

# IMU (Inertial Measurement Unit): M-G370PDG0

## Features

- Small Size, Lightweight : 24 x 24 x 10 mm, 10 grams
- Low-Noise, High-Stability
  - Gyro Bias Instability : 0.8 °/h
  - Angular Random Walk : 0.06 °/√h
- Initial Bias Error : 360 °/h (1σ) / 2 mG (1σ)
- 6 Degrees Of Freedom
  - Triple Gyroscopes : ±450 °/s
  - Tri-Axis Accelerometer : ±8 G/±16 G
- 16/32-bit Data Resolution
- Digital Serial Interface : SPI / UART
- Calibrated Stability  
(Bias, Scale Factor, Axial Alignment)
- Data Output Rate (Max.) : ~2k Sps
- External Trigger Input / External Counter Reset Input
- Delta Angle / Delta Velocity Output
- Calibration Temperature Range : -40 °C to +85 °C
- Operating Temperature Range : -40 °C to +85 °C
- Single Voltage Supply : 3.3 V
- Low Power Consumption : 16 mA (Typ.)



Product Name and Number  
M-G370PDG0 : X2G000221000100



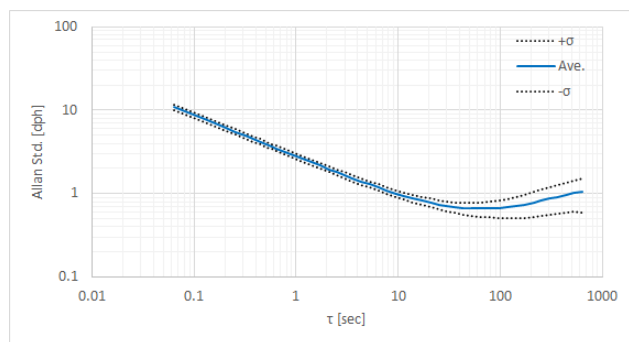
## Description

The M-G370PDG0 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 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 general-purpose SPI / UART support for host communications, the M-G370PDG0 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.

## Application

- Antenna Platform Stabilization
- Camera Gimbals
- Navigation Systems
- Vibration Control and Stabilization
- Pointing and Tracking Systems
- Autonomous Vehicle

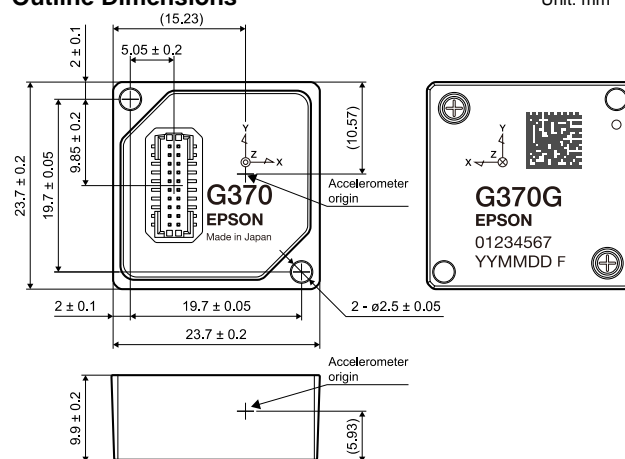
## Typical Performance Characteristic



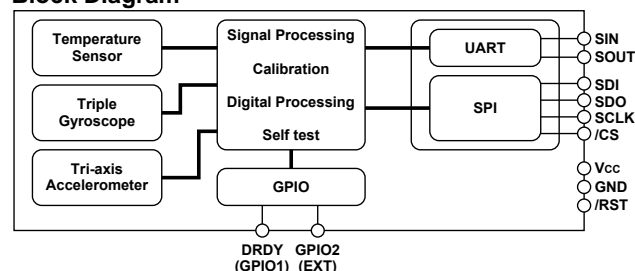
Gyro Allan Variance Characteristic

## Outline Dimensions

Unit: mm



## Block Diagram



## Table of Contents

Revision History .....	7
Ordering Information .....	8
Evaluation tools .....	8
Symbols .....	9
Notice of the Document .....	10
<b>1. Product Specifications .....</b>	<b>11</b>
1.1 Absolute Maximum Ratings .....	11
1.2 Recommended Operating Condition .....	11
1.3 Characteristics and Electrical Specifications .....	12
1.4 Timing Specifications .....	14
1.5 Connector Pin Layout and Functions .....	16
<b>2. Mechanical Dimensions .....</b>	<b>17</b>
2.1 Outline Dimensions .....	17
2.2 Connector Dimensions .....	18
<b>3. Typical Performance Characteristics .....</b>	<b>19</b>
<b>4. Basic Operation .....</b>	<b>20</b>
4.1 Connection to Host .....	20
4.2 Operation Mode .....	21
4.3 Functional Block Diagram .....	22
4.4 Data Output Timing .....	23
4.5 Data Ready Signal .....	24
4.6 Sampling Counter .....	24
4.7 GPIO .....	24
4.8 Self Test .....	24
4.9 External Trigger Input .....	25
4.10 External Counter Reset Input .....	27
4.11 Checksum .....	28
4.12 Automatic Start (For UART Auto Mode Only) .....	29
4.13 Filter .....	29
4.14 Delta Angle / Delta Velocity Output .....	32
4.15 Range Over Function .....	33
4.16 Frame Alignment Correction .....	33
<b>5. Digital Interface .....</b>	<b>35</b>
5.1 SPI Interface .....	36
5.1.1 SPI Read Timing (Normal Mode) .....	36
5.1.2 SPI Write Timing (Normal Mode) .....	37
5.1.3 SPI Read Timing (Burst Mode) .....	37
5.2 UART Interface .....	38
5.2.1 UART Read Timing (Normal Mode) .....	39
5.2.2 UART Read Timing (Burst Mode) .....	39
5.2.3 UART Write Timing .....	40
5.2.4 UART Auto Mode Operation .....	40
5.3 Data Packet Format .....	41
<b>6. User Registers .....</b>	<b>44</b>
6.1 BURST Register (Window 0) .....	47

6.2	MODE_CTRL Register (Window 0).....	47
6.3	DIAG_STAT Register (Window 0) .....	47
6.4	FLAG (ND / EA) Register (Window 0).....	49
6.5	GPIO Register (Window 0).....	49
6.6	COUNT Register (Window 0).....	50
6.7	RANGE_OVER Register (Window 0).....	50
6.8	TEMP Register (Window 0).....	50
6.9	GYRO Register (Window 0).....	51
6.10	ACCL Register (Window 0).....	51
6.11	ID Register (Window 0).....	51
6.12	DELTA_ANGLE Register (Window 0) .....	52
6.13	DELTA_VELOCITY Register (Window 0).....	52
6.14	SIG_CTRL Register (Window 1) .....	52
6.15	MSC_CTRL Register (Window 1) .....	53
6.16	SMPL_CTRL Register (Window 1).....	54
6.17	FILTER_CTRL Register (Window 1) .....	55
6.18	UART_CTRL Register (Window 1).....	56
6.19	GLOB_CMD Register (Window 1) .....	57
6.20	BURST_CTRL1 Register (Window 1) .....	57
6.21	BURST_CTRL2 Register (Window 1) .....	58
6.22	POL_CTRL Register (Window 1) .....	59
6.23	GLOB_CMD3 Register (Window 1) .....	59
6.24	ATTI_CTRL Register (Window 1).....	61
6.25	GLOB_CMD2 Register (Window 1) .....	61
6.26	R_MATRIX Register (Window 1).....	61
6.27	PROD_ID Register (Window 1).....	61
6.28	VERSION Register (Window 1) .....	62
6.29	SERIAL_NUM Register (Window 1).....	62
6.30	WIN_CTRL Register (Window 0,1) .....	62
<b>7.</b>	<b>Sample Program Sequence .....</b>	<b>63</b>
7.1	SPI Sequence .....	63
7.1.1	Power-on Sequence (SPI).....	63
7.1.2	Register Read and Write (SPI).....	63
7.1.3	Sampling Data (SPI) .....	63
7.1.4	Self Test (SPI) .....	66
7.1.5	Software Reset (SPI) .....	67
7.1.6	Flash Test (SPI).....	67
7.1.7	Flash Backup (SPI) .....	67
7.1.8	Filter Setting (SPI).....	68
7.1.9	Frame Alignment Correction (SPI) .....	68
7.2	UART Sequence .....	69
7.2.1	Power-on Sequence (UART).....	69
7.2.2	Register Read and Write (UART).....	69
7.2.3	Sampling Data (UART).....	69
7.2.4	Self Test (UART) .....	72
7.2.5	Software Reset (UART).....	72
7.2.6	Flash Test (UART).....	72
7.2.7	Flash Backup (UART) .....	73
7.2.8	Initial Backup (UART).....	73
7.2.9	Filter Setting (UART).....	74
7.2.10	Auto Start (UART only).....	74

**8. Handling Notes..... 75**

8.1 Cautions for Use ..... 75

8.2 Cautions for Storage ..... 75

8.3 Other Cautions ..... 76

8.4 Limited Warranty ..... 76

**9. Contact..... 77**

## List of Figures

Figure 1.1	SPI Write Timing and Sequence .....	14
Figure 1.2	SPI Read Timing and Sequence .....	14
Figure 1.3	SPI Read Timing and Sequence (BURST MODE) .....	15
Figure 1.4	Connector Pin Assignment .....	16
Figure 2.1	M-G370PDG0 Outline Dimensions .....	17
Figure 2.2	Header Pin Dimensions .....	18
Figure 3.1	Gyro Allan Variance Characteristic .....	19
Figure 3.2	Gyro Bias vs. Temperature Characteristic .....	19
Figure 3.3	Gyro Noise Frequency Characteristic .....	19
Figure 3.4	Accelerometer Allan Variance Characteristic .....	19
Figure 3.5	Accelerometer Bias vs. Temperature Characteristic .....	19
Figure 3.6	Accelerometer Noise Frequency Characteristic .....	19
Figure 4.1	SPI Connection .....	20
Figure 4.2	UART Connection .....	20
Figure 4.3	Operational State Diagram .....	22
Figure 4.4	Functional Block Diagram .....	22
Figure 4.5	Data Output Timing – Data Output Rate 1 kSps, FIR Filter TAP N = 2 .....	23
Figure 4.6	Data Output Timing – Data Output Rate 500 Sps, FIR Filter TAP N = 4 .....	23
Figure 4.7	Data Ready Signal Timing .....	24
Figure 4.8	External Trigger Input (UART / SPI Manual Mode) .....	26
Figure 4.9	External Trigger Input Functional Block Diagram .....	26
Figure 4.10	External Counter Reset Input .....	28
Figure 4.11	Checksum .....	28
Figure 4.12	Moving Average Filter Characteristics .....	29
Figure 4.13	FIR Kaiser Filter Typical Characteristic 1 .....	30
Figure 4.14	FIR Kaiser Filter Typical Characteristic 2 .....	30
Figure 4.15	Delta Angle / Delta Velocity .....	32
Figure 5.1	SPI Read Timing (Normal Mode) .....	36
Figure 5.2	SPI Write Timing (Normal Mode) .....	37
Figure 5.3	SPI Timing (Burst Mode) .....	37
Figure 5.4	UART Bit Format .....	38
Figure 5.5	UART Read Timing (Normal Mode) .....	39
Figure 5.6	UART Read Timing (Burst Mode) .....	40
Figure 5.7	UART Write Timing .....	40
Figure 5.8	UART Auto Mode Operation .....	40
Figure 6.1	SRAM and Flash Memory Diagram .....	57

## List of Tables

Table 1.1	Absolute Maximum Ratings	11
Table 1.2	Recommended Operating Conditions	11
Table 1.3	Sensor Specifications	12
Table 1.4	Interface Specifications	13
Table 1.5	Timing Specifications	14
Table 1.6	Pin Function Descriptions	16
Table 2.1	Header Part Number	18
Table 2.2	Socket Part Number	18
Table 4.1	External Trigger Input Timing Requirements	25
Table 4.2	External Counter Reset Input Timing	27
Table 4.3	Transient Response in Number of Samples Based on Output Data Rate vs Filter Tap	31
Table 5.1	SPI Communication Settings	36
Table 5.2	SPI Timing (Normal Mode)	36
Table 5.3	Command Format (Read)	36
Table 5.4	Response Format (Read)	37
Table 5.5	Command Format (Write)	37
Table 5.6	SPI Timing (Burst Mode)	37
Table 5.7	UART Communication Settings	38
Table 5.8	UART Timing	38
Table 5.9	UART Timing ( $t_{\text{READRATE}}$ Requirements for Burst Mode)	39
Table 5.10	Command Format (Read)	39
Table 5.11	Response Format (Read)	39
Table 5.12	Command Format (Burst Mode)	40
Table 5.13	Command Format (Write)	40
Table 5.14	UART Data Packet Format (UART Burst / Auto Mode) Example: 16-bit Output	41
Table 5.15	UART Data Packet Format (UART Burst / Auto Mode) Example: 32-bit Output	42
Table 5.16	Data Packet Format (SPI Burst Mode) Example: 16-bit Output	43
Table 5.17	Data Packet Format (SPI Burst Mode) Example: 32-bit Output	43
Table 6.1	Register Map	45
Table 6.2	SET_ERR Flag Output Rate and Filter Setting Table Evaluation Table	54
Table 6.3	Output Rate Limitations	55
Table 6.4	Delta Angle: Scale Factor & Range	60
Table 6.5	Delta Velocity: Scale Factor & Range	60

**Revision History**

Rev. No.	Date	Page	Description
Rev1.0	2023/03/17	ALL	New release.
Rev1.1	2023/06/30	P45, 46 P57 P73	Table 6.1 Register Map modify 6.21 GLOB_CMD Register (Window 1) modify 7.2.8 Initial Backup (UART) modify

## Ordering Information

The product can be ordered with the following numbers. Please inquire separately about details.

Product Model Number	Product Name	Comments
X2G000221000100	M-G370PDG0	This product.

## Evaluation tools

Evaluation tools can be provided for this product. Please inquire separately about details.

Product Model Number	Product Name	Comments
X2H000021000100	M-G32EV031	Breakout Board for IMU.
X2H000021000200	M-G32EV041	USB Evaluation Board for IMU. An option to connect M-G330/366/370PDG0 to the USB port of a PC.



## Symbols



- **Compliant with the EU RoHS directive**

\* About products without the Pb-Free label

Product terminals are lead-free but the internal components of the product contain lead (high melting point solder lead as well as the lead contained in the glass of an electronic component are both not applicable under the EU RoHS directive).

## Notice of the Document

### Evaluation board/kit and Development tool important notice

1. This evaluation board/kit or development tool is designed for use for engineering evaluation, demonstration, or development purposes only. Do not use it for other purposes. It is not intended to meet the requirements of design for finished products.
2. This evaluation board/kit or development tool is intended for use by an electronic engineer and is not a consumer product. The user should use it properly and in a safe manner. Seiko Epson does not assume any responsibility or liability of any kind of damage and/or fire caused by the use of it. The user should cease to use it when any abnormal issue occurs even during proper and safe use.
3. The part used for this evaluation board/kit or development tool may be changed without any notice.

### NOTICE: PLEASE READ CAREFULLY BELOW BEFORE USE THIS DOCUMENT

The content of this document is subject to change without notice.

