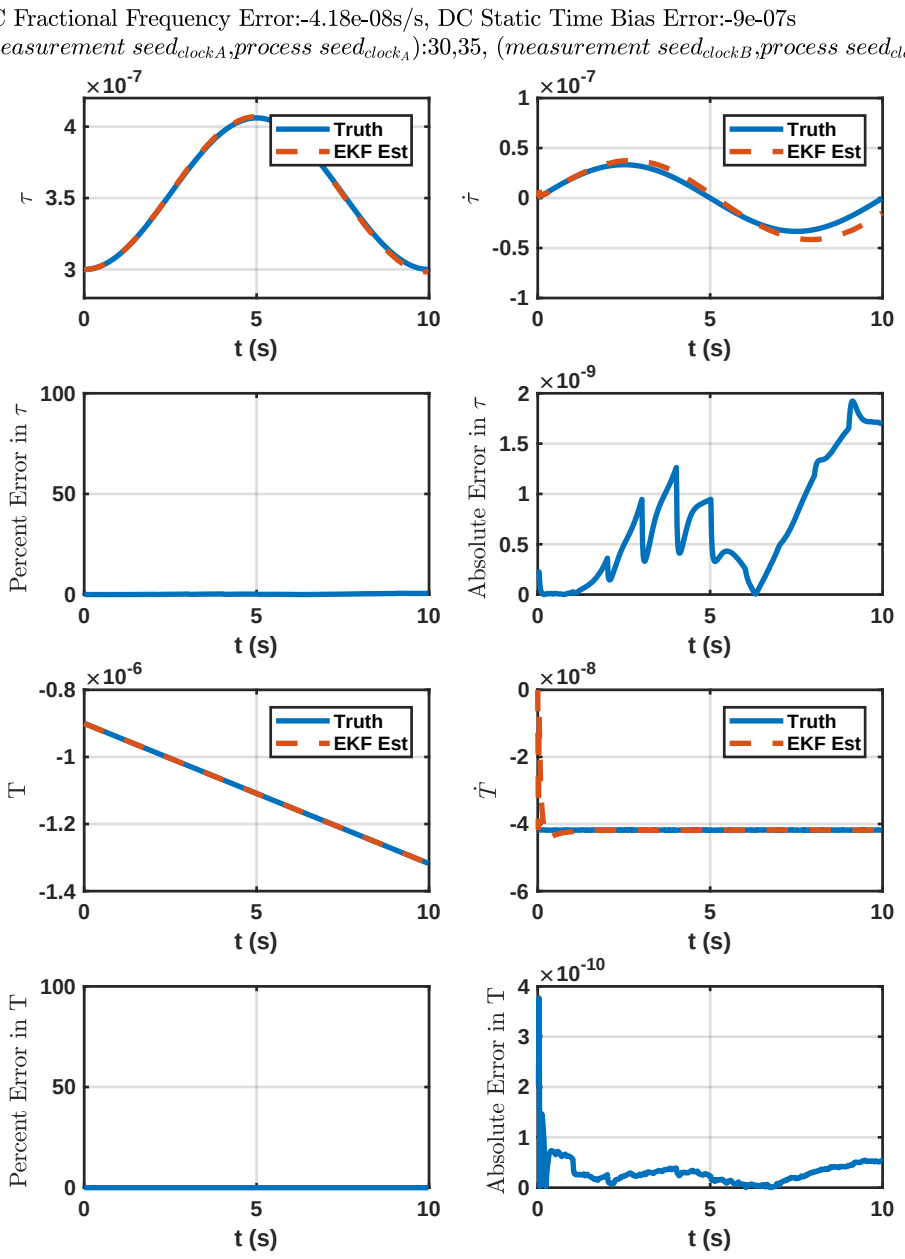

Figure B.96: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

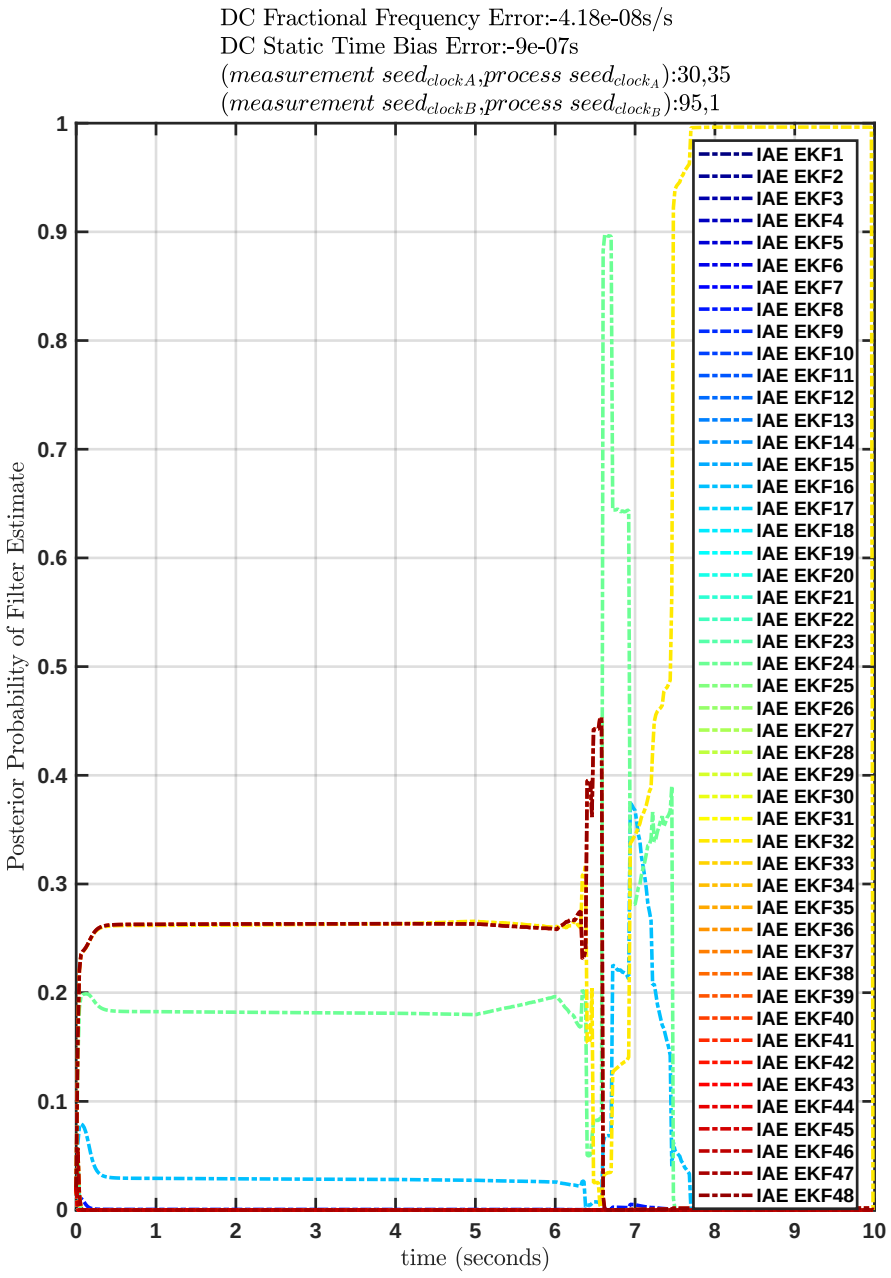

Figure B.97: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:-5.38e-08s/s, DC Static Time Bias Error:-9.5e-07s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):13,89, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):53,33

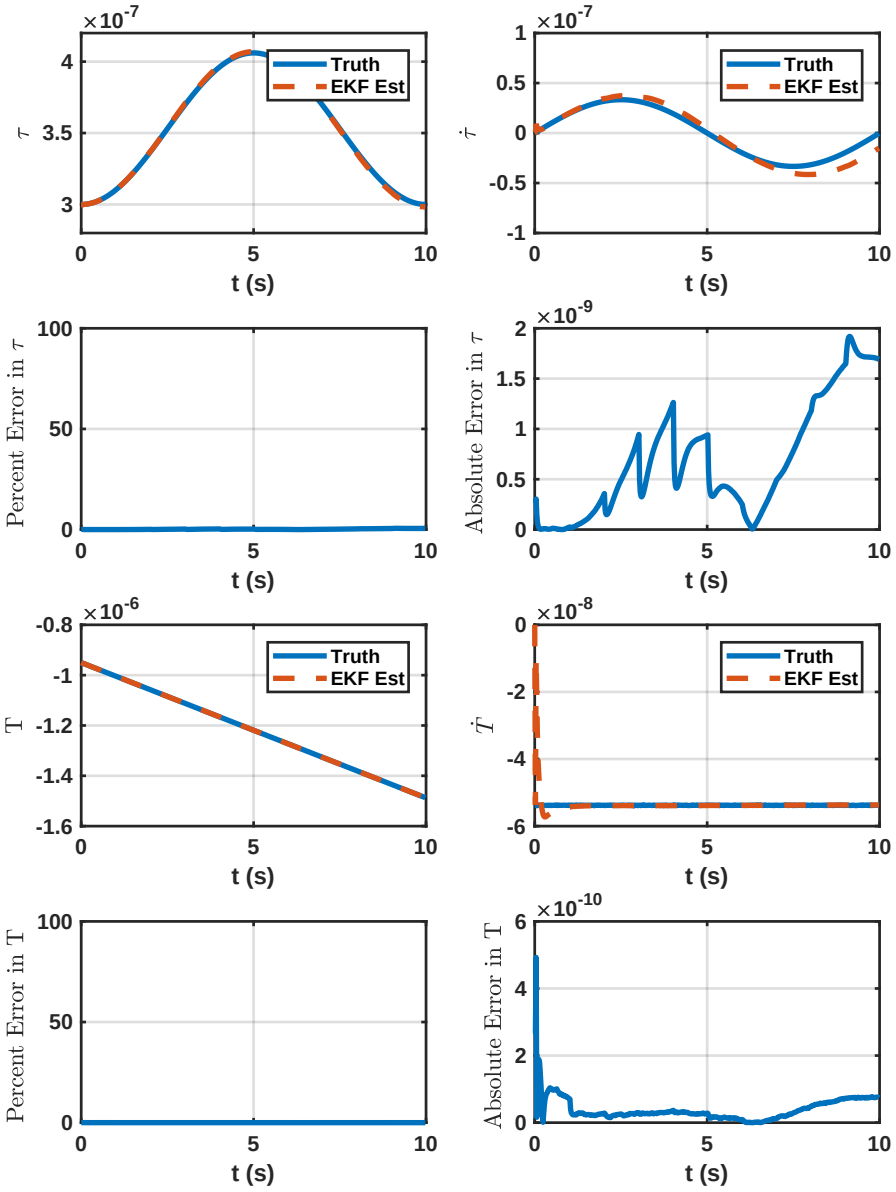

Figure B.98: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

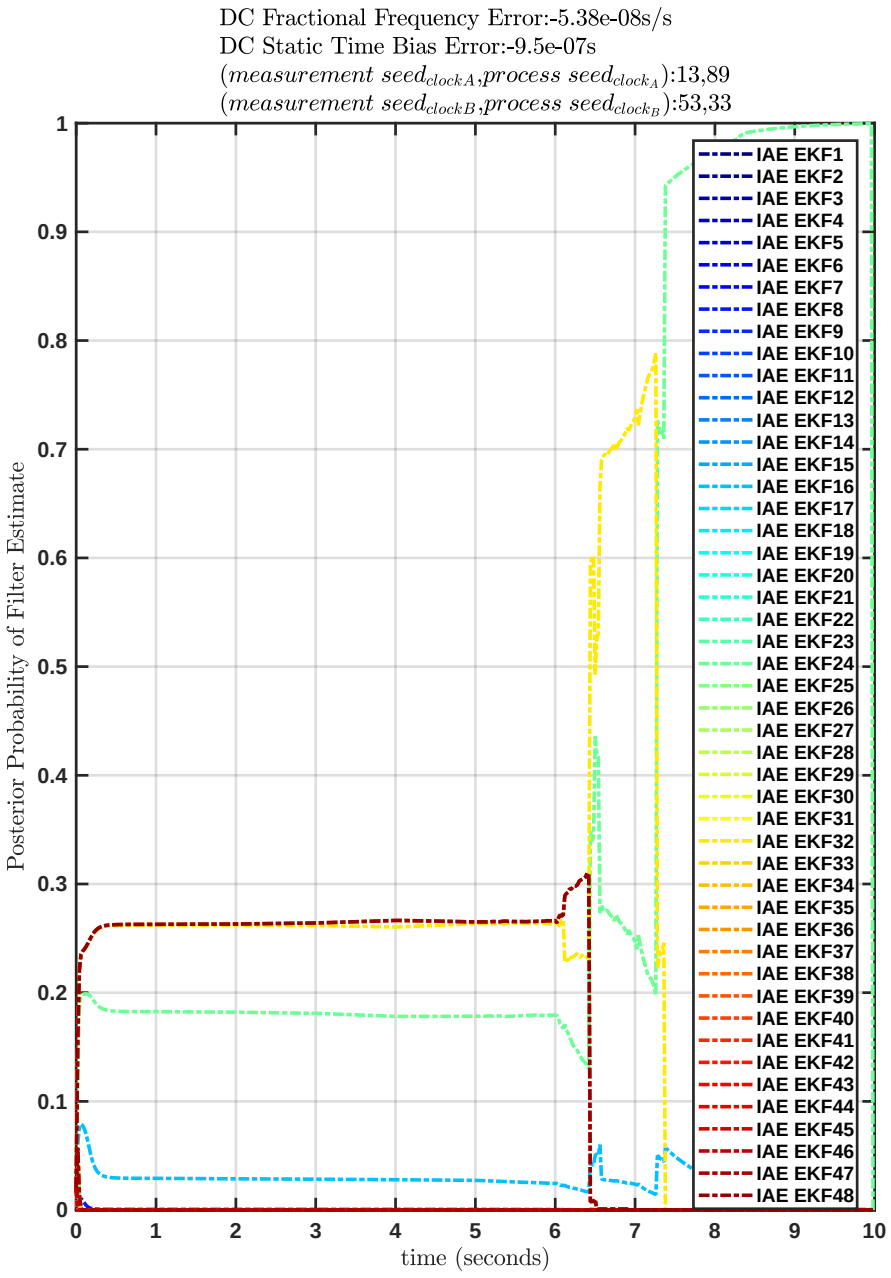

Figure B.99: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:-1.59e-08s/s, DC Static Time Bias Error:-7.61e-06s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):30,50, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):11,30

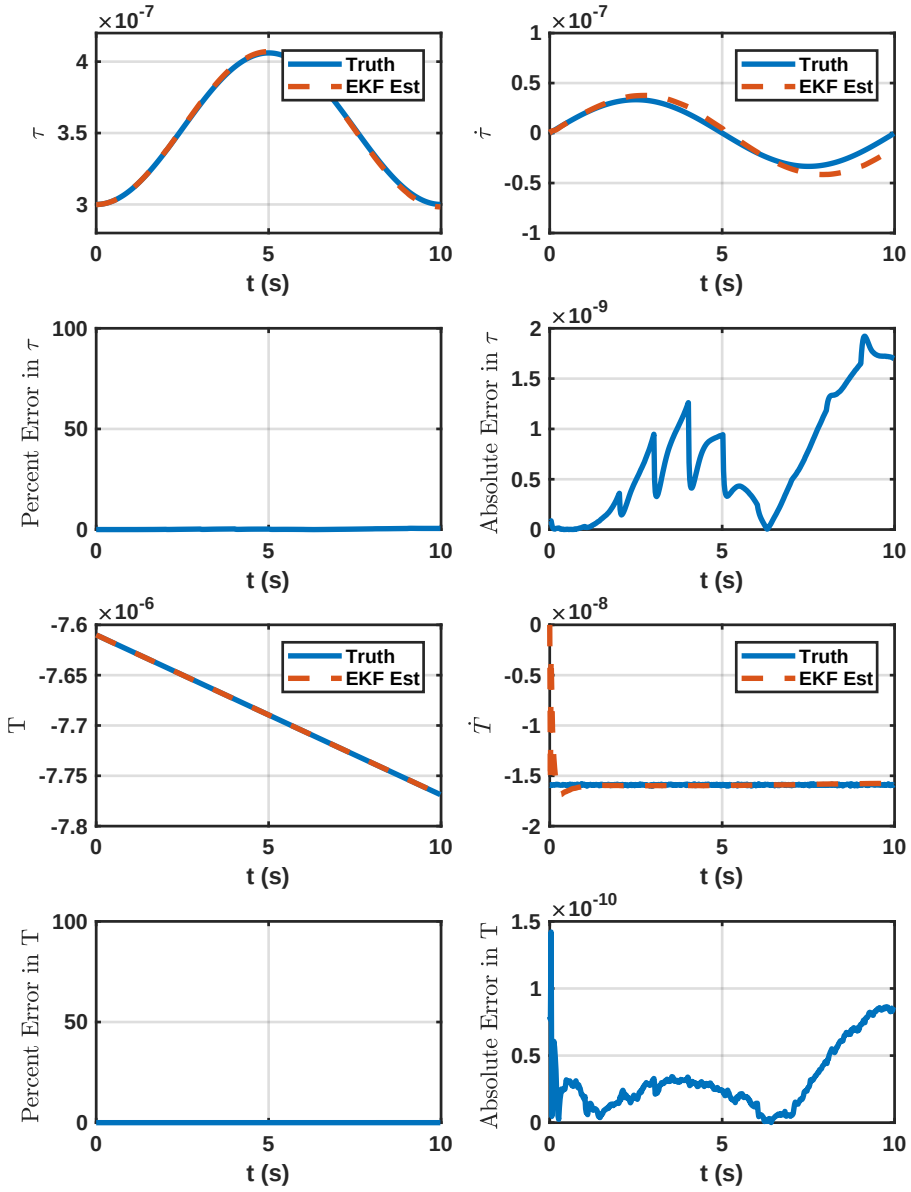

Figure B.100: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

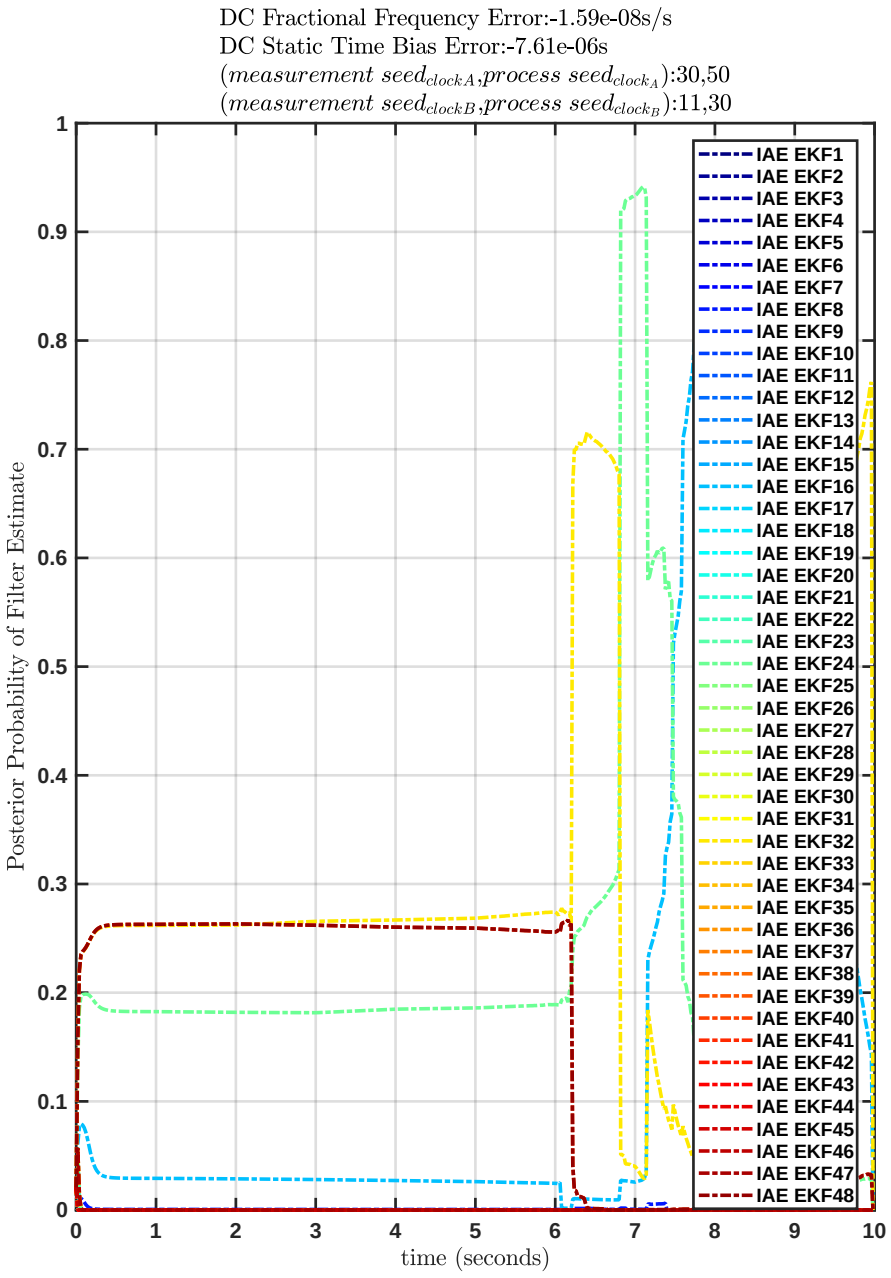

Figure B.101: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

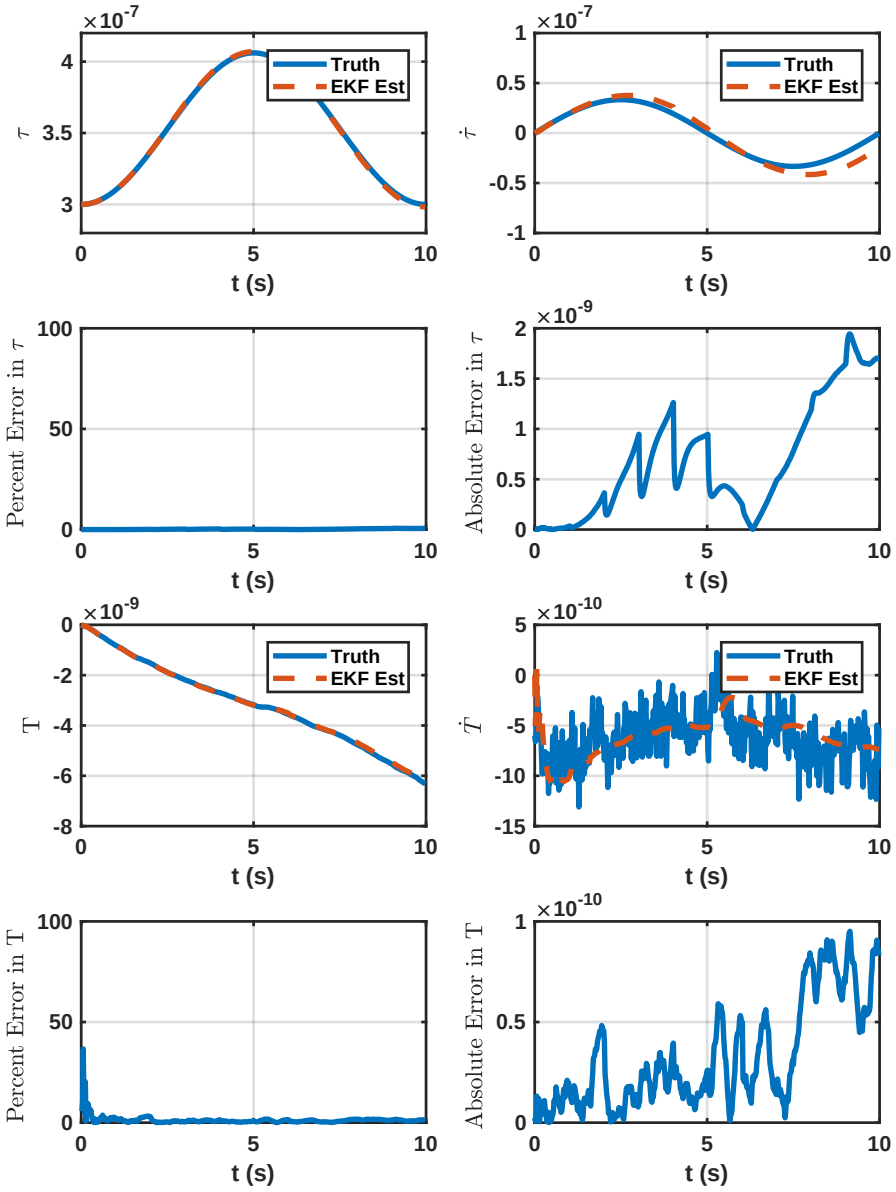

Figure B.102: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

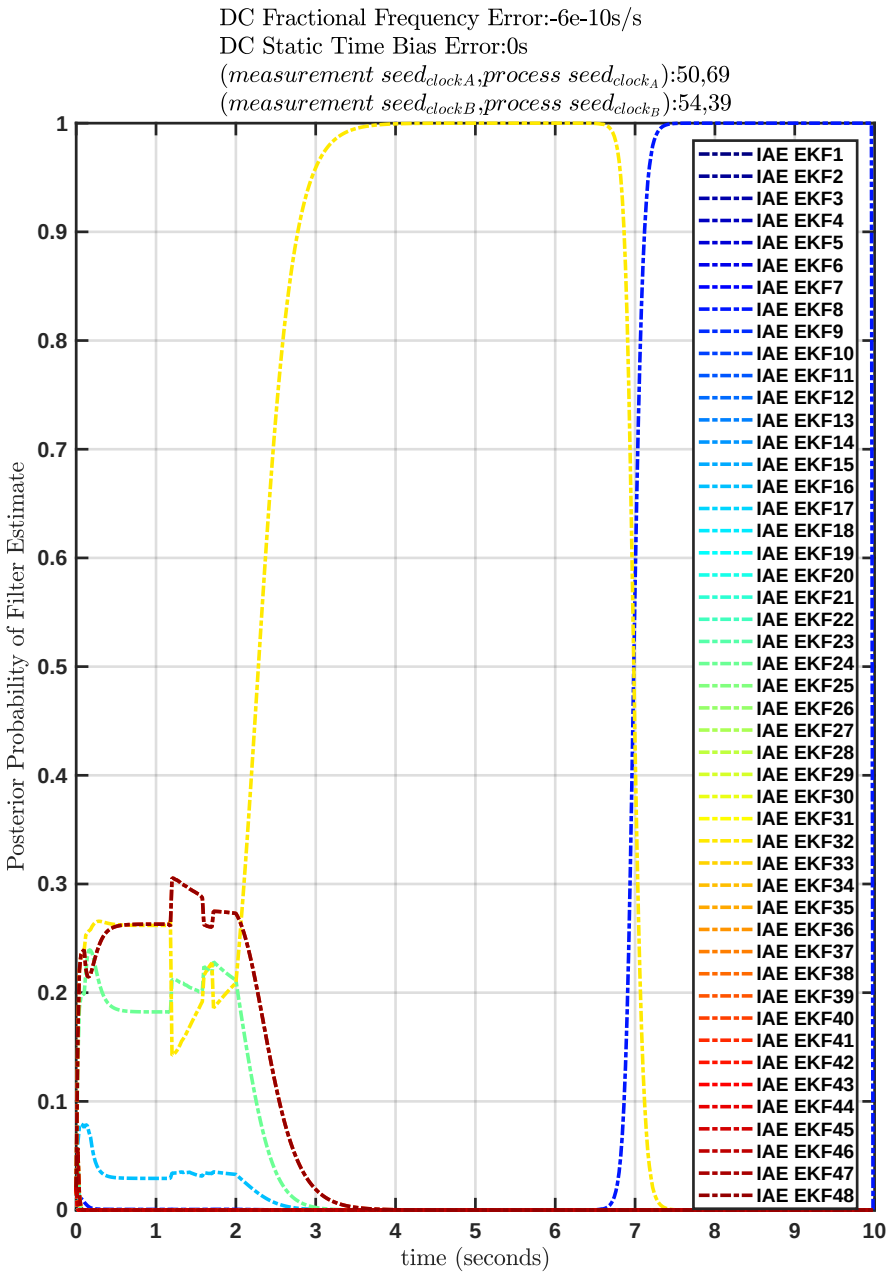

Figure B.103: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:0s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):82,50, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):73,86

