Simple Kalman Filter: Design Example

Jay A. Farrell[†]

***Abstract*— This article presents a (simple) Kalman filter design along with its solution. The article should come along with a data set. The intent is as follows:**

- **Readers complete the design themselves and implement it in Matlab or Python using the provided data.**
- **The solution is provided so that readers can get help when they get stuck. Please try to solve the problem on your own. You miss the valuable learning and implementation experience if you simply read the solution.**
- **Readers should buy a micro-processor (or equivalent) and IMU to implement the real-time code to try this out with a stationary IMU.**
- **The next steps would be to implement an AHRS (Chapter 10 in [1]) followed by a full INS (Chapter 11 in [1]).**

Please do not contact me with questions. I am providing this document and data as an educational service, because many students try to jump into significantly harder Kalman filter designs without ever trying and understanding a simple design, which often leads to frustration later.

If you find errors in the document or implementation, please do let me know.

I. Problem Statement

You are given measurements from a (stationary) accelerometer. The model of the time evolution of the scalar position $p(t)$ and velocity $v(t)$ as a function of the acceleration $a(t)$ is

$$\dot{p}(t) = v(t) \tag{1}$$
$$\dot{v}(t) = a(t). \tag{2}$$

The accelerometer measurement $\tilde{u} \in \Re$ is modeled as

$$\tilde{u}(t) = a(t) - b(t) - n(t) \tag{3}$$

where $a(t)$ is the acceleration, $b(t)$ is a sensor bias, and $n(t)$ is white random measurement noise with power spectral density Q_n (See Section 4.4.2 in [1]). The bias is modeled as a first-order Gauss-Markov process (see Section 4.6 in [1])

$$\dot{b}(t) = -\lambda b(t) + \omega(t) \tag{4}$$

where $\lambda \geq 0$ and $\omega(t)$ is white random measurement noise with power spectral density Q_ω.

The state vector is $\boldsymbol{x}(t) = [p(t), v(t), b(t)]^\top$.

Problem 1. Starting from a zero initial state, compute a (dead reckoning) estimate of the position and velocity by integrating the estimated acceleration

$$\hat{a}(t) = \tilde{u}(t) + \hat{b}(t)$$

through the system model of eqns. (1-2).

[†] Professor at the Dept. of Electrical Engineering, UC Riverside. {farrell}@ece.ucr.edu.

Problem 2. The state estimates will drift from their true values (zero). Plot the drift versus time. Explain why these plots have the shape that they do.

Problem 3. The estimated state vector is defined as

$$\hat{\boldsymbol{x}}(t) = \left[\hat{p}(t), \hat{v}(t), \hat{b}(t)\right]^\top.$$

Define the error state to be $\delta\boldsymbol{x} = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)$. Show that the continuous-time model for the error state is

$$\delta\dot{p}(t) = \delta v(t), \tag{5}$$
$$\delta\dot{v}(t) = \delta b(t) + n(t), \tag{6}$$
$$\delta\dot{b}(t) = -\lambda\delta b(t) + \omega(t). \tag{7}$$

Define the matrices $\mathbf{F}$ and $\mathbf{G}$ such that this model can be written as

$$\delta\dot{\boldsymbol{x}}(t) = \mathbf{F}\delta\boldsymbol{x}(t) + \mathbf{G}\boldsymbol{\omega}(t) \tag{8}$$

where $\boldsymbol{\omega}(t) = [\omega(t), n(t)]^\top$.

Problem 4. Manufacturer specification sheets often show noise power spectral densities in non-ANSI units. This problem works through the unit conversions. In the following, the notation Q_n and Q_ω denote the (constant as a function of frequency) power spectral density of the white noise processes $n(t)$ and $\omega(t)$, respectively.

Assume that $Q_n \doteq \sigma_n^2 = 1.0 \times 10^{-4} \frac{(m/s)^2}{s}$ in ANSI units. The parameter σ_n is the velocity random walk (VRW) parameter and may be given in various other units: $\frac{(m/s/s)}{\sqrt{Hz}}$, $\frac{(m/s)}{\sqrt{s}}$ or $\frac{(m/s)}{\sqrt{Hr}}$. What are the ANSI units of σ_n? Find the numeric value of σ_n in these alternative units.

Problem 5. Find the VRW parameter on the attached manufacturer spreadsheet and convert its value to ANSI units.

Problem 6. Assuming a sample period of $T = 1.0$ seconds, find the state transition matrix $\boldsymbol{\Phi}$ and driving noise covariance matrix $\mathbf{Q}_d$ for the equivalent discrete-time model:

$$\delta\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}\delta\boldsymbol{x}_k + \boldsymbol{\omega}_k \tag{9}$$

where $\boldsymbol{\omega}_k \sim N(\mathbf{0}, \mathbf{Q}_d)$.