1. This document may not be copied, reproduced, or used for any other purposes, in whole or in part, without the consent of Seiko Epson Corporation("Epson").
2. Before purchasing or using Epson products, please contact with our sales representative for the latest information and be always sure to check the latest information published on Epson's official web sites and sources.
3. Information provided in this document such as application circuits, programs, usage, etc., are for reference purpose only. Please use the application circuits, programs, usage, etc. in the design of your equipment or systems at your own responsibility. Epson makes no guarantees against any infringements or damages to any third parties' intellectual property rights or any other rights resulting from the information. This document does not grant you any licenses, intellectual property rights or any other rights with respect to Epson products owned by Epson or any third parties.
4. Epson is committed to constantly improving quality and reliability, but semiconductor products in general are subject to malfunction and failure. In using Epson products, you shall be responsible for safe design in your products; your hardware, software and systems are designed enough to prevent any harm or damages to life, health or property even if any malfunction or failure might be caused by Epson products. In designing of your products with using Epson products, please be sure to check and comply with the latest information regarding Epson products (this document, specifications, data sheets, manuals, Epson's web site, etc.). When using the information included in the above materials such as product data, chart, technical contents, programs, algorithms and application circuit examples, you shall evaluate your products both in stand-alone basis and within your overall systems. You shall be solely responsible for deciding whether or not to adopt and use Epson products.
5. Epson has prepared this document and programs provided in this document carefully to be accurate and dependable, but Epson does not guarantee that the information and the programs are always accurate and complete. Epson assumes no responsibility for any damages which you incurred by due to misinformation in this document and the programs.
6. No dismantling, analysis, reverse engineering, modification, alteration, adaptation, reproduction, etc., of Epson products is allowed.
7. Epson products have been designed, developed and manufactured to be used in general electronic applications (office equipment, communications equipment, measuring instruments, home electronics, etc.) and applications individually listed in this document ("General Purpose"). Epson products are NOT intended for any use beyond the General Purpose that requires particular/higher quality or reliability in order to refrain from causing any malfunction or failure leading to harm to life, health or serious property damage or severe impact on society, including, but not limited to listed below. Therefore, you are advised to use Epson products only for the General Purpose. Should you desire to buy and use Epson products for the particular purpose other than the General Purpose, Epson makes no warranty and disclaims with respect to Epson products, whether express or implied, including without limitation any implied warranty of merchantability or fitness for any particular purpose.  
[Particular purpose]  
Space equipment (artificial satellites, rockets, etc.)  
Transportation vehicles and their control equipment (automobiles, aircraft, trains, ships, etc.)  
Medical equipment (other than applications individually listed in this document) / Relay equipment to be placed on sea floor  
Power station control equipment / Disaster or crime prevention equipment / Traffic control equipment / Financial equipment  
Other applications requiring similar levels of reliability as the above
8. Epson products listed in this document and our associated technologies shall not be used in any equipment or systems that laws and regulations in Japan or any other countries prohibit to manufacture, use or sell. Furthermore, Epson products and our associated technologies shall not be used for developing military weapons of mass destruction, military purpose use, or any other military applications. If exporting Epson products or our associated technologies, you shall comply with the Foreign Exchange and Foreign Trade Control Act in Japan, Export Administration Regulations in the U.S.A (EAR) and other export-related laws and regulations in Japan and any other countries and follow the required procedures as provided by the relevant laws and regulations.
9. Epson assumes no responsibility for any damages (whether direct or indirect) caused by or in relation with your non-compliance with the terms and conditions in this document.
10. Epson assumes no responsibility for any damages (whether direct or indirect) incurred by any third party that you assign, transfer, loan, etc., Epson products.
11. For more details or other concerns about this document, please contact our sales representative.
12. Company names and product names listed in this document are trademarks or registered trademarks of their respective companies.

2022.08

©Seiko Epson Corporation 2023. All rights reserved.

# 1. Product Specifications

## 1.1 Absolute Maximum Ratings

Table 1.1 Absolute Maximum Ratings

Parameter	Min.	Typ.	Max.	Unit
V <sub>CC</sub> to GND	-0.3	—	4.8	V
Digital Input Voltage to GND	-0.3	—	V <sub>CC</sub> + 0.3	V
Digital Output Voltage to GND	-0.3	—	V <sub>CC</sub> + 0.3	V
Storage Temperature Range	-40	—	85	°C
Acceleration / Shock (Half-sine, 0.5 ms)	—	—	1000	G

### Precautions about ESD

Electrostatic discharge (ESD) may damage the product.

When you store or handle the product, take appropriate preventive measures against electrostatic discharge (ESD). Damages caused by electrostatic discharge (ESD) range from small performance degradation, partial malfunction, to complete breakdown.

This is a high-precision product. Even small performance degradation may cause the product not to conform to the specifications.

## 1.2 Recommended Operating Condition

Table 1.2 Recommended Operating Conditions

Parameter	Condition	Min.	Typ.	Max.	Unit
V <sub>CC</sub> to GND		3.15	3.3	3.45	V
Digital Input Voltage to GND		GND	—	V <sub>CC</sub>	V
Digital Output Voltage to GND		-0.3	—	V <sub>CC</sub> + 0.3	V
Calibration Temperature Range	Performance parameters are applicable	-40	—	85	°C
Operating Temperature Range		-40	—	85	°C

### 1.3 Characteristics and Electrical Specifications

Table 1.3 Sensor Specifications

$T_A = 25\text{ }^{\circ}\text{C}$ ,  $V_{CC} = 3.3\text{ V}$ , angular rate =  $0\text{ }^{\circ}/\text{s}$ ,  $\leq \pm 1\text{ G}$ , unless otherwise noted

Parameter	Test Conditions / Comments	Min.	Typ.	Max.	Unit
<b>GYRO SENSOR</b>					
Sensitivity					
Output Range		—	$\pm 450$	—	$^{\circ}/\text{s}$
Scale Factor	16 bits	-0.2%	66	+0.2%	LSB/( $^{\circ}/\text{s}$ )
	32 bits	-0.2%	$66 \times (2^{16})$	+0.2%	
Nonlinearity (Best fit straight line)	$1\sigma$	—	0.05	—	% of FS
Misalignment	$1\sigma$ , Axis-to-axis, $\Delta = 90^{\circ}$ ideal	—	0.01	—	$^{\circ}$
Bias					
Initial Error	$1\sigma$ , $-40\text{ }^{\circ}\text{C} \leq T_A \leq +85\text{ }^{\circ}\text{C}$	—	360	—	$^{\circ}/\text{h}$
Repeatability	$1\sigma$ , Turn-on to turn-on <sup>*3</sup>	—	36	—	$^{\circ}/\text{h}$
Bias Instability	Average	—	0.8	—	$^{\circ}/\text{h}$
Angular Random Walk	Average	—	0.06	—	$^{\circ}/\sqrt{\text{h}}$
Linear Acceleration Effect	Average	—	18	—	( $^{\circ}/\text{h}$ )/G
Noise Density	$f = 10\text{ Hz}$ to $20\text{ Hz}$	—	4.7	—	( $^{\circ}/\text{h}$ )/ $\sqrt{\text{Hz}}$ , rms
Frequency Property					
3 dB Bandwidth		—	189	—	Hz
<b>ACCELEROMETERS</b>					
Sensitivity					
Output Range		—	$\pm 8/\pm 16$ <sup>*4</sup>	—	G
Scale Factor	16 bits, Output Range $\pm 8$	-0.1%	4	+0.1%	LSB/mG
	32 bits, Output Range $\pm 8$	-0.1%	$4 \times (2^{16})$	+0.1%	
	16 bits, Output Range $\pm 16$	-0.1%	2	+0.1%	
	32 bits, Output Range $\pm 16$	-0.1%	$2 \times (2^{16})$	+0.1%	
Nonlinearity (Best fit straight line)	$1\sigma$ , $< 1\text{ G}$	—	0.1	—	% of FS
Misalignment	$1\sigma$ , Axis-to-axis, $\Delta = 90^{\circ}$ ideal	—	0.01	—	$^{\circ}$
Bias					
Initial Error	$1\sigma$ , $-40\text{ }^{\circ}\text{C} \leq T_A \leq +85\text{ }^{\circ}\text{C}$	—	2	—	mG
Repeatability	$1\sigma$ , Turn-on to turn-on <sup>*3</sup>	—	2	—	mG
Bias Instability	Average	—	24	—	$\mu\text{G}$
Velocity Random Walk	Average	—	0.02	—	(m/s)/ $\sqrt{\text{h}}$
Noise Density	$f = 10\text{ Hz}$ to $20\text{ Hz}$	—	50	—	$\mu\text{G}/\sqrt{\text{Hz}}$ , rms
Frequency Property					
3 dB Bandwidth		—	333	—	Hz
<b>TEMPERATURE SENSOR</b>					
Scale Factor <sup>*1*2</sup>	Output = 0 @ $+25\text{ }^{\circ}\text{C}$	—	0.00390625	—	$^{\circ}\text{C}/\text{LSB}$

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

\*2) This is the temperature scale factor for the upper 16 bits (TEMP\_HIGH).

\*3) Turn-on to turn-on / Day by day, estimated variation during 5 consecutive days.

\*4) Selectable by register setting.

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.

• The Typ. values in the specifications are average values or  $1\sigma$  values.

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

• Acceleration characteristics do not depend on the output range.

Table 1.4 Interface Specifications

 $T_A = 25\text{ }^{\circ}\text{C}$ ,  $V_{CC} = 3.3\text{ V}$ , unless otherwise noted

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
<b>LOGIC INPUTS<sup>*1</sup></b>					
Positive Trigger Voltage	LVC MOS Schmitt	1.2	—	2.52	V
Negative Trigger Voltage	LVC MOS Schmitt	0.75	—	1.98	V
Hysteresis Voltage	LVC MOS Schmitt	0.3	—	—	V
Logic 1 Input Current, $I_{INH}$	$V_{IH} = 3.3\text{ V}$	—	0.1	—	$\mu\text{A}$
Logic 0 Input Current, $I_{INL}$	$V_{IL} = 0\text{ V}$	—	0.1	—	$\mu\text{A}$
Input Capacitance, $C_{IN}$		—	8	—	pF
RST Voltage Range		0	—	$V_{CC} + 0.3$	V
RST High-level Input Voltage, $V_{IH}$		$0.8 \times V_{CC}$	—	—	V
RST Low-level Input Voltage, $V_{IL}$		—	—	$0.2 \times V_{CC}$	V
RST Low Pulse Width		100	—	—	ms
Pull-up Resistor		32	80	224	k $\Omega$
<b>DIGITAL OUTPUTS<sup>*1</sup></b>					
Output High Voltage, $V_{OH}$	$I_{SOURCE} = 1.4\text{ mA}$ LVC MOS	$V_{CC} - 0.4$	—	—	V
Output Low Voltage, $V_{OL}$	$I_{SINK} = 1.4\text{ mA}$ LVC MOS	—	—	0.4	V
<b>FUNCTIONAL TIMES<sup>*2</sup></b>					
Time until data is available					
Power-On Start-Up Time		—	—	800	ms
Reset Recovery Time		—	—	800	ms
Flash Test Time		—	—	30	ms
Flash Backup Time		—	—	200	ms
Self Test Time		—	—	80	ms
Filter Setting Time		—	—	1	ms
Data Output Rate	$DOUT\_RATE = 0x00$	—	—	2000	Sps
Clock Accuracy		—	—	$\pm 0.001$	%
Power Supply	Operating voltage range, $V_{CC}$	3.15	3.3	3.45	V
Power Supply Current		—	16	—	mA

\*1) Digital I/O signal pins operate at 3.3 V inside the unit. All digital I/O signal pins (except RST) can tolerate 5 V input.

\*2) These specifications do not include the effect of temperature fluctuation and response time of the internal filter.

Note) The specifications above are not included in the factory test items but their characteristic is confirmed.

## 1.4 Timing Specifications

Table 1.5 Timing Specifications

$T_A = 25\text{ }^{\circ}\text{C}$ ,  $V_{CC} = 3.3\text{ V}$ , unless otherwise noted

Parameter	Description	Min.	Typ.	Max.	Unit
<b>NORMAL MODE</b>					
$f_{SCLK}$		0.01	—	2.0	MHz
$t_{STALL}$	Stall period between data	20	—	—	$\mu\text{s}$
$t_{WRITERATE}$	Write rate	40	—	—	$\mu\text{s}$
$t_{READRATE}$	Read rate	40	—	—	$\mu\text{s}$
<b>BURST MODE</b>					
$f_{SCLK}$		0.01	—	1.0	MHz
$t_{STALL1}$	Stall period between data	45	—	—	$\mu\text{s}$
$t_{STALL2}$	Stall period between data	4	—	—	$\mu\text{s}$
$t_{READRATE2}$	Read rate	32	—	—	$\mu\text{s}$
<b>COMMON</b>					
$t_{CS}$	Chip select to clock edge	10	—	—	ns
$t_{DAV}$	SO valid after SCLK edge	—	—	80	ns
$t_{DSU}$	SI setup time before SCLK rising edge	10	—	—	ns
$t_{DHD}$	SI hold time after SCLK rising edge	10	—	—	ns
$t_{SCLKR}$ , $t_{SCLKF}$	SCLK rise/fall times	—	—	20	ns
$t_{DR}$ , $t_{DF}$	SO rise/fall times	—	—	20	ns
$t_{SFS}$	High after SCLK edge CS	80	—	—	ns

Note) The specifications above are not included in the factory test items but their characteristic is confirmed.

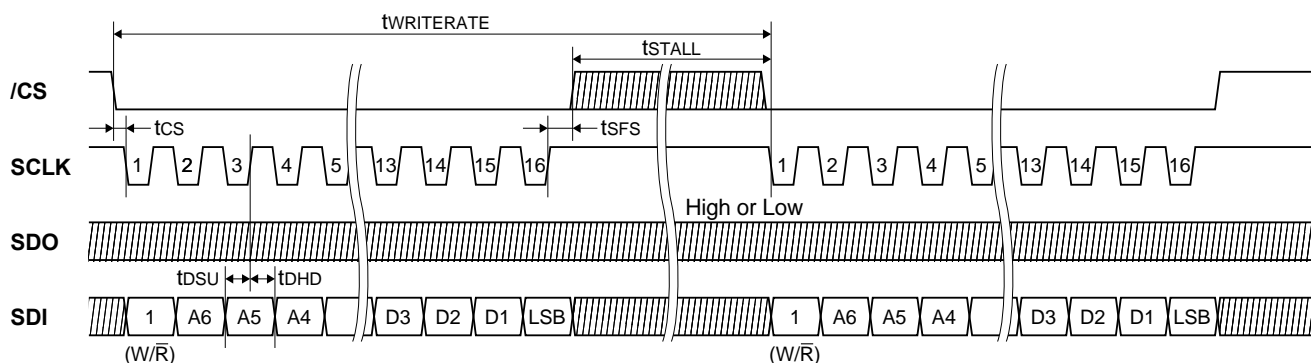


Figure 1.1 SPI Write Timing and Sequence

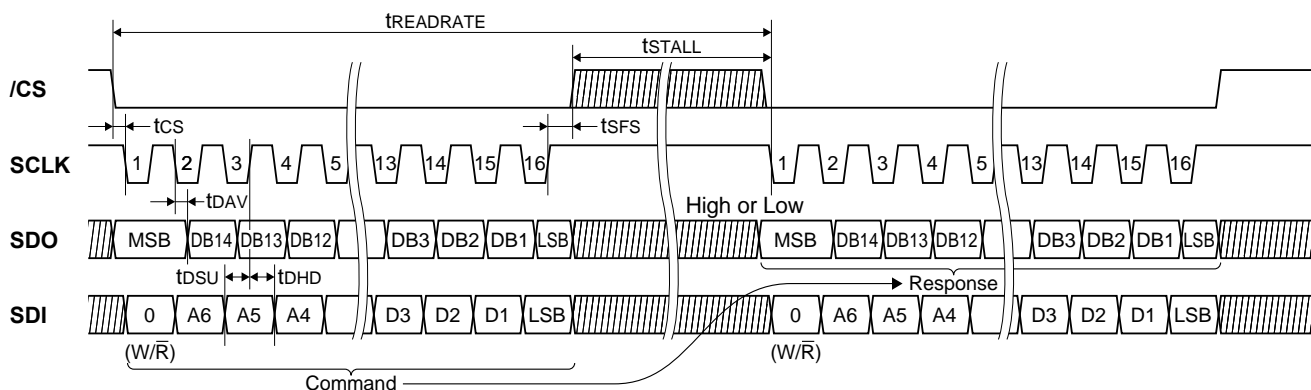
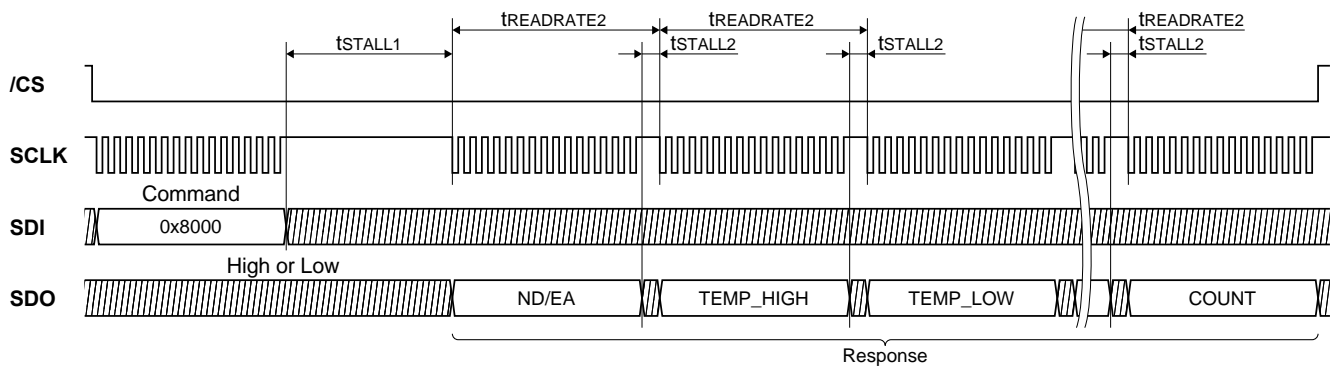