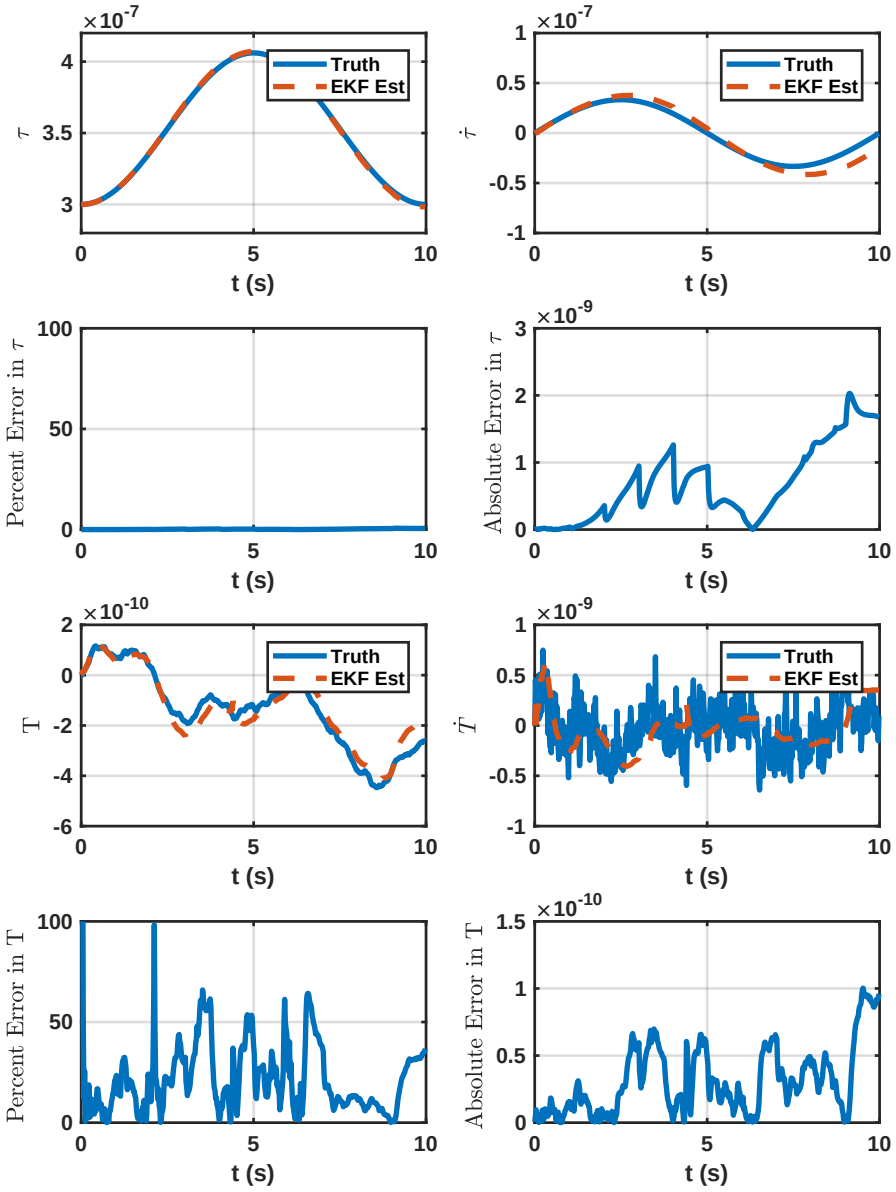

Figure B.104: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

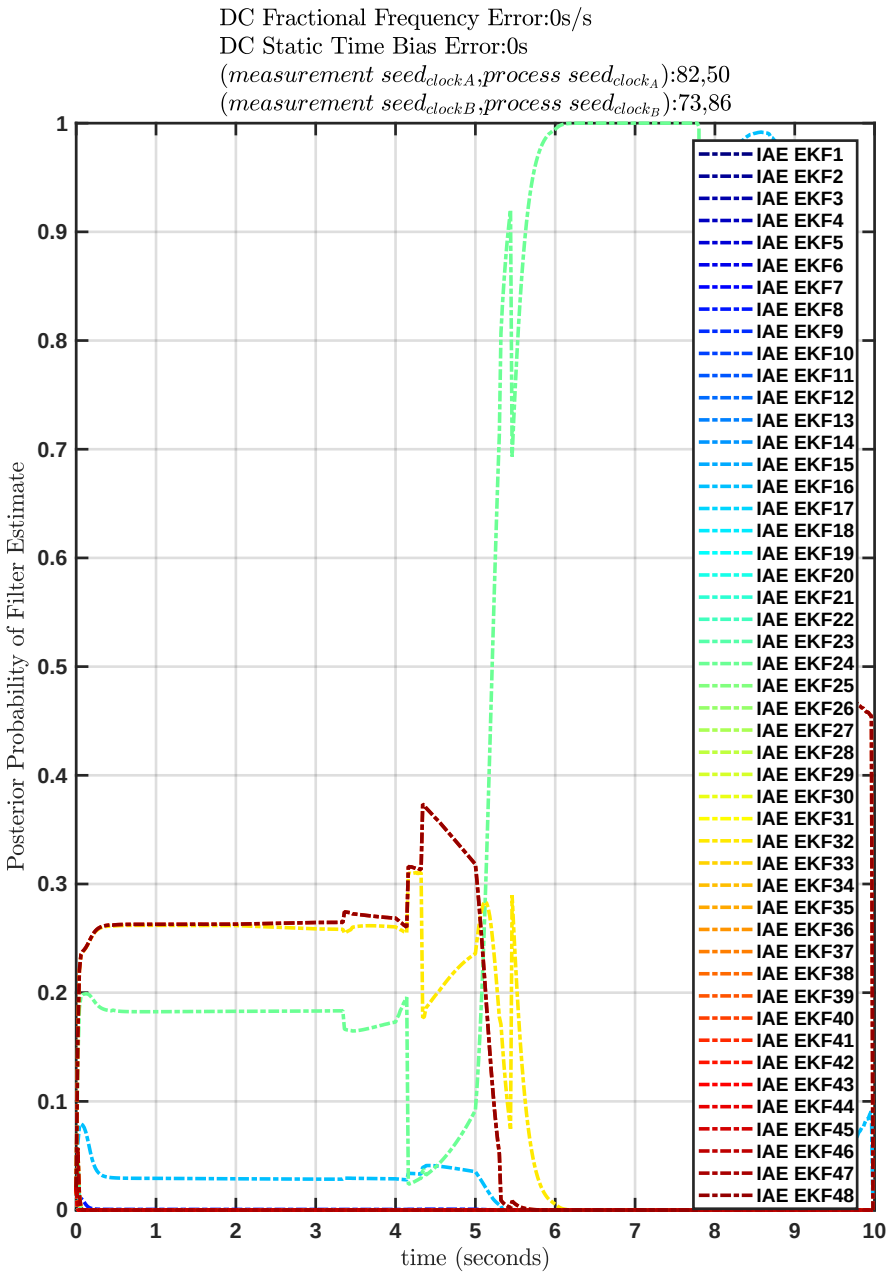

Figure B.105: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:-7e-10s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):43,100, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):67,86

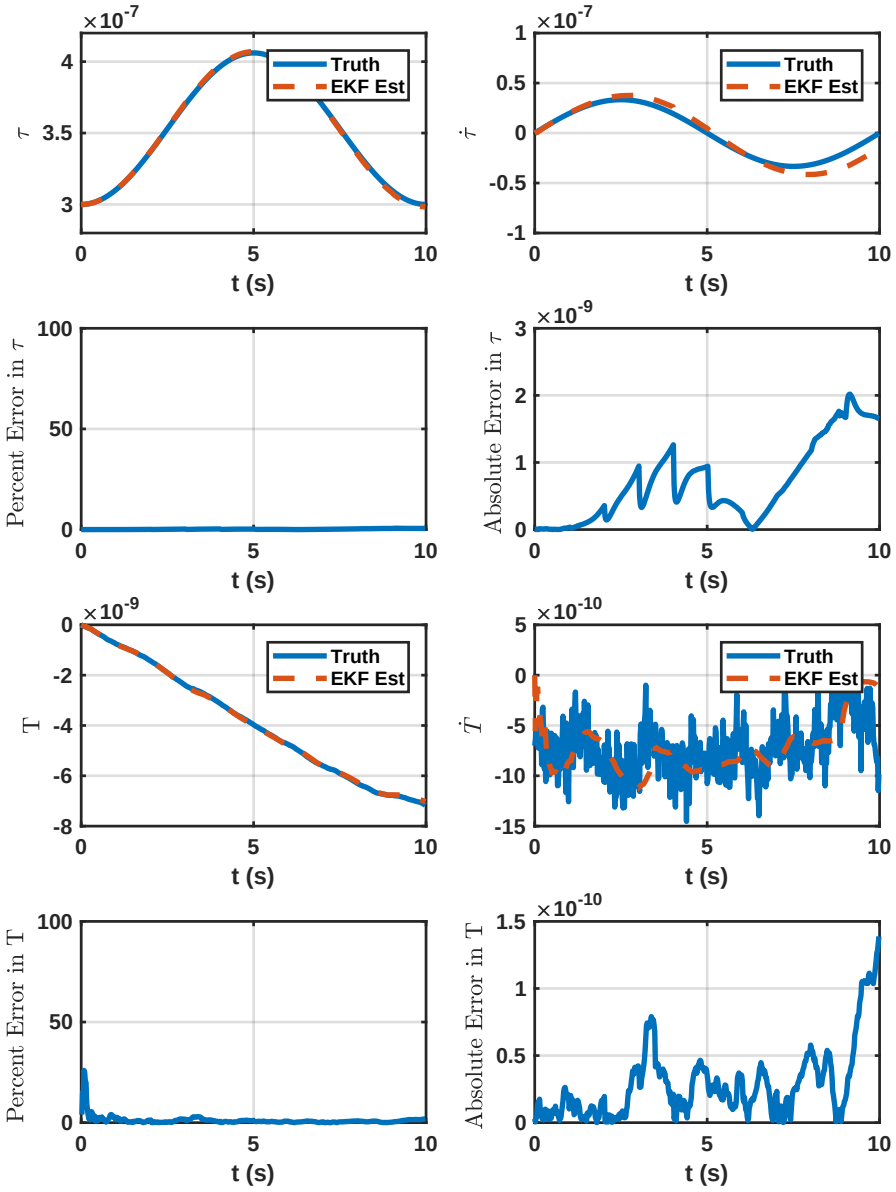

Figure B.106: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

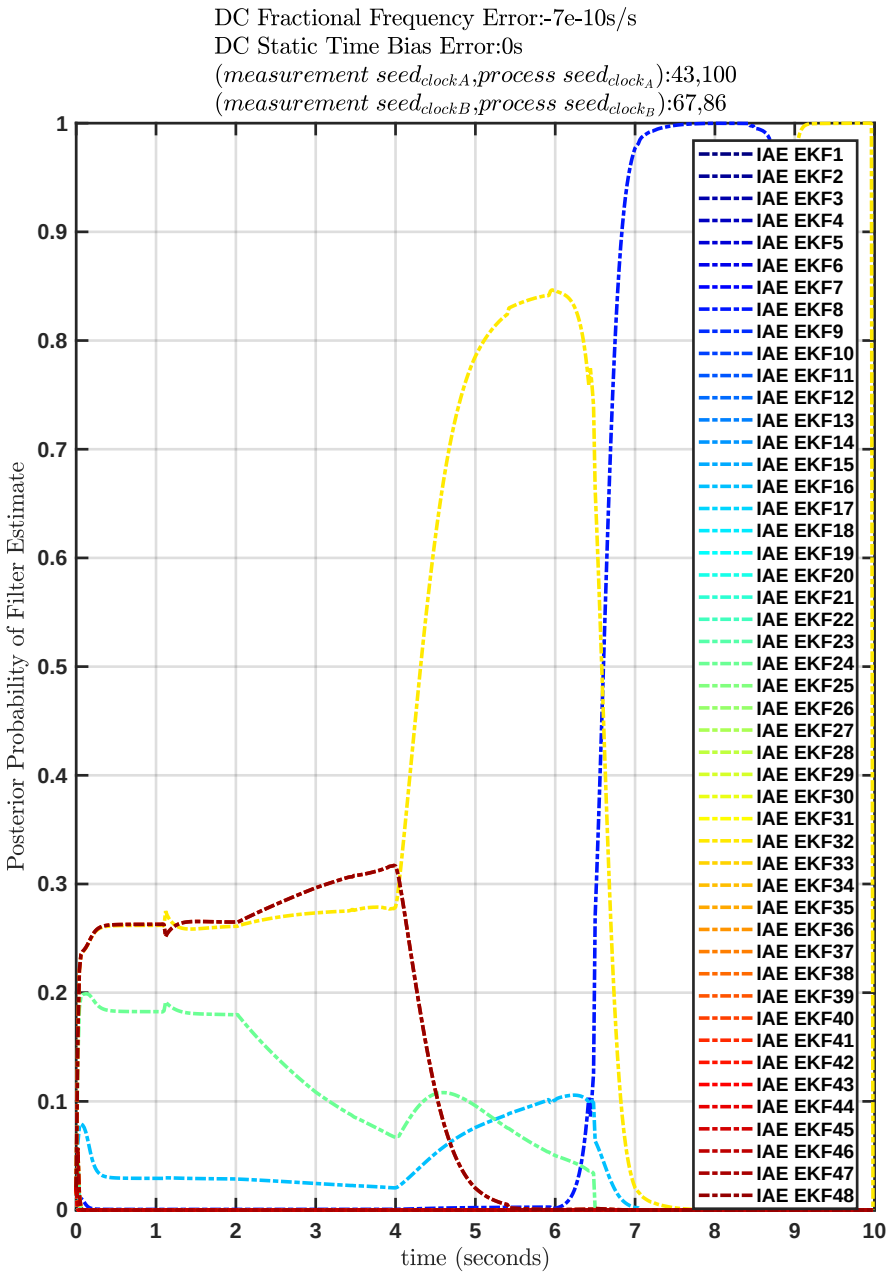

Figure B.107: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:-6e-10s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):50,69, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):54,39

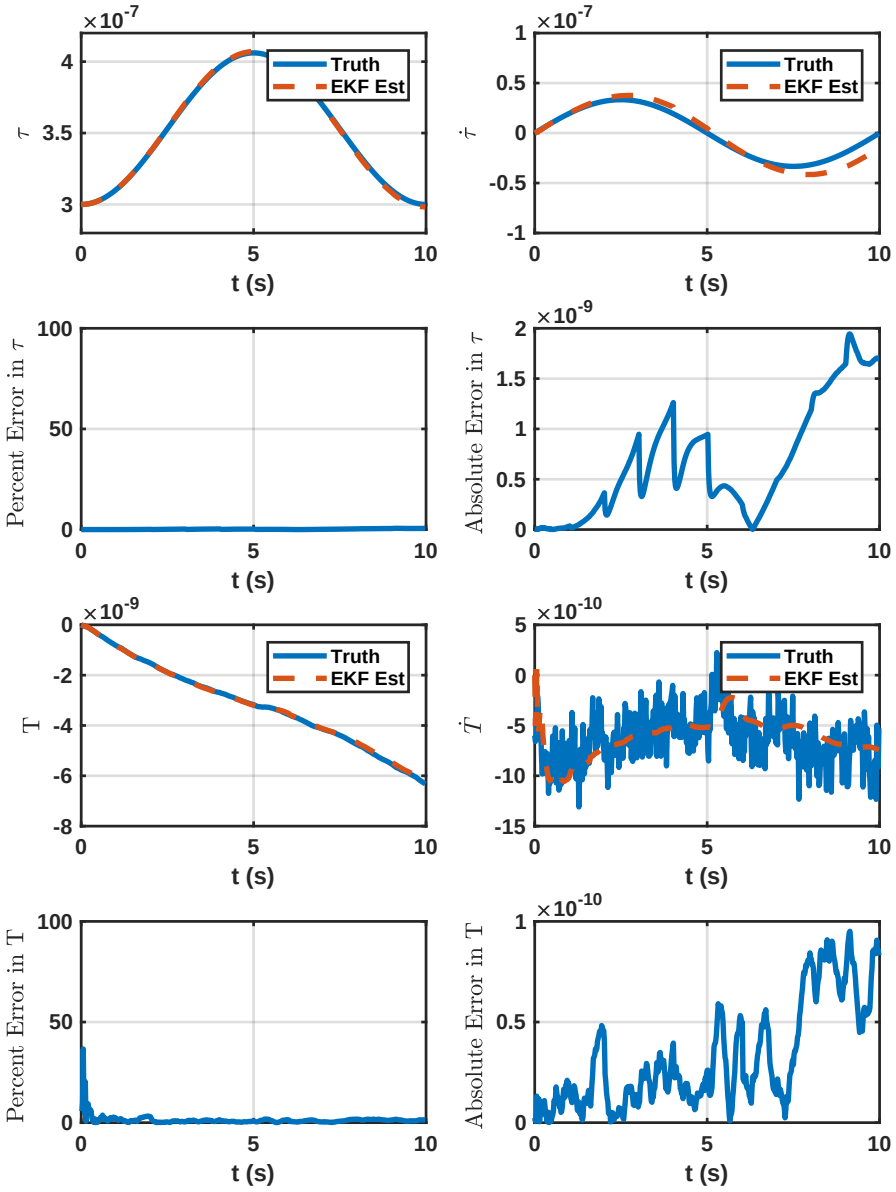

Figure B.108: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

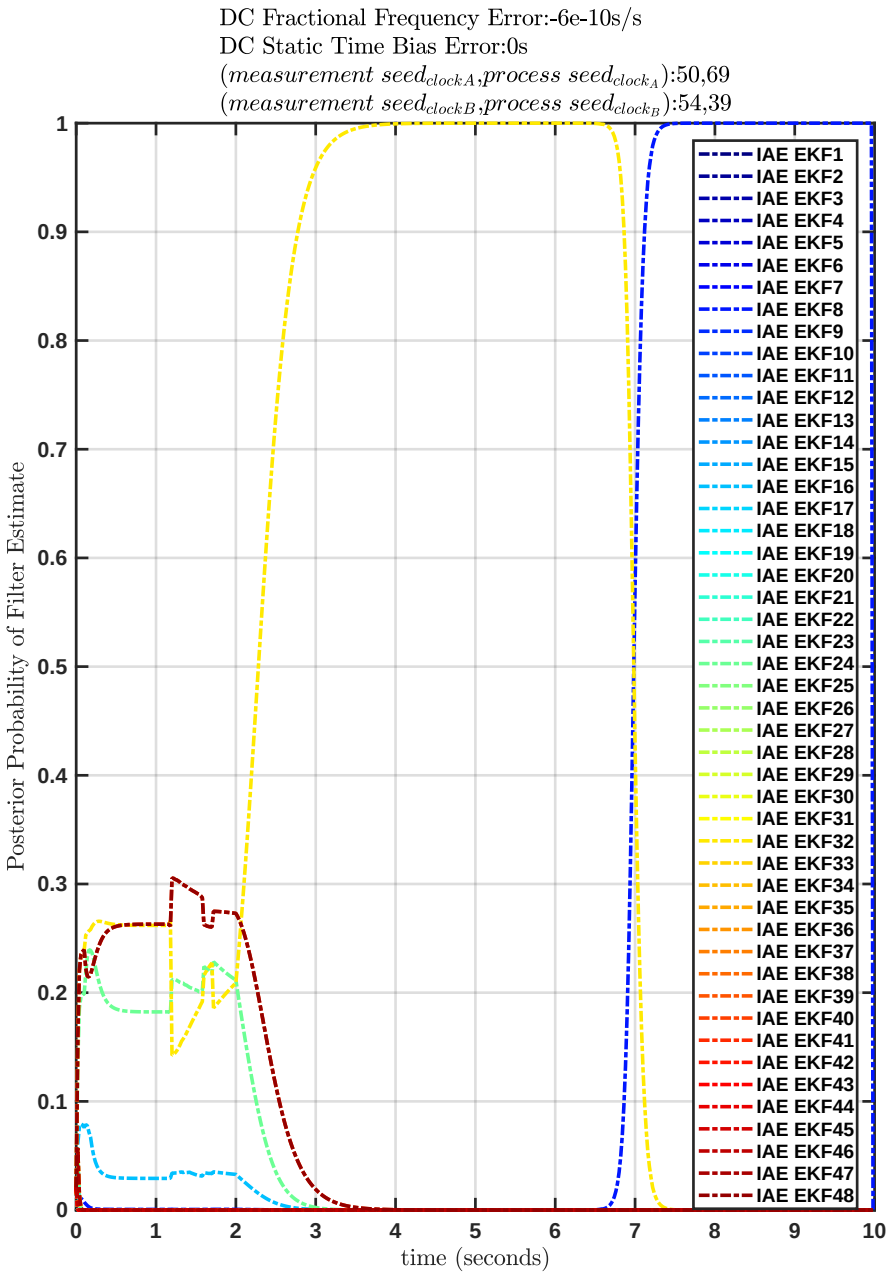

Figure B.109: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:1e-10s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):38,67, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):44,63

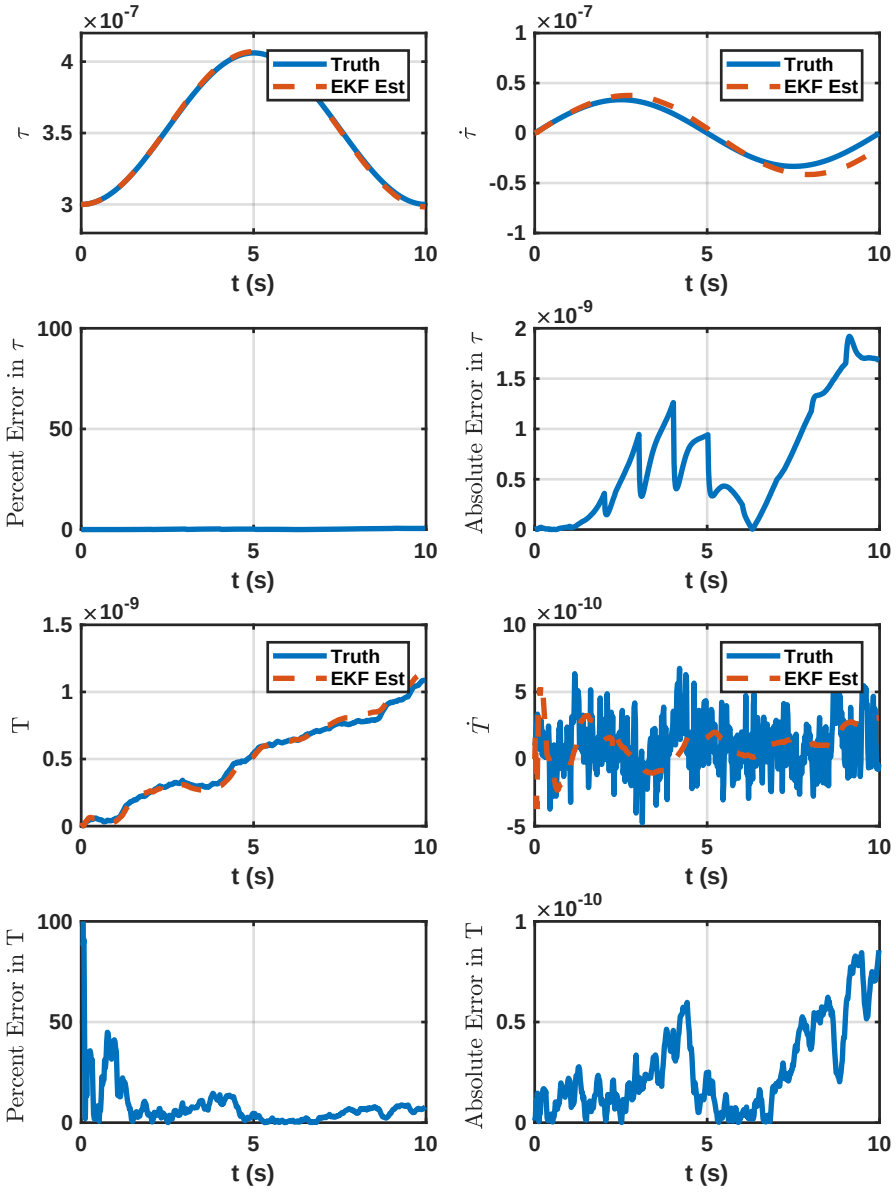

Figure B.110: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

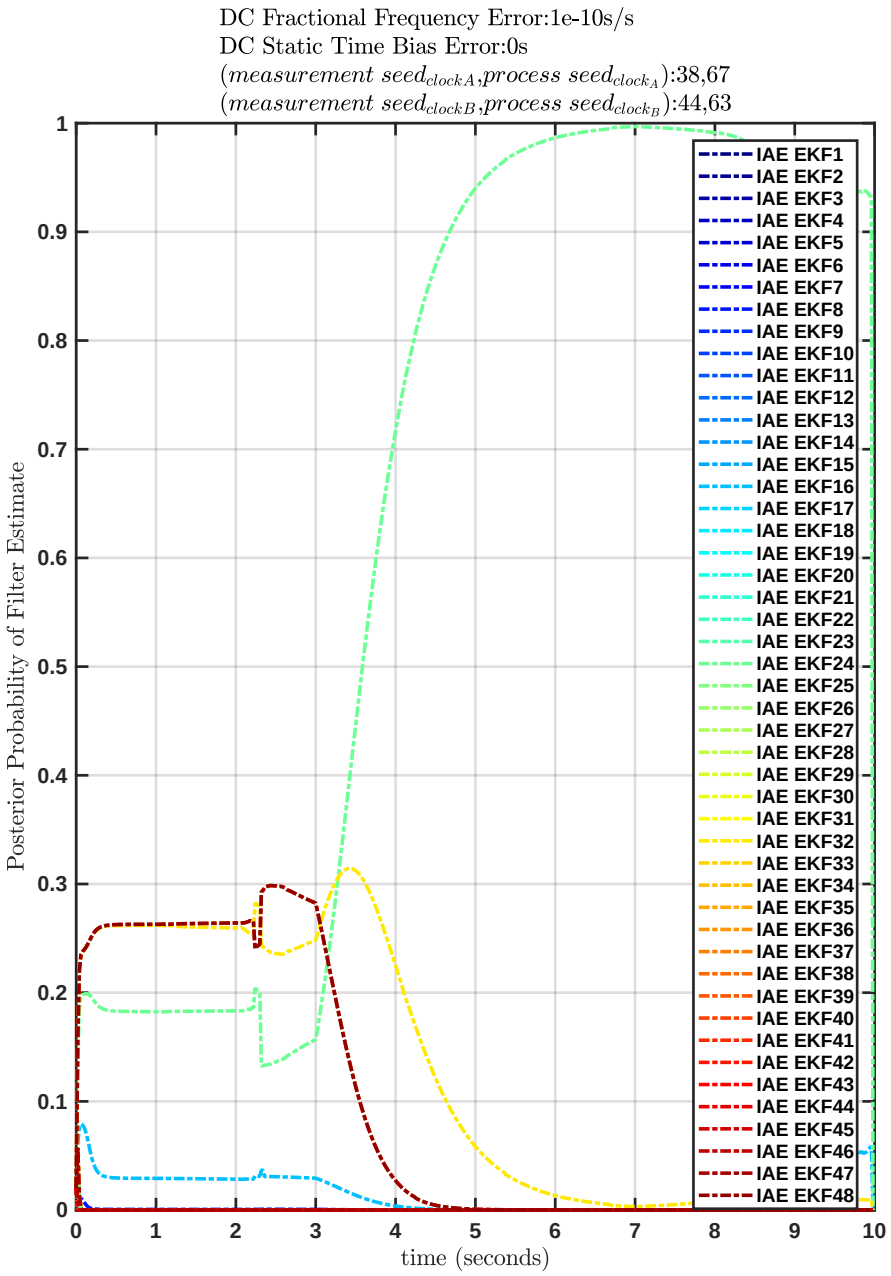

Figure B.111: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:0s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):40,59, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):68,49

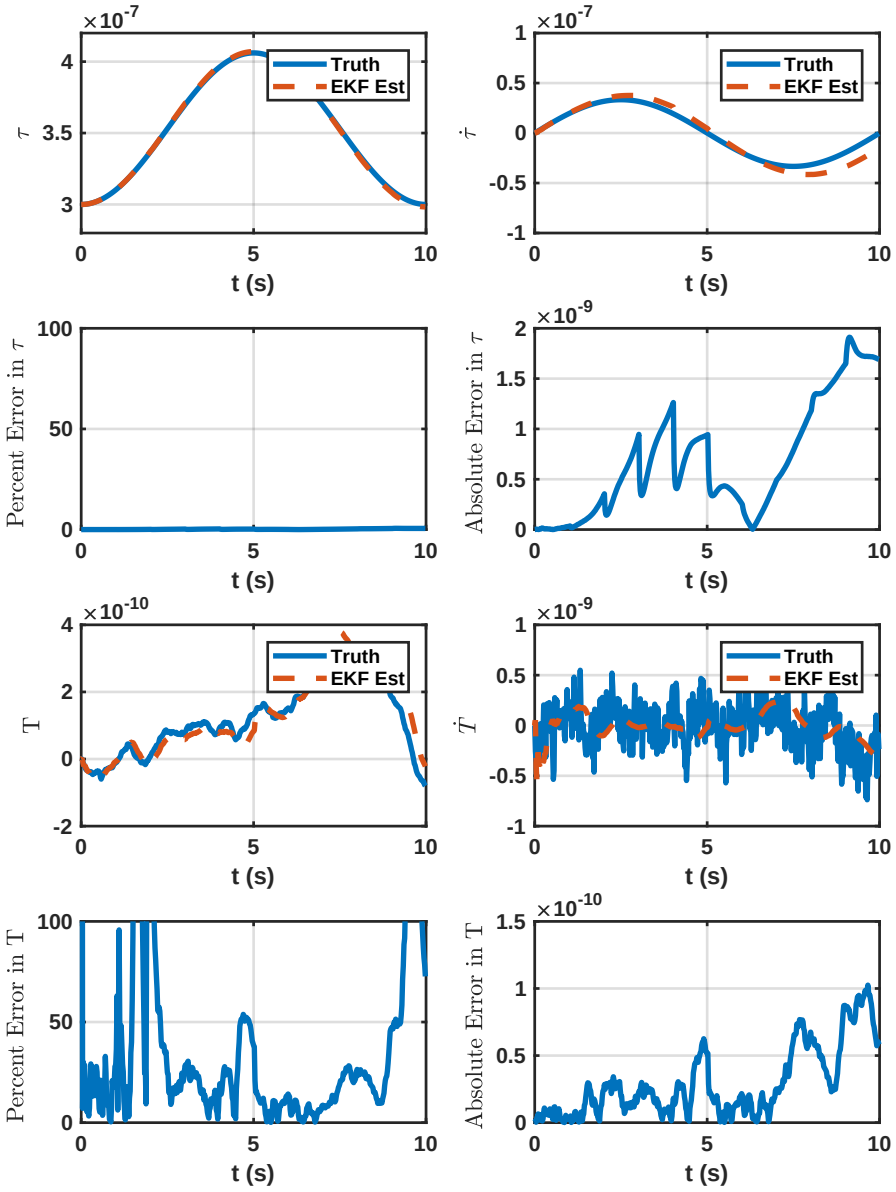

Figure B.112: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

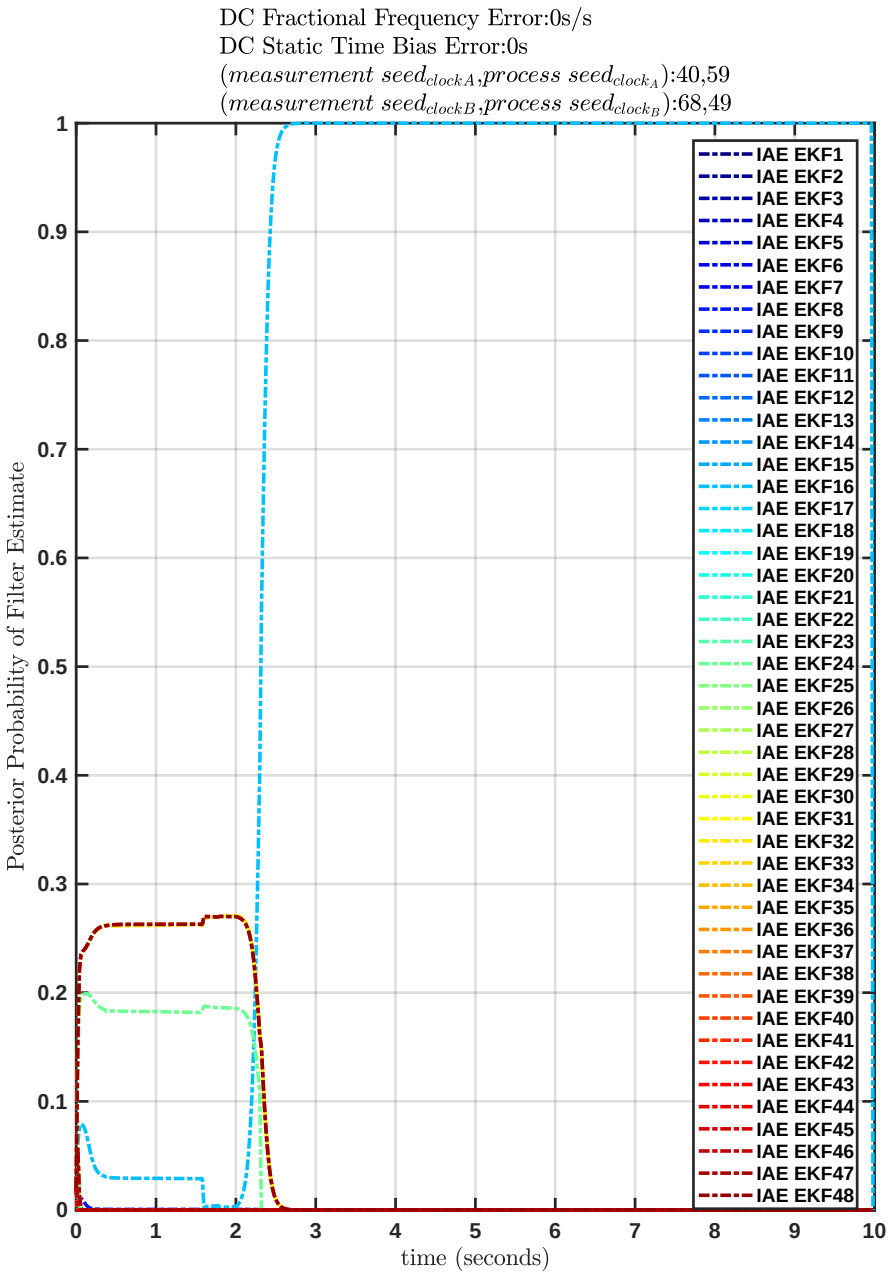

Figure B.113: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:1e-09s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):42,50, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):62,79