For numeric computations, use the ANSI parameters from Problem 3 and let $\lambda = 0.001Hz$, $Q_n = 4.4 \times 10^{-7} \frac{(m/s)^2}{s}$, and $Q_\omega = 8 \times 10^{-6} \frac{(m/s/s)^2}{s}$.

Problem 7. Design and implement a Kalman filter to estimate the state of the system assuming that a position measurement occurs with a frequency $F_s = \frac{1}{T}$:

$$y_k = y(kT) = p(kT) + \eta(kT)$$

where $\eta_k \sim N(0, R)$ is a assumed to be white.

To implement the measurements, we will use the fact that the accelerometer is stationary and define its initial position to be zero. Therefore, each y_k has a value of zero. The only error in this 'measurement' is due to vibration of the table, which is assumed to have magnitude significantly less than a millimeter. Therefore the attached Matlab implementation uses a value of $R = \left(1.0 \times 10^{-5} m\right)^2$.

Consider the following questions:

1) Is the state of the system observable from the position measurement?
2) How many position measurements will be required before the bias becomes observable?
3) Plot both the time history of the residual r_k and the time history of the normalized residual $\eta_k = \frac{r_k}{\sqrt{S_k}}$ where $S_k = cov(r_k)$. Look at the first few seconds. Can you explain their shapes? Is the residual sequence white?
4) Plot the histogram of the residual r_k and the normalized residual $\eta_k = \frac{r_k}{\sqrt{S_k}}$. The normalized residual should be a standard normal random variable. Is it?
5) Plot position, velocity, and bias versus time. Focus on the first and last few seconds separately. Can you explain the shapes.

When plotting any random quantity, always also plot plus and minus the (computed) standard deviation of that variable. This is necessary for any proper discussion of the variable.

II. Solutions

Solution 1. The estimates of position $\hat{p}(t)$ and velocity $\hat{v}(t)$ are computed by integrating

$$\begin{aligned} \dot{\hat{p}}(t) &= \hat{v}(t) && (10)\\ \dot{\hat{v}}(t) &= \tilde{u}(t) - \hat{b}(t) && (11)\\ \dot{\hat{b}}(t) &= 0 && (12) \end{aligned}$$

Using Euler integration, in discrete-time, the algorithm is

$$\begin{aligned} \hat{p}_{k+1} &= \hat{p}_k + \hat{v}_k T && (13)\\ \hat{v}_{k+1} &= \hat{v}_k + \tilde{u}_k T - \hat{b}_k T && (14)\\ \hat{b}_{k+1} &= \hat{b}_k && (15) \end{aligned}$$

where $t_k = kT$ and $\boldsymbol{x}_k = \boldsymbol{x}(t_k) = \boldsymbol{x}(kT)$. These equations can be integrated through the duration of the data.

More advanced integration algorithms (i.e., predictor-corrector, Runge-Kutte) are possible. Try then and compare the results.

Solution 2. Even though the accelerometer is stationary, the velocity and position estimates grow (approximately) linearly and parabolicly with time. This is because the accelerometer bias is unknown and distinct from $\hat{b}$; therefore, the bias error is large (and relatively constant). The integral of a constant is a line, so the velocity estimate grows linearly, with a slope approximately equal to the value of the bias. The second integral of a bias is a parabola, which explains the shape of the position error.

Initialize the value of $\hat{b}(0)$ to some reasonable value (e.g., the first acceleration measurement or the average of the acceleration measurements), reintegrate the data. Can you explain the results?

Solution 3. Differencing eqns. (1), (2), and (4) with equations (10-12) respectively, using eqn. (3) to eliminate $\tilde{u}$, yields eqns. (5-7). See Section 1.1.1 in [1].

Eqn. (**??**) is equivalent to eqns. (10-12) for

$$\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\lambda \end{bmatrix} \text{ and } \mathbf{G} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}. \qquad (16)$$

Solution 4. In ANSI units $\sigma_n = 1.0 \times 10^{-2} \frac{(m/s)}{\sqrt{s}}$. This is equivalent to

$$\sigma_n = 1.0 \times 10^{-2} \frac{(m/s)}{\sqrt{s}} \frac{\sqrt{s}}{\sqrt{s}} = 1.0 \times 10^{-2} \frac{(m/s/s)}{\sqrt{Hz}}.$$

Similarly, this is equivalent to

$$\sigma_n = 1.0 \times 10^{-2} \frac{(m/s)}{\sqrt{s}} \frac{60\sqrt{s}}{\sqrt{Hr}} = 6.0 \times 10^{-1} \frac{(m/s)}{\sqrt{Hr}}.$$

Note that it is Q_n that is needed to specify the continuous-time stochastic model, so the designer reads σ_n, the VRW parameter, from the manufacturer specification sheet, converts it to ANSI units, squares it to compute Q_n, and proceeds with the design.