Figure 1.2 SPI Read Timing and Sequence

**Figure 1.3 SPI Read Timing and Sequence (BURST MODE)**

## 1.5 Connector Pin Layout and Functions

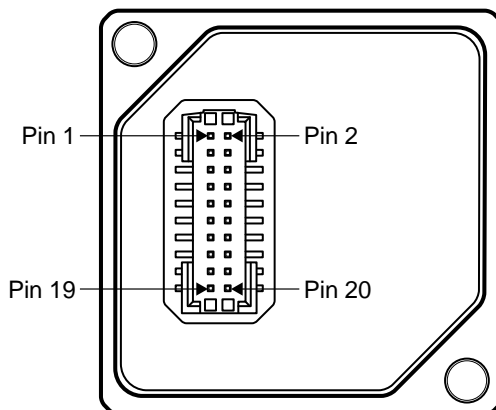


Figure 1.4 Connector Pin Assignment

Table 1.6 Pin Function Descriptions

Pin No.	Mnemonic	Type <sup>*1</sup>	Description
1	SCLK	I	SPI Serial Clock <sup>*2</sup>
2	SDO	O	SPI Data Output <sup>*2</sup>
5	SDI	I	SPI Data Input <sup>*2</sup>
6	/CS	I	SPI Chip Select <sup>*2</sup>
7	SOUT	O	UART Data Output <sup>*2</sup>
9	SIN	I	UART Data Input <sup>*2</sup>
13	DRDY (GPIO1)	I/O	Data Ready <sup>*3</sup> (General Purpose I/O1)
14	GPIO2 (EXT)	I/O	General Purpose I/O2 <sup>*4</sup> (External Trigger Input or External Counter Reset Input)
16	/RST	I	Reset <sup>*5</sup>
10, 11, 12	V <sub>CC</sub>	S	Power Supply 3.3 V
3, 4, 8, 15	GND	S	Ground
17, 18, 19, 20	NC	N/A	Do Not Connect

\*1) Pin Type I: Input, O: Output, I/O: Input/Output, S: Supply, N/A: Not Applicable

\*2) Connect either SPI or UART but not both. Connecting both SPI and UART at the same time may result in malfunction of the device. Regarding unused pins, please connect unused input pins to V<sub>CC</sub> through resistor.

\*3) Regarding Pin function selection, please refer to the **DRDY\_ON** at register MSC\_CTRL[0x02 (W1)] bit[2]

\*4) Regarding Pin function selection, please refer to the **EXT\_SEL** at register MSC\_CTRL[0x02 (W1)] bit[7:6]

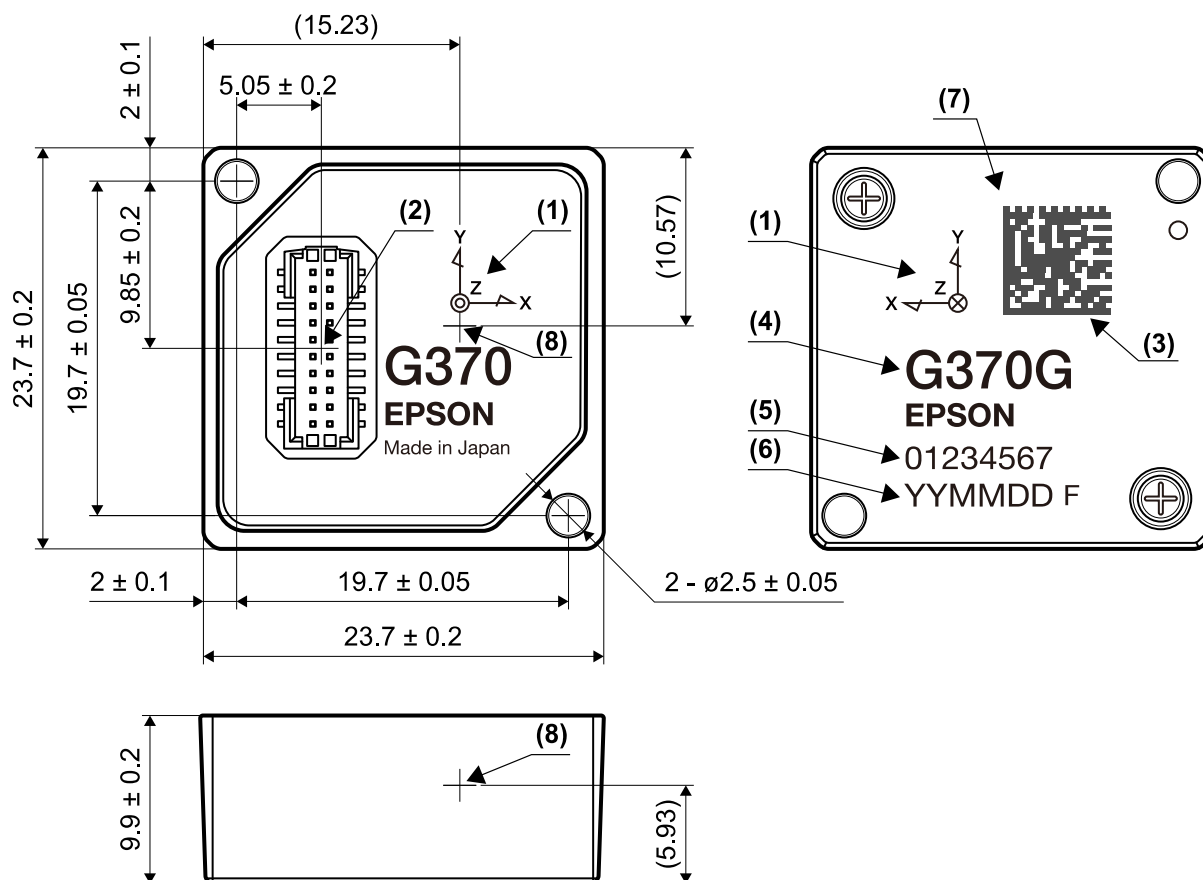
\*5) If the /RST pin is not used, keep the pin at High (V<sub>CC</sub>) voltage level.

Note) All input pins have weak pull up resistors inside the IMU.



## 2. Mechanical Dimensions

### 2.1 Outline Dimensions



\* (data) reference dimensions

(Unit: mm)

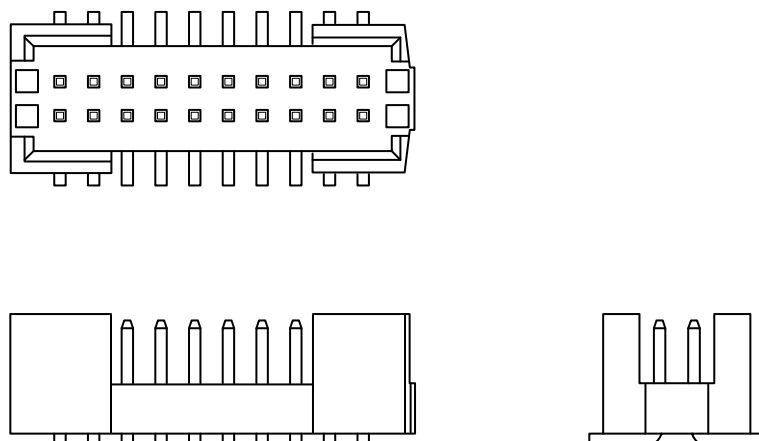
No.	Description
(1)	Definition of sensor axes
(2)	Connector position
(3)	Matrix code (Data Matrix) Including product name, serial number, date, and factory code
(4)	Product name
(5)	Serial number
(6)	Date and factory code
(7)	Frame ground
(8)	Accelerometer origin

**Figure 2.1 M-G370PDG0 Outline Dimensions**

The right-hand screw rotation direction of the gyro sensor is positive (+).

## 2.2 Connector Dimensions

Figure 2.2 and Table 2.1 describes the connector manufacturer and the model number of the header built into the IMU.



**Figure 2.2 Header Pin Dimensions**

**Table 2.1 Header Part Number**

Maker	Parts Number	RoHS Compliant
Samtec	FTMH-110-02-L-DV-ES	Yes

\* END SHROUDS is MOLDED TO POSITION END SHROUDS

Table 2.2 shows the connector manufacturer and the model number of the recommended socket used at the host side.

**Table 2.2 Socket Part Number**

Maker	Parts Number	RoHS Compliant
Samtec	CLM-110-02-H-D	Yes
Samtec	CLM-110-02-L-D	Yes

### 3. Typical Performance Characteristics

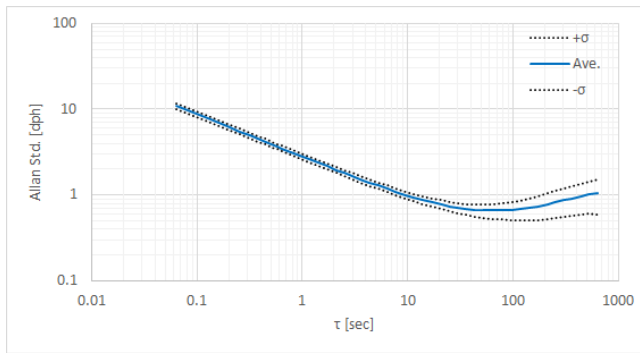


Figure 3.1 Gyro Allan Variance Characteristic

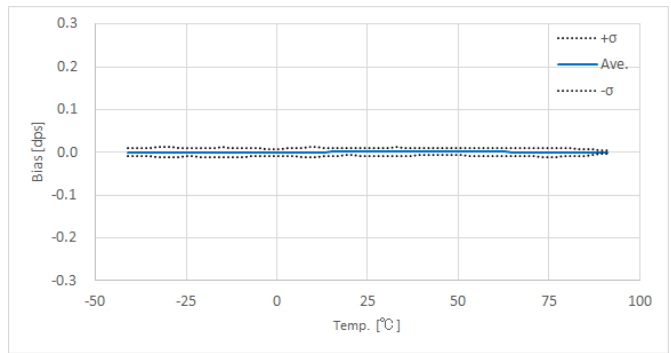


Figure 3.2 Gyro Bias vs. Temperature Characteristic

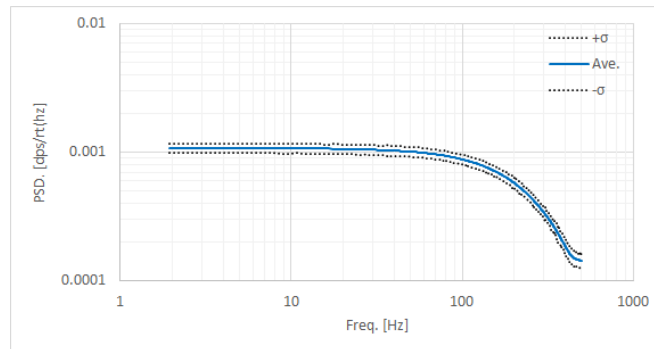


Figure 3.3 Gyro Noise Frequency Characteristic

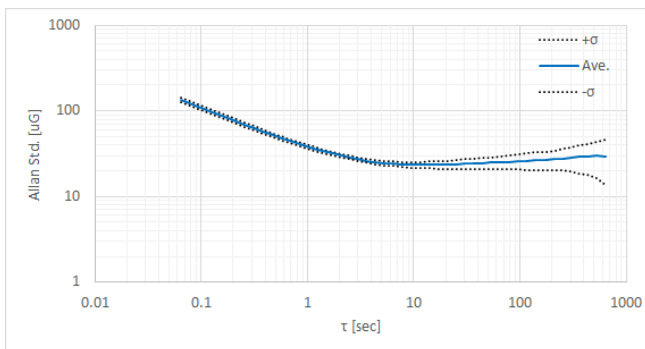


Figure 3.4 Accelerometer Allan Variance Characteristic

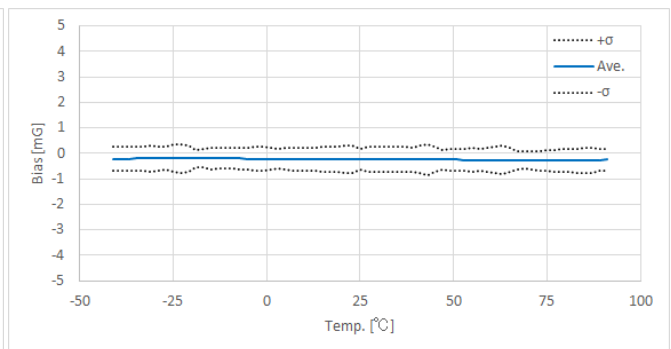


Figure 3.5 Accelerometer Bias vs. Temperature Characteristic

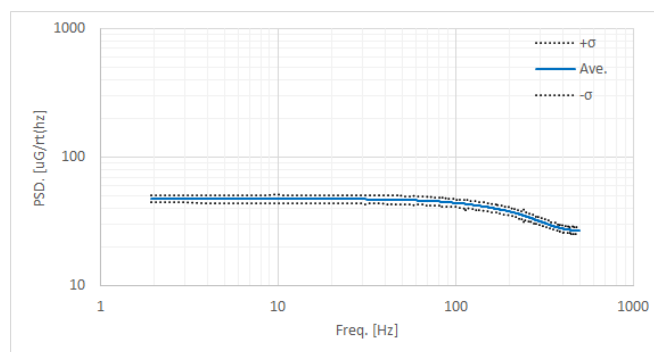


Figure 3.6 Accelerometer Noise Frequency Characteristic

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

## 4. Basic Operation

### 4.1 Connection to Host

The device is connected to the host via SPI or UART. The following is an example of the connection.