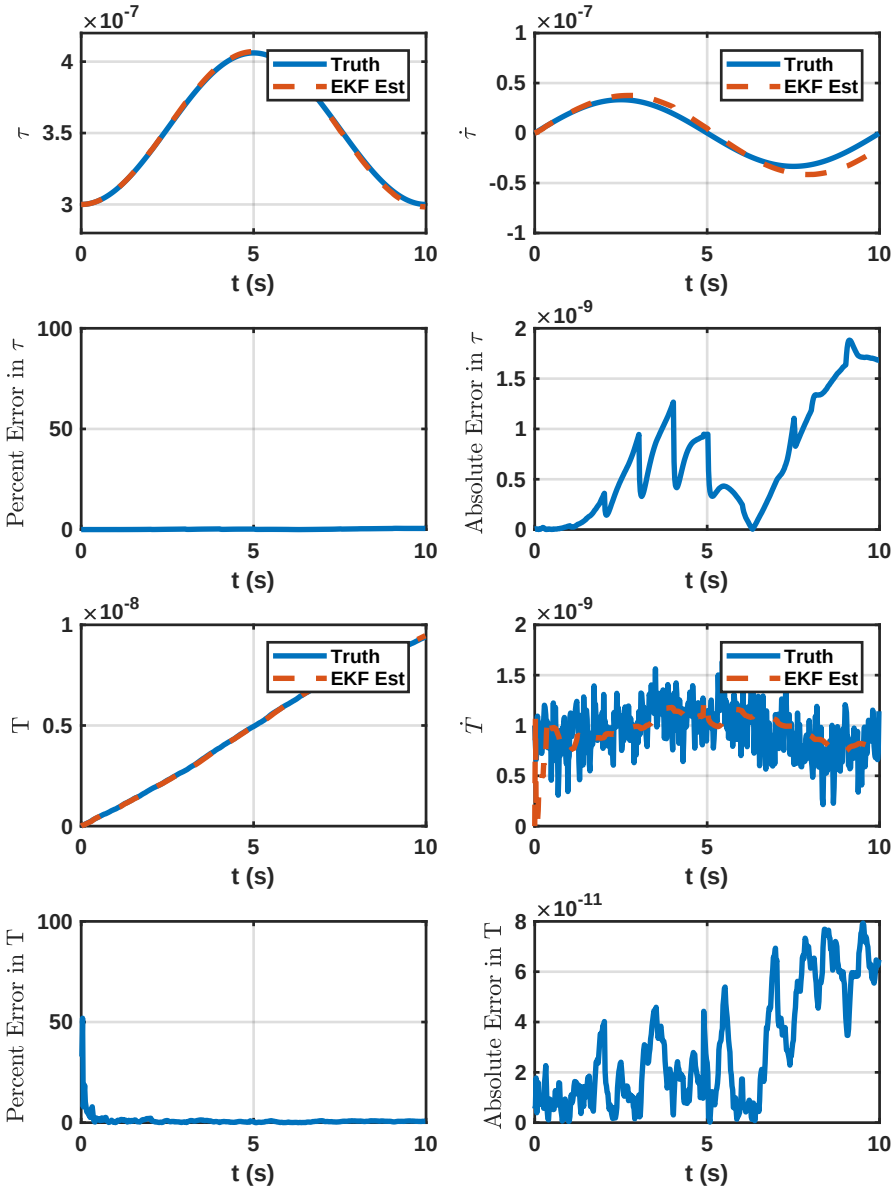

Figure B.114: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

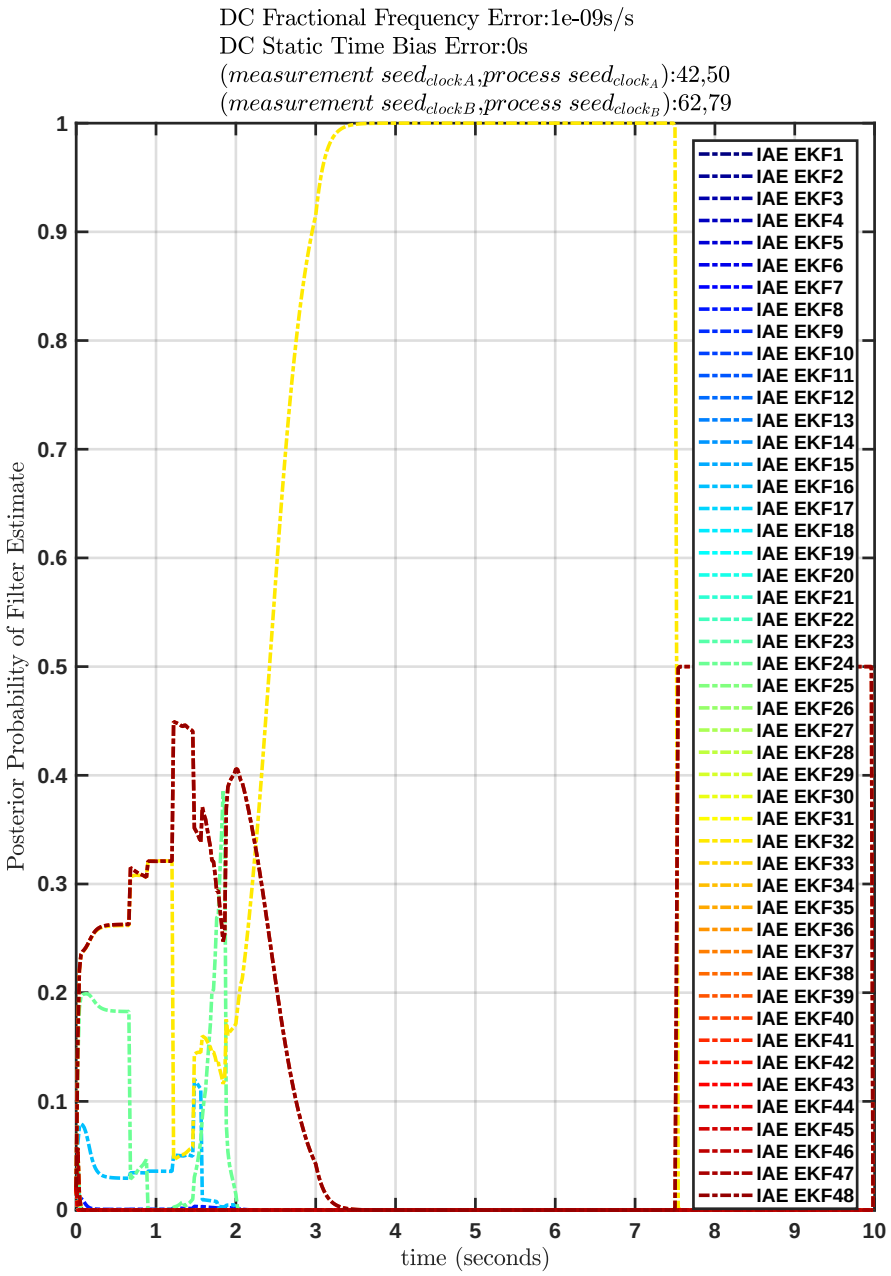

Figure B.115: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:2e-10s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clockA}$,$process\ seed_{clockA}$):49,0, ($measurement\ seed_{clockB}$,$process\ seed_{clockB}$):36,16

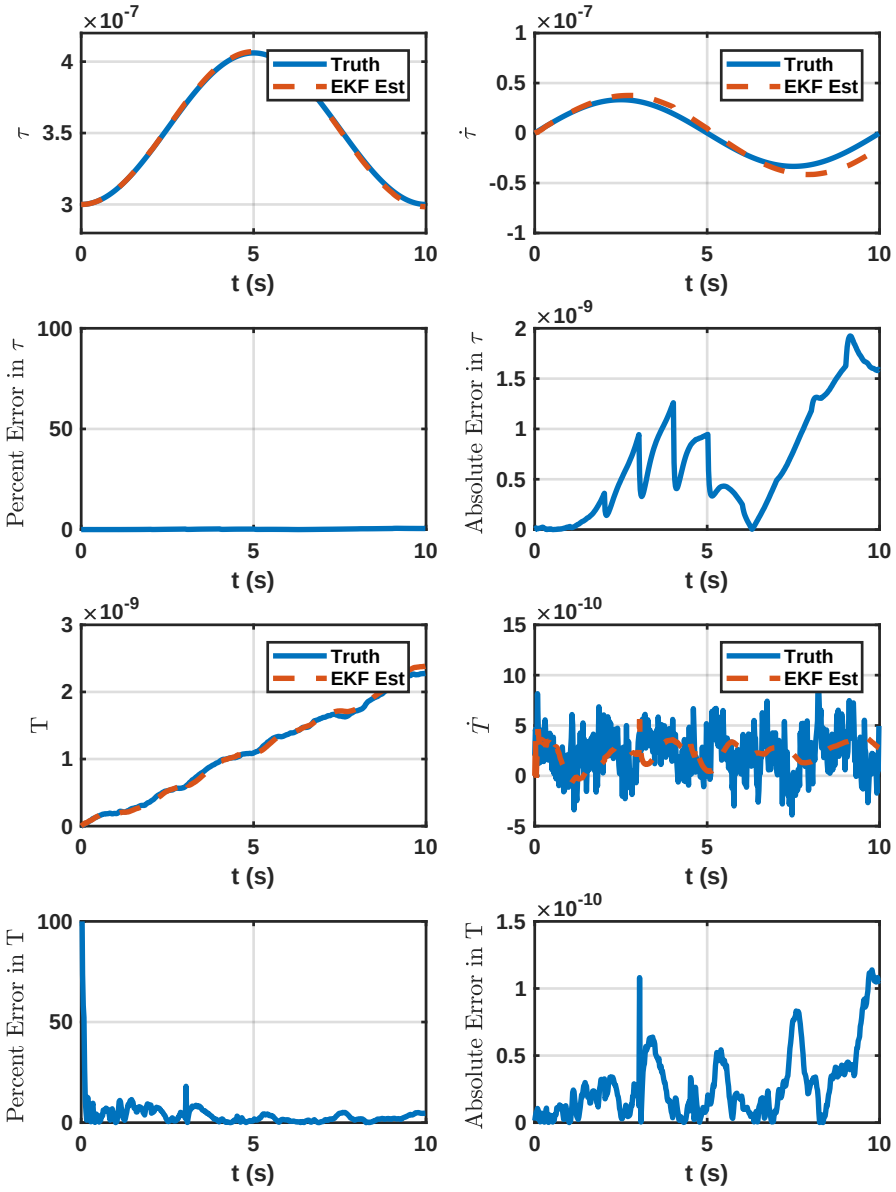

Figure B.116: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

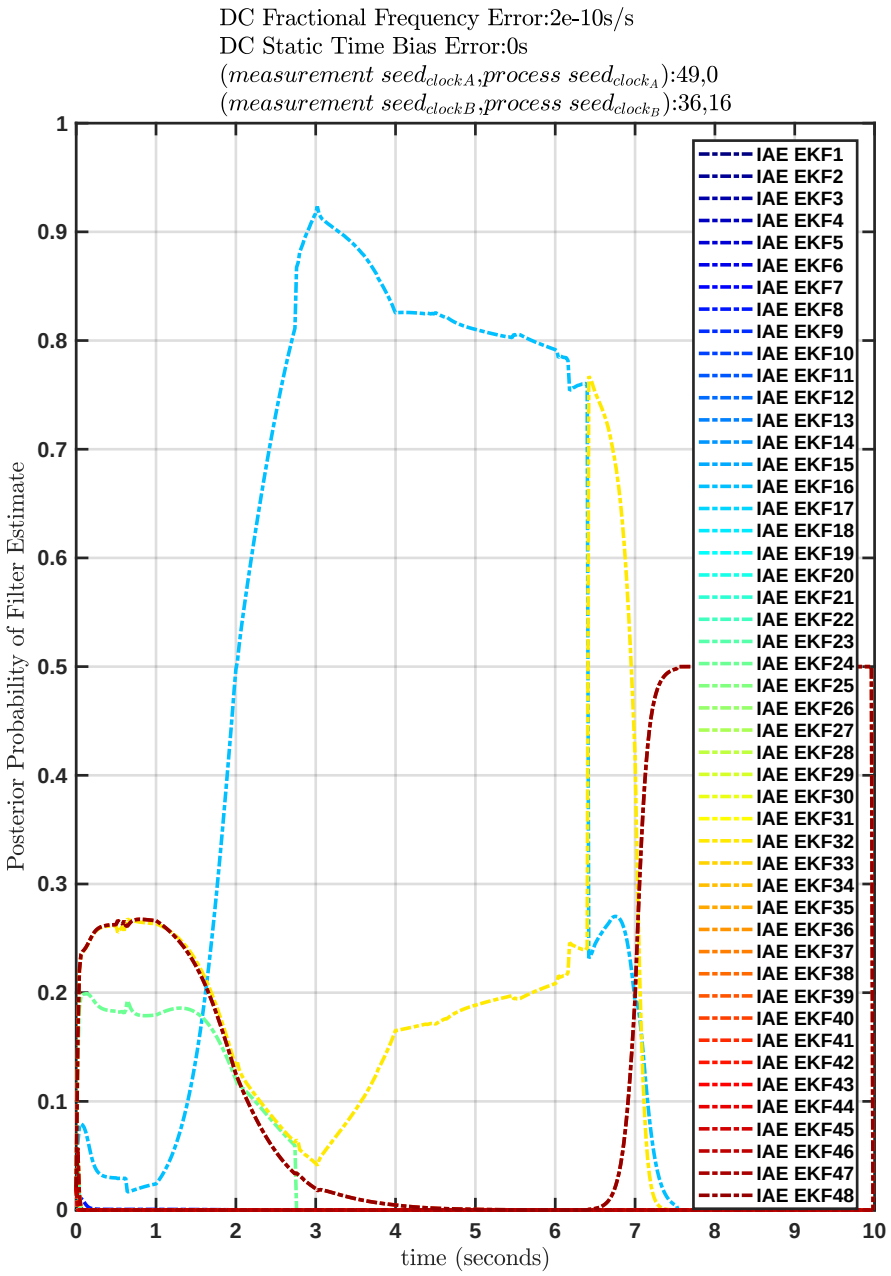

Figure B.117: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

DC Fractional Frequency Error:5.31e-08s/s, DC Static Time Bias Error:-9.5e-07s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):13,89, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):53,33

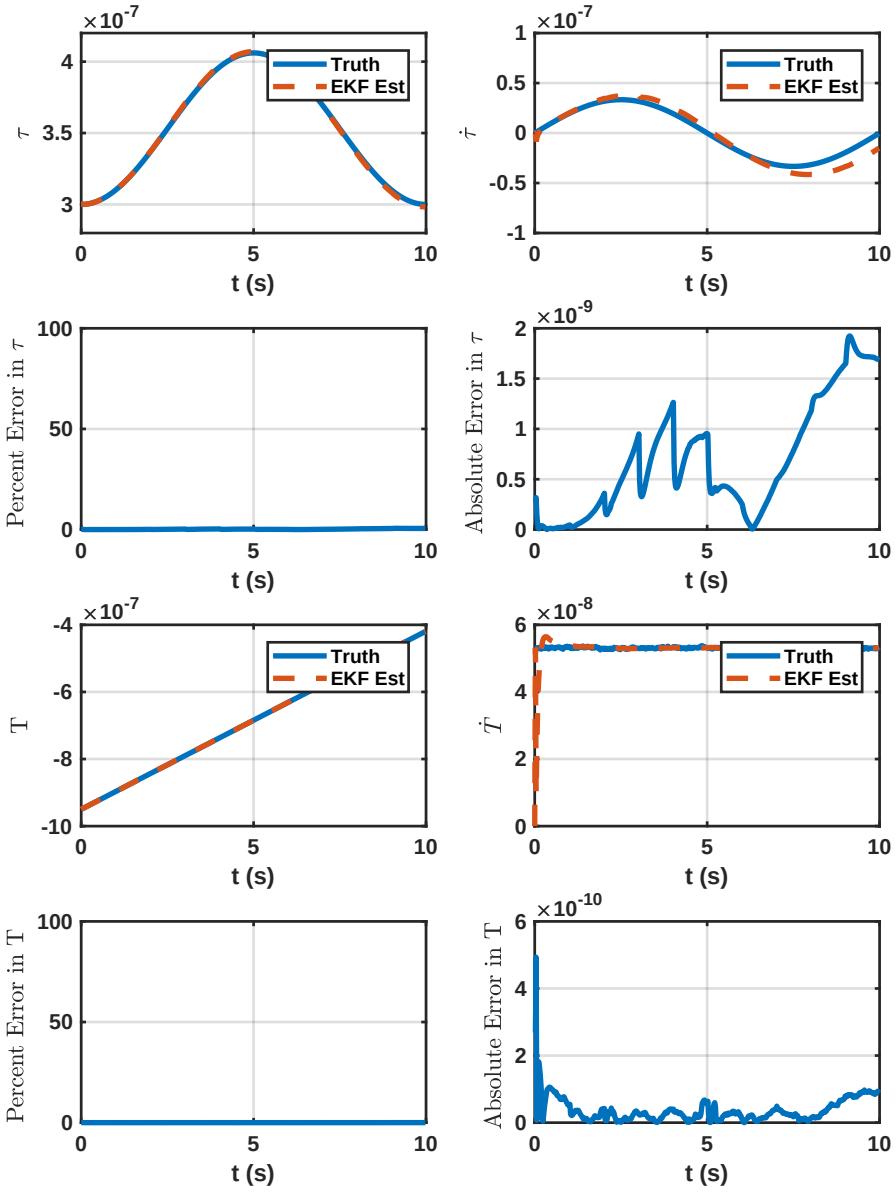

Figure B.118: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Kasdin Approximants.

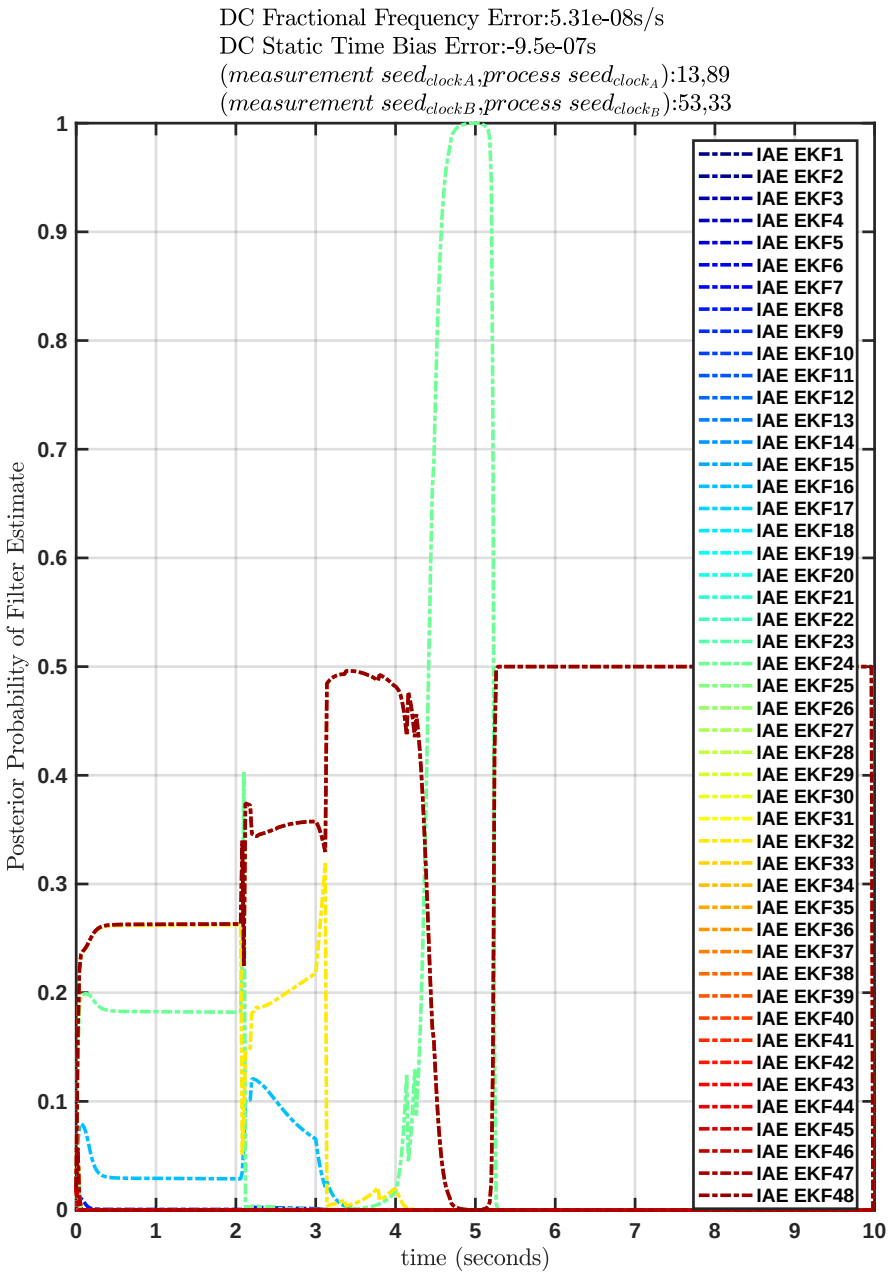

Figure B.119: Posterior Probability of State Estimates From Each Filter Bank Constituent Using Kasdin Approximants.

B.3 Monte Carlo Analysis of Bank of IAE Extended Kalman Filter Applied to HPPC System Using Oustaloup Noise

The σ diffusion coefficients used in this experiment are the same as the previous section, shown in table B.3. The only difference is that Oustaloup approximants are used as in equation 3.19 instead of the Kasdin Method.

DC Fractional Frequency Error:-7.83e-08s/s, DC Static Time Bias Error:-5.65e-06s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):50,69, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):54,39

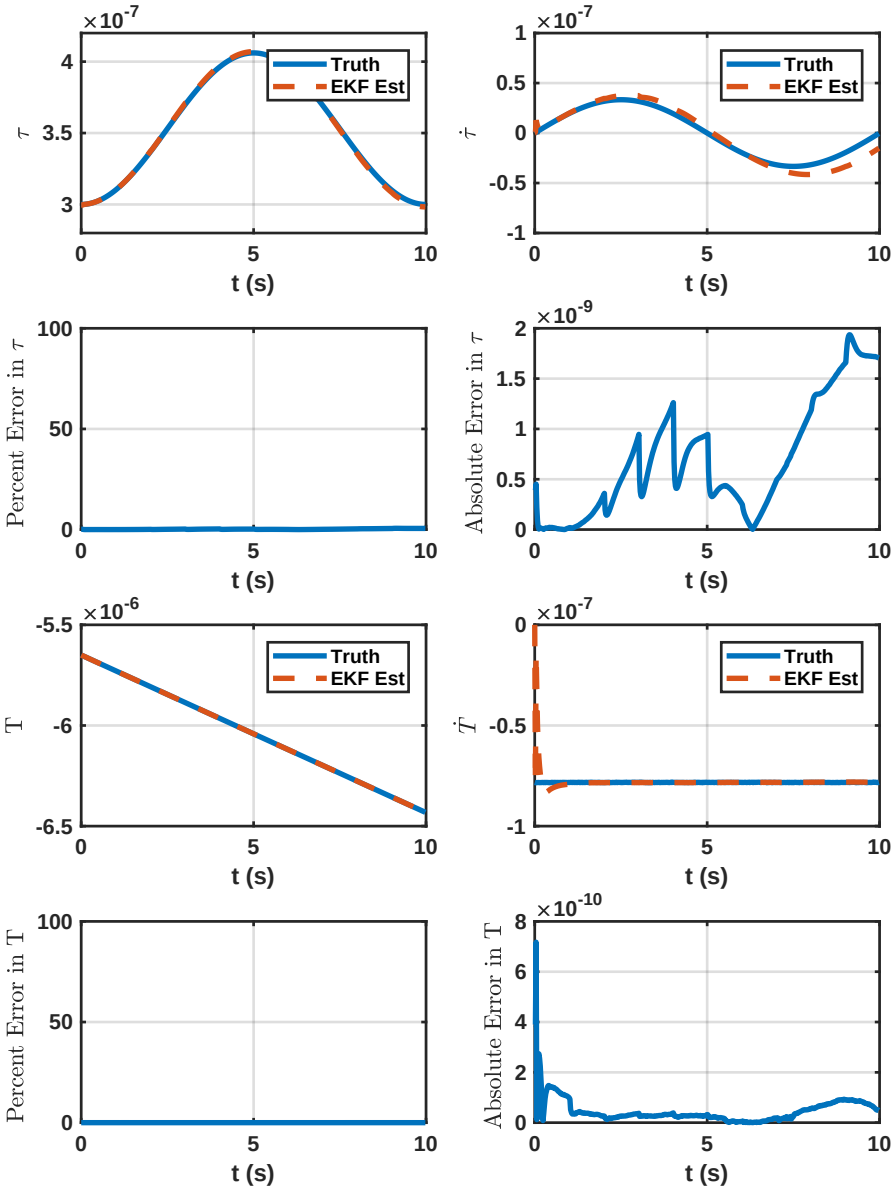

Figure B.120: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

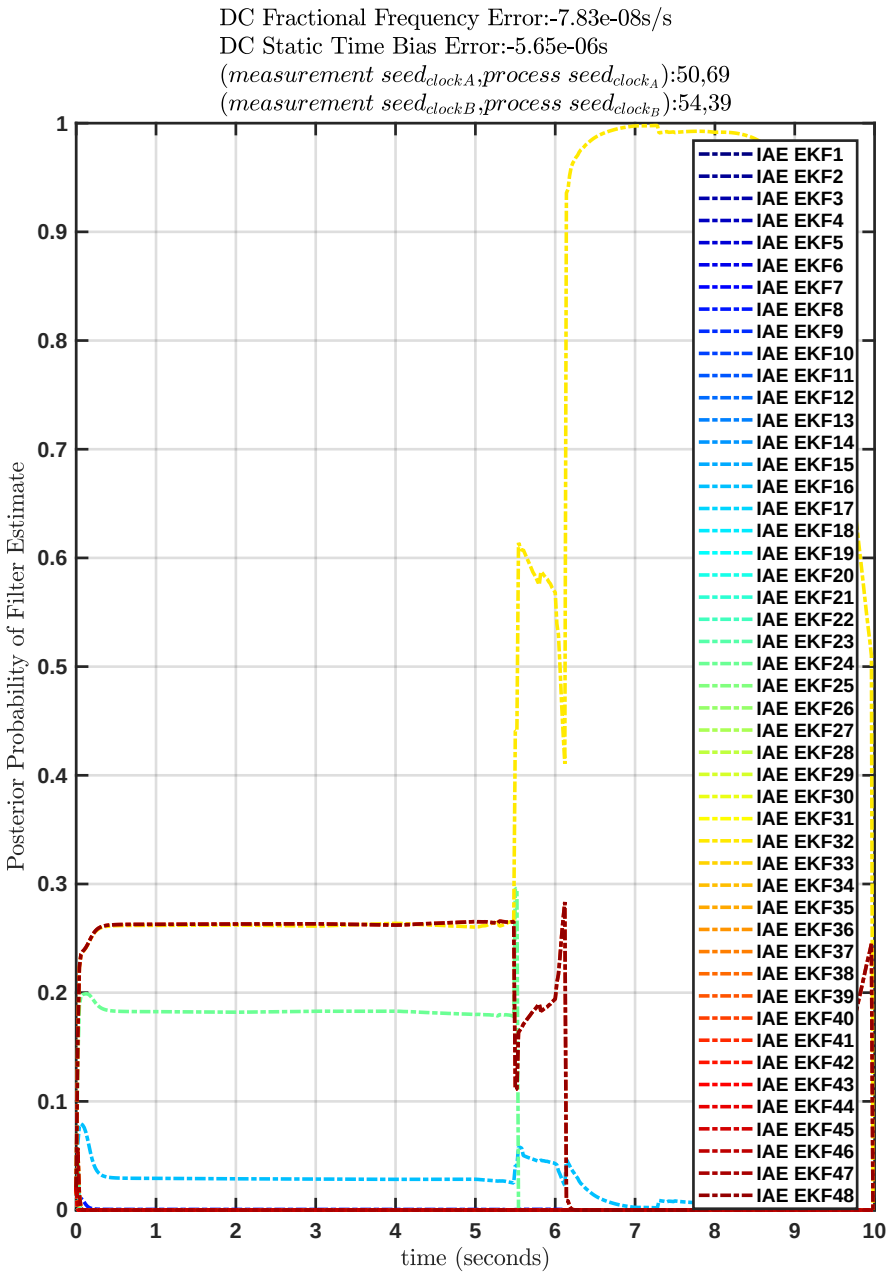

Figure B.121: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

DC Fractional Frequency Error:-5.19e-08s/s, DC Static Time Bias Error:-9.18e-06s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):38,67, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):44,63