Solution 5. On page 2 of the specification sheet, the manufacturer states that the VRW parameter is $\sigma_n = 0.04 \frac{m/s}{\sqrt{Hr}}$. The ANSI value is

$$\begin{aligned} \sigma_n &= 4.0 \times 10^{-2} \frac{(m/s)}{\sqrt{Hr}} = 4.0 \times 10^{-2} \frac{(m/s)}{\sqrt{Hr}} \frac{\sqrt{Hr}}{60\sqrt{s}} \\ &= 6.67 \times 10^{-4} \frac{(m/s)}{\sqrt{s}}. \end{aligned}$$

Therefore, $Q_n = 4.4 \times 10^{-7} \frac{(m/s)^2}{s}$.

Solution 6. This step is worked out in detail using symbols in Section 4.9.3 of [1] for the case of $\lambda = 0.0$.

Alternatively, for the stated numeric values and a given value of T, defining $\mathbf{Q} = \begin{bmatrix} Q_\omega & 0 \\ 0 & Q_n \end{bmatrix}$, numeric values for $\mathbf{\Phi}$ and $\mathbf{Q}_d$ can be computed using eqns. (4.113-4.115) in [1]. Different values of T yield different results. For $T = 1.0s$, the results are

$$\mathbf{\Phi} = \begin{bmatrix} 1.0000 & 1.0000 & 0.4998 \\ 0.0000 & 1.0000 & 0.9995 \\ 0.0000 & 0.0000 & 0.9990 \end{bmatrix}$$

and

$$\mathbf{Q}_d = 10^{-6} \times \begin{bmatrix} 0.5464 & 1.2193 & 1.3320 \\ 1.2193 & 3.1047 & 3.9960 \\ 1.3320 & 3.9960 & 7.9920 \end{bmatrix}.$$

Solution 7. The Kalman filter implementation will look something like the following:

1) Precompute the constants $\mathbf{\Phi}$, $\mathbf{Q}_d$, R, and $\mathbf{H} = [1, 0, 0]$.
2) Initialize the state error covariance matrix $\mathbf{P}_0$ and state estimate $\boldsymbol{x}_0$.
3) Enter a loop that processes the IMU data
   a) Integrate the state vector one time step forward using the accelerometer data.
   b) When the time advances T seconds from the last measurement (i.e., $t = kT$), implement the Kalman filter measurement update:
      i) Compute the predicted measurement: $\hat{y}_k = \mathbf{H}\hat{x}_k$
      ii) Compute the residual measurement: $r_k = y_k - \hat{y}_k$ (where $y_k = 0$ for all k)
      iii) Advance the error covariance to the measurement time: $\mathbf{P}_k = \mathbf{\Phi}\mathbf{P}_{k-1}\mathbf{\Phi}^\top + \mathbf{Q}_d$
      iv) Compute the variance of the residual:
      $$S_k = \mathbf{H}\mathbf{P}_k\mathbf{H}^\top + R$$
      v) Compute the Kalman gain: $\mathbf{K} = \mathbf{P}_k\mathbf{H}^\top S_k^{-1}$
      vi) Compute the updated state estimate:
      $$\boldsymbol{x}_k = \boldsymbol{x}_k + \mathbf{K}r_k$$
      vii) Compute the erro covariance at the measurement time, after including the information from the measurement: $\mathbf{P}_k = (\mathbf{I} - \mathbf{K}\mathbf{H})\,\mathbf{P}_k$

   There are various alternative implementations that may be more computationally efficient or numerically stable. This implementation is straightforward and easy to understand. Experiment with alternative implementations. Their performance should be identical.

4) Plot and analyze the results.

The only unspecified quantity in the above description is the matrix $\mathbf{P}_0$. Because the initial position and velocity are known, their initial covariance will each be zero. The standard deviation of the initial bias is given on the manufacturer data sheet as $\sigma_b = 8mg = 7.2 \times 10^{-2} \frac{m}{s^2}$. Therefore, $\mathbf{P}_0 = diag([0, 0, \sigma_b^2])$.