NOTE: • Connect either SPI or UART but not both. Connecting both SPI and UART at the same time may result in malfunction of the device.

- Refer to Table 1.6 Pin Function Descriptions for the connection of unused pins.

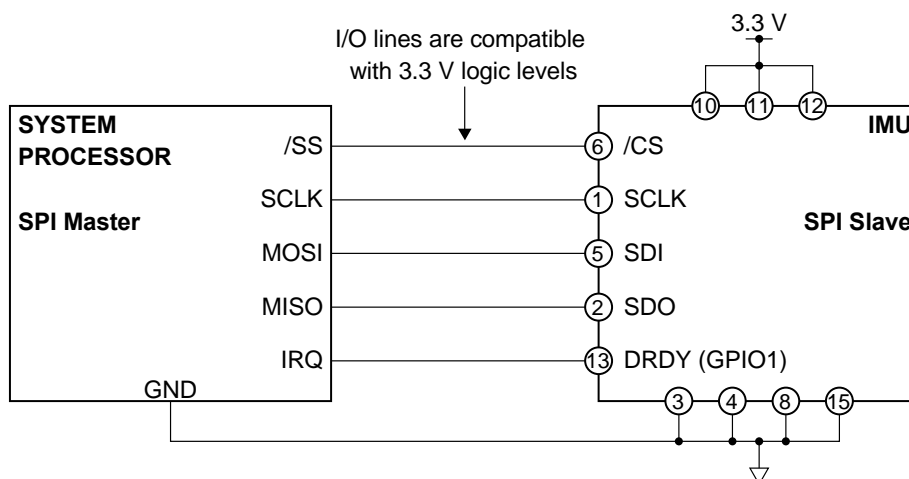


Figure 4.1 SPI Connection

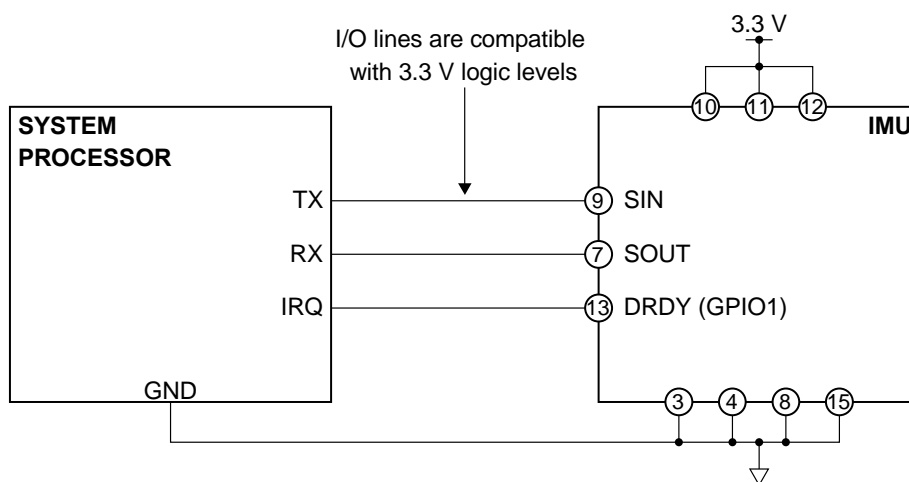


Figure 4.2 UART Connection

## 4.2 Operation Mode

The device has the following two operation modes. Only when the UART interface is used, Sampling mode has two submodes: Manual mode and Auto mode.

- (1) Configuration mode
- (2) Sampling mode
  - Manual mode
  - Auto mode (for UART only)

Immediately after a hardware reset or power-on, internal initialization starts. During the internal initialization, all the register values and states of external pins are undefined. After the internal initialization is completed, the device goes into Configuration mode. Configure various operational settings in Configuration mode <sup>(\*)1</sup>. After configuration is completed, go to the sampling mode to read out the temperature, angular rate, and acceleration data. To change the operation mode, write to **MODE\_CMD** (MODE\_CTRL[0x02 (W0)] bit[9:8]). When software reset is executed by writing "1" to **SOFT\_RST** (GLOB\_CMD[0x0A (W1)] bit[7]), internal initialization is executed and then the device goes into Configuration mode regardless of the current operation mode.

When the UART interface is used, writing to **UART\_AUTO** (UART\_CTRL[0x08 (W1)] bit[0]) can switch between the Manual mode and the Auto mode <sup>(\*)2</sup>.

NOTE: When the SPI interface is used, Manual mode must be selected. Otherwise, the device does not work properly.

\*1) Make sure that the device is in Configuration mode when you write to the registers to configure operational settings. In Sampling mode, writing to registers is ignored except the following cases.

- Writing to **MODE\_CMD** (MODE\_CTRL[0x02 (W0)] bit[9:8])
- Writing to **GPIO\_DATA** (GPIO[0x08 (W0)] bit[9:8])
- Writing to **SOFT\_RST** (GLOB\_CMD[0x0A (W1)] bit[7])
- Writing to **WINDOW\_ID** (WIN\_CTRL[0x7E (W0/W1)] bit[7:0])

\*2) The following explains register notation used in this document.

For example, MODE\_CTRL[0x02 (W0)] bit[9:8] refers to:

- MODE\_CTRL: Register Name
- [0x02 (W0)]: First number is the Register Address, (W0) refers to Window Number "0"
- bit[9:8]: Bits from 9 to 8

\*3) While the device is in UART Auto mode and sensor sampling is active, register read access is not supported. Otherwise, the sampling data transmitted in the UART Auto mode will be corrupted by the response data from the register read.

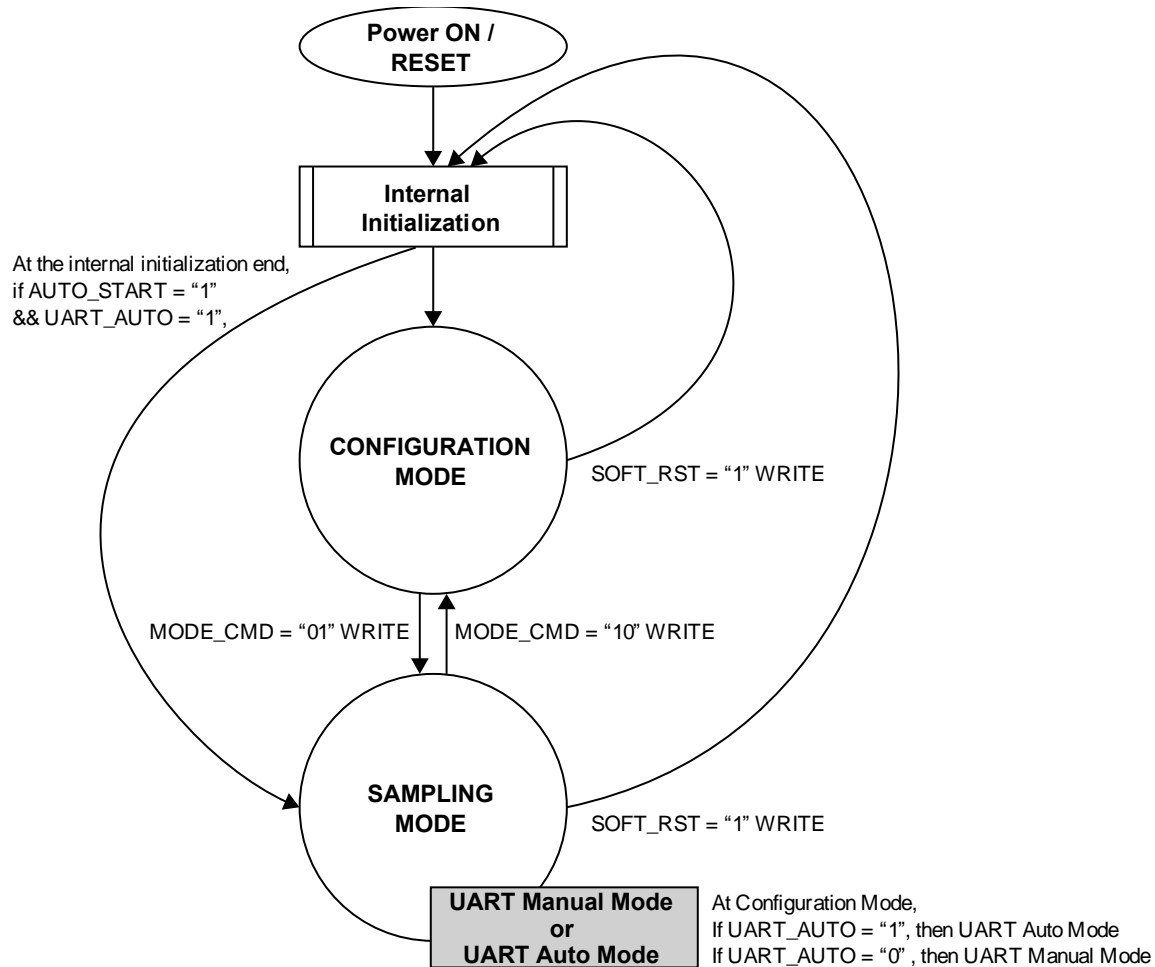


Figure 4.3 Operational State Diagram

### 4.3 Functional Block Diagram

Fc: 189 Hz, 2nd order  
Update Rate: 10 kHz

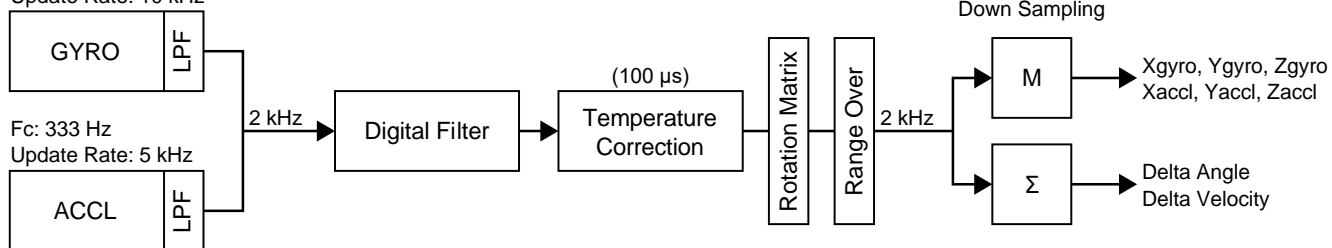


Figure 4.4 Functional Block Diagram

## 4.4 Data Output Timing

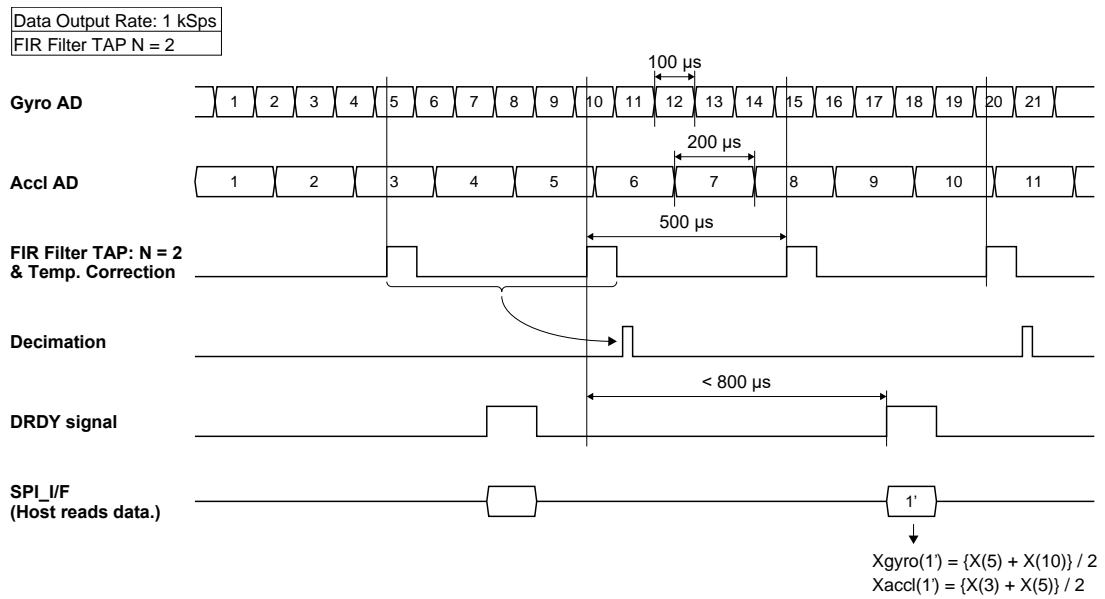


Figure 4.5 Data Output Timing – Data Output Rate 1 kSps, FIR Filter TAP N = 2

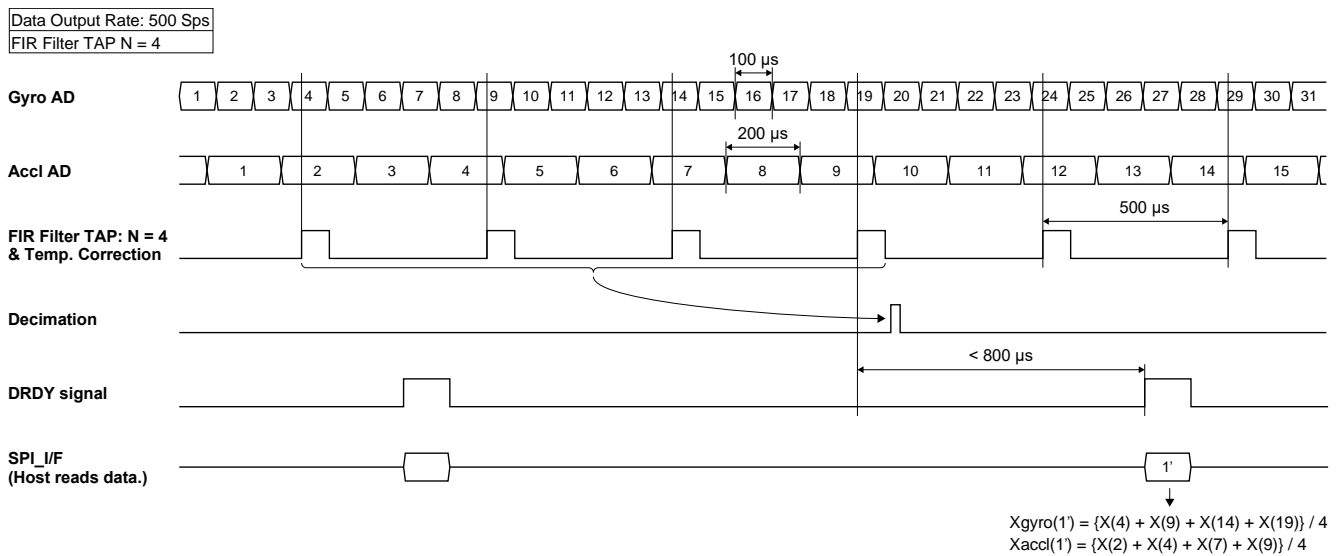


Figure 4.6 Data Output Timing – Data Output Rate 500 Sps, FIR Filter TAP N = 4

## 4.5 Data Ready Signal

The Data Ready signal is asserted when one sampling cycle completes and the registers are updated with new sensor values. When the sensor values are read out, the Data Ready signal becomes negated. In case of UART Auto mode, the Data Ready signal becomes negated just before data is output.

The Data Ready signal is output to the pin when **DRDY\_ON** (MSC\_CTRL[0x02 (W1)] bit[2]) is set to "1". The polarity of the signal can be changed by writing to **DRDY\_POL** (MSC\_CTRL[0x02 (W1)] bit[1]).