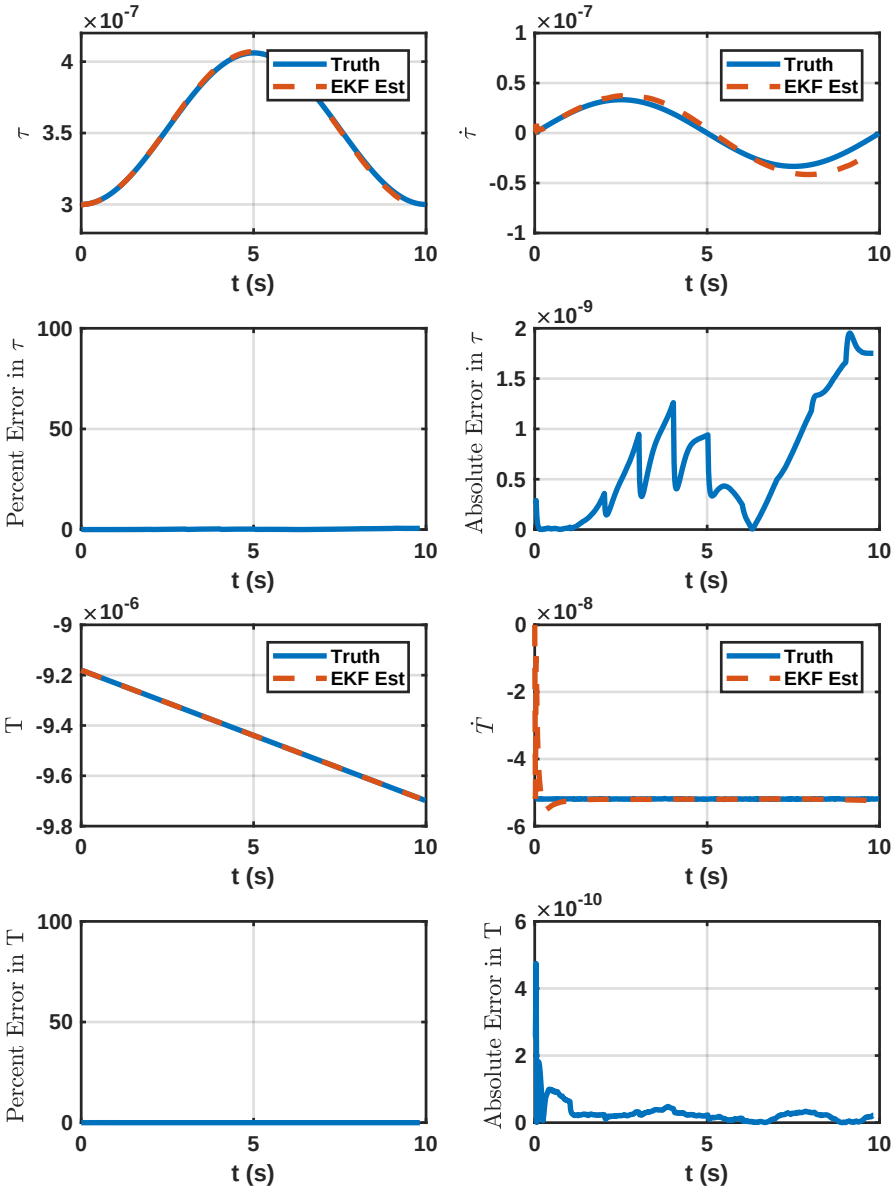

Figure B.122: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

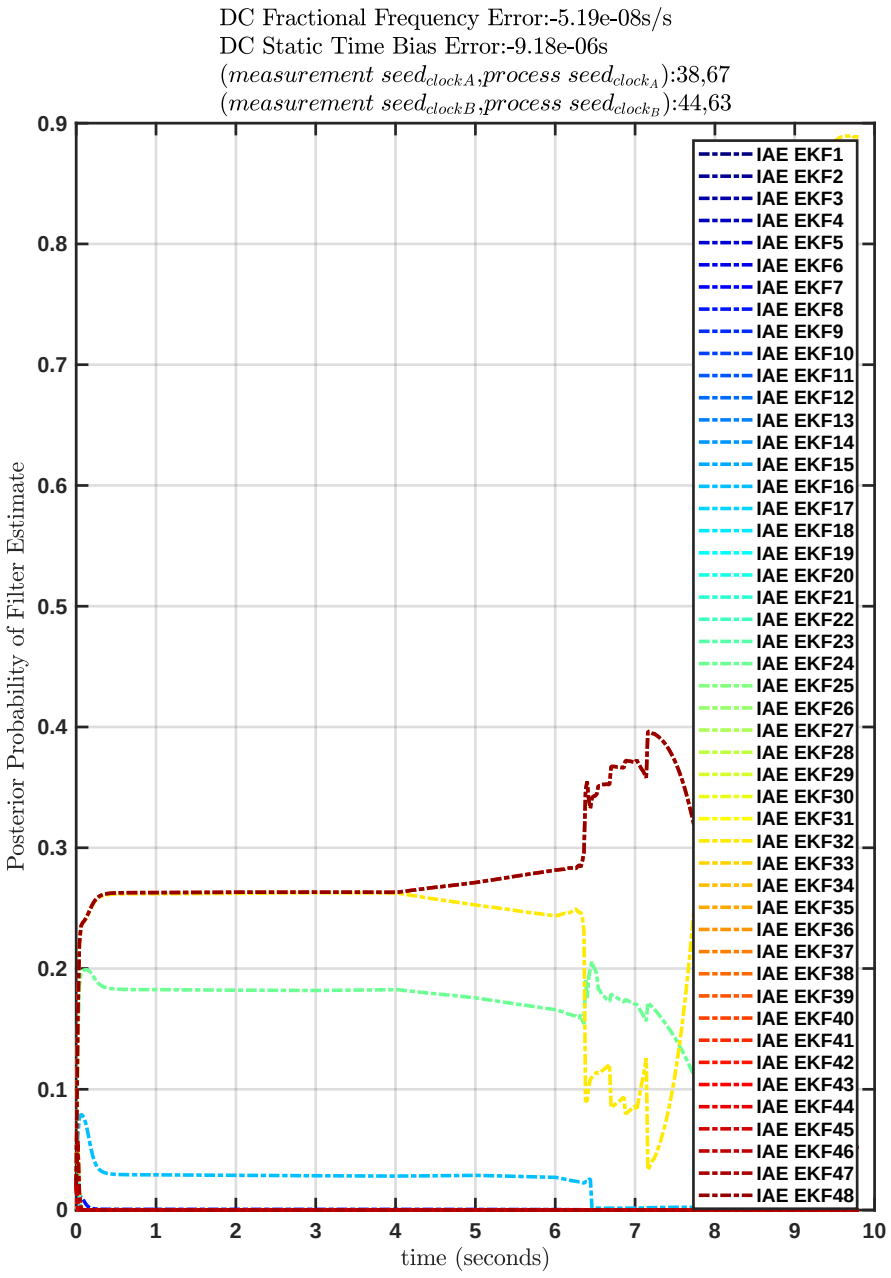

Figure B.123: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

DC Fractional Frequency Error:-6.38e-08s/s, DC Static Time Bias Error:-1.91e-06s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):13,39, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):85,95

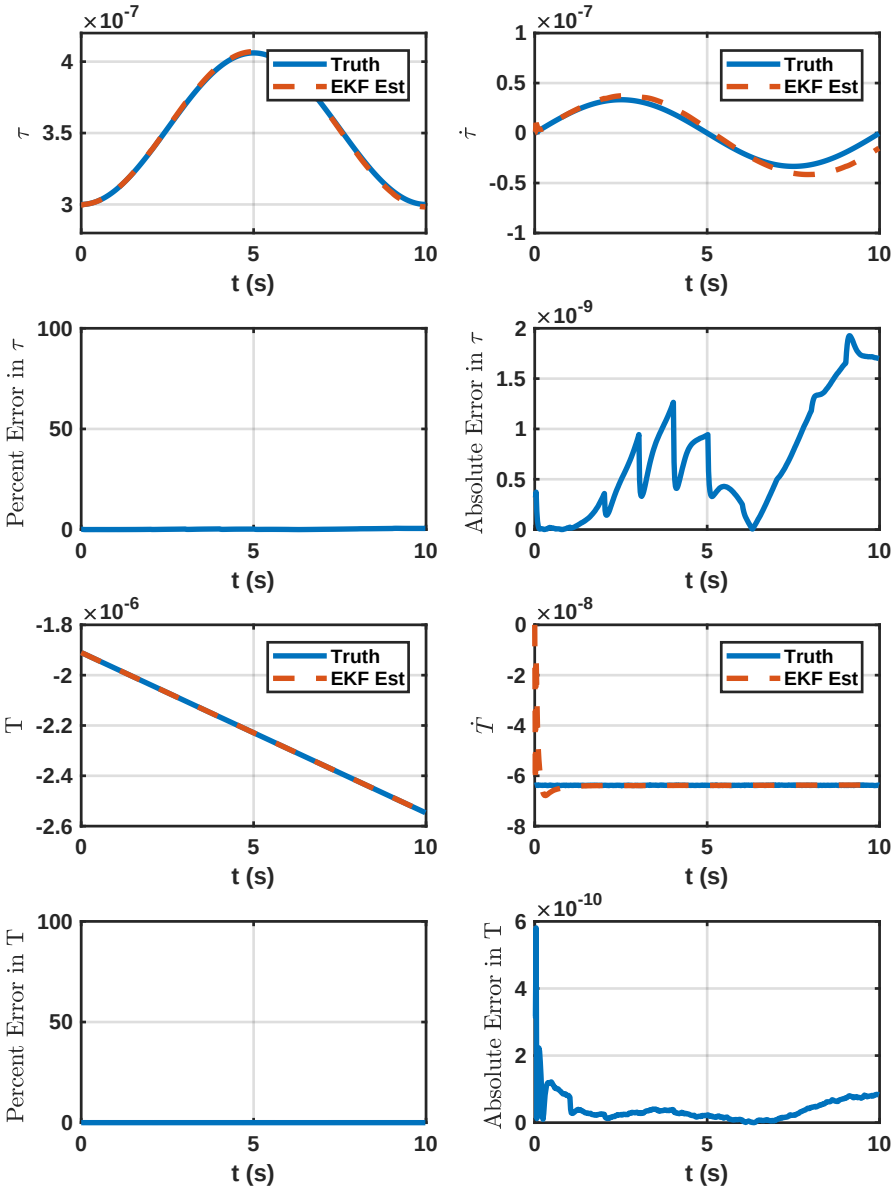

Figure B.124: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

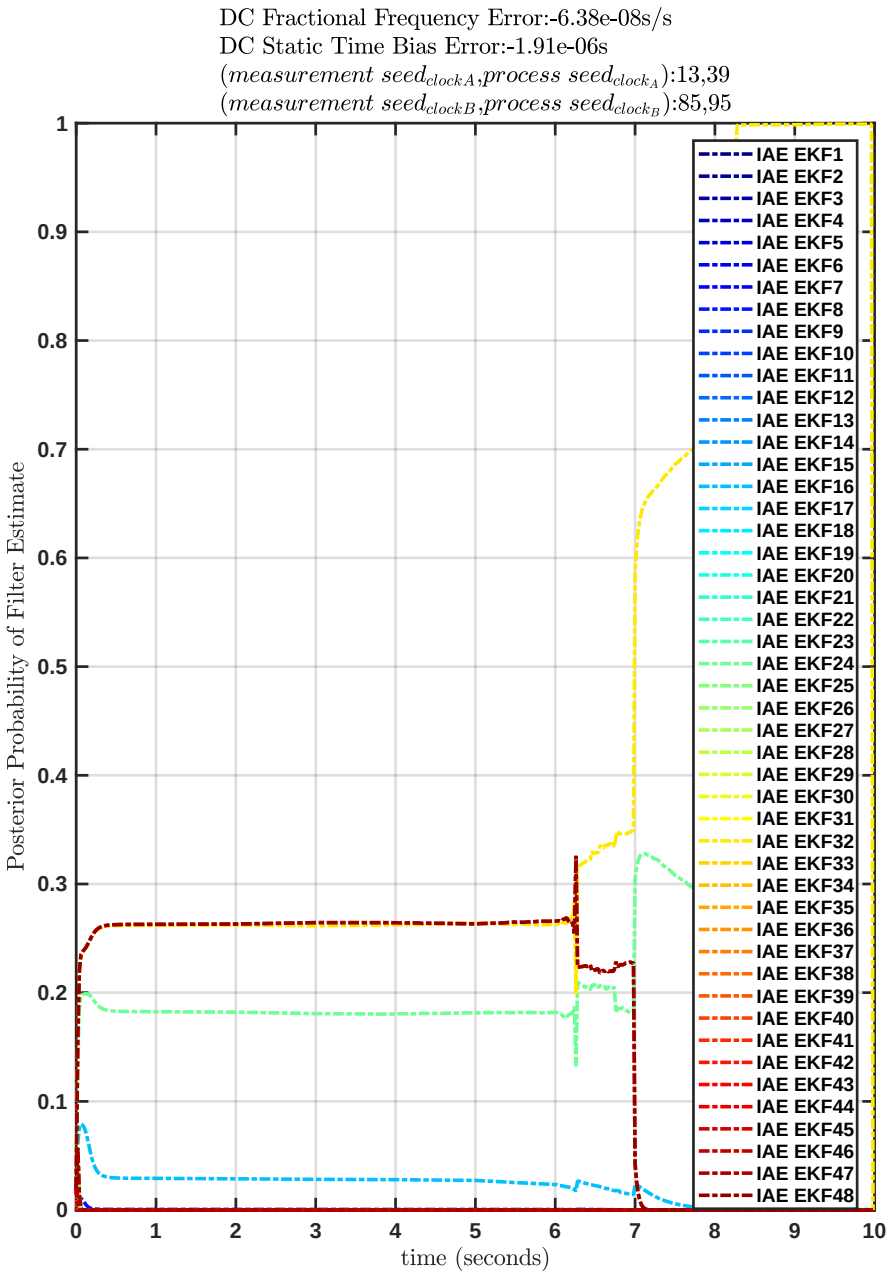

Figure B.125: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

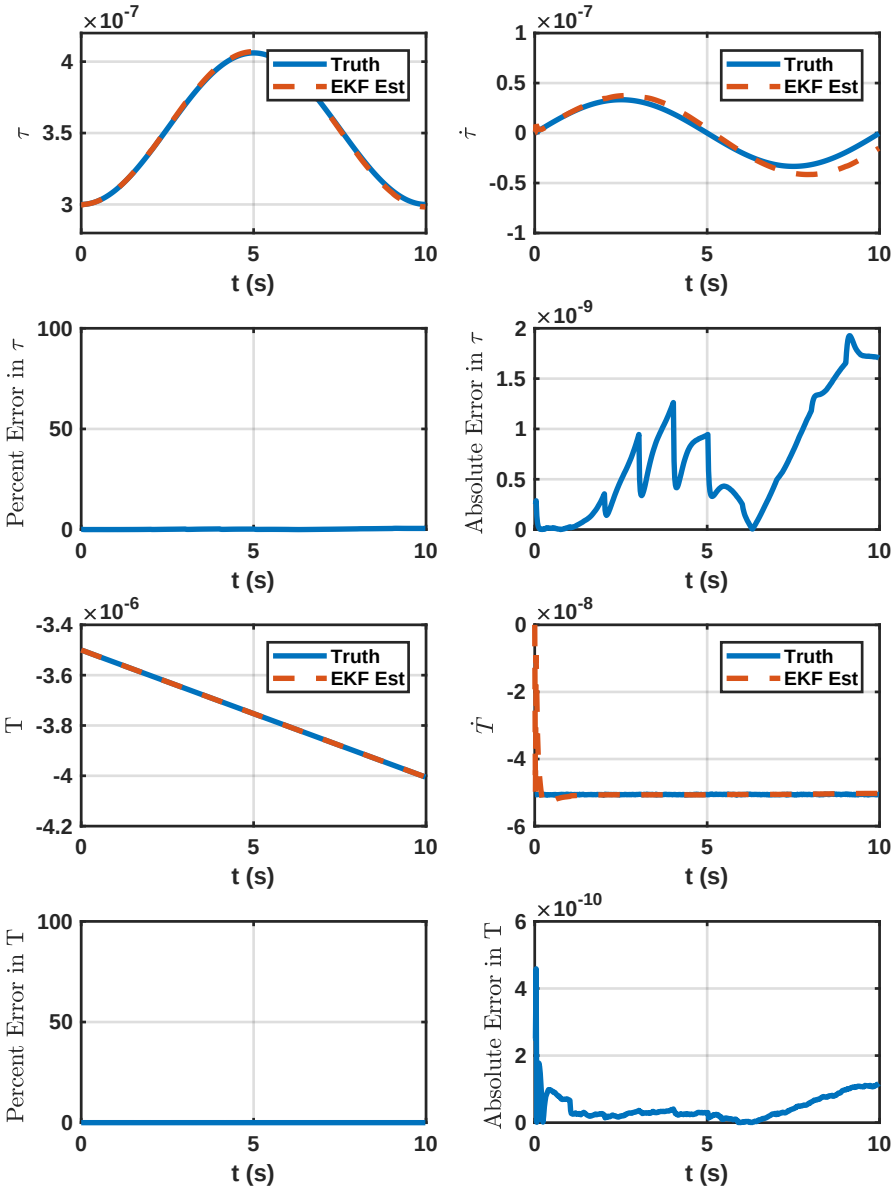

Figure B.126: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

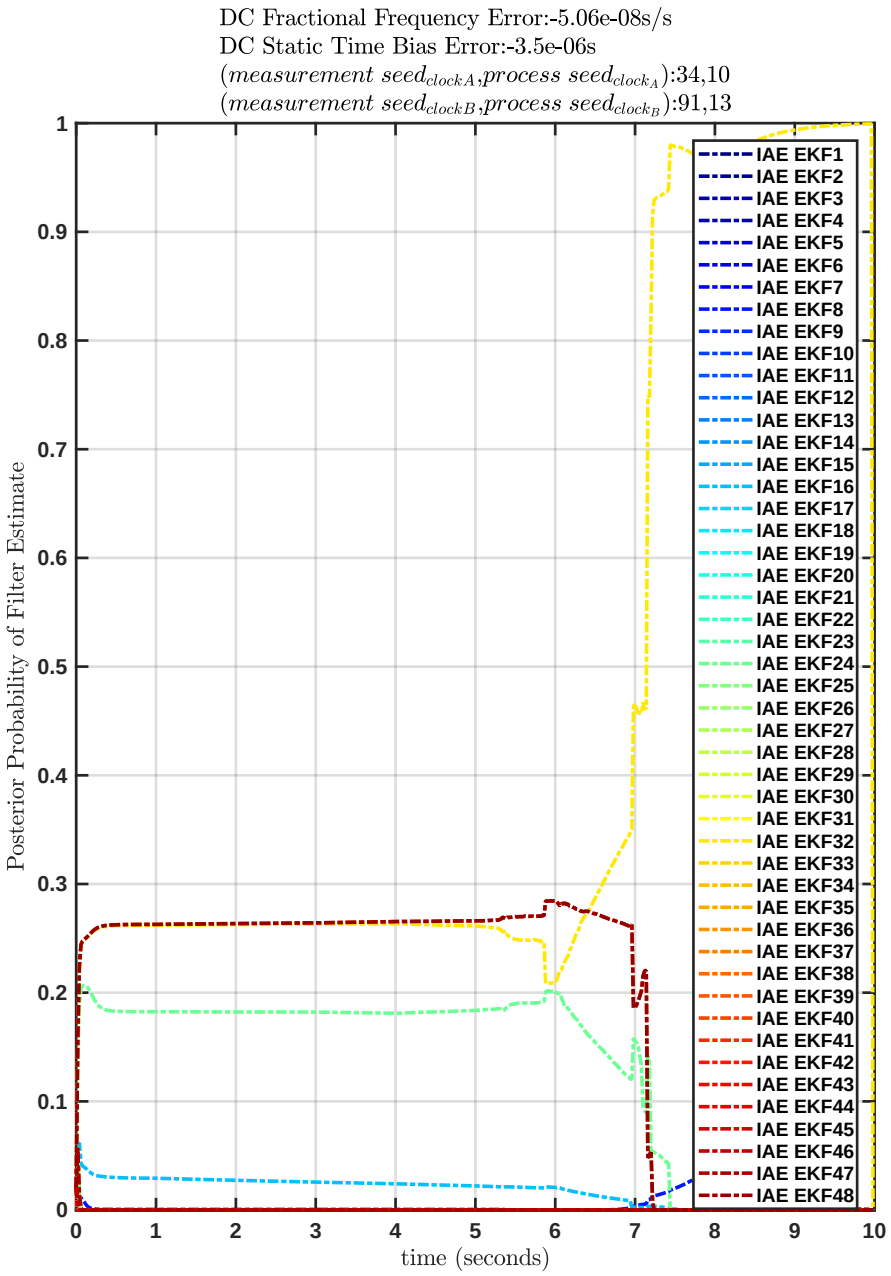

Figure B.127: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

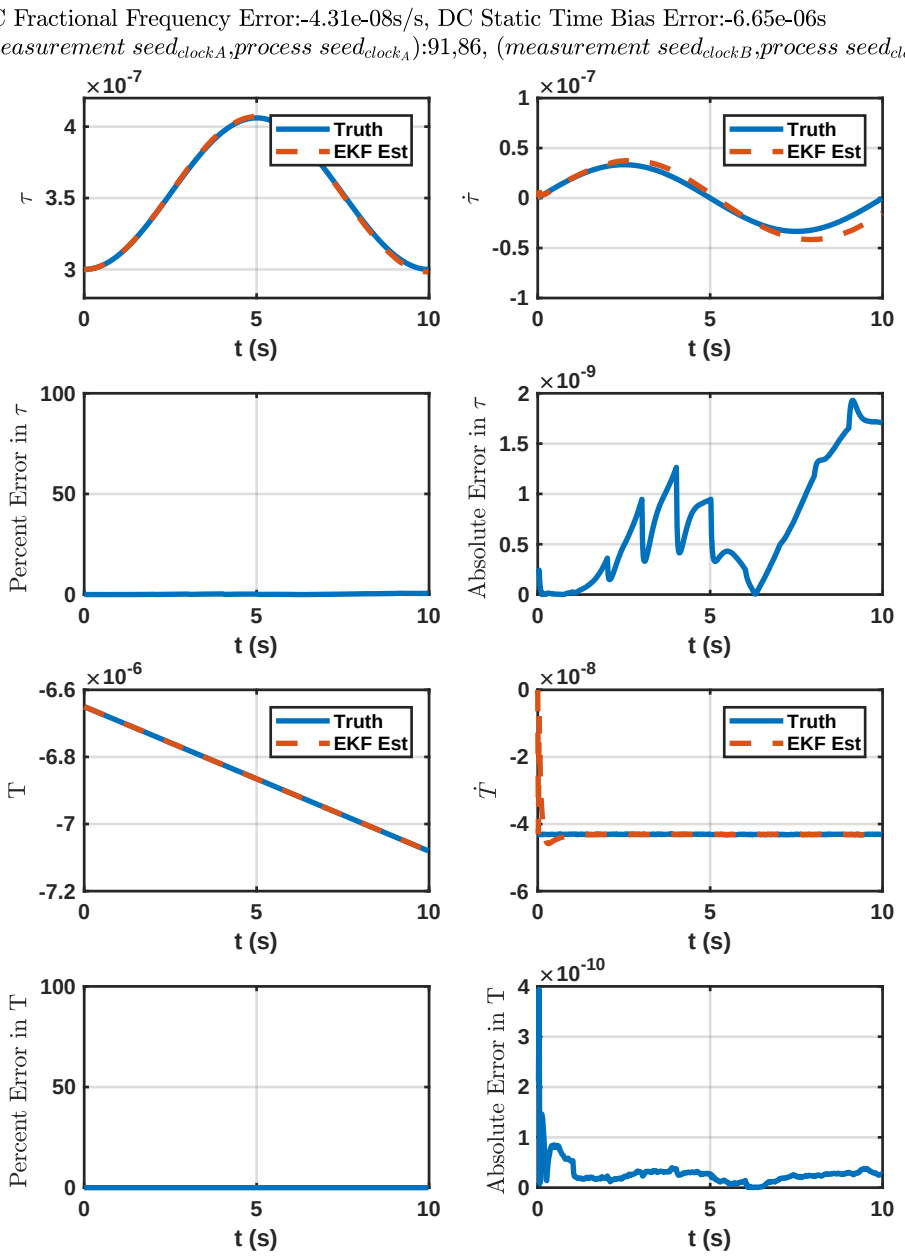

Figure B.128: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

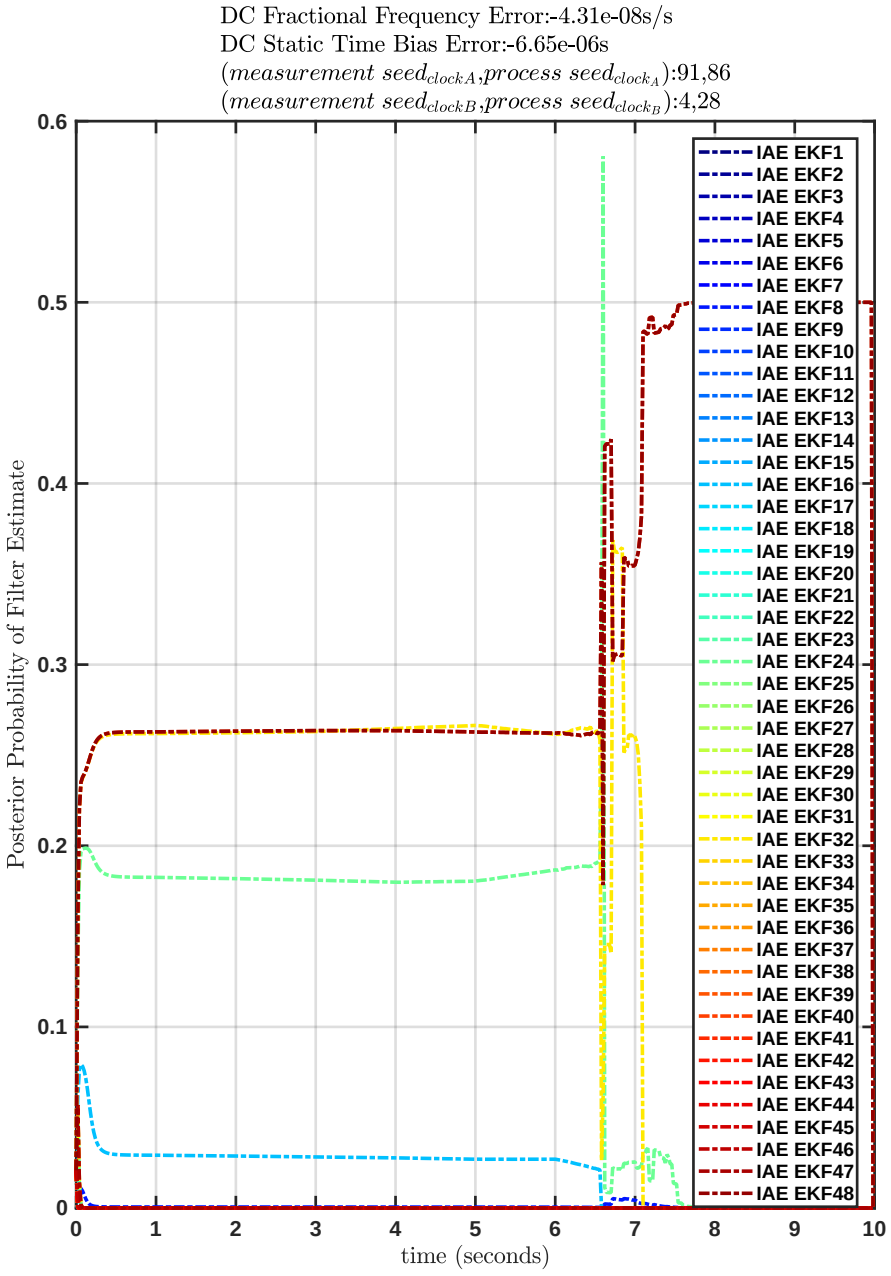

Figure B.129: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

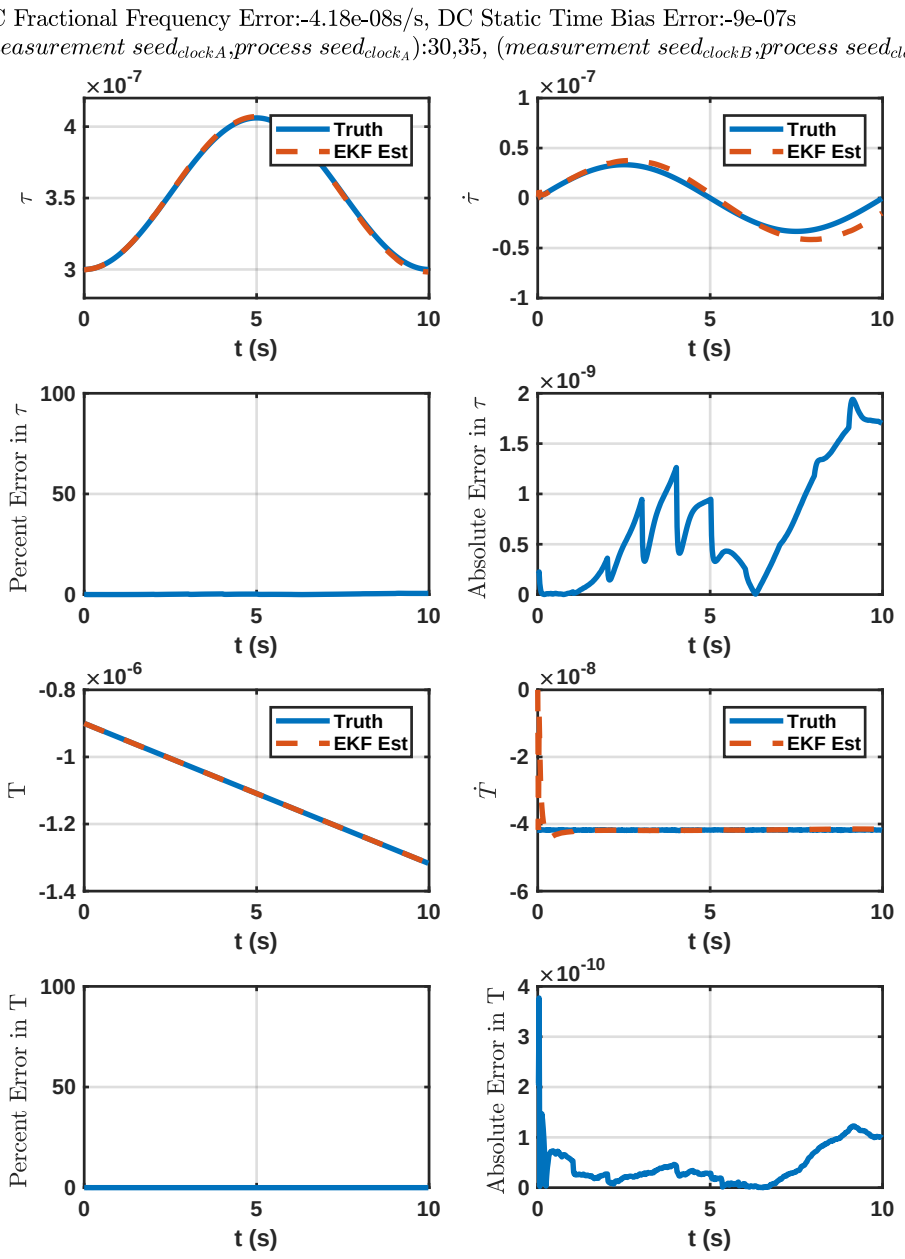

Figure B.130: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

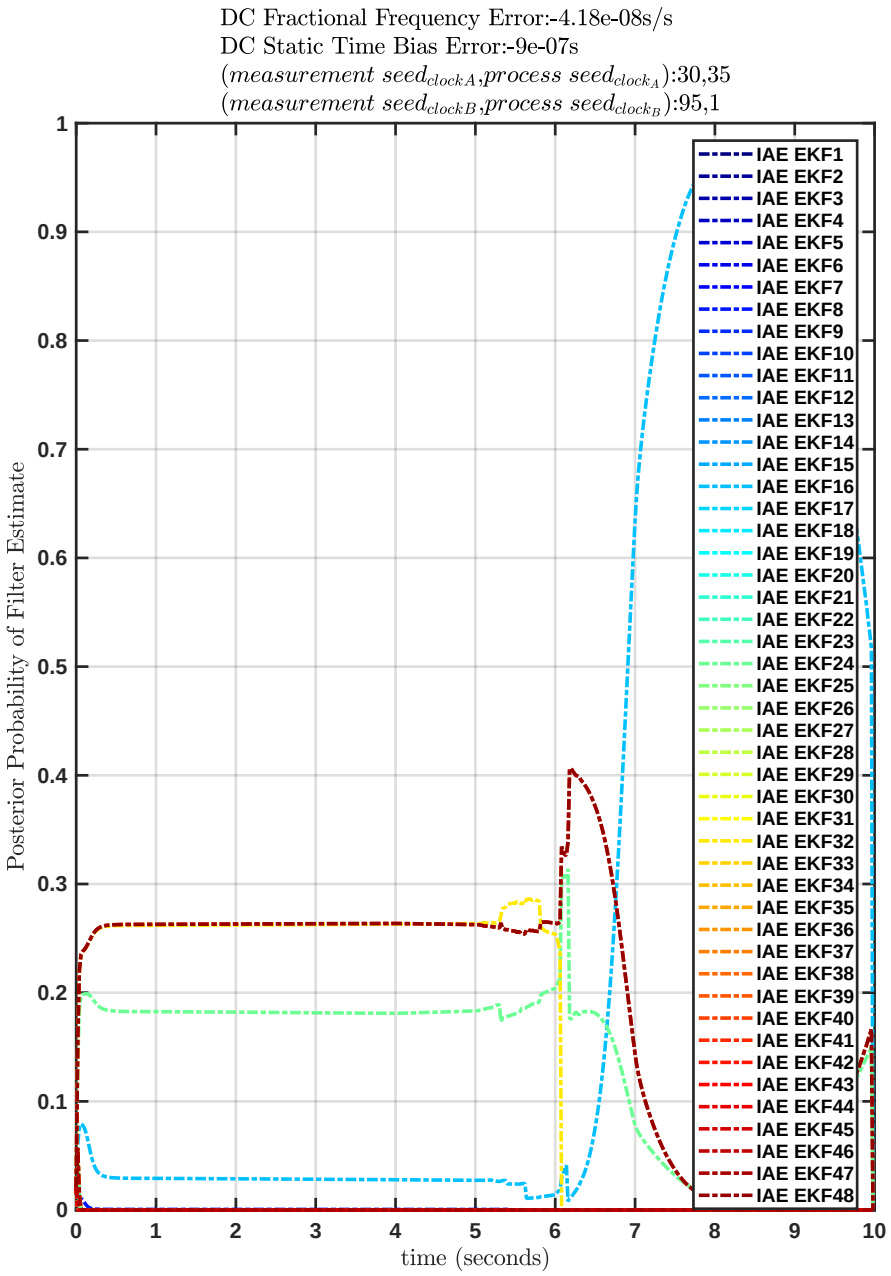

Figure B.131: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

DC Fractional Frequency Error:-7e-10s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):43,100, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):67,86