The implemented Kalman filter is in the file 'SimpleKF_IMU_data.m. In that file, T is defined using the IMU sample rate for the computation of $\mathbf{\Phi}$ and $\mathbf{Q}_d$. That definition allows $\mathbf{P}(t)$ to be computed at the IMU sample rate which is useful for the plots that follow as it shows the accumulation of uncertainty through integration and the reduction of uncertainty following position measurements. If $\mathbf{\Phi}$ and $\mathbf{Q}$ are accumulated over the a one second interval, then the values computed in Solution 6 should be recovered.

Answers to the specific questions are as follows:

1) The observability analysis can be performed in either continuous or discrete time. In continuous time, the observability matrix is

$$\mathbf{O} = \begin{bmatrix} \mathbf{H} \\ \mathbf{HF} \\ \mathbf{HFF} \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

which is full rank, showing that the system is observable. Note that the stochastic portions of the model do not affect the observability analysis.

2) The answer depends on the assumed the initial uncertainty which is reflected through the error covariance $\mathbf{P}_0$. For most instances of the initial uncertainty, three measurements would be required. In the particular case where the initial position and velocity are zero, the bias can be estimated from a single position measurement.

3) The normalized and unnormalized residuals are shown in the left column of Fig. II. For the unnormalized residual, there is one large value of about -0.04 at $t = 1.0$ that results in a large axis scaling. The bottom row shows the normalized residuals. The graph versus time does not show any obvious time correlations.

4) The histograms of the normalized and unnormalized residuals are shown in Fig. II. Ignoring the first 10 residuals, to avoid the initial estimation transients, the upper histogram on the right shows the distribution of the unnormalized residuals, with most values less than a millimeter. The histogram of all the normalized residuals is on the bottom right, which shows that the variance is less than the value of 1.0 that is expected, indicating that the system is not quite tuned correctly.

5) Plots of the state (errors) versus time for the first and last ten seconds is shown in Fig. II. For the bias, the error is defined by subtracting the final estimated value from all the previous values. Both the velocity error and its covariance grow linearly and rapidly in the first second. Both the position error and its covariance grow parabolically during the first second. These facts allow the bias to be estimated so that on all future steps, the rates of growth are significantly less. The right column of figures shows the steady state operation for the specified filter parameters. In each second, A few millimeters of position is accumulated through the IMU integration process, far less that the 4 centimeters of error accumulated in the first second, before the bias was estimated.

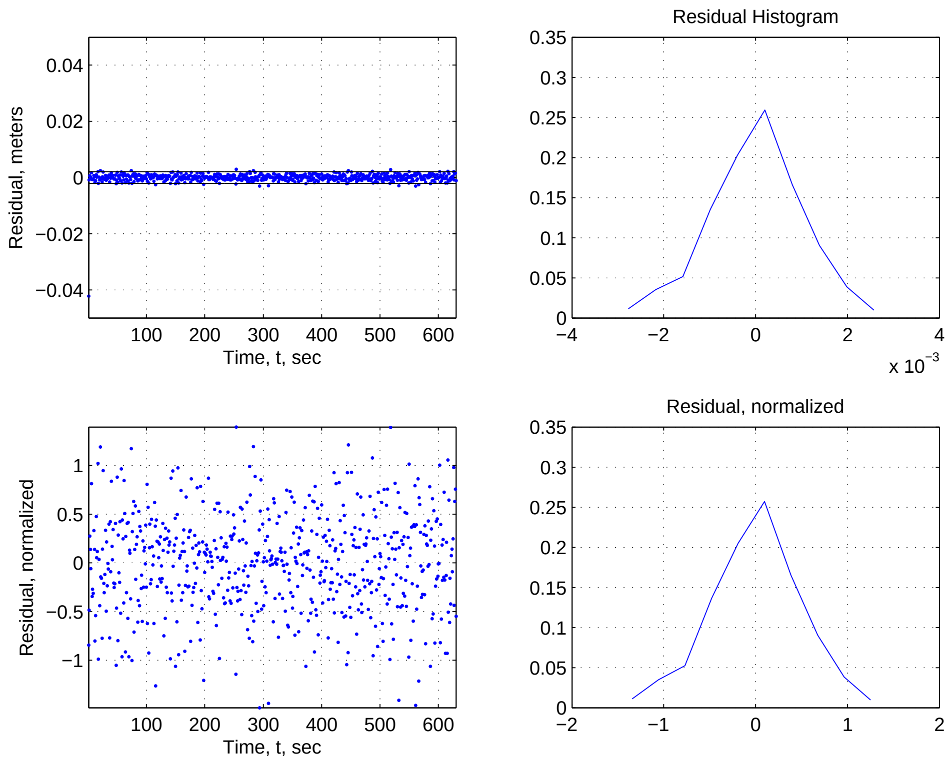


Fig. 1. Residual plots.