The Data Ready signal is the logical sum of all the ND flags corresponding to each sensor value. If all the ND flags are disabled in **ND\_EN** (SIG\_CTRL[0x00 (W1)] bit[15:9] [7:2]), the Data Ready signal will not be asserted. On the other hand, if all the sensor values enabled in **ND\_EN** (SIG\_CTRL[0x00 (W1)] bit[15:9]) are not read out, the Data Ready signal is kept asserted and never becomes negated.

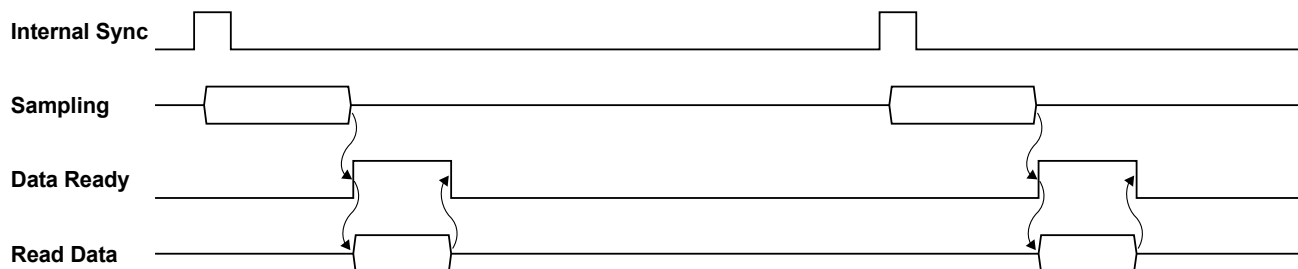


Figure 4.7 Data Ready Signal Timing

## 4.6 Sampling Counter

By reading the COUNT[0x0A (W0)] register, the counter value can be read which is incremented based on the sampling completion timing of the internal A/D converter. The count interval is 500  $\mu$ sec/count and is based on the precision of the internal reference oscillator (crystal).

Additionally, during UART / SPI Burst mode and in UART Auto mode, the counter value can be included in the burst response by setting **COUNT\_OUT** (BURST\_CTRL1[0x0C (W1)] bit[1]). For information about the response format, refer to 5.3 Data Packet Format.

## 4.7 GPIO

The device has two general purpose I/O ports (GPIO). By accessing the GPIO[0x08 (W0)] register, the direction (in/out) of each port can be configured and data can be read/written to. The GPIO port can be read in the normal mode, and also in the UART Burst mode or UART Auto mode.

GPIO1 is shared with the Data Ready signal. The switch between GPIO1 and the Data Ready signal can be controlled by **DRDY\_ON** (MSC\_CTRL[0x02 (W1)] bit[2]). When "0" is written to **DRDY\_ON**, GPIO1 acts as a general purpose I/O port.

GPIO2 is shared with the EXT signal (External Trigger Input or External Counter Reset). The switch of GPIO2 and the EXT signal can be controlled by **EXT\_SEL** (MSC\_CTRL[0x02 (W1)] bit[7:6]). When "00" is written to **EXT\_SEL**, GPIO2 acts as a general purpose I/O port.

## 4.8 Self Test

The self test function can be used to check whether the outputs of the gyroscope and the accelerometer are within the pre-determined range and operating properly. For the gyroscope, the test result is OK if the bias of the output for each X-, Y-, or Z-axis is close to zero when the device is not moving. For the accelerometer, the test result is OK if the absolute value of the output as a three-dimensional vector is equal to the gravitational acceleration. When performing the self test, make sure the device does not move during the test and the test is conducted in a place without vibration.

For information about the execution time of the self test, see "Self Test Time" in Table 1.4 Interface Specifications.

To use the self test function, see the descriptions of **SELF\_TEST** (MSC\_CTRL[0x02 (W1)] bit[10]) and **ST\_ERR\_ALL** (DIAG\_STAT[0x04 (W0)] bit[1]).

NOTE: When executing the self-test, be sure to disable the external trigger function.

MSC\_CTRL[0x02 (W1)] bit[7:6] = "00" or "01"



## 4.9 External Trigger Input

The External Trigger Input function provides control of the sample data output timing by using an externally supplied input pulse signal to the GPIO2 (EXT) pin. By enabling the **EXT\_SEL** (MSC\_CTRL[0x02 (W1)] bit[7:6]), GPIO2 pin can be used as the External Trigger Input pin. The polarity of the External Trigger Input is a positive pulse.

To enable the External Trigger Input function, program the settings as shown below.

- Set **EXT\_SEL** (MSC\_CTRL[0x02 (W1)] bit[7:6]) to "11" to enable the external trigger input function.
- For normal angular rate and linear acceleration output (**ATTI\_ON** (ATTI\_CTRL[0x15 (W1)] bit [9]) = "0"), set **FILTER\_SEL** according to the External Trigger Input frequency to FILTER\_CTRL[0x06 (W1)] bit[4:0]. The valid **FILTER\_SEL** settings are described in Table 6.2 for a selected **DOUT\_RATE**. But in this case, the table refers to the valid combination of External Trigger Input frequency and **FILTER\_SEL** (instead of **DOUT\_RATE** sps). If the External Trigger Input frequency is between two **DOUT\_RATE** sps settings, choose the lower **DOUT\_RATE** sps setting to determine supported **FILTER\_SEL** settings.

NOTE: For normal angular rate and linear acceleration output, the minimum External Trigger Input frequency is 15.625 Hz.

- For Delta Angle / Delta Velocity output (**ATTI\_ON** = "1"), **FILTER\_SEL** can be set to TAP = 0 or other values depending on the application.

When this function is active, the operation is as follows:

For UART Auto mode:

The FIR Filter & Temp Correction processing is performed at the internal 2 kHz rate timing for Gyro and Accl sampling data. When the external trigger input signal is asserted, the current sampling data after the FIR Filter & Temp Correction processing has been performed is automatically sent to the host.

For UART / SPI Manual mode and SPI Burst mode

The FIR Filter & Temp Correction processing is performed at the internal 2 kHz rate timing for Gyro and Accl sampling data. When the external trigger input signal is asserted, the current sampling data after FIR Filter & Temp Correction processing is updated in each register, and the Data Ready signal is asserted. The host should read the sampling data in synchronization with Data Ready.

The External Trigger Input timing requirements and diagrams are shown in Table 4.1, Figure 4.8, and Figure 4.9.

**Table 4.1 External Trigger Input Timing Requirements**

Parameter	Description	Min.	Max.	Unit
t <sub>ETW</sub>	External Trigger Input width	100	–	ns
t <sub>ETC</sub> <sup>*1</sup>	External Trigger Input cycle	1	1000	ms
t <sub>ETA2T_G</sub>	Time from Gyro ADC's completion to External Trigger Input (This is a precision of External Sync.)	0	600	μs
t <sub>ETA2T_A</sub>	Time from Accl ADC's completion to External Trigger Input (This is a precision of External Sync.)	0	700	μs
t <sub>ETD1</sub> <sup>*2</sup>	Delay time from External Trigger Input to DRDY asserted	–	300	μs

\*1) For normal angular rate and linear acceleration output (**ATTI\_ON** = "0"), the minimum External Trigger Input frequency is 15.625 Hz, because the strongest low pass filtering is moving average TAP = 128.

\*2) This does not include group delay of the internal filter.

Data Output Rate: 1 kSps  
FIR Filter TAP N = 2

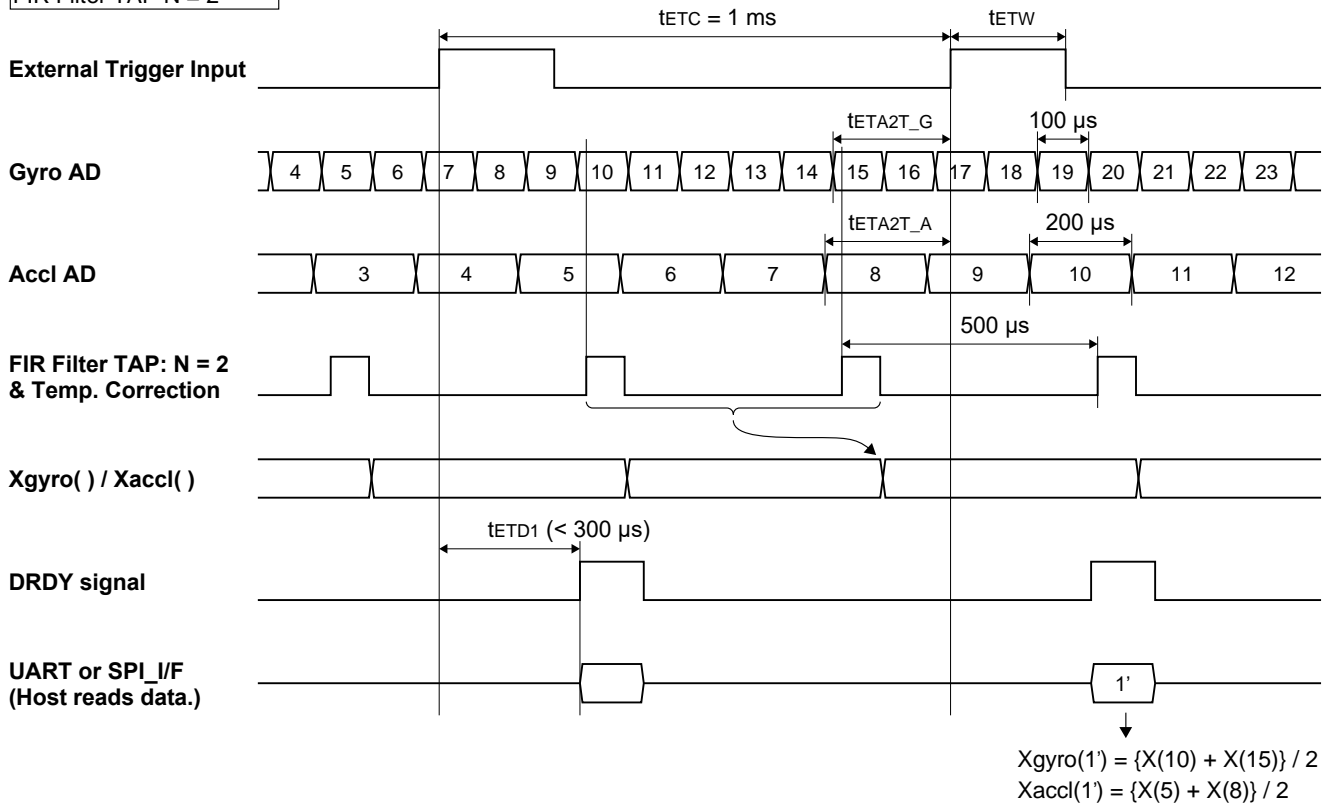


Figure 4.8 External Trigger Input (UART / SPI Manual Mode)

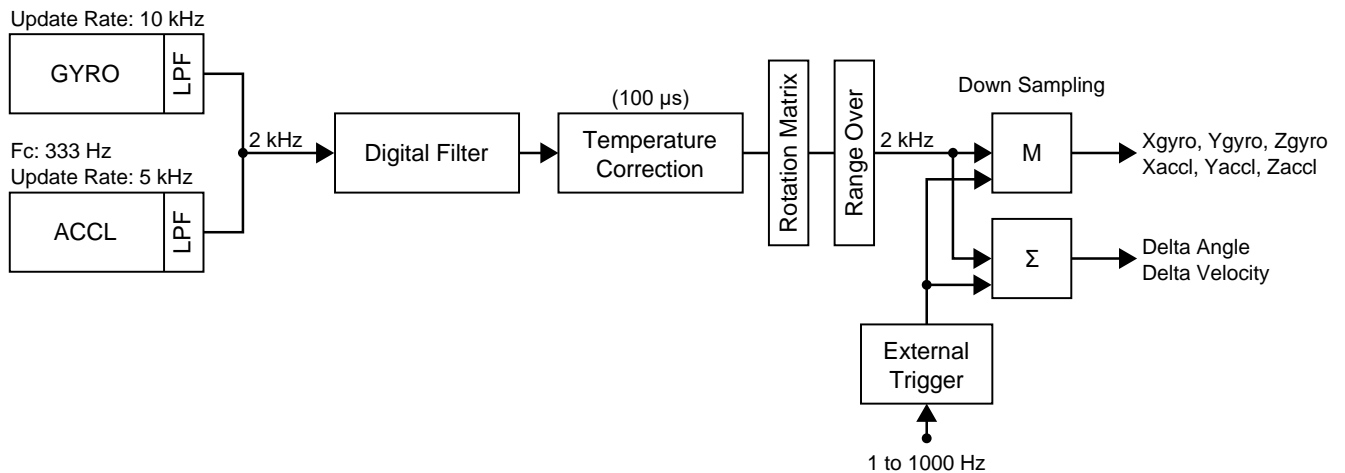


Figure 4.9 External Trigger Input Functional Block Diagram

#### 4.10 External Counter Reset Input

The External Counter Reset Input function can be used to measure the time offset from an externally supplied input trigger on the GPIO2 (EXT) pin to the completion of the next ADC sampling group.

This function is enabled by writing to **EXT\_SEL** (MSC\_CTRL[0x02 (W1)] bit [7:6]) to select GPIO2 for use as an External Counter Reset Input pin. The active polarity of the input signal is positive pulse.

The following describes the operation when this function is active:

- The IMU has an internal 16-bit up counter incrementing at 62.5 kHz.
- The counter begins counting starting from 0 <sup>(\*)1</sup> when Sampling mode begins. The counting resolution is 16  $\mu$ s.
- The counter can be reset by assertion of an external signal on the External Counter Reset Input pin. After the counter is reset, the count value is cleared and begins incrementing again from 0.
- The counter value is transferred at the time of the ADC sampling completion and stored in the COUNT[0x0A (W0)] register before the Data Ready signal is asserted.
- The Host can obtain the time offset from assertion of the most recent External Counter Reset Input signal to the ADC sampling completion time by reading the sampling data with the counter value when Data Ready signal is asserted.
- The counter stops counting <sup>(\*)2</sup> when Sampling mode is stopped.
- The counter will roll over and increment from 0 again if the count value increments past 65535.

\*1) Enter Sampling mode from Configuration mode

\*2) Leave Sampling mode and enter Configuration mode

NOTE: When the External Counter Reset Input function is enabled, the COUNT[0x0A (W0)] register stores the counter value instead of the sampling count.

The timing specification and timing diagram for the External Counter Reset Input function are shown in Table 4.2 and Figure 4.10.

**Table 4.2 External Counter Reset Input Timing**

Parameter	Description	Min.	Max.	Unit
t <sub>ERW</sub>	External Reset Input Width	100	–	ns
t <sub>ERC</sub>	External Reset Input Cycle	1	1000	ms
t <sub>ER2A</sub>	Time from External Reset Input to ADC's completion	(count <sup>*1</sup> x 16) + $\Delta$ <sub>ER2A</sub>		$\mu$ s
$\Delta$ <sub>ER2A</sub>	Precision of t <sub>ER2A</sub>	-150	150	$\mu$ s

\*1) The count value is read from register COUNT[0x0A (W0)] as indicated.