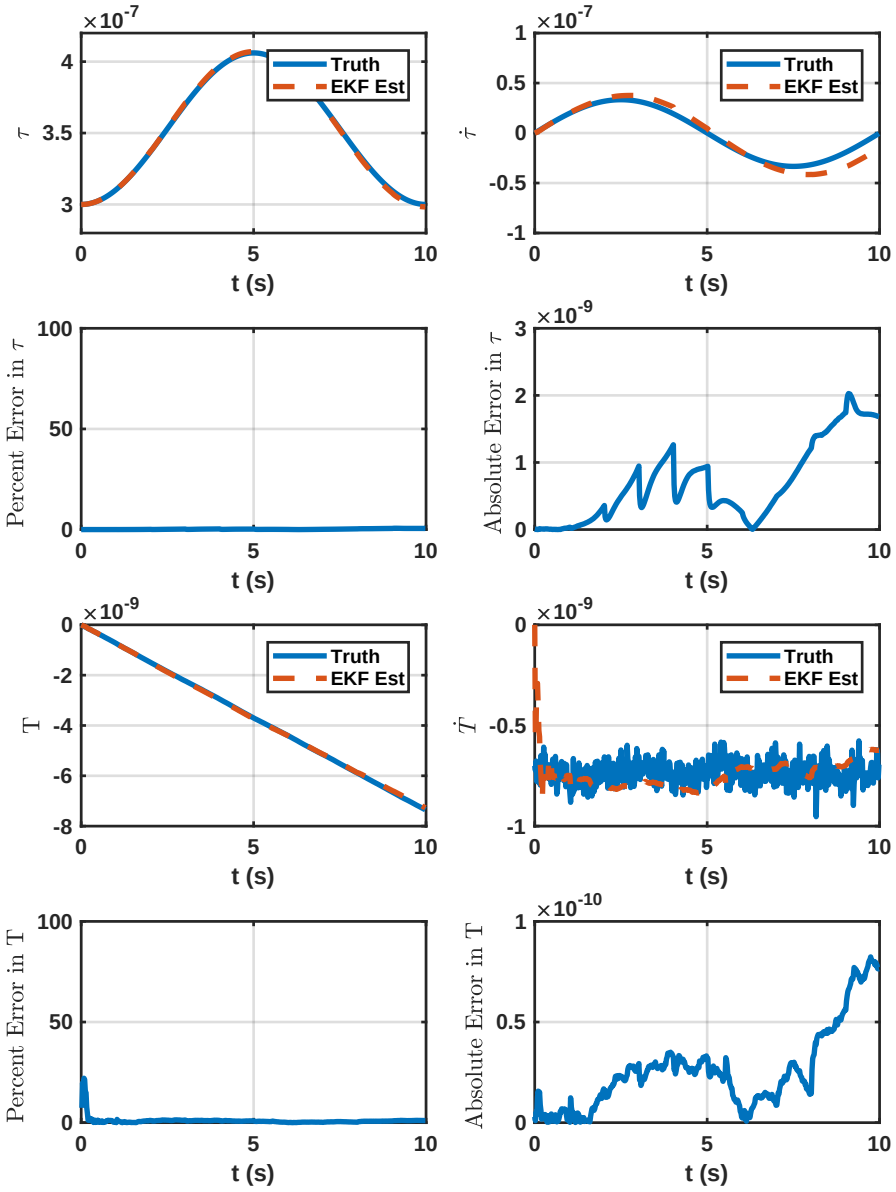

Figure B.132: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

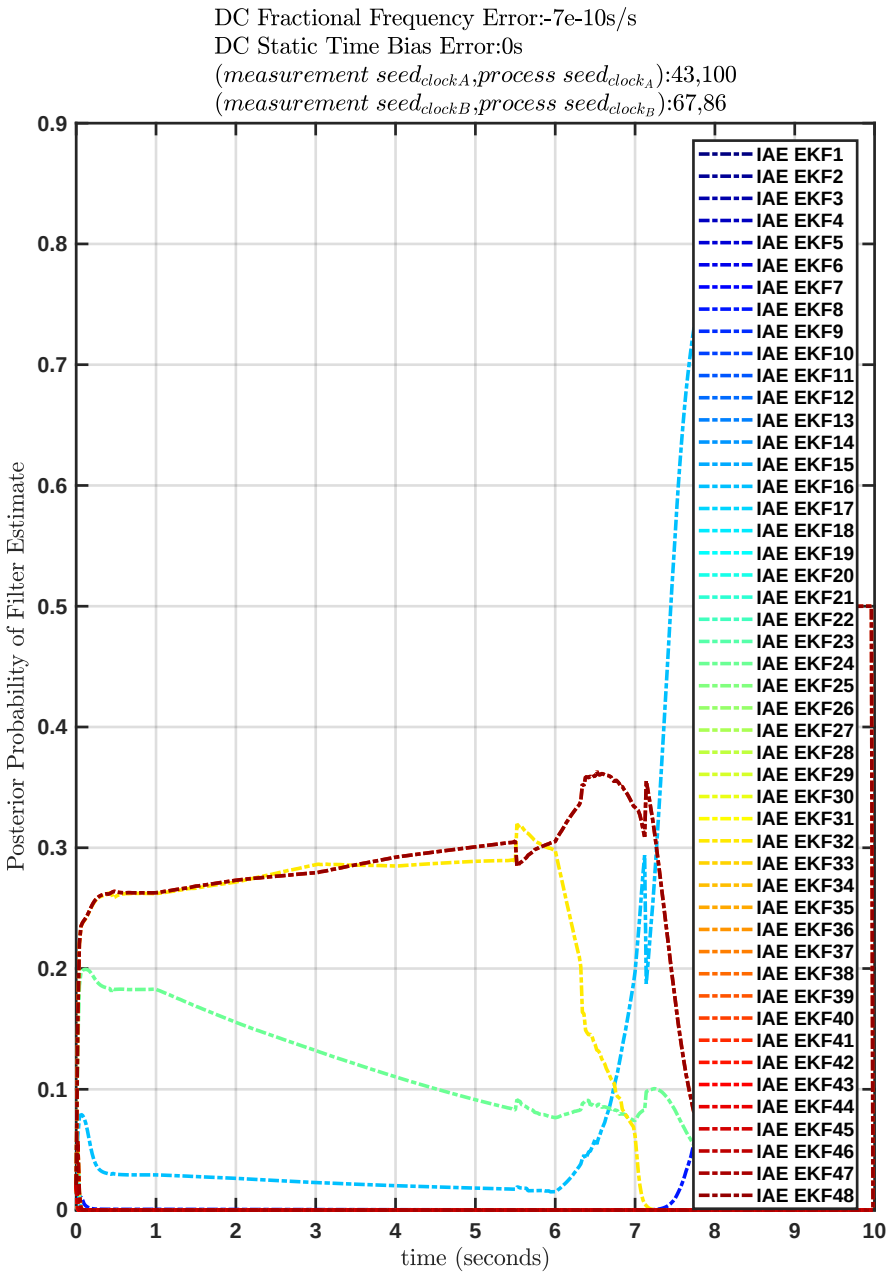

Figure B.133: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

DC Fractional Frequency Error:0s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):82,50, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):73,86

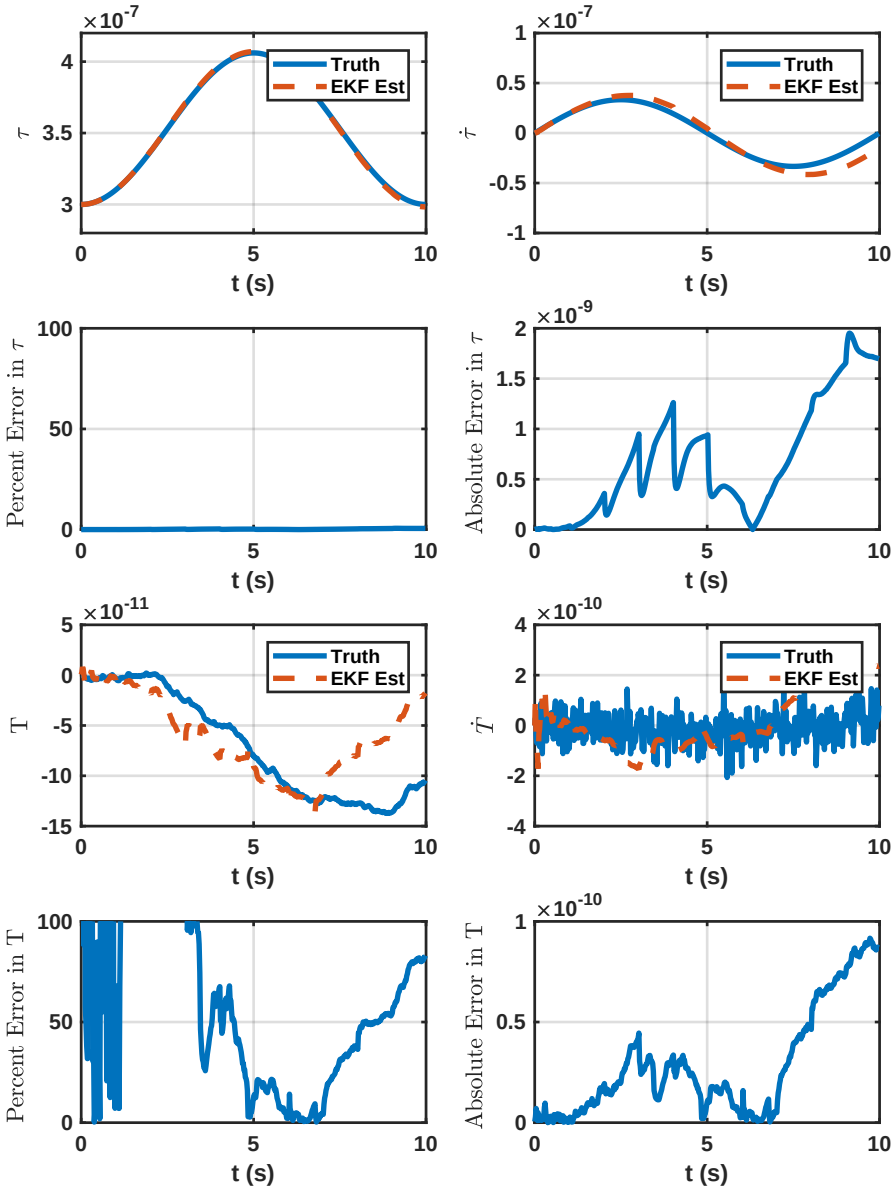

Figure B.134: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

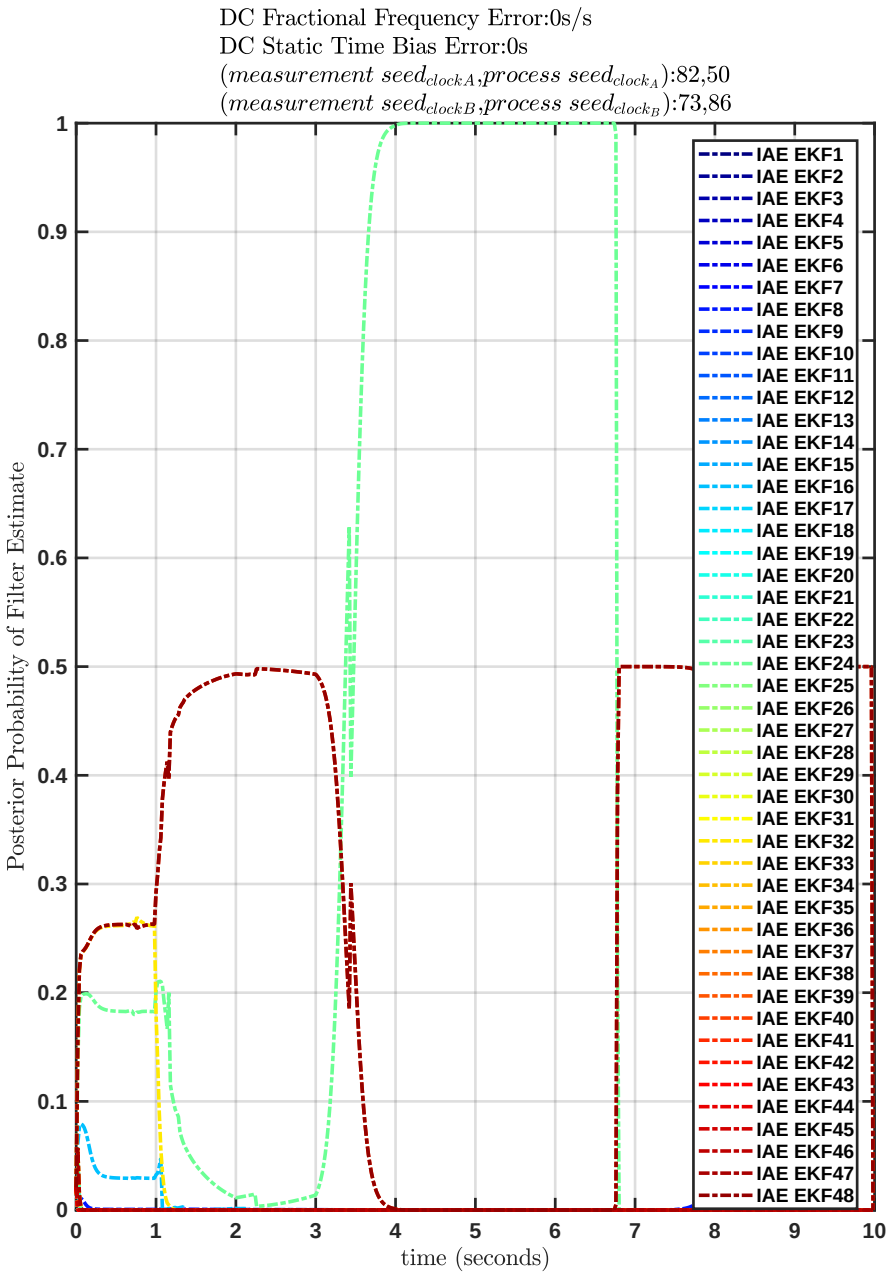

Figure B.135: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

DC Fractional Frequency Error:7e-10s/s, DC Static Time Bias Error:0s
($measurement\ seed_{clockA}$,$process\ seed_{clock_A}$):92,3, ($measurement\ seed_{clockB}$,$process\ seed_{clock_B}$):90,61

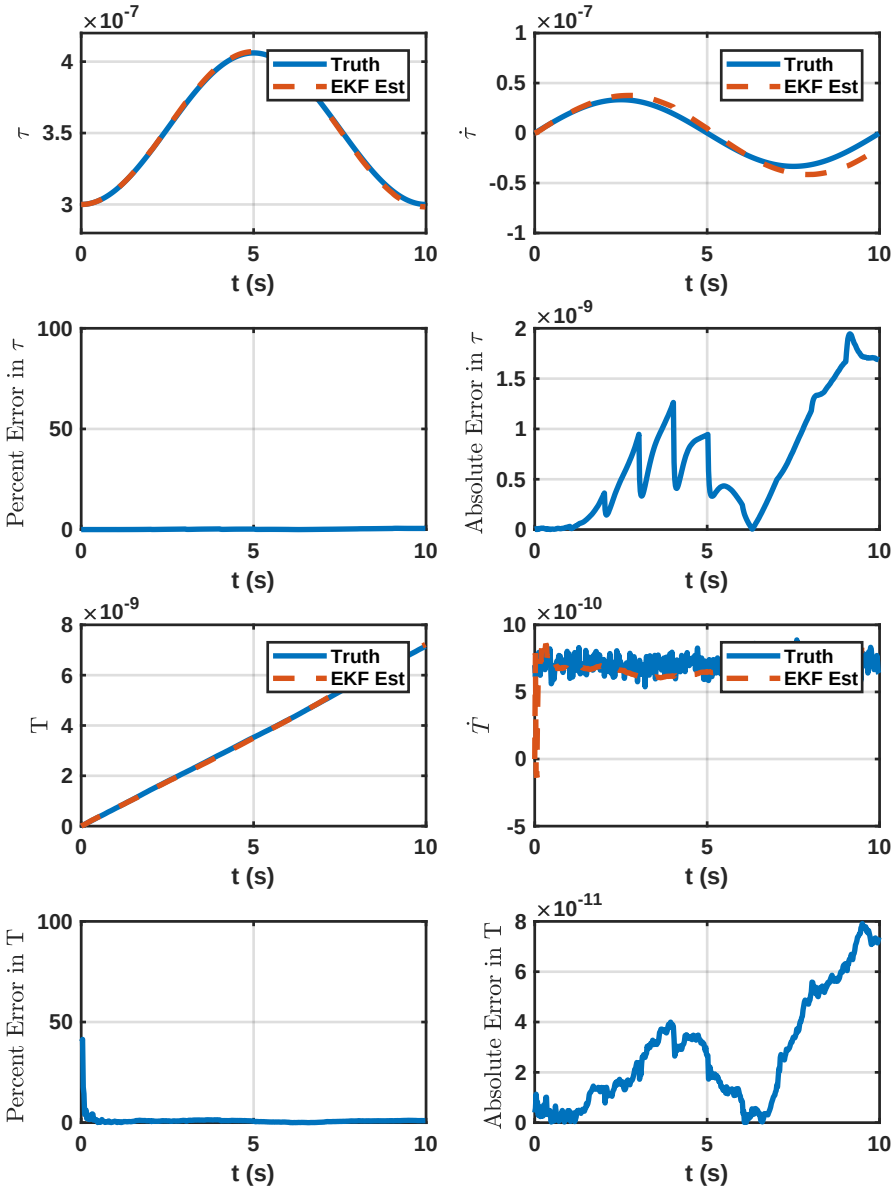

Figure B.136: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

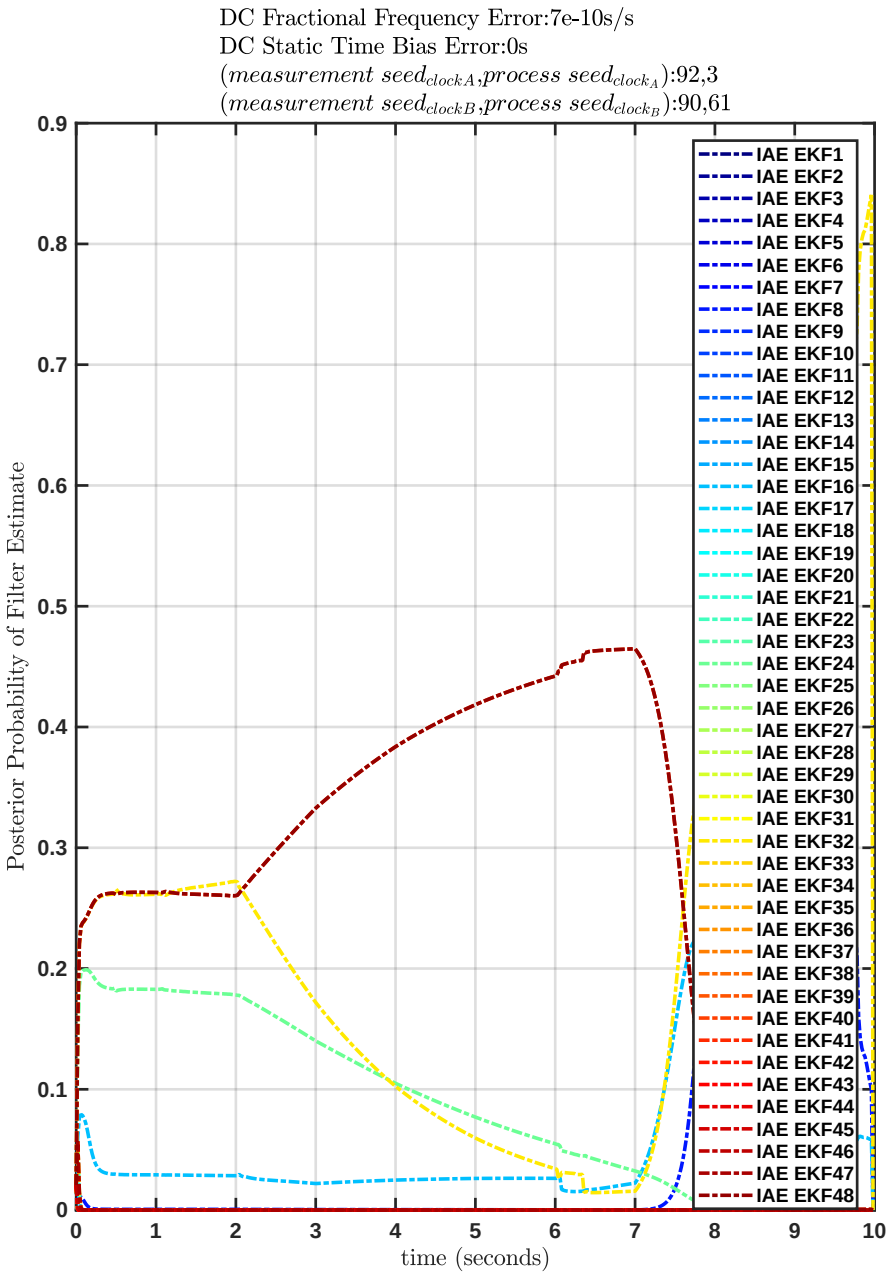

Figure B.137: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

B.4 Monte Carlo Results for HPPC Filter with Oscillator States only - Longer Simulations

The σ diffusion coefficients used in this experiment are the same as the previous section, shown in table B.3. The only difference is that the simulations are ran for 5 minutes instead of 10 seconds.

$$var(\tau)_k = [(1e-5)1_{1\times 8}, (1e-6)1_{1\times 8}, (1e-7)1_{1\times 8}, (1e-8)1_{1\times 8}, (1e-9)1_{1\times 8}, (1e-10)1_{1\times 10}] \quad \text{(B.11)}$$

$$var(T)_k = 1e-16[(1e-5)1_{1\times 8}, (1e-6)1_{1\times 8}, (1e-7)1_{1\times 8}, (1e-8)1_{1\times 8}, (1e-9)1_{1\times 8}, (1e-10)1_{1\times 10}] \quad \text{(B.12)}$$

$$var(t_{A,Rx})_k = [(1e-5)1_{1\times 8}, (1e-6)1_{1\times 8}, (1e-7)1_{1\times 8}, (1e-8)1_{1\times 8}, (1e-9)1_{1\times 8}, (1e-10)1_{1\times 8}, (1e-11)1_{1\times 8}, (1e-12)1_{1\times 8}] \quad \text{(B.13)}$$

$$var(t_{B,Rx})_k = var(t_{A,Rx}) \quad \text{(B.14)}$$

Where $1_{1\times n}$ is a vector of length n, composed of all 1's. Where $1_{1\times n}$ is a vector of length n, composed of all 1's. Then, each initial covariance matrix is computed for each filter bank entry, for all k

$$Q_o = diag([var(\tau)_k LA, var(\tau)_k LA, var(\tau)_k LA, var(T)_k [LA1e6LA1e2LA]]) \quad \text{(B.15)}$$