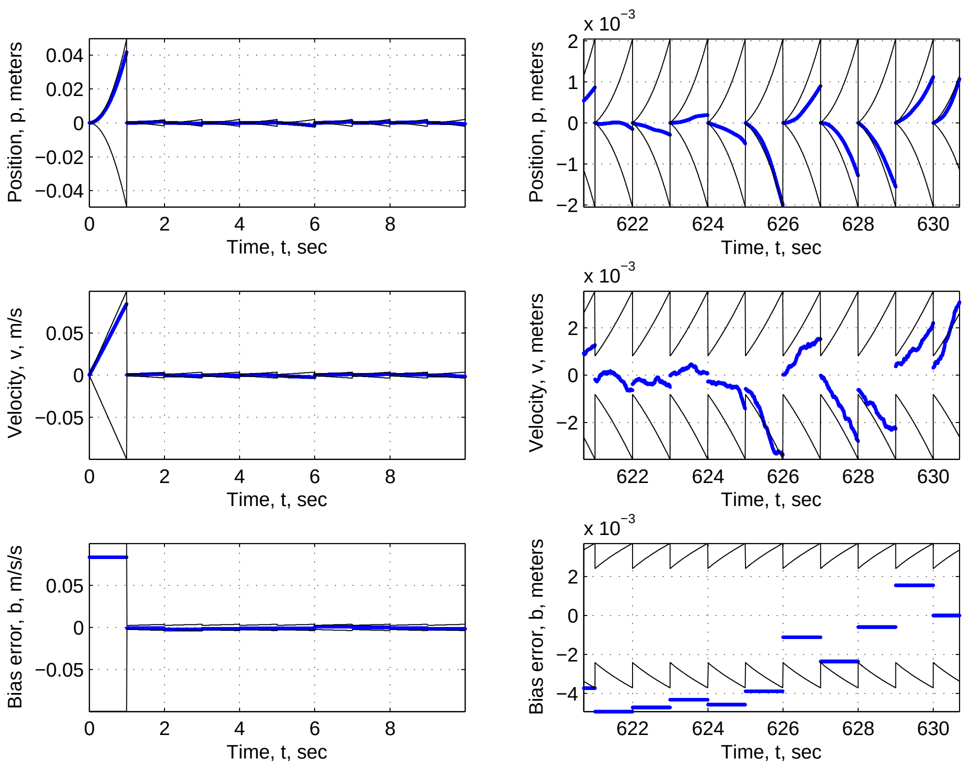


Fig. 2. Estimated state plots.

M-G350-PD11

EPSON
EXCEED YOUR VISION

IMU (Inertial Measurement Unit)

■ GENERAL DESCRIPTION

The M-G350-PD11 is a small form factor inertial measurement unit (IMU) with 6 degrees of freedom: triaxial angular rates and linear accelerations, and provides high-stability and high-precision measurement capabilities with the use of high-precision compensation technology. A variety of calibration parameters are stored in a memory of the IMU, and are automatically reflected in the measurement data being sent to the application after the power of the IMU is turned on. With a general-purpose SPI/UART supported for host communication, the M-G350-PD11 reduces technical barriers for users to introduce inertial measurement and minimizes design resources to implement inertial movement analysis and control applications.

The features of the IMU such as high stability, high precision, and small size make it easy to create and differentiate applications in various fields of industrial systems.