Data Output Rate: 1 kSps  
FIR Filter TAP N = 2

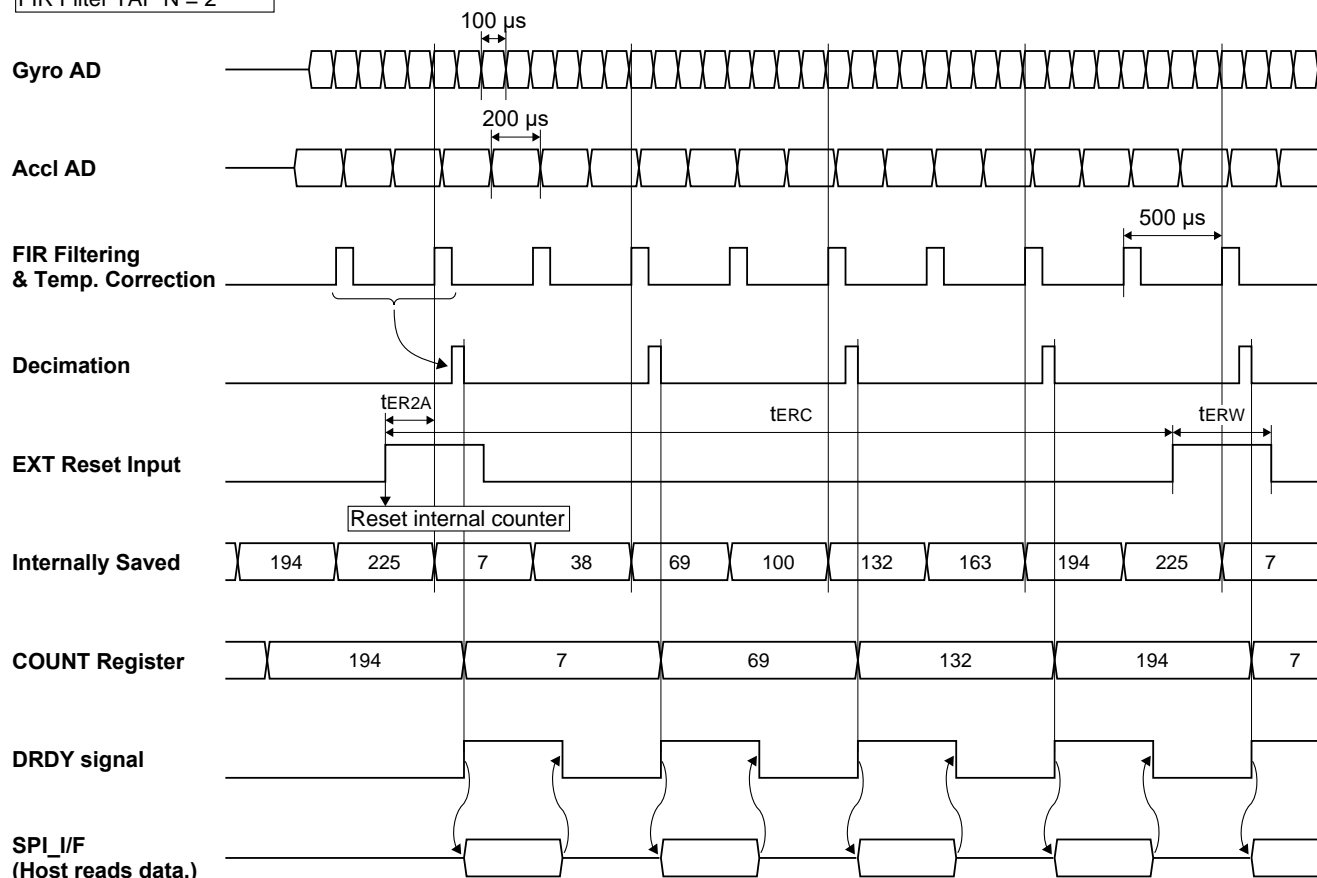


Figure 4.10 External Counter Reset Input

#### 4.11 Checksum

A checksum can be appended to the response data during UART / SPI Burst mode or UART Auto mode by enabling this function in **CHKSM\_OUT** (BURST\_CTRL1[0x0C (W1)] bit[0]).

The checksum range of the data content is calculated immediately after the address byte (AD = 0x80) of the response data up to (not including) the delimiter byte (CR = 0x0D). The calculation method of checksum is a simple addition of the data content in units of 16 bits, and the resulting sum is truncated to 16 bits and appended as checksum just before the delimiter byte (CR = 0x0D).

Example:

Because the sum total is "611B4" for the following response data stream, the checksum is "11B4":  
"FE01 C455 4000 0052 33C0 0043 7BC8 004A 2608 FD73 3AA0 FF75 4C30 1F53 8FD0 0600 0014"

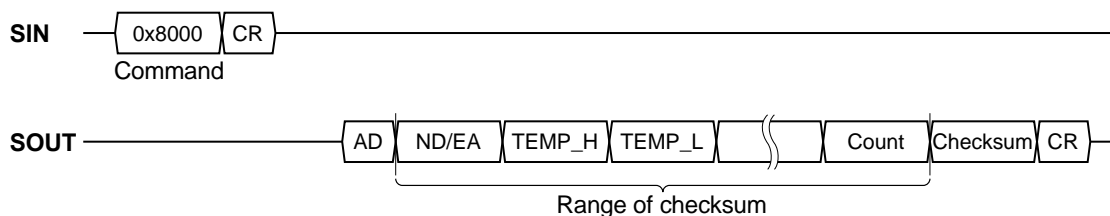


Figure 4.11 Checksum

## 4.12 Automatic Start (For UART Auto Mode Only)

The Automatic Start function, when enabled, allows the device to automatically enter Sampling mode after completing internal initialization when power is supplied or the IMU is reset. This function is designed to be used in conjunction with the UART Auto mode. Please refer to Figure 4.3 for the state transition.

Follow the procedure below to enable the Automatic Start function:

- Write “1” to both **UART\_AUTO** (bit[0]) and **AUTO\_START** (bit[1]) of **UART\_CTRL**[0x08 (W1)].
- Store the current register settings to non-volatile memory by writing “1” to **FLASH\_BACKUP** (GLOB\_CMD[0x0A (W1)] bit[3]). After completion of the **FLASH\_BACKUP** command, confirm the results by reading **FLASH\_BU\_ERR** (DIAG\_STAT[0x04 (W0)] bit[0]).
- The IMU will automatically enter Sampling mode after the power supply is cycled, or a hardware reset, or a software reset command is executed.

The Automatic Start function can be enabled simultaneously with the External Trigger Input function.

Follow the procedure below to enable the Automatic Start with External Trigger Input function:

- Write “1” to both **UART\_AUTO** (bit[0]) and **AUTO\_START** (bit[1]) of **UART\_CTRL**[0x08 (W1)].
- Write “10” to **EXT\_SEL** (MSC\_CTRL[0x02 (W1)] bit[7:6]) to enable the External Trigger Input. Please connect the external trigger input signal to the GPIO2 pin.
- Store the current register settings to non-volatile memory by writing “1” to **FLASH\_BACKUP** (GLOB\_CMD[0x0A (W1)] bit[3]). After completion of the **FLASH\_BACKUP** command, confirm the results by reading **FLASH\_BU\_ERR** (DIAG\_STAT[0x04 (W0)] bit[0]).
- The IMU will automatically enter Sampling mode after the power supply is cycled, or a hardware reset, or a software reset command is executed.

## 4.13 Filter

This device contains built-in user configurable digital filters that are applied to the sensor data. The type of filter (moving average filter or FIR Kaiser filter) and the numbers of TAPs can be set with the **FILTER\_CTRL**[0x06 (W1)] register.

### (1) Moving Average Filter

TAP setting can be N = 2, 4, 8, 16, 32, 64, or 128.

Figure 4.12 shows the characteristics of this filter.

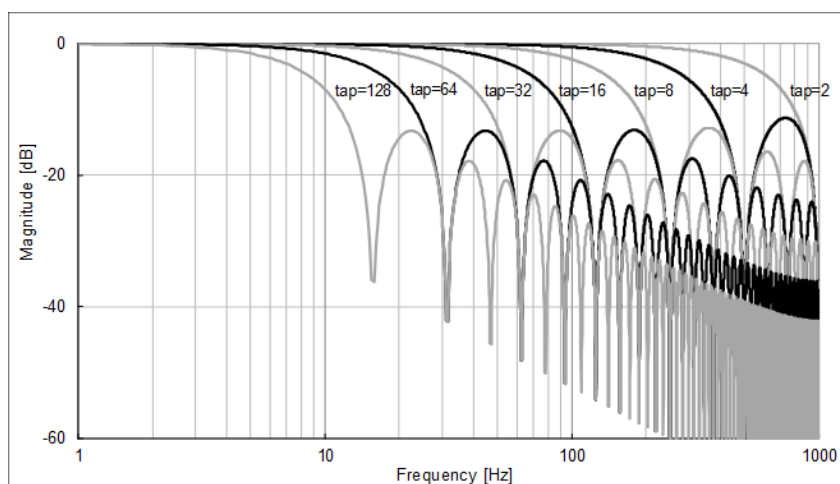


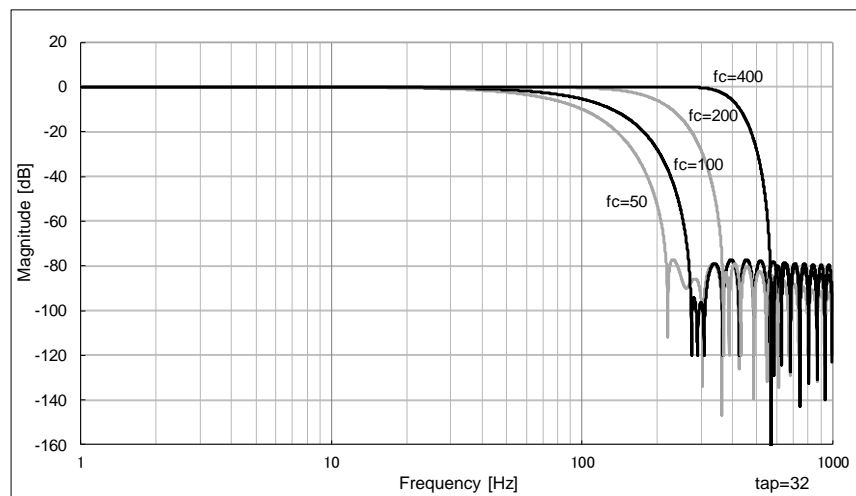
Figure 4.12 Moving Average Filter Characteristics

**(2) FIR Kaiser Filter**

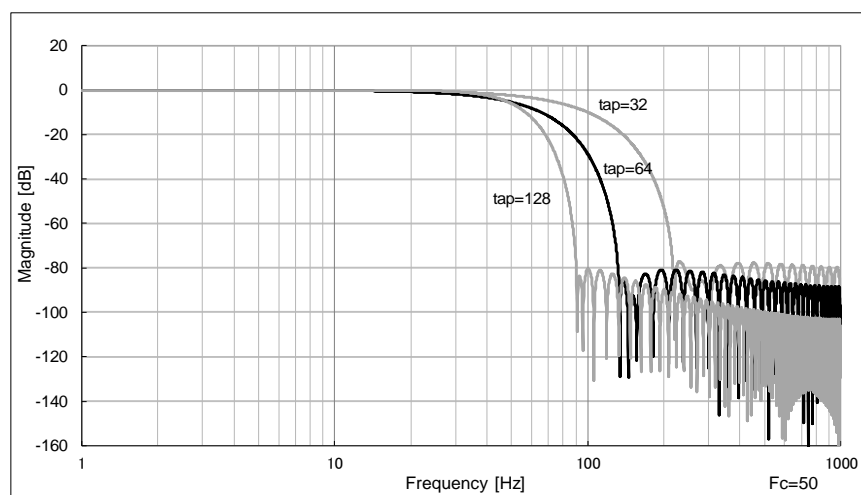
Uses Kaiser Window (parameter = 8).

TAP setting can be N = 32, 64, or 128 with cutoff frequency  $f_c = 50, 100, 200,$  or  $400\text{Hz}$ .

Figure 4.13 and Figure 4.14 show the typical characteristic of this filter.



**Figure 4.13 FIR Kaiser Filter Typical Characteristic 1**



**Figure 4.14 FIR Kaiser Filter Typical Characteristic 2**

Please note that the transient response of the digital filter is a maximum of 127 samples from the sampling start time and varies depending on the output data rate and the filter tap setting. Refer to Table 4.3 which describes the transient response in terms of number of samples for valid combinations of output data rate and filter tap setting.

**Table 4.3 Transient Response in Number of Samples Based on Output Data Rate vs Filter Tap**

	TAP0	TAP2	TAP4	TAP8	TAP16	TAP32	TAP64	TAP128
2000 sps	0	1	3	7	15	31	63	127
1000 sps		0	1	3	7	15	31	63
500 sps			0	1	3	7	15	31
400 sps				1	3	6	12	25
250 sps				0	1	3	7	15
200 sps					1	3	6	12
125 sps					0	1	3	7
100 sps						1	3	6
80 sps						1	2	5
62.5 sps						0	1	3
50 sps							1	3
40 sps							1	2
31.25 sps							0	1
25 sps								1
20 sps								1
15.625 sps								0

	TAP32 Fc50	TAP32 Fc100	TAP32 Fc200	TAP32 Fc400	TAP64 Fc50	TAP64 Fc100	TAP64 Fc200	TAP64 Fc400	TAP128 Fc50	TAP128 Fc100	TAP128 Fc200	TAP128 Fc400
2000 sps	31	31	31	31	63	63	63	63	127	127	127	127
1000 sps	15	15	15	15	31	31	31	31	63	63	63	63
500 sps	7	7	7		15	15	15		31	31	31	
400 sps	6	6	6		12	12	12		25	25	25	
250 sps	3	3			7	7			15	15		
200 sps	3	3			6	6			12	12		
125 sps	1				3				7			
100 sps	1				3				6			
80 sps												
62.5 sps												
50 sps												
40 sps												
31.25 sps												
25 sps												
20 sps												
15.625 sps												

#### 4.14 Delta Angle / Delta Velocity Output

Delta-Angle / Delta-Velocity Output is the function to output integrated angle increments and integrated velocity increments by mathematical accumulation of the angular rate and linear acceleration sensor values.

The Delta Angle register and Delta Velocity register[0x64–0x7B (W0)] represent the 32-bit numerical integration of angular rate and linear acceleration value before the down-sampling block. The integration of angular rate and linear acceleration value is reset when the Host reads the Delta Angle register and Delta Velocity register or automatically in UART Auto mode after sending the output data.

For example, in case of down-sampling ratio 4:1 (**DOUT\_RATE** of **SMPL\_CTRL**[0x05 (W1)] = 0x02), the integration data of four x-axis angular rate and x-axis linear acceleration values after filtering and temperature correction is stored in **XLDTA\_HIGH** and **XLDTA\_LOW** (Register[0x64–0x67 (W0)]), and **XLDTV\_HIGH** and **XLDTV\_LOW** (Register[0x70–0x73 (W0)]) respectively in 32-bit data format. Figure 4.15 shows the timing diagram in the case of down-sampling ratio 4:1, 500 sps and Filter TAP: N = 0.

- Point A: By reading **XLDTA\_HIGH** and **XLDTA\_LOW** from the Host at point A' (2 ms before point A), the integration of angular rate and acceleration value is reset to 0.
- Point B: "55" is output after integration of values 15, 20, 25, -5.