$$P_o = Q_o \quad \text{(B.16)}$$

$$R_o = diag([var(t_{A,Rx})_k, var(t_{B,Rx})_k]) \quad \text{(B.17)}$$

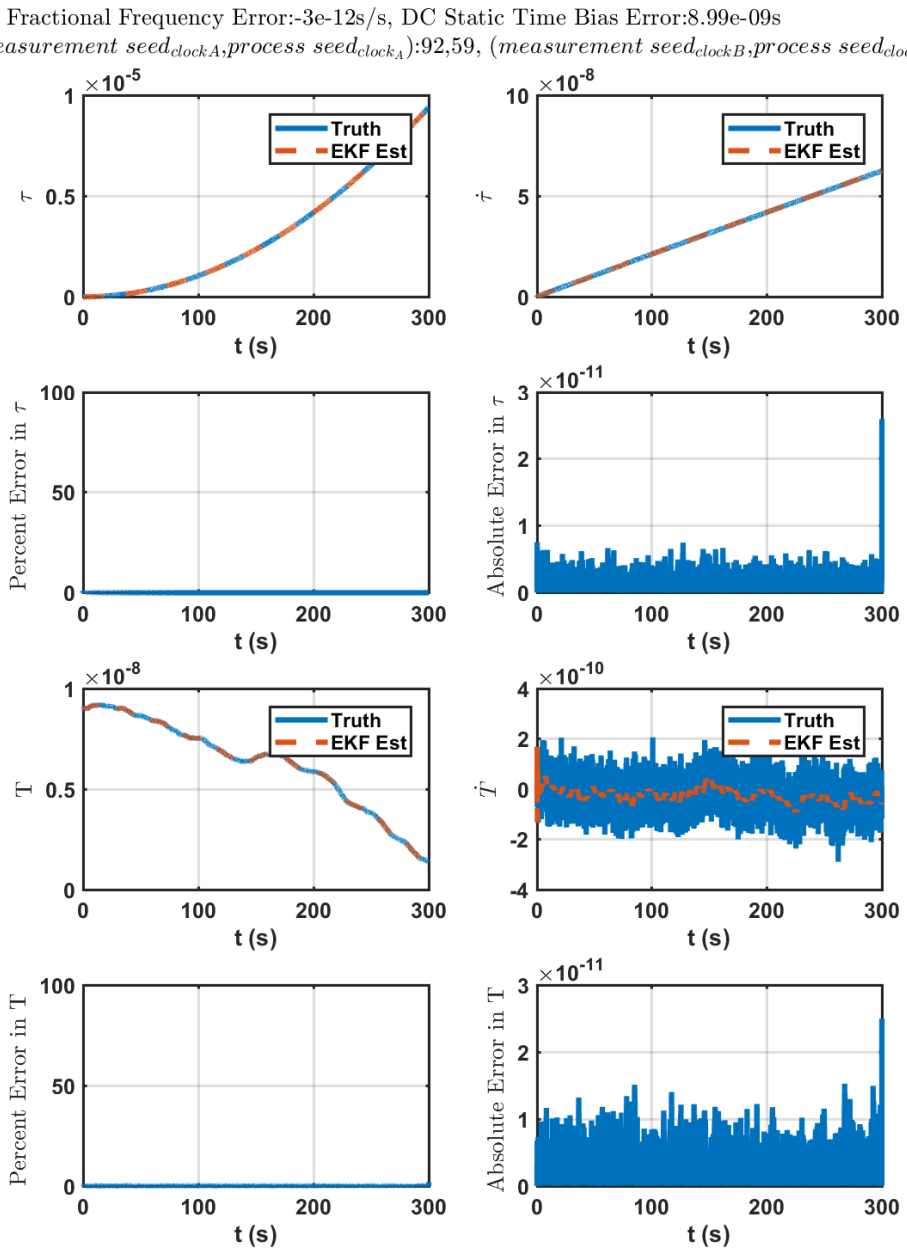

Figure B.138: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

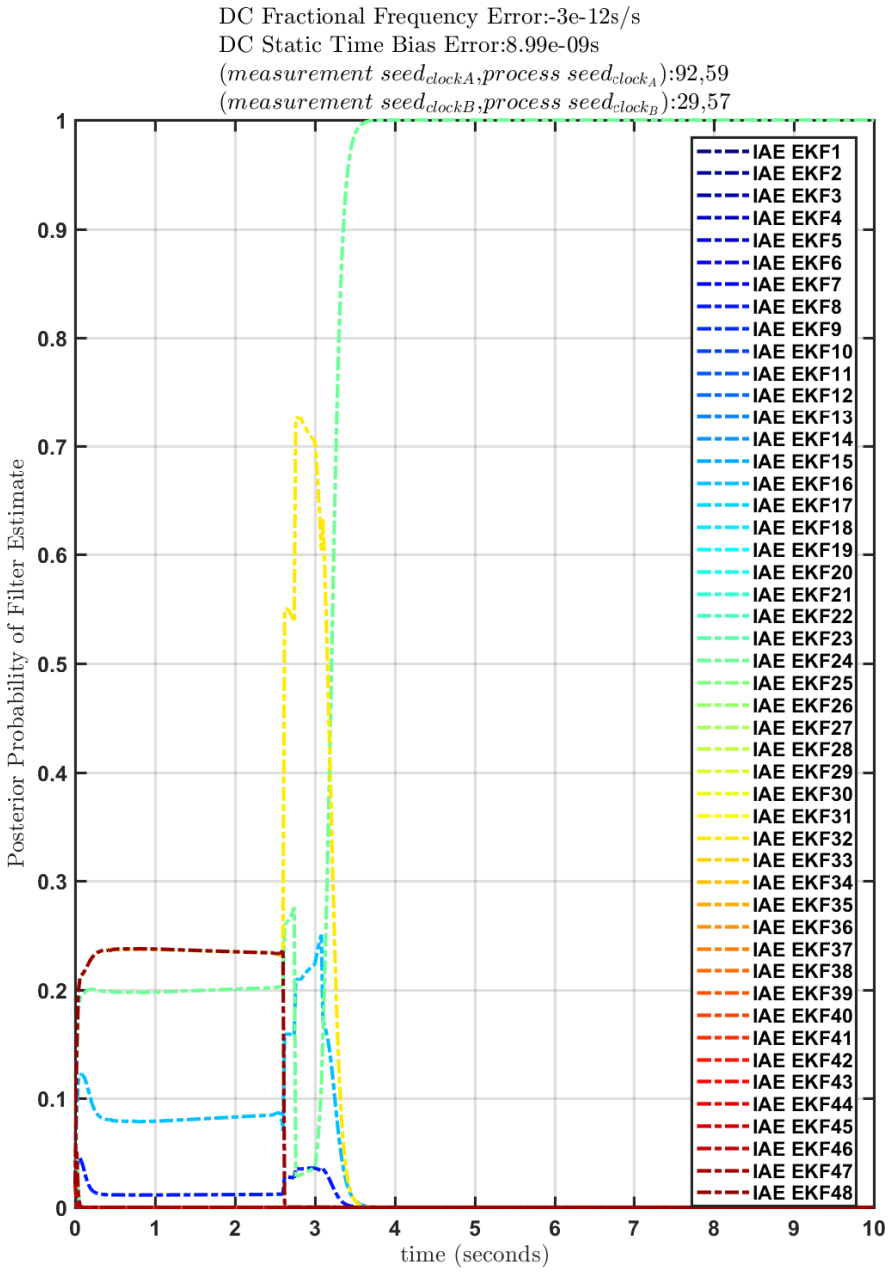

Figure B.139: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

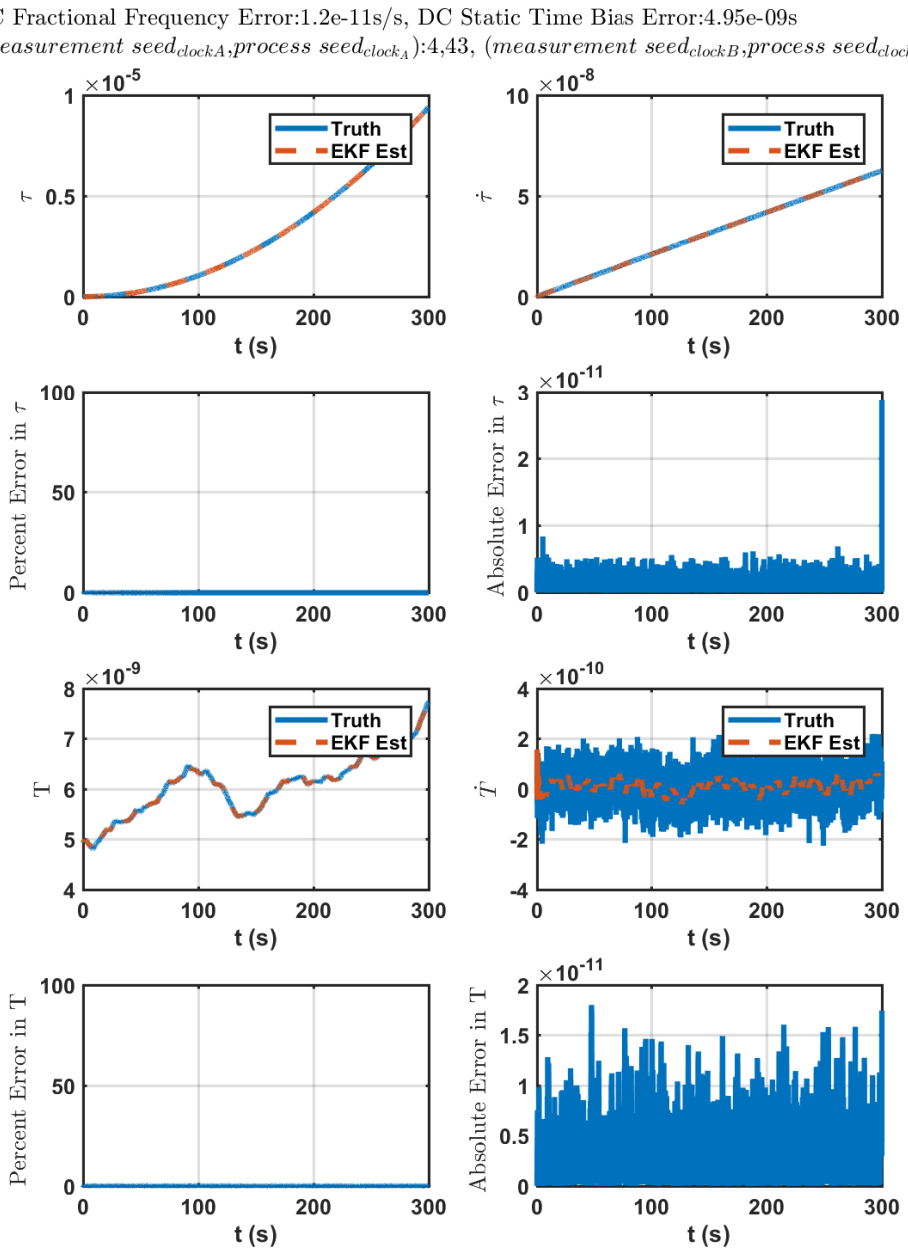

Figure B.140: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

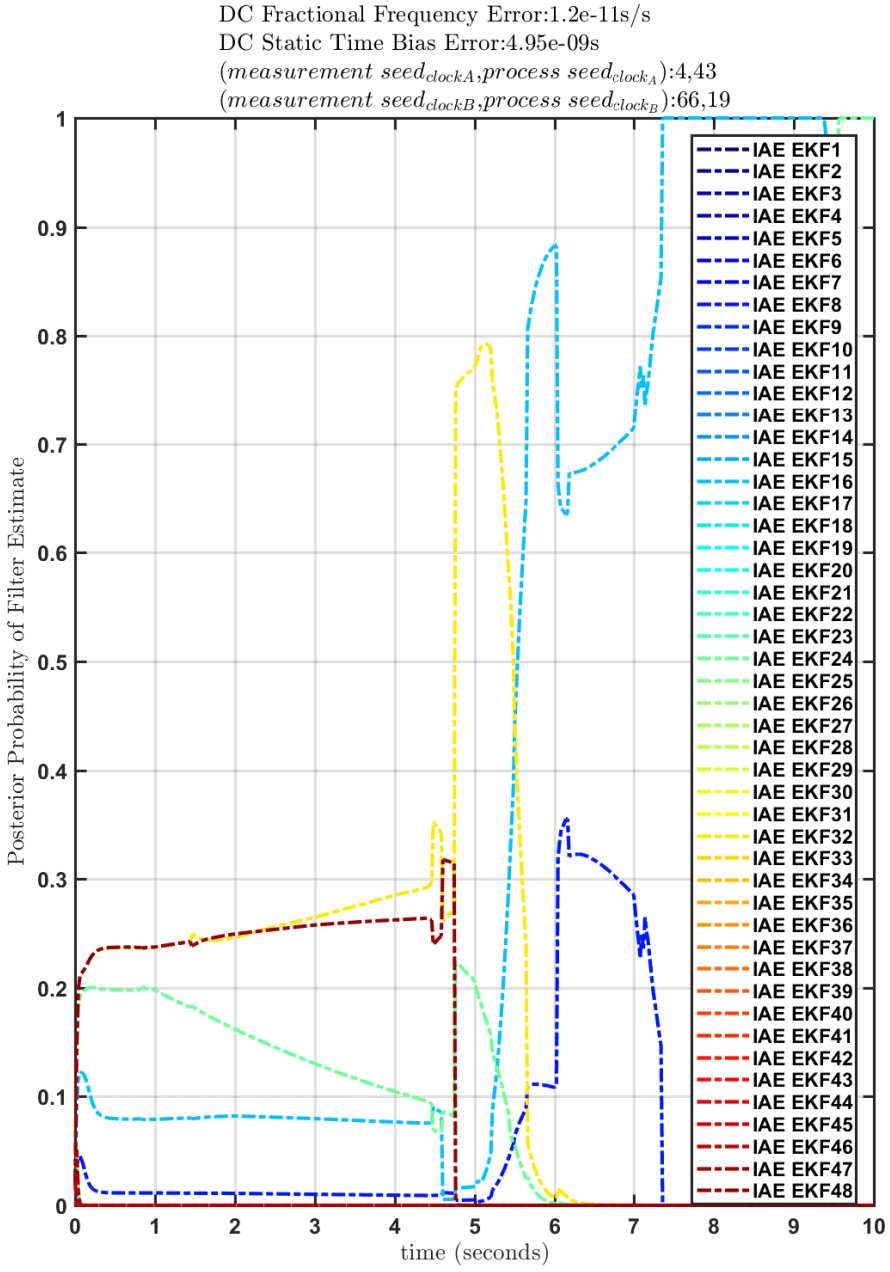

Figure B.141: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

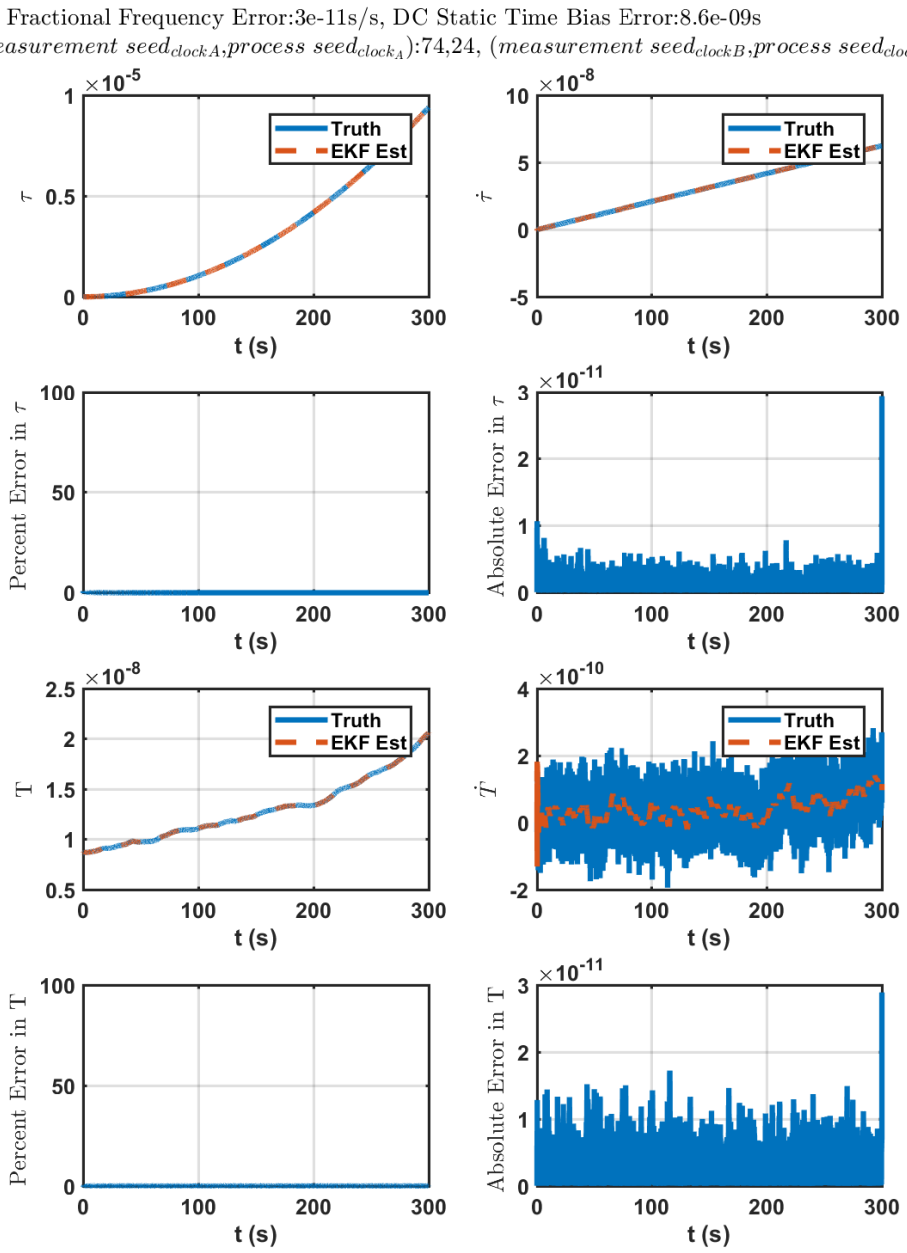

Figure B.142: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

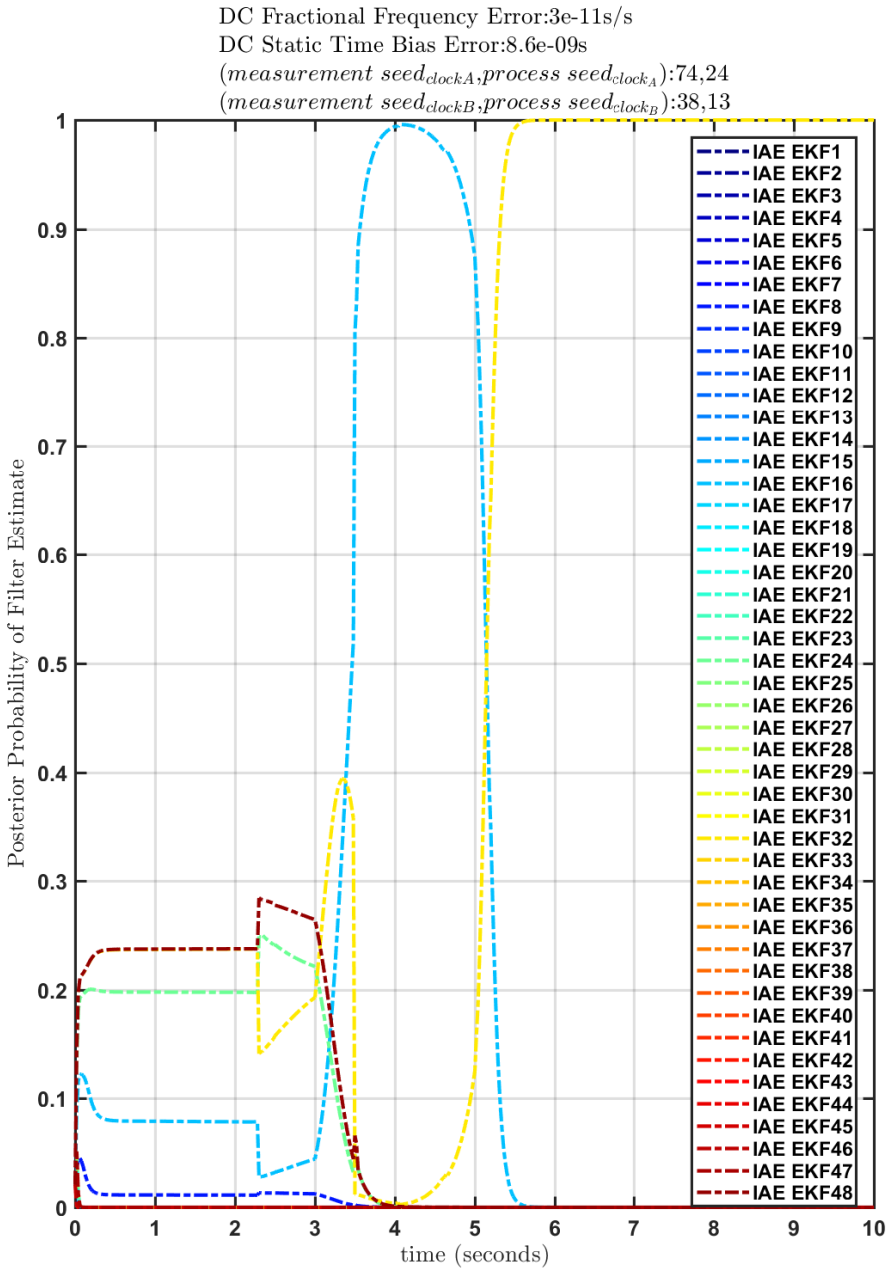

Figure B.143: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

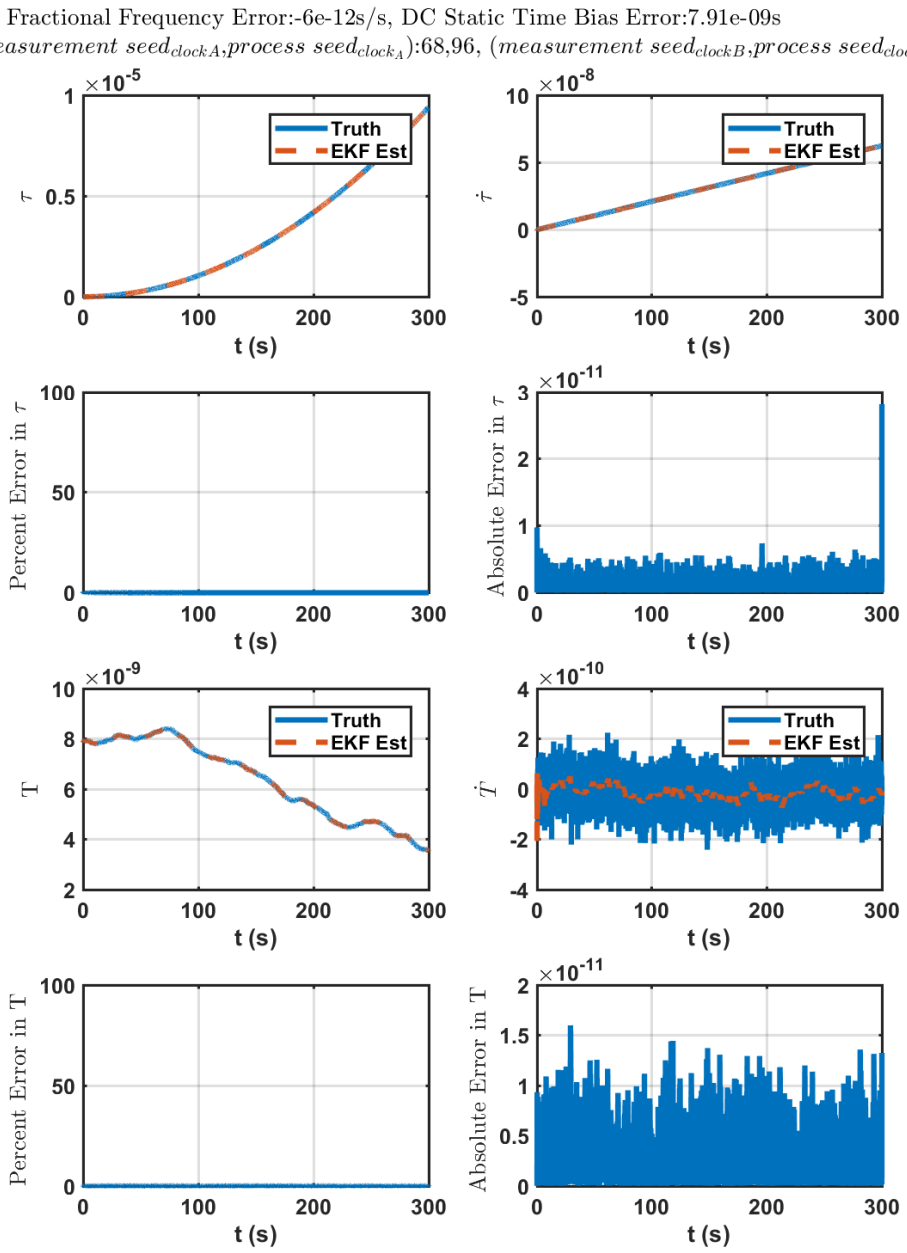

Figure B.144: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

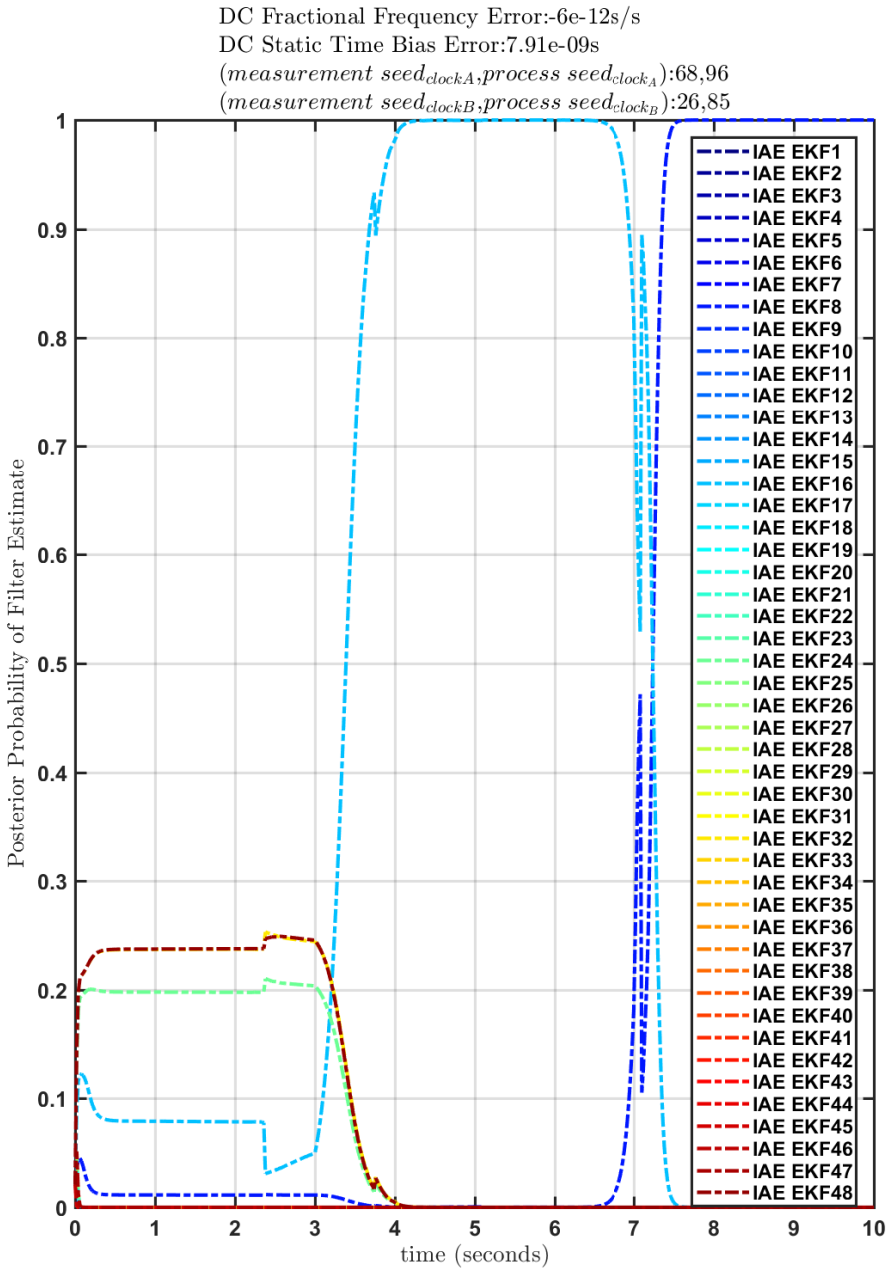

Figure B.145: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

DC Fractional Frequency Error:8e-12s/s, DC Static Time Bias Error:4.87e-09s
($measurement\ seed_{clock_A}$,$process\ seed_{clock_A}$):24,63, ($measurement\ seed_{clock_B}$,$process\ seed_{clock_B}$):40,