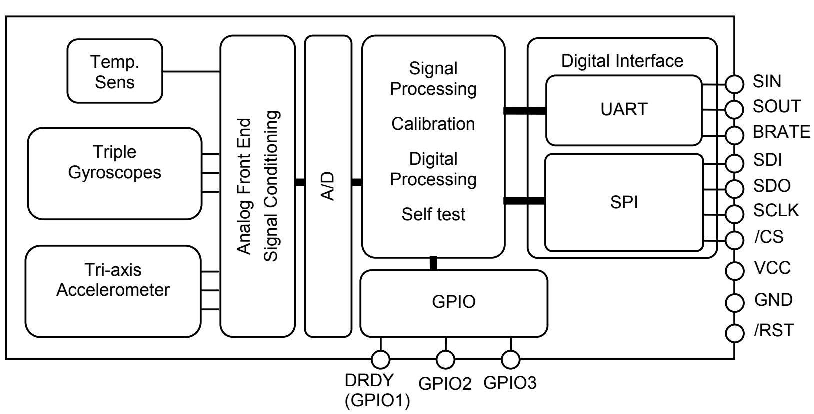
■ FEATURES

- Small Size, Lightweight : 24x24x10mm, 7grams
- Low-Noise, High-stability
  - Gyro Bias Instability : 6 deg/hr
  - Angular Random Walk : 0.2 deg/$\sqrt{\text{hr}}$
- Initial Bias Error : to 0.5 deg/s (1σ)
- 6 Degrees Of Freedom
  - Triple Gyroscopes : ±300 deg/s,
  - Tri-Axis Accelerometer : ±3 G
- 16bit data resolution
- Digital Serial Interface : SPI / UART
- Calibrated Stability (Bias, Scale Factor, Axial alignment)
- Data output rate : to 1k Sps
- Calibration temperature range : −20°C to +70°C
- Operating temperature range : −40°C to +85°C
- Single Voltage Supply : 3.3 V
- Low Power Consumption : 30mA (Typ.)

■ APPLICATIONS

- Motion analysis and control
- Unmanned systems
- Navigation systems
- Vibration control and stabilization
- Pointing and tracking systems

■ FUNCTIONAL BLOCK DIAGRAM



■ SENSOR SECTION SPECIFICATION

T_A=25°C, VCC=3.3V, angular rate=0 deg/s, ≤±1G, unless otherwise noted.

Parameter	Test Conditions / Comments	Min.	Typ.	Max.	Unit
GYRO SENSOR					
Sensitivity					
Dynamic Range	—	±300	—	—	deg/s
Sensitivity	—	Typ-0.5%	0.0125	Typ+0.5%	(deg/s)/LSB
Temperature Coefficient	1 σ, −20°C ≤ T_A ≤ +70°C	—	10	—	ppm/°C
Nonlinearity	Best fit straight line	—	0.1	—	% of FS
Misalignment	1 σ, Axis-to-axis, Δ = 90° ideal	—	±0.1	—	deg
Bias					
Initial Error	±1 σ	—	0.5	—	deg/s
Temperature Coefficient	1 σ, −20°C ≤ T_A ≤ +70°C		0.03		(deg/s)/°C
(Linear approximation)			0.001		
In-Run Bias Stability	1 σ	—	6	—	deg/hr
Angular Random Walk	1 σ	—	0.2	—	deg/ $\sqrt{hr}$
Linear Acceleration Effect			<0.01		(deg/s)/G
Noise					
Noise Density	1 σ, f = 10 to 20 Hz, no filtering	—	0.004	—	(deg/s)/ $\sqrt{Hz}$, rms
Frequency Property					
3 dB Bandwidth	—	—	133	—	Hz
ACCELEROMETERS					
Sensitivity					
Dynamic Range	—	±3	—	—	G
Sensitivity	—	Typ-0.5%	0.125	Typ+0.5%	mG/LSB
Temperature Coefficient	1σ, −20°C ≤ T_A ≤ +70°C	—	20	—	ppm/°C
Nonlinearity	≤ 1G , Best fit straight line	—	0.1	—	% of FS
Misalignment	1 σ, Axis-to-axis, Δ = 90° ideal	—	0.03	—	deg
Bias					
Initial Error	±1 σ	—	8	—	mG
Temperature Coefficient	1 σ, −20°C ≤ T_A ≤ +70°C		0.4		mG/°C
(Linear approximation)			0.02		
In-Run Bias Stability	1 σ	—	0.1	—	mG
Velocity Random Walk	1 σ	—	0.04	—	(m/sec)/ $\sqrt{hr}$
Noise					
Noise Density	1 σ, f = 10 to 20 Hz, no filtering	—	0.1	—	mG/ $\sqrt{Hz}$, rms
Frequency Property					
3 dB Bandwidth	—	—	148	—	Hz
TEMPERATURE SENSOR					
Scale Factor *1	Output = -15214(0xC492) @ +25°C	—	0.0042725	—	°C/LSB

*1) This is a reference value used for internal temperature compensation. We provide no guarantee that the value gives an absolute value of the internal temperature.

Note) The values in the specifications are based on the data calibrated at the factory. The values may change according to the way the product is used.

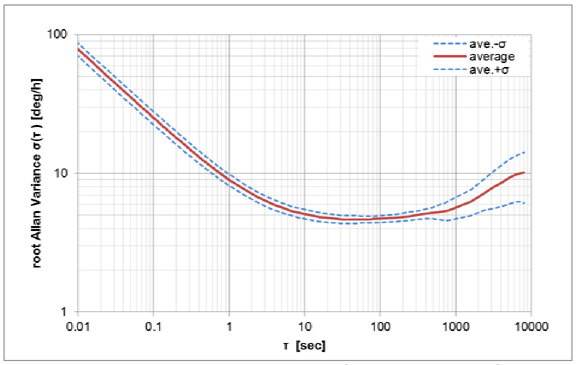
Note) The Typ values in the specifications are average values or 1σ values.