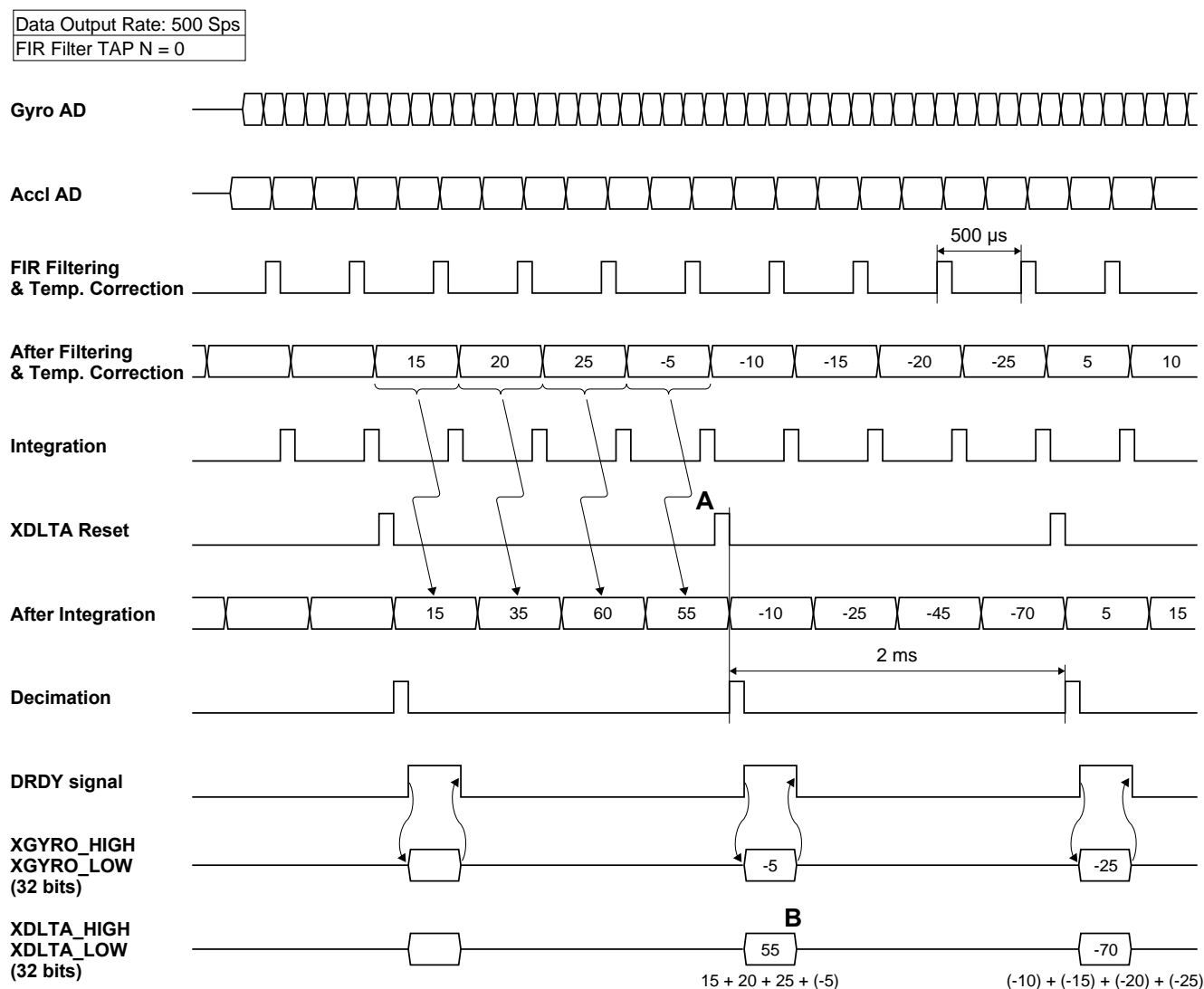


Figure 4.15 Delta Angle / Delta Velocity



As described above, the numerical integration of angular rate and linear acceleration values is reset by reading Delta Angle register and Delta Velocity register by the Host or automatically in UART Auto mode after sending the output data. However, when selecting 16-bit output mode in **DLTA\_BIT** (BURST\_CTRL2[0x0E (W1)] bit[11]) or **DLTV\_BIT** (BURST\_CTRL2[0x0E (W1)] bit[10]), the upper 16 bits of the value will be output and lower 16 bits will be added to the next integration cycle.

NOTE:

- Delta Angle / Delta Velocity output is restricted and not supported for SPI Normal mode reads.
- Delta Angle and Gyro sensor output cannot be used together, and Delta Velocity and Accelerometer sensor cannot be used together.
- When Delta Angle / Delta Velocity is enabled, the **FILTER\_SEL** can be set to TAP = 0 (or other value regardless of **DOUT\_RATE** setting)

To enable Delta Angle / Delta Velocity output, program the settings as shown below.

1. Enable burst data output by writing "1" to **DLTA\_OUT** or **DLTV\_OUT** (BURST\_CTRL1[0x0D (W1)] bit[11:10]).
2. Set the required bit length for **DLTA\_BIT** or **DLTV\_BIT** (BURST\_CTRL2[0x0F (W1)] bit[11:10]).
3. Set the required scale factor for Delta Angle and/or Delta Velocity, respectively in **DLTA\_RANGE\_CTRL** or **DLTV\_RANGE\_CTRL** (GLOB\_CMD3[0x12 (W1)] bit[7:4] & bit[3:0]).

NOTE: By setting an appropriate scale factor, overflow due to accumulation of angular velocity and acceleration values can be avoided. The appropriate scale factor depends on the operating conditions of the application.

4. Write "1" to **ATTI\_ON** (ATTI\_CTRL[0x15 (W1)] bit[9]) to enable the Delta Angle / Delta Velocity output function. Section 5.3 provides an example of the Delta Angle / Delta Velocity output data fields for burst output in data packet format.

## 4.15 Range Over Function

This device supports the notification when a range over condition is detected in the sensor data. The range over detection threshold is equal to the dynamic range for both the Gyro sensor and the Accelerometer sensor described in 1.3 *Characteristics and Electrical specifications*.

Gyro Sensor:  $\pm 450$  [deg/s]

Accelerometer:  $\pm 7.5$  [G] for Output Range  $\pm 8$  G (**A\_RANGE\_CTRL** of GLOB\_CMD3[0x13 (W1)] bit[8] = "0")  
 $\pm 15$  [G] for Output Range  $\pm 16$  G (**A\_RANGE\_CTRL** of GLOB\_CMD3[0x13 (W1)] bit[8] = "1")

Detection is performed by "Range Over" block in the processing order as described in Figure 4.4 Functional Block Diagram. The host can confirm that a range over has occurred by reading the **RO** bit of FLAG (ND / EA) in the burst read data or the register FLAG[0x06 (W0)] bit[8]. The source of the range over occurrence can be confirmed by reading RANGE\_OVER[0x0C (W0)] bit[13:8] and bit[0]. The **RO** bits (RANGE\_OVER[0x0C (W0)] bit[13:8] bit[0]) are reset by reading the register, so that any subsequent range over detection can be resumed during sampling.

Refer to FLAG[0x06 (W0)], RANGE\_OVER[0x0C (W0)] for register operation.

## 4.16 Frame Alignment Correction

The frame alignment of the three-axis gyro triad and accelerometer triad can be corrected by using the R\_MATRIX function. The matrix coefficients are 16-bit fixed-point numbers with 2 bits representing the integer part including the sign bit, and 14 bits representing the fractional part. The default coefficient values for the R\_MATRIX, if unchanged, will result in an identity matrix.

- R\_MATRIX\_M\*\*[0x38–0x49 (W1)]: The coefficients for the 3 x 3 rotation matrix for gyro triad.

The coefficient setting register is common to angular velocity and acceleration.

Frame alignment correction of the gyroscope triad is represented by the 3 x 3 matrix multiplication operation:

$$\begin{pmatrix} Gx \\ Gy \\ Gz \end{pmatrix} = \begin{pmatrix} R\_MATRIX\_M11 & R\_MATRIX\_M12 & R\_MATRIX\_M13 \\ R\_MATRIX\_M21 & R\_MATRIX\_M22 & R\_MATRIX\_M23 \\ R\_MATRIX\_M31 & R\_MATRIX\_M32 & R\_MATRIX\_M33 \end{pmatrix} \begin{pmatrix} gx \\ gy \\ gz \end{pmatrix}$$

G: Gyroscope output data after the rotation matrix operation.

Data is output to the X-, Y-, and ZGYRO registers ([0x12–0x1D (W0)]).

M: Misalignment 3 x 3 matrix

g: Measurement data (3 axes)

Frame alignment correction of the accelerometer triad is represented by the 3 x 3 matrix multiplication operation:

$$\begin{pmatrix} Ax \\ Ay \\ Az \end{pmatrix} = \begin{pmatrix} R\_MATRIX\_M11 & R\_MATRIX\_M12 & R\_MATRIX\_M13 \\ R\_MATRIX\_M21 & R\_MATRIX\_M22 & R\_MATRIX\_M23 \\ R\_MATRIX\_M31 & R\_MATRIX\_M32 & R\_MATRIX\_M33 \end{pmatrix} \begin{pmatrix} ax \\ ay \\ az \end{pmatrix}$$

A: Accelerometer output data after the rotation matrix operation.

Data is output to the X-, Y-, and ZACCL registers ([0x1E–0x29 (W0)]).

M: Misalignment 3 x 3 matrix

a: Measurement data (3 axes)

NOTE: The initial value for M (misalignment matrix) is the identity matrix. When M is unmodified, G = g, A = a.

Refer to 7.1.9 *Frame Alignment Correction (SPI)* for an example of setting the rotation matrix coefficients.

NOTE: The frame alignment correction has effect on the Delta Angle / Delta Velocity output function.

## 5. Digital Interface

This device has the following two external interfaces.

- (1) SPI interface
- (2) UART interface

The SPI interface and the UART interface have almost the same functions, except additionally the UART interface supports Auto mode function. Because both interfaces are always active, the user needs only to connect the desired interface pins SPI or UART, without needing any hardware pin configuration or selection.

NOTE: Connecting both SPI and UART at the same time is not supported and may result in malfunction of the device.

The registers inside the device are accessed via the SPI or UART interfaces.

In this document, data sent to the device is called a "Command" and data sent back in response to the command is called a "Response". There are two types of commands: write command and read command. The write command has no response. The write command always writes to the internal register in 8-bit words. The response to the read command, i.e., the data from the internal register, is always read in 16-bit words.

When reading from the registers, there is a special mode called the Burst mode in addition to the Normal mode.

When the IMU output data rate is high (i.e., 1000 sps), it is possible to exceed the bandwidth of the host interface and causes the data transmission to be incorrect. In this case, the user must balance the transmission data rate and the bandwidth capability of the host interface.

Adjust the following settings accordingly to optimize the host interface bandwidth:

- For the UART, adjust the baud rate in **BAUD\_RATE** (UART\_CTRL[0x08 (W1)] bit[9:8]).
- For the SPI, adjust the host side SPI clock frequency and SPI wait time.

Adjust the following settings accordingly to optimize the transmission data rate:

- The transmission data rate is affected by the data output rate setting in **DOUT\_RATE** (SMPL\_CTRL[0x04 (W1)] bit[15:8]).
- The transmission data rate is also affected by the number of output bytes included in Burst mode read transfer. The adjustment to the number of output bytes is in registers BURST\_CTRL1[0x0C (W1)] and BURST\_CTRL2[0x0E (W1)].

Several concrete examples for setting the transmission data rate and host interface bandwidth are shown below:

(1) For UART and 32-bit output:

Baudrate: 921600 baud, Data output rate: 2000 sps

- **BAUD\_RATE** (UART\_CTRL[0x08 (W1)] bit[9:8]) = "10": 921600 baud
- **UART\_AUTO** (UART\_CTRL[0x08 (W1)] bit[0]) = "1": UART Auto Mode
- **DOUT\_RATE** (SMPL\_CTRL[0x04 (W1)] bit[15:8]) = 0x00: 2000 sps
- BURST\_CTRL1[0x0C (W1)] = 0xF006: FLAG, TEMP, angle rate, acceleration, GPIO, and COUNT outputs
- BURST\_CTRL2[0x0E (W1)] = 0x7000: TEMP, angle rate, and acceleration are all output in 32 bits.

Baudrate: 460800 baud, Data output rate: 1000 sps

- **BAUD\_RATE** (UART\_CTRL[0x08 (W1)] bit[9:8]) = "00": 460800 baud
- **UART\_AUTO** (UART\_CTRL[0x08 (W1)] bit[0]) = "1": UART Auto Mode
- **DOUT\_RATE** (SMPL\_CTRL[0x04 (W1)] bit[15:8]) = 0x01: 1000 sps
- BURST\_CTRL1[0x0C (W1)] = 0xF006: FLAG, TEMP, angle rate, acceleration, GPIO, and COUNT outputs
- BURST\_CTRL2[0x0E (W1)] = 0x7000: TEMP, angle rate, and acceleration are all output in 32 bits.

(2) For SPI and 32-bit output:

- SPI Interface Transmission Setting:  $f_{SCLK} = 1 \text{ MHz}$  and  $t_{STALL} = 24 \mu\text{s}$  for normal mode
- **DOUT\_RATE** (SMPL\_CTRL[0x04 (W1)] bit[15:8]) = 0x01: 1000 sps
- BURST\_CTRL1[0x0C (W1)] = 0xF006: FLAG, TEMP, angle rate, acceleration, GPIO, and COUNT outputs
- BURST\_CTRL2[0x0E (W1)] = 0x7000: TEMP, angle rate, and acceleration are all output in 32 bits.

(3) For UART and 16-bit output:

- **BAUD\_RATE** (UART\_CTRL[0x08 (W1)] bit[9:8]) = "00": 460800 baud
- **UART\_AUTO** (UART\_CTRL[0x08 (W1)] bit[0]) = "1": UART Auto Mode
- **DOUT\_RATE** (SMPL\_CTRL[0x04 (W1)] bit[15:8]) = 0x01: 1000 sps
- BURST\_CTRL1[0x0C (W1)] = 0xF006: FLAG, TEMP, angle rate, acceleration, GPIO, and COUNT outputs
- BURST\_CTRL2[0x0E (W1)] = 0x0000: TEMP, angle rate, and acceleration are all output in 16 bits.

(4) For SPI and 16-bit output:

- SPI Interface Transmission Setting:  $f_{SCLK} = 1\text{ MHz}$  and  $t_{STALL} = 24\text{ }\mu\text{s}$  for normal mode
- **DOUT\_RATE** (SMPL\_CTRL[0x04 (W1)] bit[15:8]) = 0x01: 1000 sps
- BURST\_CTRL1[0x0C (W1)] = 0xF006: FLAG, TEMP, angle rate, acceleration, GPIO, and COUNT outputs
- BURST\_CTRL2[0x0E (W1)] = 0x0000: All TEMP, angle rate, and acceleration are all output in 16 bits.

## 5.1 SPI Interface

Table 5.1 shows the communication settings of SPI interface and Table 5.2 shows the SPI timing for Normal mode.