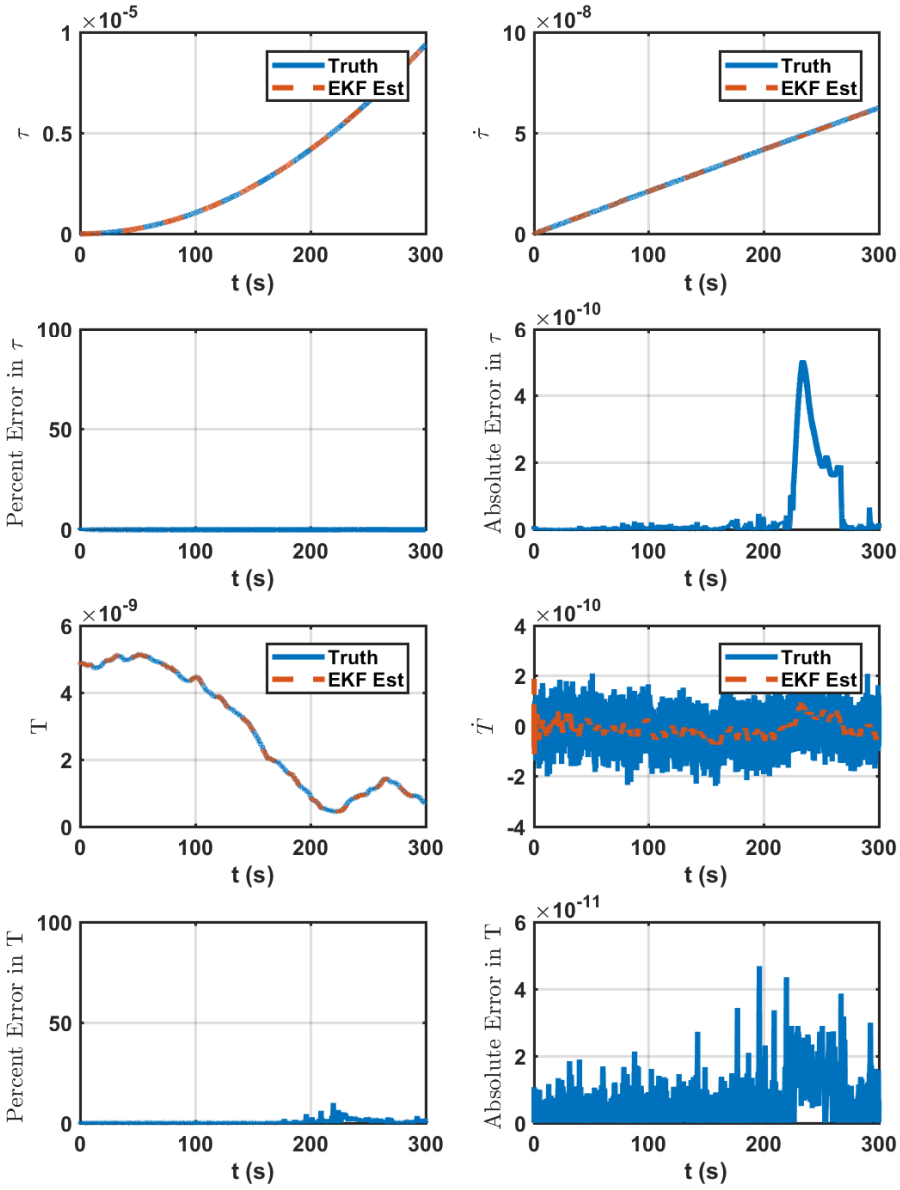

Figure B.146: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

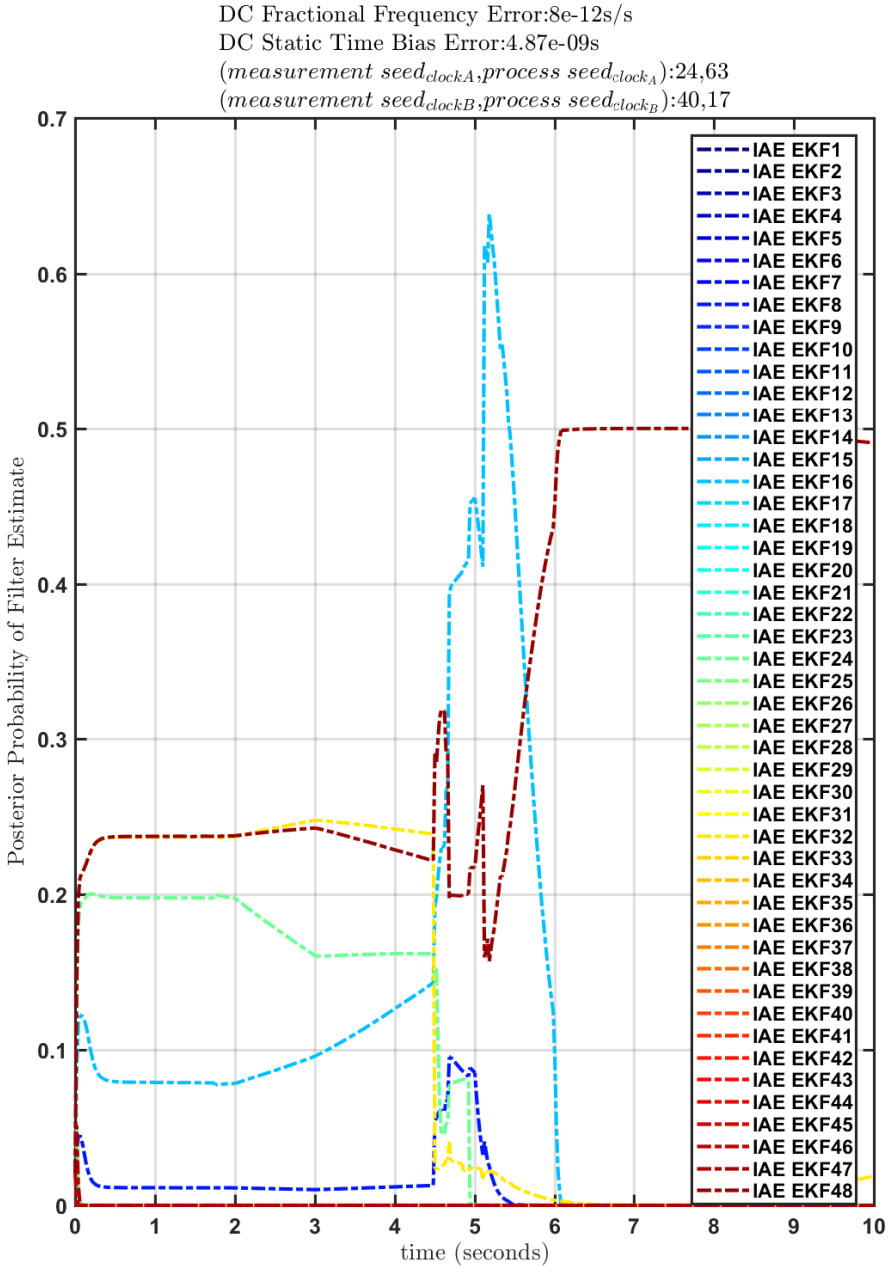

Figure B.147: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

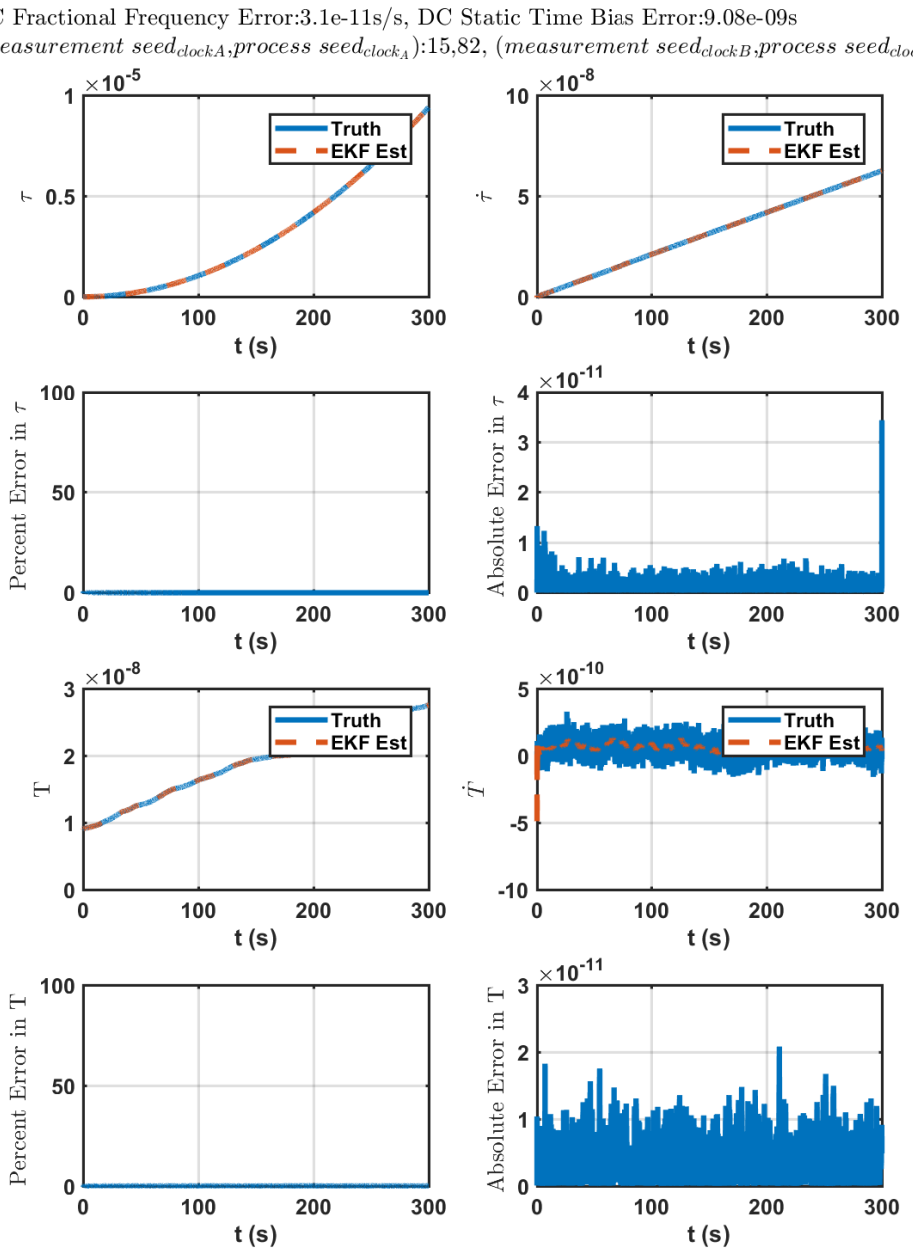

Figure B.148: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

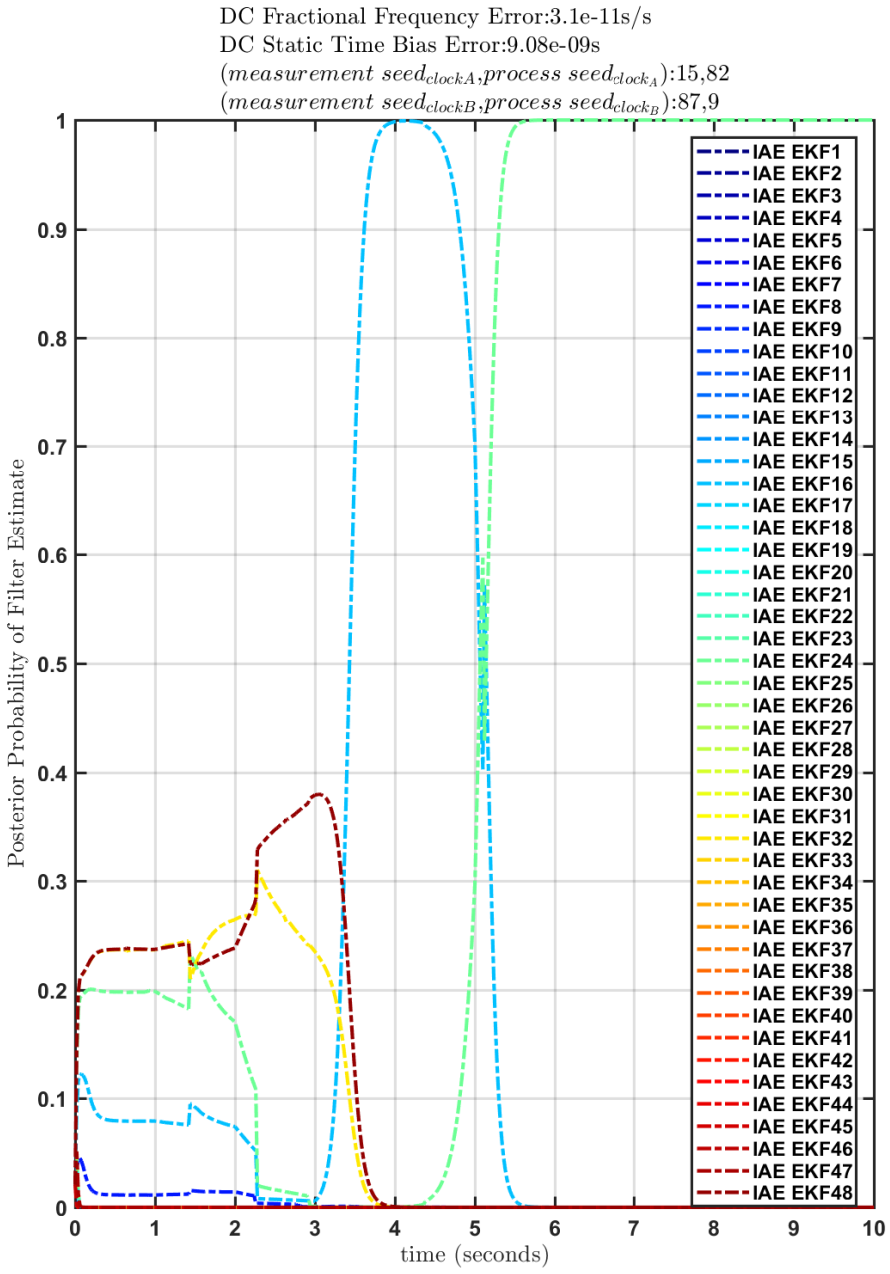

Figure B.149: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

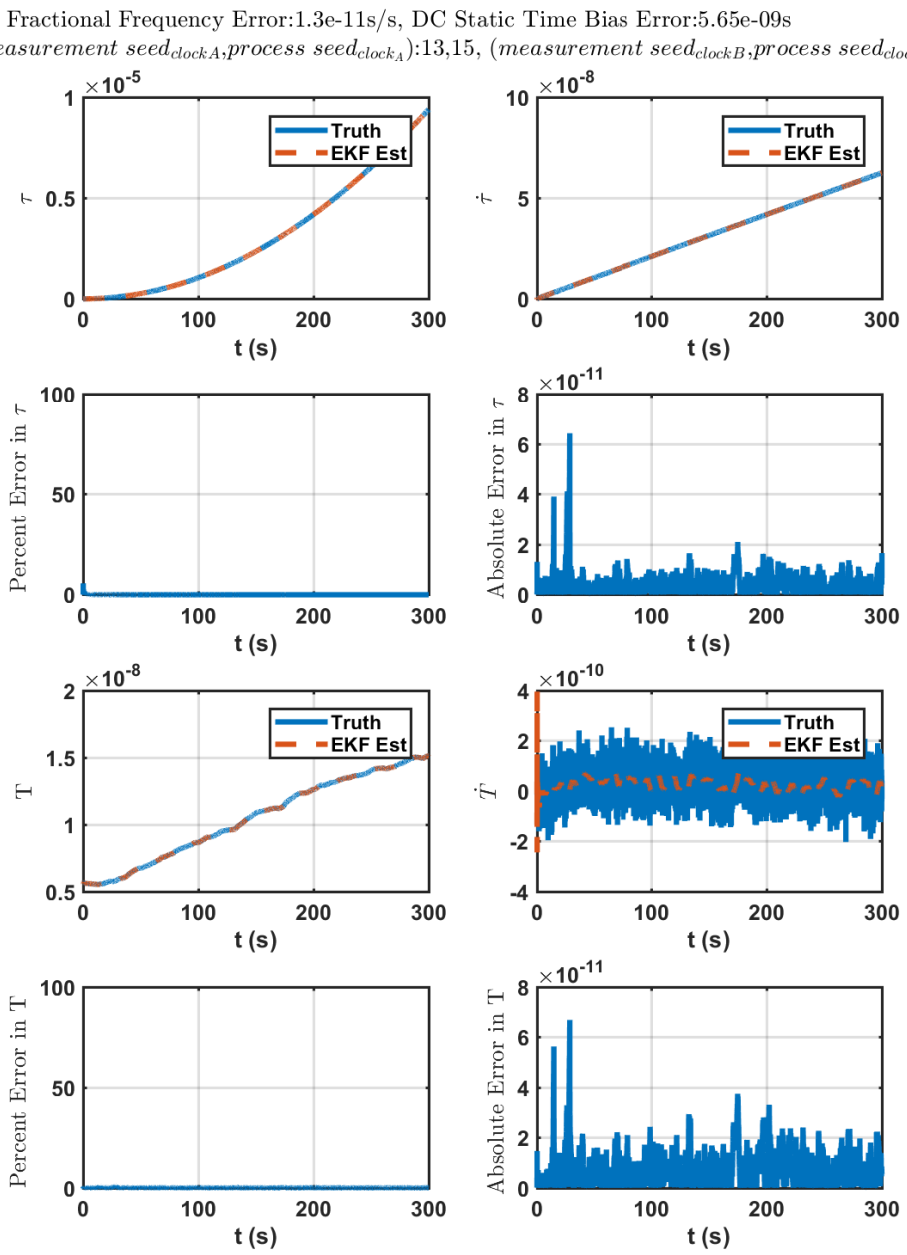

Figure B.150: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

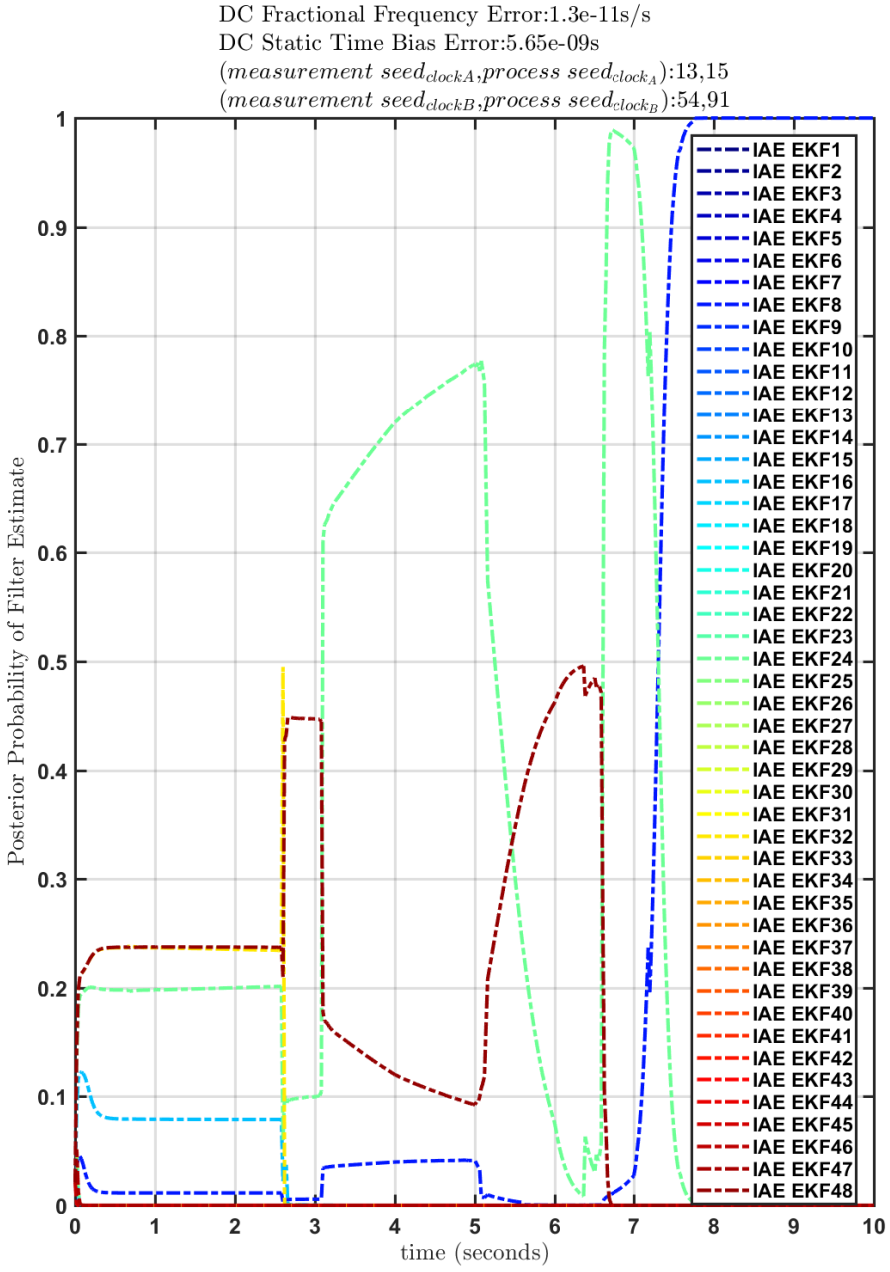

Figure B.151: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

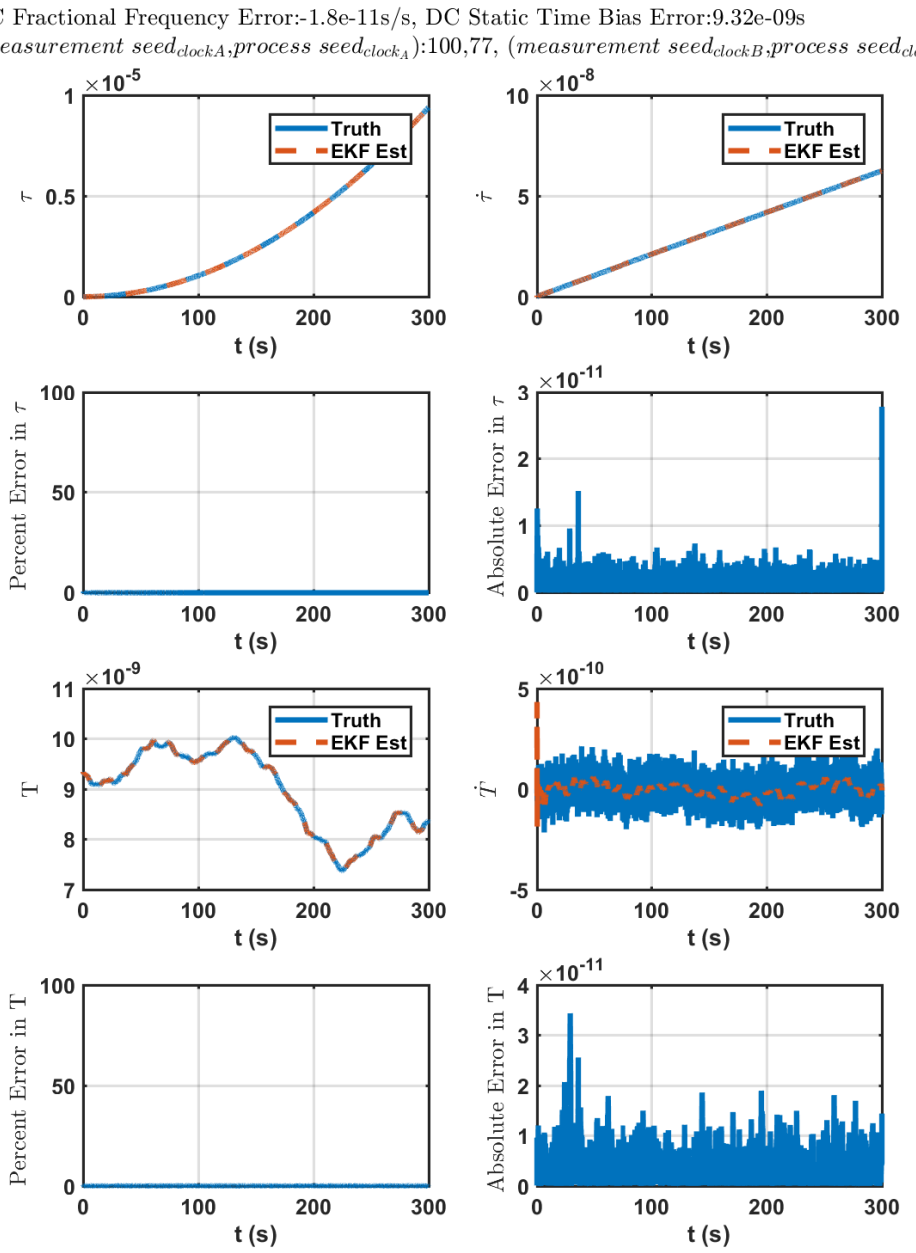

Figure B.152: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

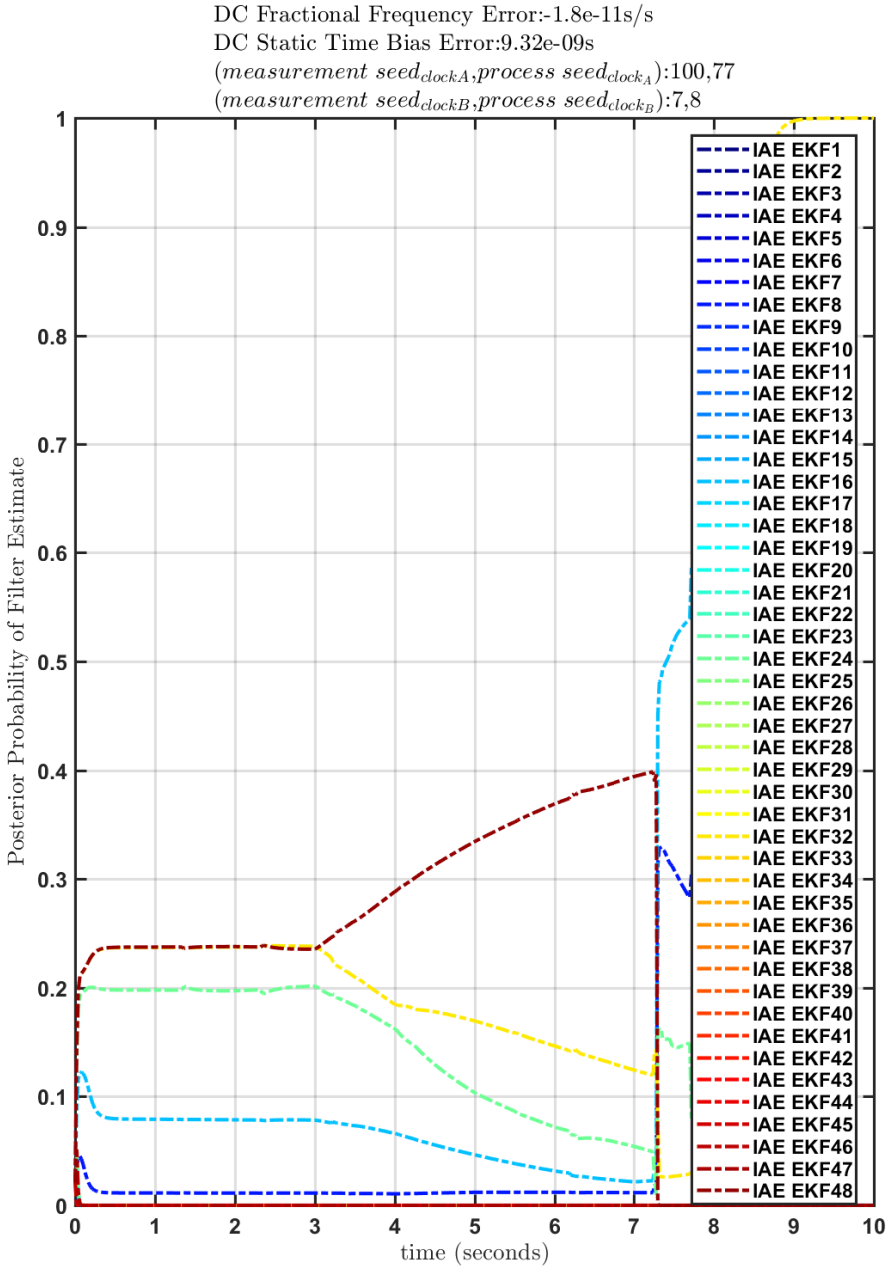

Figure B.153: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

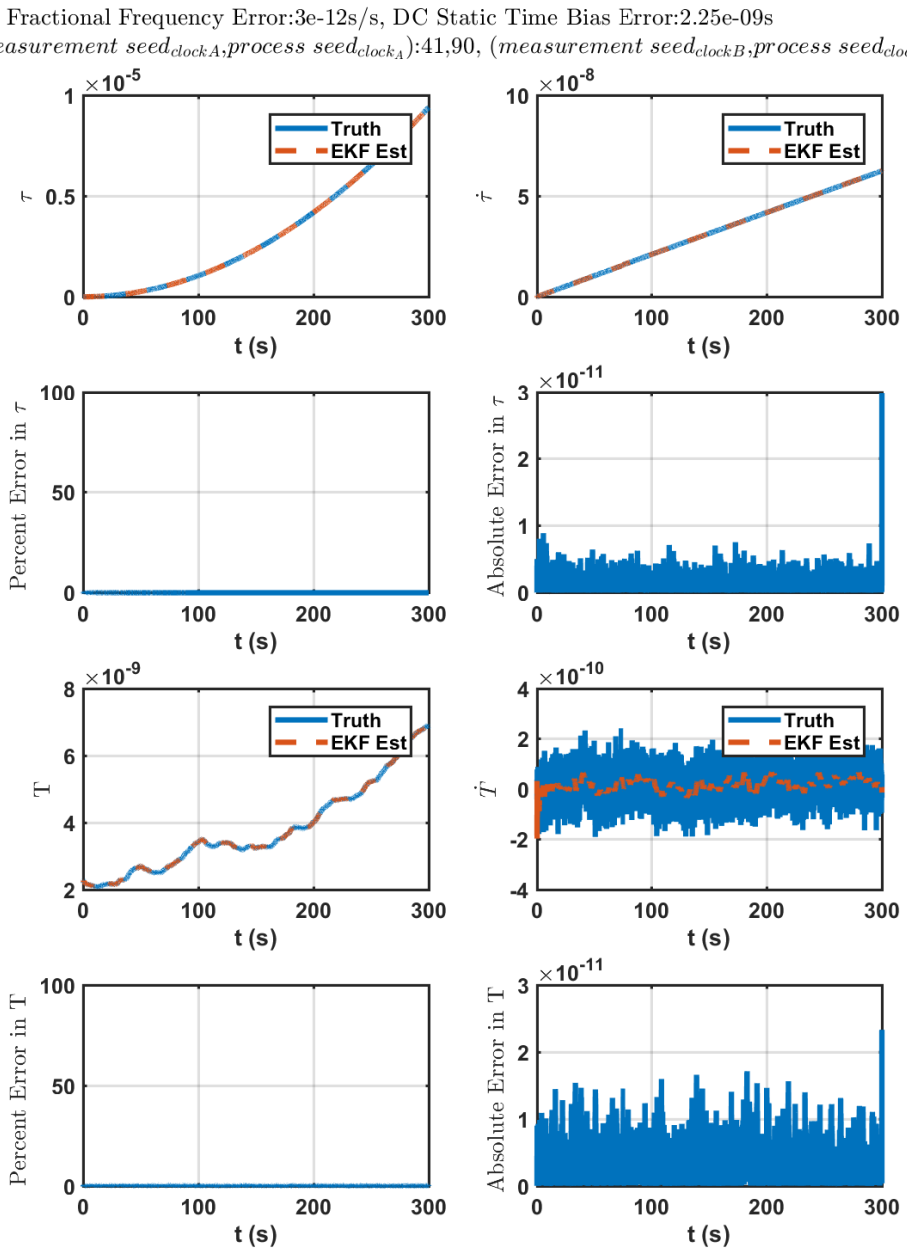

Figure B.154: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

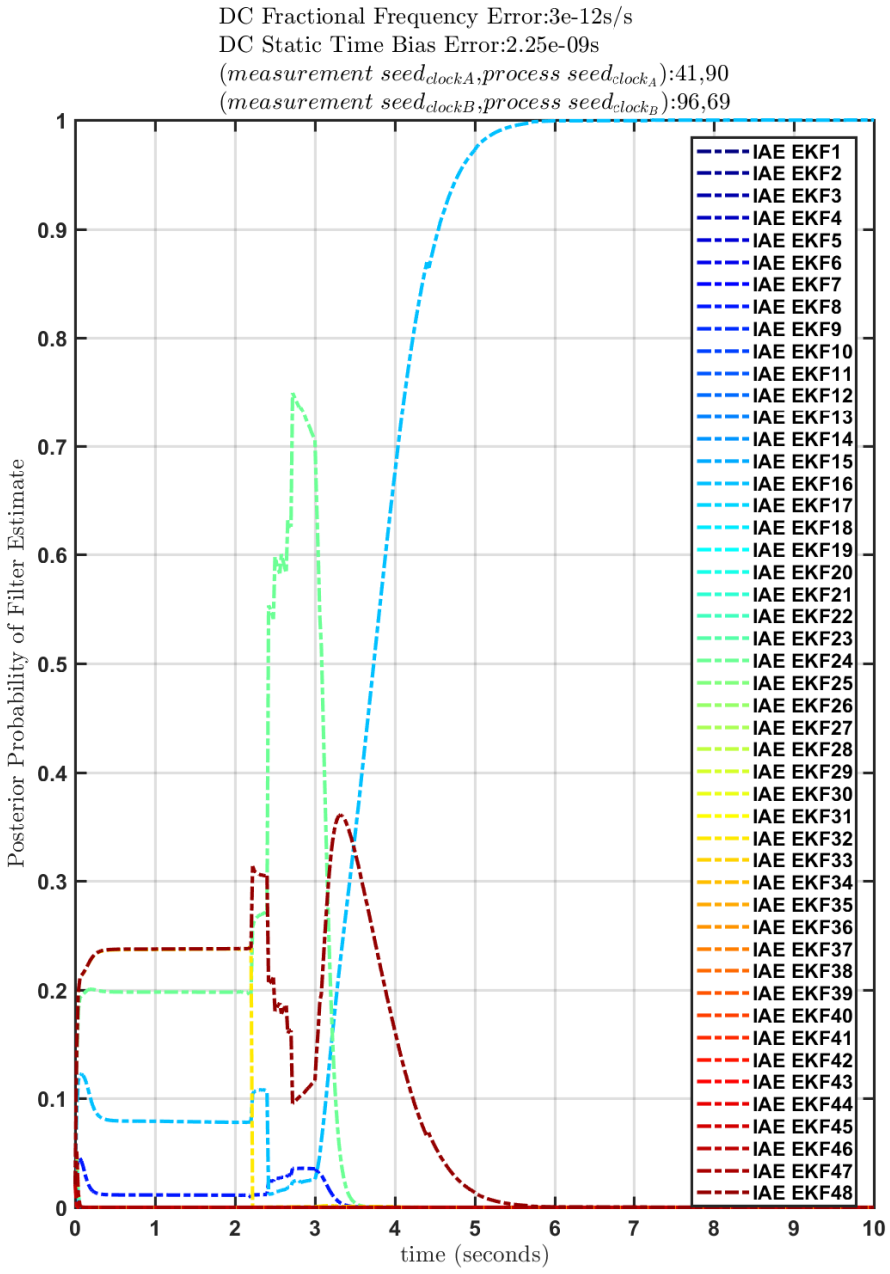

Figure B.155: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

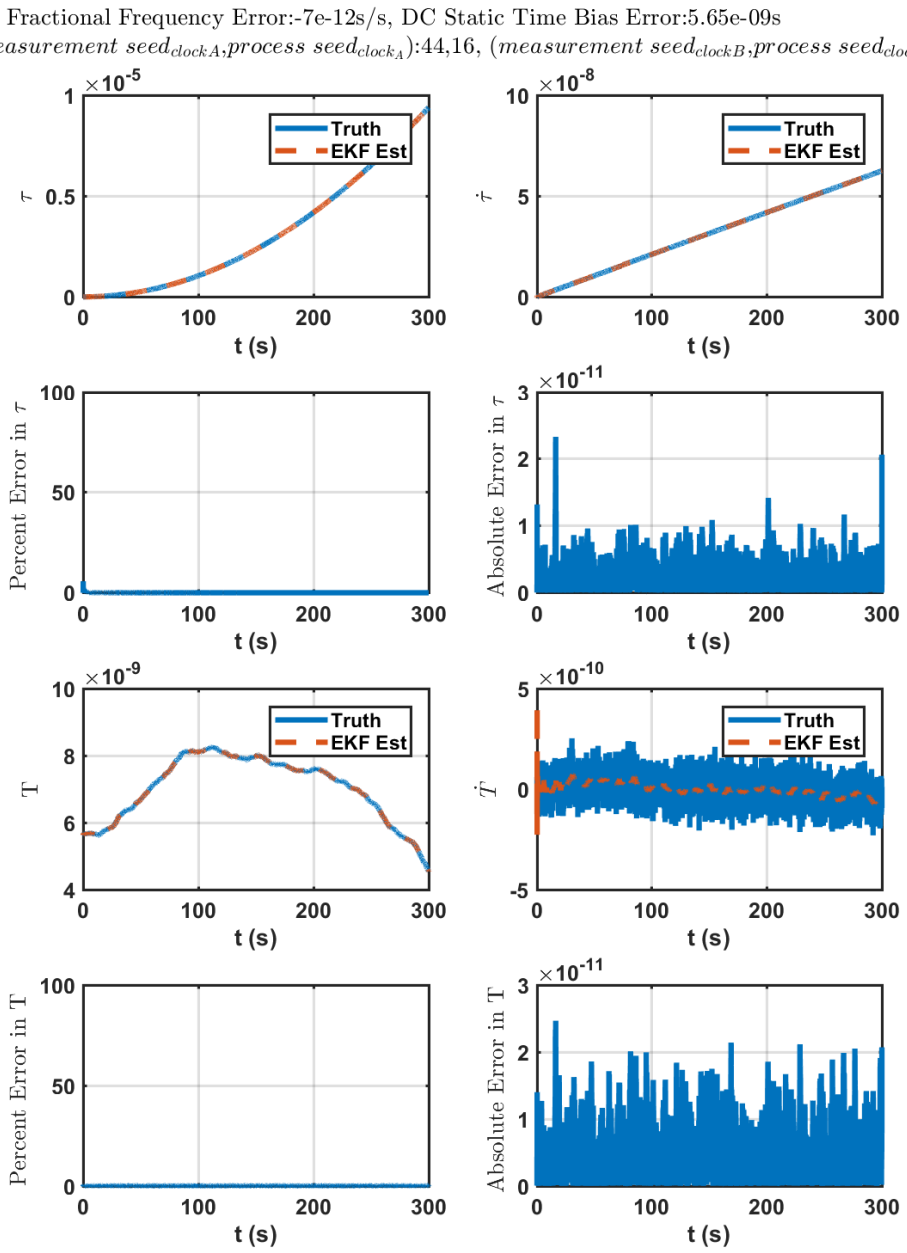