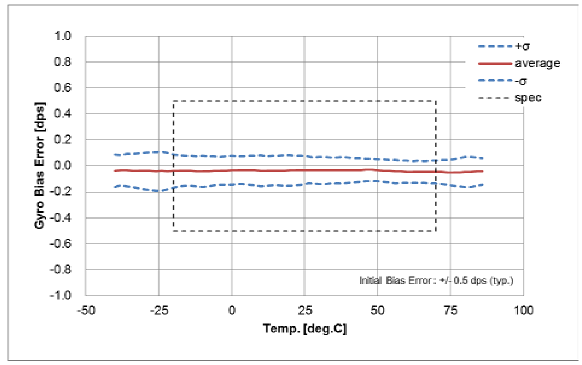
Note) Unless otherwise noted, the Max / Min values in the specifications are design values or Max / Min values at the factory tests.

■ RECOMMENDED OPERATING CONDITION

Parameter	Condition	min	Typ	Max	Unit
VCC to GND		3.15	3.3	3.45	V
Digital Input Voltage to GND		GND		VCC	V
Digital Output Voltage to GND		−0.3		VCC +0.3	V
Calibration temperature range	Performance parameters are applicable	−20		70	°C
Operating Temperature Range		−40		85	°C

■ TYPICAL PERFORMANCE CHARACTERISTICS

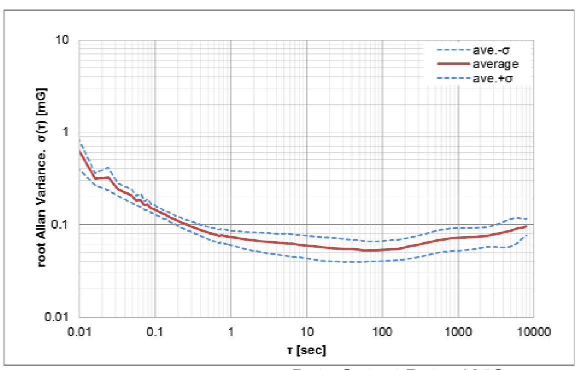


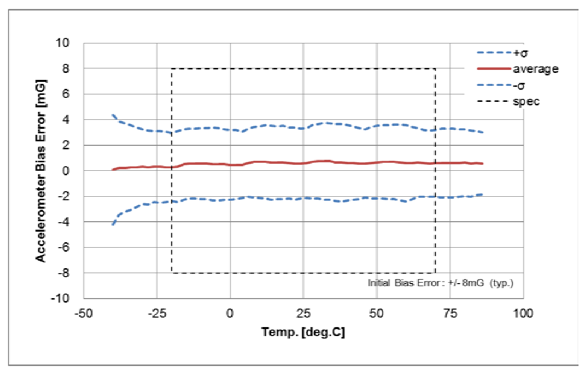


Data Output Rate: 125Sps
Average Filter TAP: N=16

Gyro Allan Variance Characteristic (N=9)

Gyro Bias vs. Temperature Characteristic (N=40)

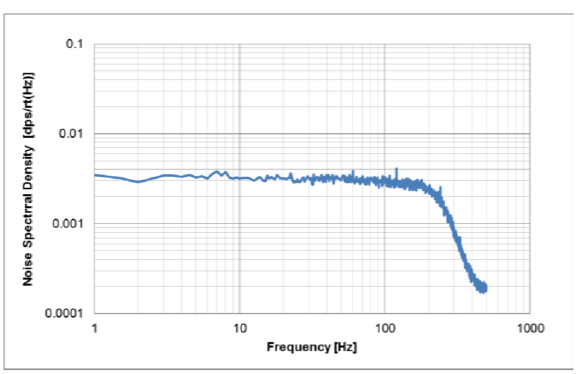


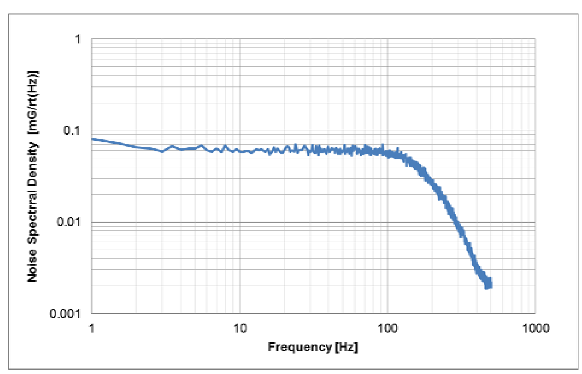


Data Output Rate: 125Sps
Average Filter TAP: N=16

Accelerometer Allan Variance Characteristic (N=9)

Accelerometer Bias vs. Temperature Characteristic (N=40)



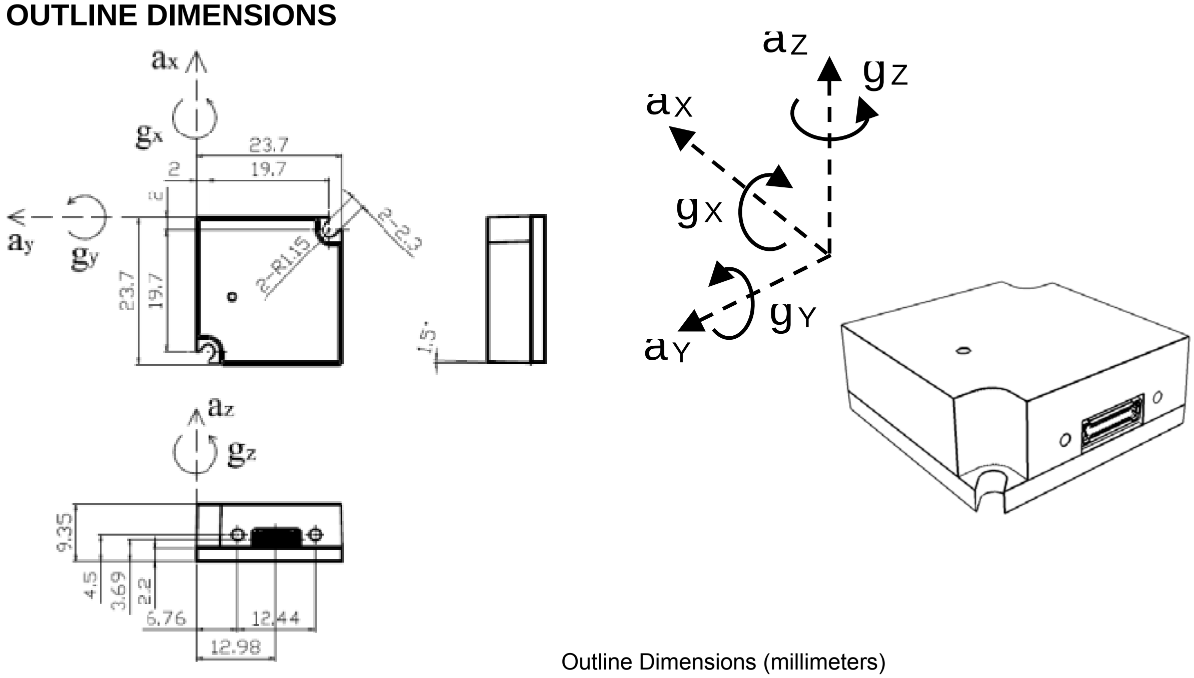


Gyro Noise Frequency Characteristic

Accelerometer Noise Frequency Characteristic

The product characteristics shown above are just examples and are not guaranteed as specifications.

■ OUTLINE DIMENSIONS



Outline Dimensions (millimeters)

NOTICE:

No part of this material may be reproduced or duplicated in any form or by any means without the written permission of Seiko Epson. Seiko Epson reserves the right to make changes to this material without notice. Seiko Epson does not assume any liability of any kind arising out of any inaccuracies contained in this material or due to its application or use in any product or circuit and, further, there is no representation that this material is applicable to products requiring high level reliability, such as, medical products. Moreover, no license to any intellectual property rights is granted by implication or otherwise, and there is no representation or warranty that anything made in accordance with this material will be free from any patent or copyright infringement of a third party. This material or portions thereof may contain technology or the subject relating to strategic products under the control of the Foreign Exchange and Foreign Trade Law of Japan and may require an export license from the Ministry of Economy, Trade and Industry or other approval from another government agency.

All brands or product names mentioned herein are trademarks and/or registered trademarks of their respective companies.

©Seiko Epson Corporation 2012, All rights reserved

SEIKO EPSON CORPORATION

M Project

421-8 Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-42-587-8291 FAX: +81-42-587-5117

Document code:412136501
First issue August, 2011 in Japan
Revised Oct, 2012 in Japan
Rev.20121023

References

[1] J. A. Farrell, *Aided Navigation: GPS with High Rate Sensors*. McGraw Hill, 2008.