Figure B.156: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

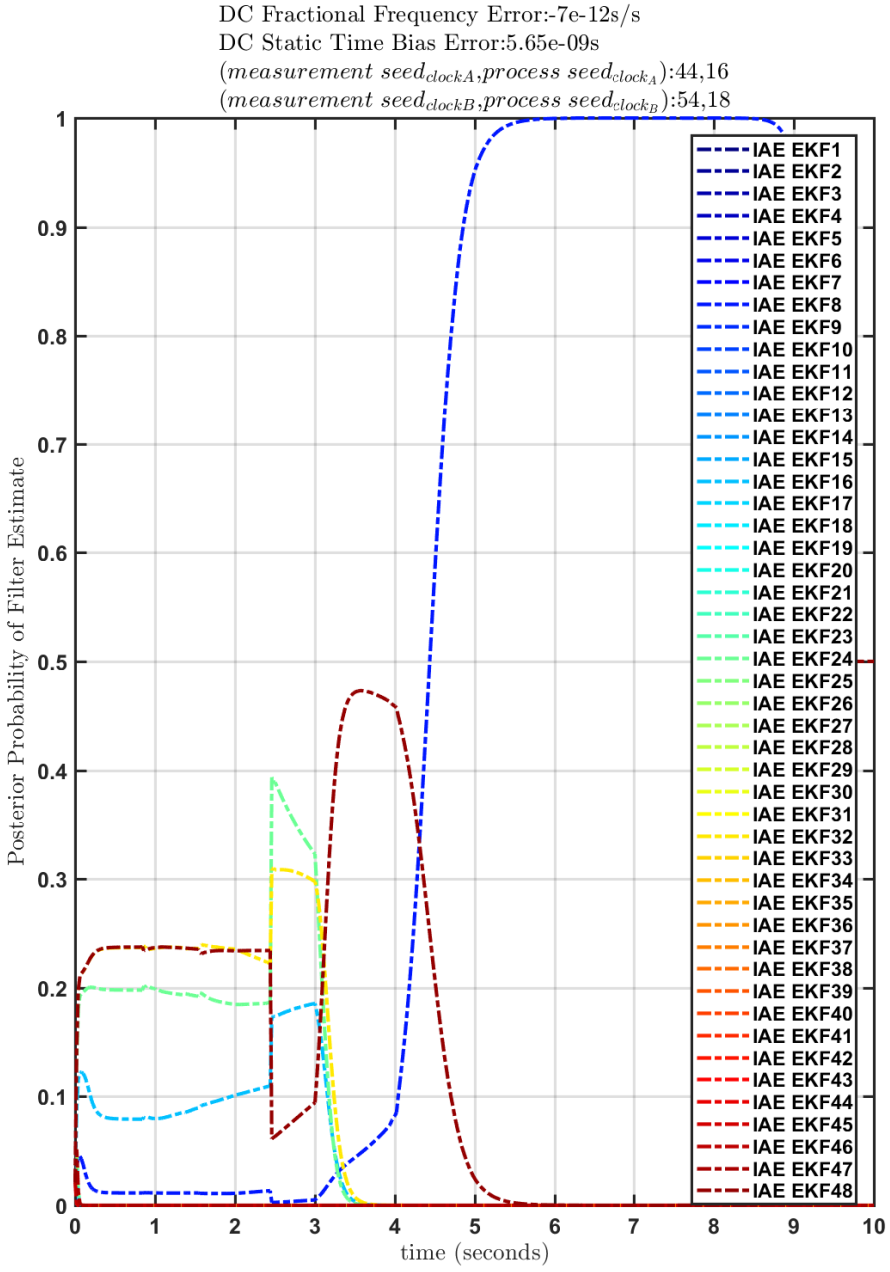

Figure B.157: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

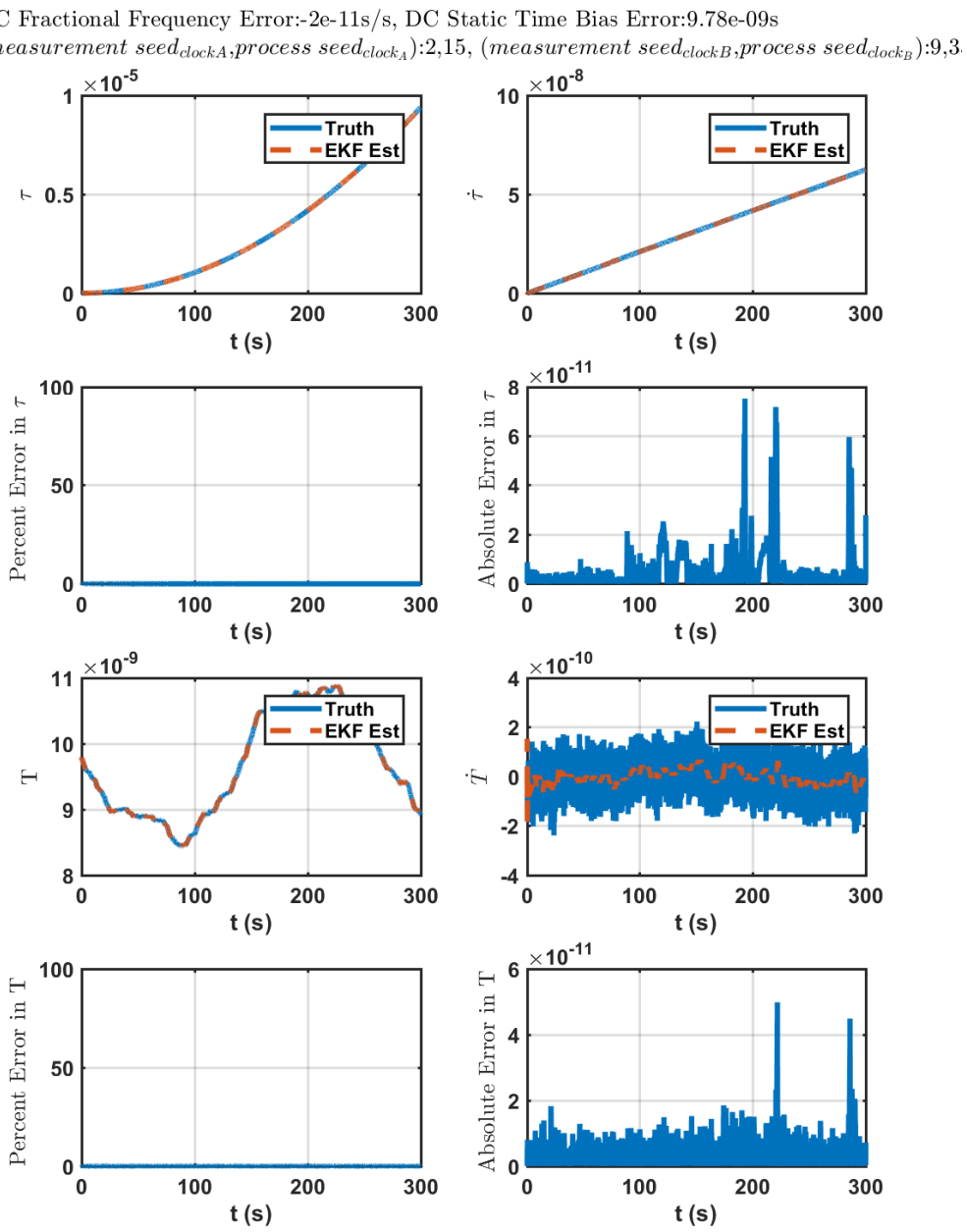

Figure B.158: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

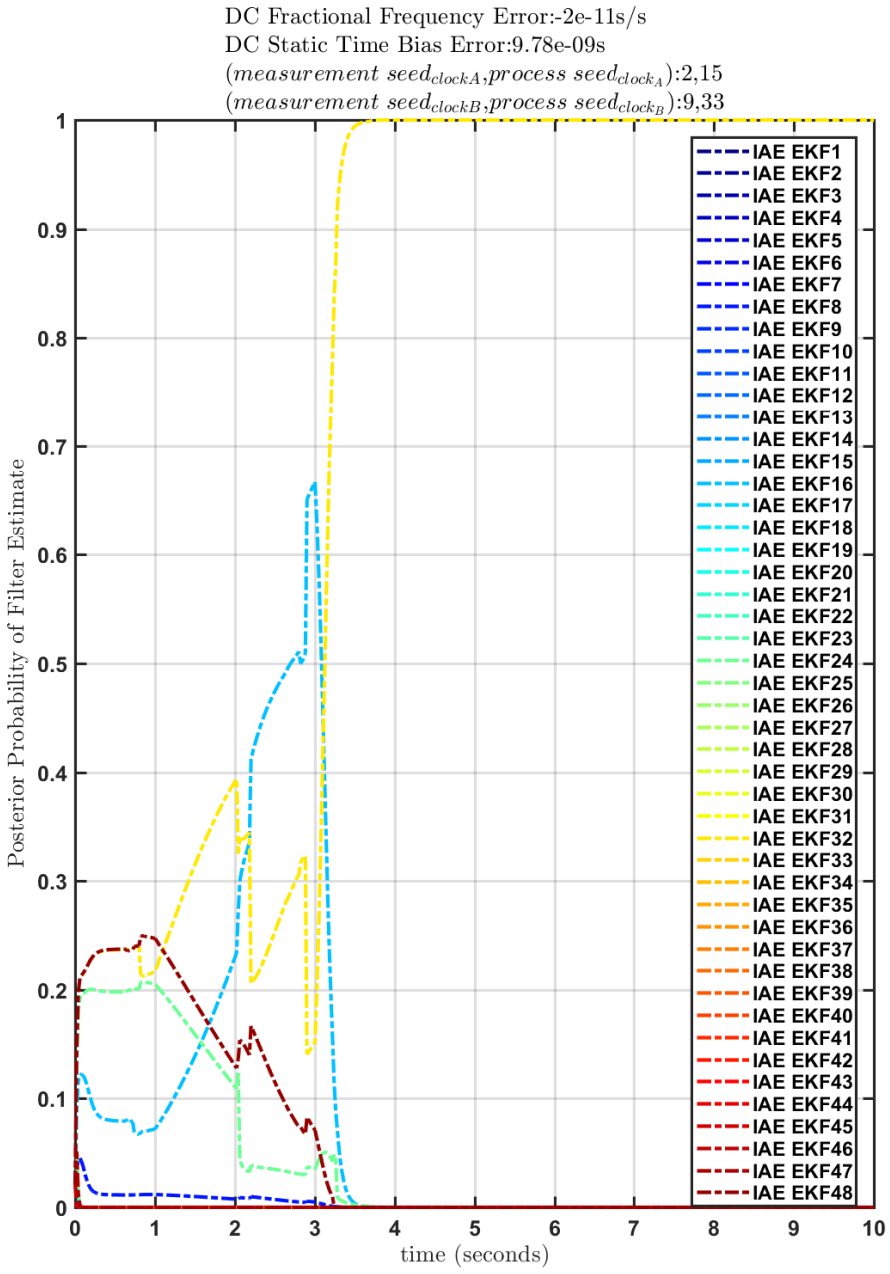

Figure B.159: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

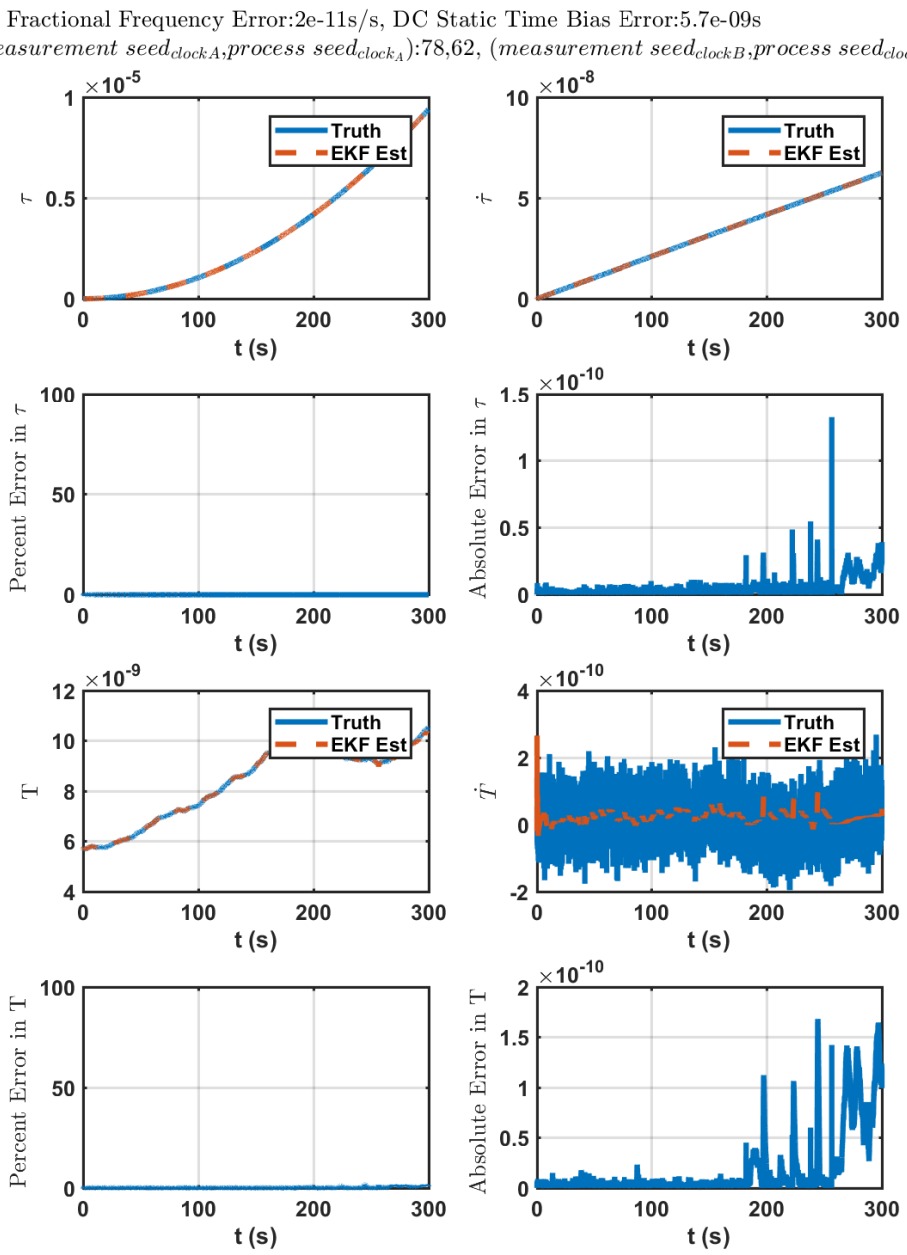

Figure B.160: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

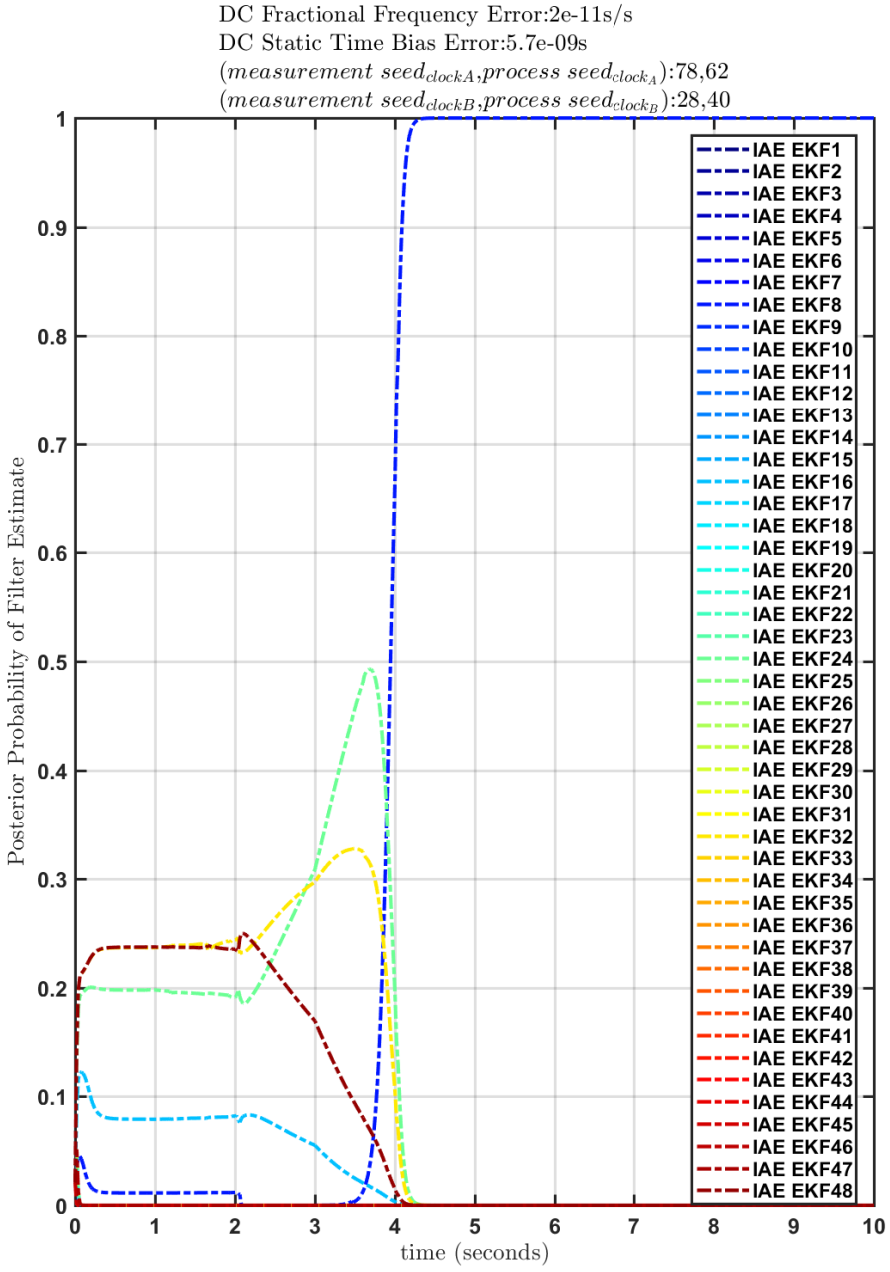

Figure B.161: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

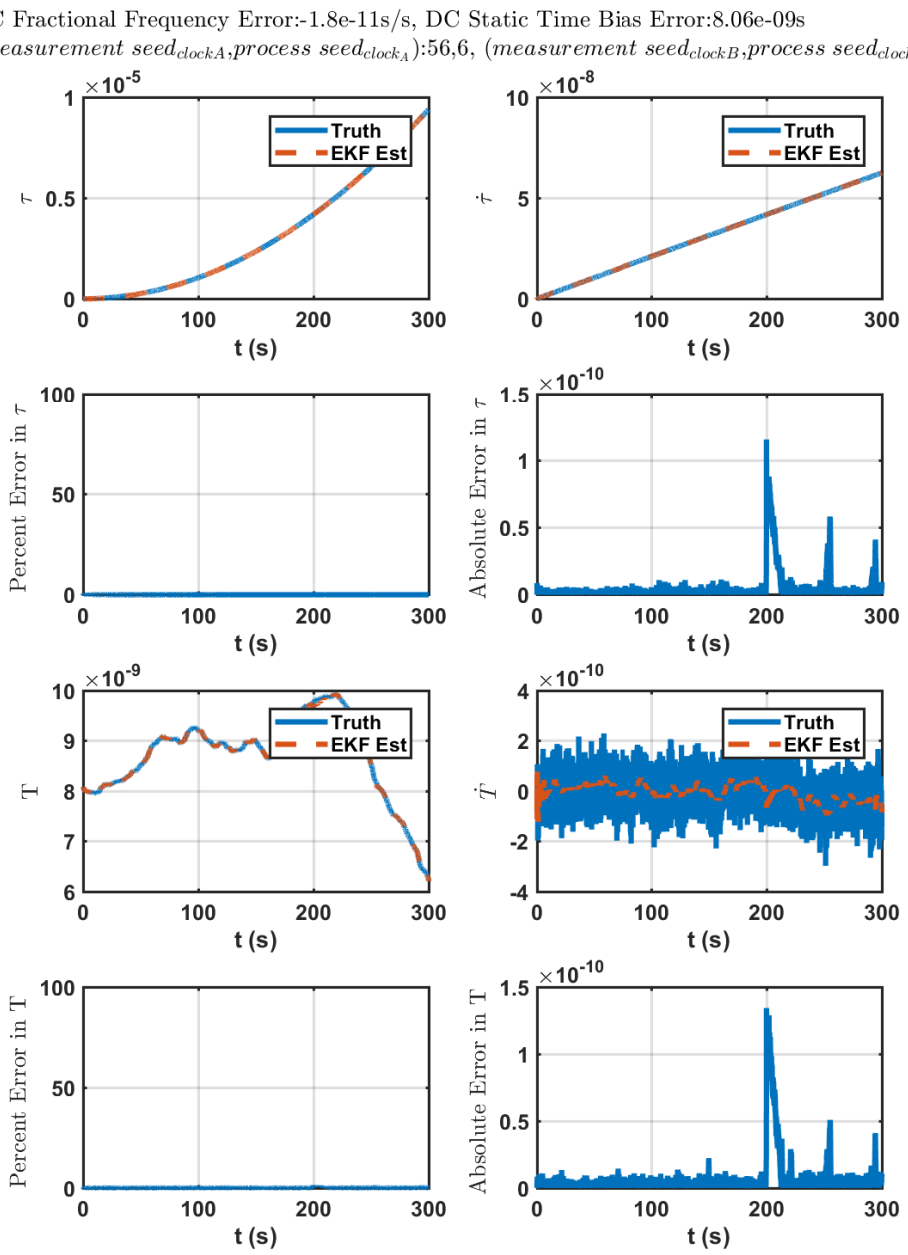

Figure B.162: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

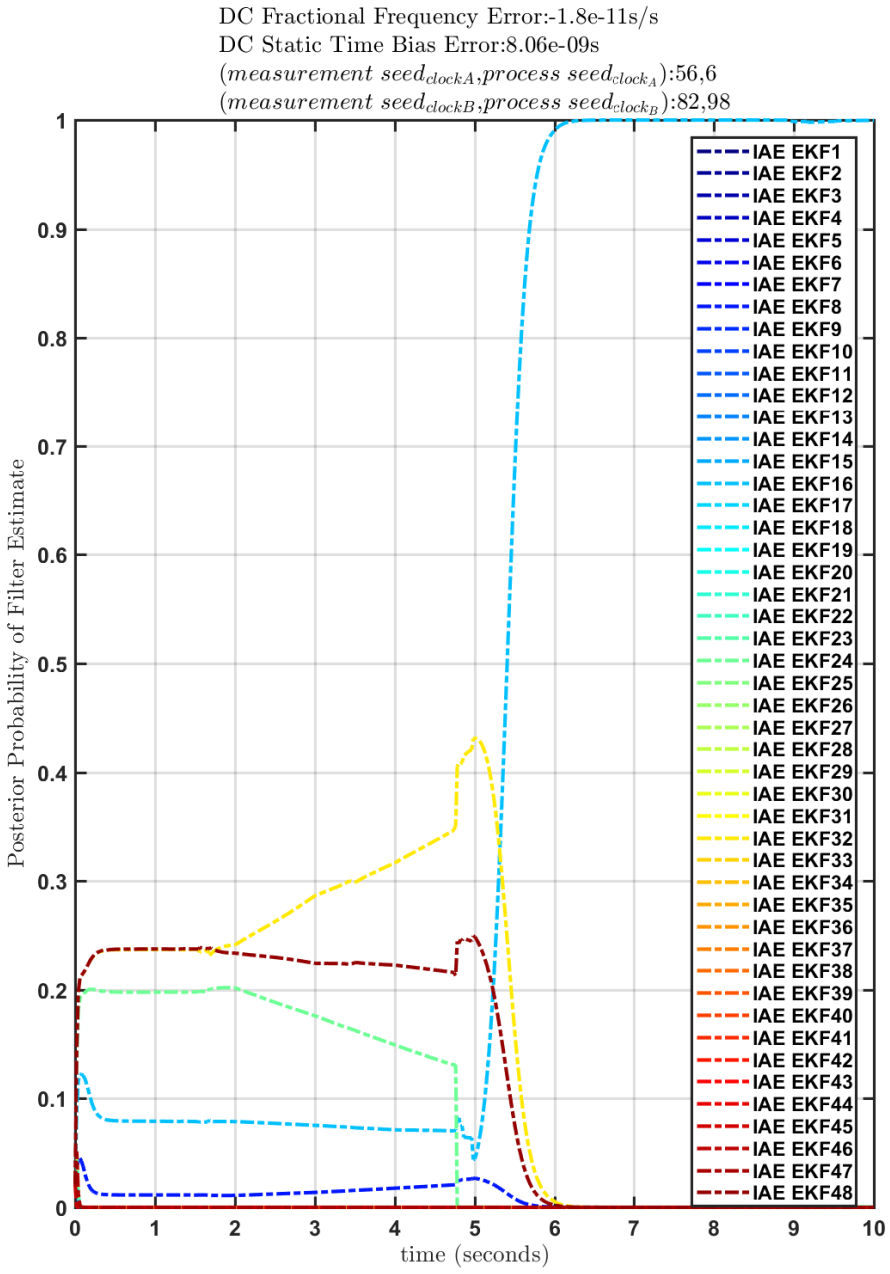

Figure B.163: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.

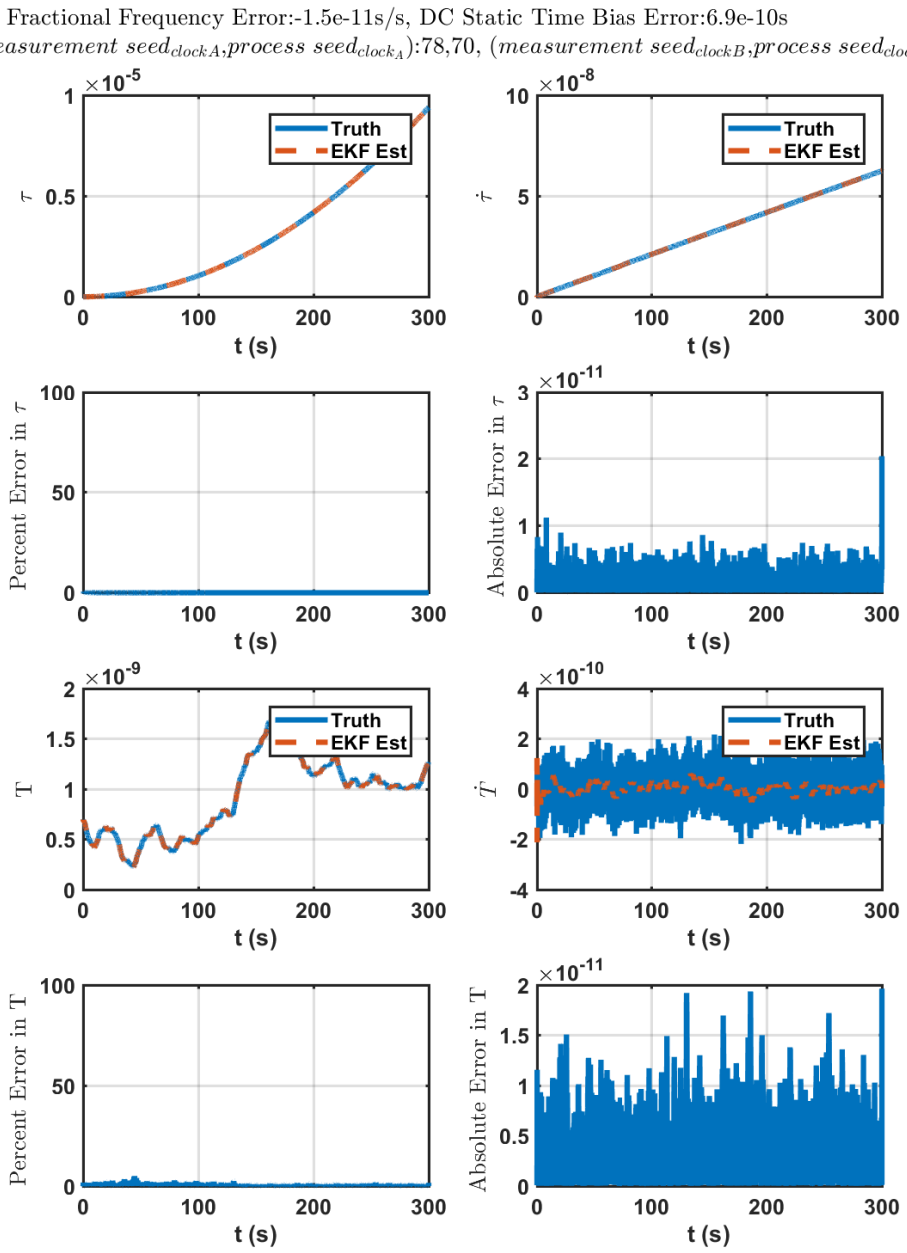

Figure B.164: State Estimates For HPPC Timing Protocol Adaptive Extended Kalman Filter Using Oustaloup Approximants.

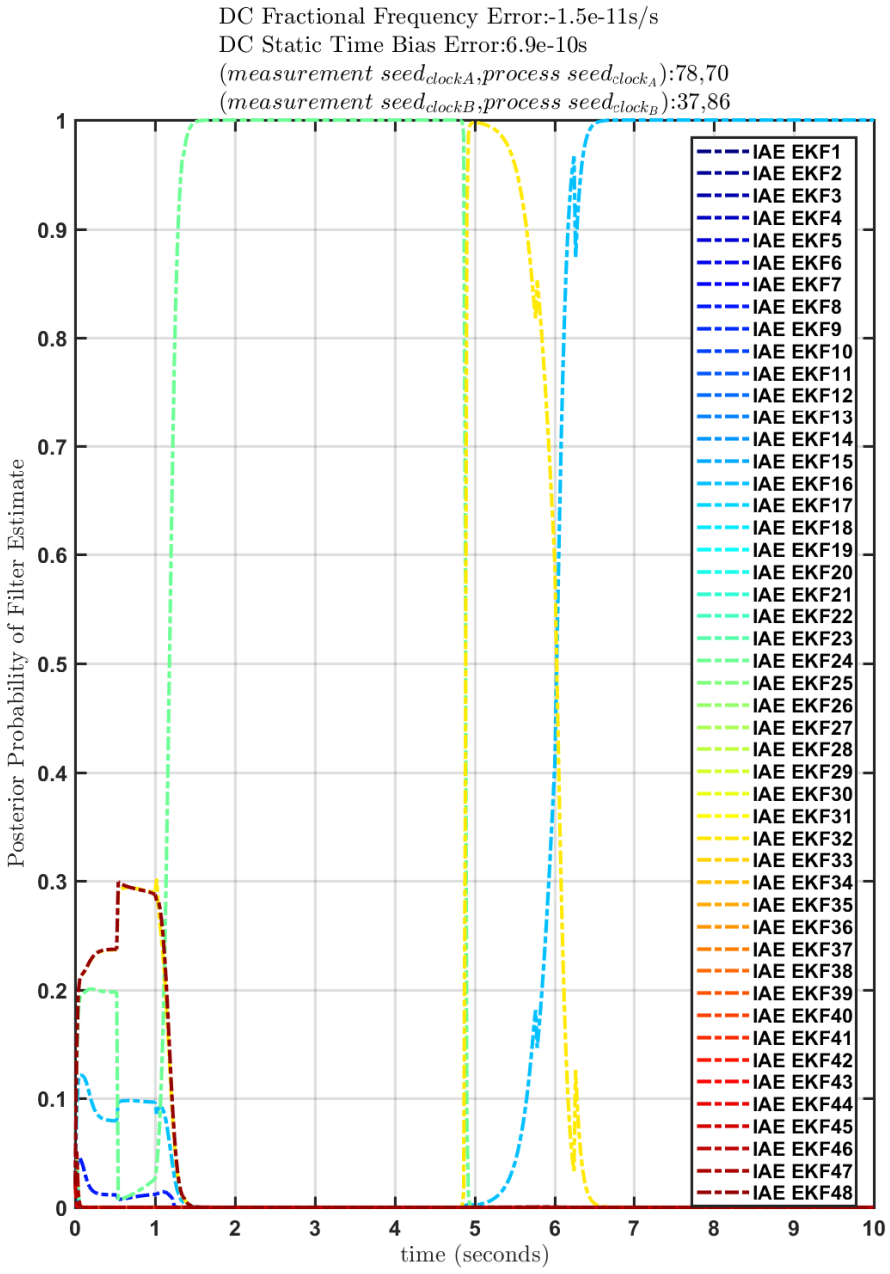

Figure B.165: Posterior Probability Of State Estimates From Each Filter Bank Constituent Using Oustaloup Approximants.