**Table 5.1 SPI Communication Settings**

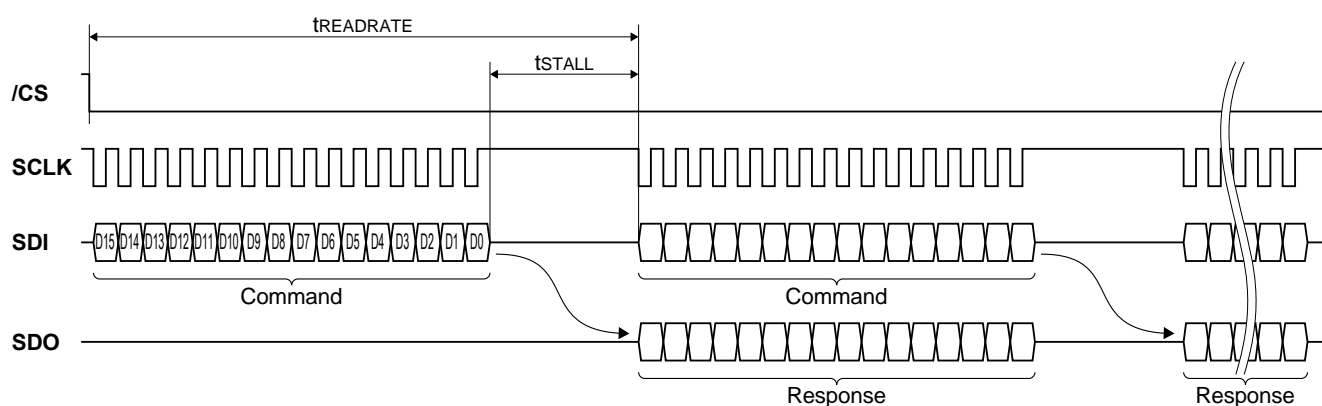
Parameter	Set value
Mode	Slave
Word length	16 bits
Phase	Rising edge
Polarity	Negative logic

**Table 5.2 SPI Timing (Normal Mode)**

Parameter	Min.	Max.	Unit
$f_{SCLK}$	0.01	2.0	MHz
$t_{STALL}$	20	—	$\mu\text{s}$
$t_{WRITERATE}$	40	—	$\mu\text{s}$
$t_{READRATE}$	40	—	$\mu\text{s}$

### 5.1.1 SPI Read Timing (Normal Mode)

The response data to a read command, i.e., the data from the internal register, is always returned in 16-bit words. The SPI interface supports sending the next command during the same bus cycle as receiving a response to the read command (full-duplex).



**Figure 5.1 SPI Read Timing (Normal Mode)**

**Table 5.3 Command Format (Read)**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	A[6:0]							XX							

A[6:0]: Register address (even address)

XX: Don't Care

Table 5.4 Response Format (Read)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D[15:8]								D[7:0]							

D[15:8]: Register read data (upper byte)

D[7:0]: Register read data (lower byte)

### 5.1.2 SPI Write Timing (Normal Mode)

A write command to a register has no response. Unlike register reading, registers are written in 8-bit words.

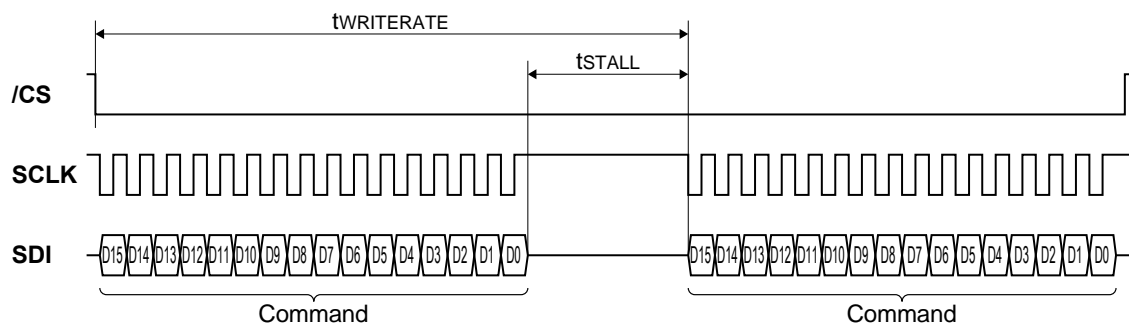


Figure 5.2 SPI Write Timing (Normal Mode)

Table 5.5 Command Format (Write)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	A[6:0]							D[7:0]							

A[6:0]: Register address (even address)

D[7:0]: Register write data

### 5.1.3 SPI Read Timing (Burst Mode)

Burst mode access of read data is supported using a “Burst Read Command” by writing 0x00 in **BURST\_CMD** (BURST[0x00 (W0)] bit[7:0]). In Burst mode, ND flag / EA flag, temperature sensor value, 3-axis gyroscope sensor value, 3-axis acceleration sensor value, GPIO, etc. can be consecutively sent as a response. The response format for the burst read output data is configured by register setting in BURST\_CTRL1[0x0C (W1)] and BURST\_CTRL2[0x0E (W1)]. Please refer to 5.3 Data Packet Format for the response format.

Table 5.6 SPI Timing (Burst Mode)

Parameter	Min.	Max.	Unit
fSCLK	0.01	1.0	MHz
tSTALL1	45	—	μs
tSTALL2	4	—	μs
tREADRATE2	32	—	μs

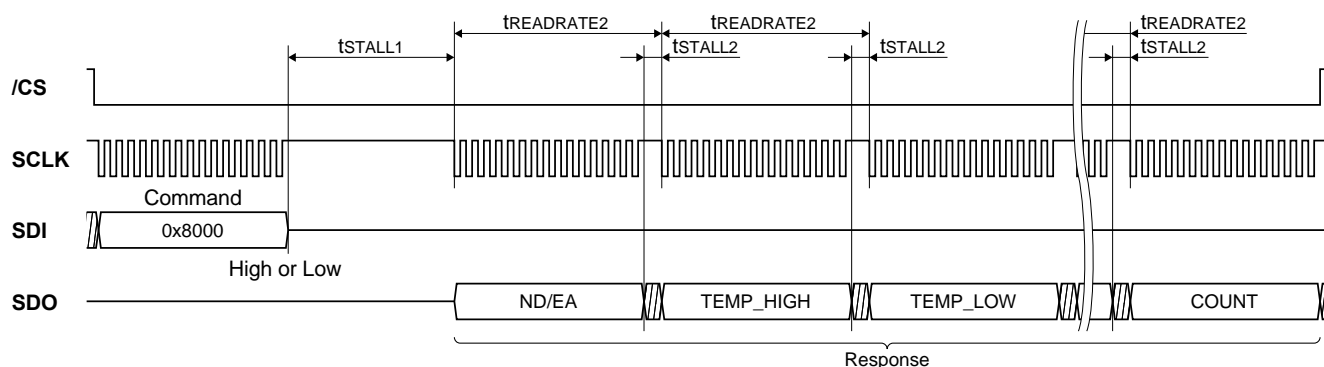


Figure 5.3 SPI Timing (Burst Mode)

NOTE: SPI Burst Read mode can support  $t_{STALL2} = 0 \mu s$  under the following conditions.

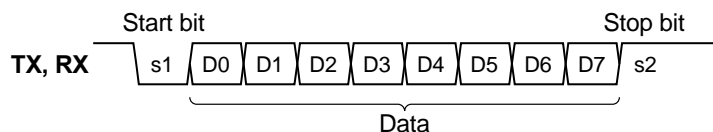
- BURST\_CTRL1[0x0C (W1)]: Set value 0xF003: FLAG, Temperature, Angular Rate, Acceleration, Count, Checksum  
BURST\_CTRL2[0x0E (W1)]: Set value 0x3000: Angular Rate & Acceleration are output in 32 bits.
- $f_{SCLK}$ : Maximum 500 kHz
- $t_{STALL1}$ : Minimum 45  $\mu s$ .
- $t_{STALL2}$ : 0  $\mu s$
- $t_{READRATE2}$ : Minimum 32  $\mu s$ .
- External Counter Reset or External Trigger Input is not supported  
MSC\_CTRL[0x02 (W1)] bit[7:6] = "00"

## 5.2 UART Interface

Table 5.7 shows the supported UART communication settings and Figure 5.4 shows the UART bit format. Please refer to **BAUD\_RATE** (UART\_CTRL[0x08 (W1)] bit[9:8]) for changing the baud rate setting.

**Table 5.7 UART Communication Settings**

Parameter	Set value
Transfer rate	230.4k bps / 460.8 kbps / 921.6kbps
Start	1 bit
Data	8 bits
Stop	1 bit
Parity	None
Delimiter	CR (0x0D)



**Figure 5.4 UART Bit Format**

For the UART interface, a delimiter (1 byte) is placed at the end of each command (by the host) and response (by the IMU). In addition for responses, the address (1 byte) specified by the command is added (by the IMU) to the beginning of the response. Table 5.8 and Table 5.9 show the timings of UART.

**Table 5.8 UART Timing**

Parameter		Manual mode				Auto mode		Unit
		Normal mode		Burst mode				
		Min.	Max.	Min.	Max.	Min.	Max.	
tSTALL	(230.4 kbps)	—	25	—	70	—	—*2	μs
tSTALL	(460.8 kbps)	—	25	—	70	—	—*2	μs
tSTALL	(921.6 kbps)	—	25	—	70	—	—*2	μs
tWRITERATE	(230.4 kbps)	350	—	—	—	350	—	μs
tWRITERATE	(460.8 kbps)	200	—	—	—	200	—	μs
tWRITERATE	(921.6 kbps)	150	—	—	—	150	—	μs
tREADRATE	(230.4 kbps)	350	—	*1	—	—*2	—	μs
tREADRATE	(460.8 kbps)	200	—	*1	—	—*2	—	μs
tREADRATE	(921.6 kbps)	150	—	*1	—	—*2	—	μs

\*1) Please refer to Table 5.9.

\*2) Register reading is not supported while in Sampling mode with UART Auto mode enabled.

**Table 5.9 UART Timing ( $t_{\text{READRATE}}$  Requirements for Burst Mode)**

Parameter	Burst Mode (Min.)	Unit
$t_{\text{READRATE}}$ (230.4kbps)	$300 + (43.4 \times B)$	$\mu\text{s}$
$t_{\text{READRATE}}$ (460.8kbps)	$200 + (21.7 \times B)$	$\mu\text{s}$
$t_{\text{READRATE}}$ (921.6kbps)	$150 + (10.9 \times B)$	$\mu\text{s}$

B = Number of receive data bytes (AD (address) and CR (delimiter) are not included).

Example  $t_{\text{READRATE}}$  Calculation:

BURST\_CTRL1[0x0C (W1)]: Set value 0xF006

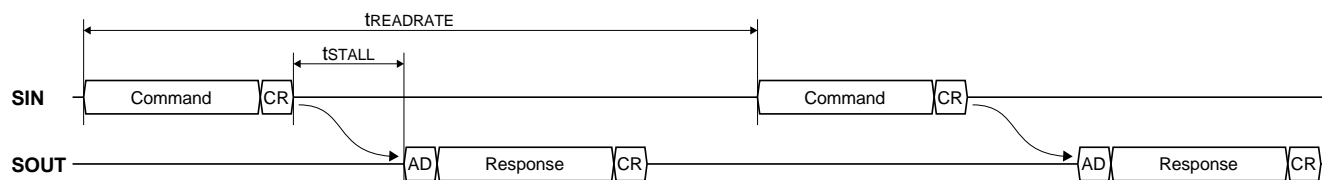
BURST\_CTRL2[0x0E (W1)]: Set value 0x7000

B = 34 bytes for the above stated register setting

$t_{\text{READRATE}}$  (460.8 kbps) =  $200 + (21.7 \times 34) = 937.8 (\mu\text{s})$

### 5.2.1 UART Read Timing (Normal Mode)

The response to the read command, i.e., the data from the internal register, is always returned 16-bit data at a time. The register address (AD) comes at the beginning of the response, for example, 0x02 for the MODE\_CTRL[0x02 (W0)] register.

**Figure 5.5 UART Read Timing (Normal Mode)****Table 5.10 Command Format (Read)**

First byte								Second byte								Third byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	A[6:0]							XX							0x0D								

A[6:0]: Register address (even address)

XX: Don't Care

0x0D: Delimiter

**Table 5.11 Response Format (Read)**

First byte								Second byte								Third byte								Fourth byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	A[6:0]							D[15:8]							D[7:0]							0x0D									

A[6:0]: Register address (even address)

D[15:8]: Register read data (upper byte)

D[7:0]: Register read data (lower byte)

0x0D: Delimiter

### 5.2.2 UART Read Timing (Burst Mode)

Burst mode access of read data is supported using a "Burst Read Command" by writing 0x00 in **BURST\_CMD** (BURST[0x00 (W0)] bit[7:0]). In Burst mode, ND flag / EA flag, temperature sensor value, 3-axis gyroscope sensor value, 3-axis acceleration sensor value, GPIO, etc. can be consecutively sent as a response. The response format for the burst read output data is configured by register setting in BURST\_CTRL1[0x0C (W1)] and BURST\_CTRL2[0x0E (W1)]. Please refer to 5.3 Data Packet Format for the response format.

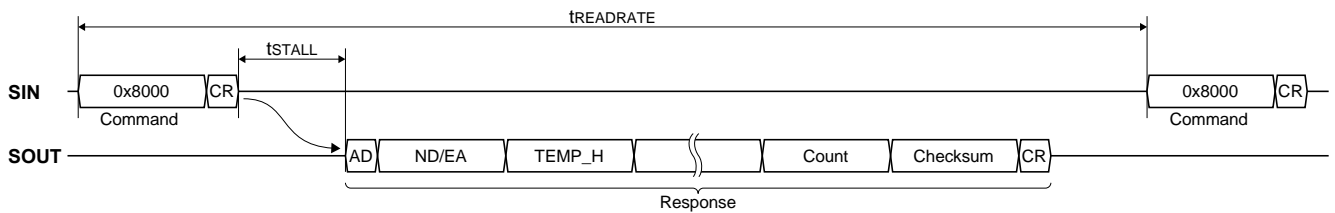


Figure 5.6 UART Read Timing (Burst Mode)

Table 5.12 Command Format (Burst Mode)

First byte								Second byte								Third byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0x80								0x00								0x0D							

0x80: Burst Command

0x00: Burst Data 0x00

0x0D: Delimiter

### 5.2.3 UART Write Timing

A write command to a register will have no response. Unlike register reading, registers are written in 8-bit words.

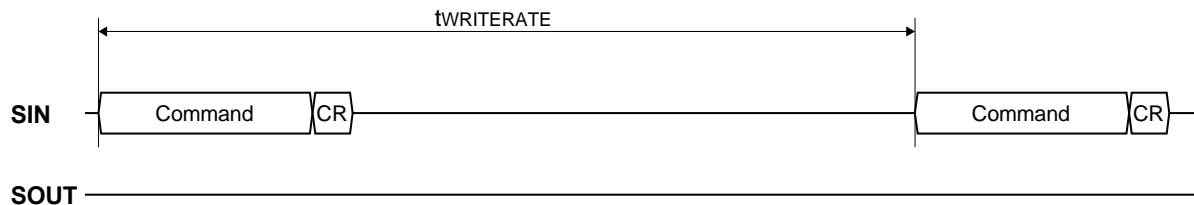


Figure 5.7 UART Write Timing

Table 5.13 Command Format (Write)

First byte								Second byte								Third byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	A[6:0]							D[7:0]								0x0D							

A[6:0]: Register address (even number or odd number)

D[7:0]: Register write data

0x0D: Delimiter

### 5.2.4 UART Auto Mode Operation

When UART Auto mode is active, all sensor outputs are sent as burst transfer automatically at the programmed output data rate without the request from the Host. For information about the response format, see 5.3 *Data Packet Format*. The response format for the burst read output data is configured by register setting in BURST\_CTRL1[0x0C (W1)] and BURST\_CTRL2[0x0E (W1)].



Figure 5.8 UART Auto Mode Operation



### 5.3 Data Packet Format

The following Table 5.14 and Table 5.15 show examples of the data packet format sent to the host in UART Burst mode or UART Auto mode. Table 5.16 and Table 5.17 show the data packet formats sent to the host in SPI Burst mode.

**Table 5.14 UART Data Packet Format (UART Burst / Auto Mode) Example: 16-bit Output**

BURST\_CTRL1[0x0C (W1)] = 0xF007 / BURST\_CTRL2[0x0E (W1)] = 0x0000

Byte No.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	ADDRESS	0x80							
2	ND	ND (Temp)	ND (XGyro)	ND (YGyro)	ND (ZGyro)	ND (XACCL)	ND (YACCL)	ND (ZACCL)	RO
3	EA	ND (XDLTA)	ND (YDLTA)	ND (ZDLTA)	ND (XDLTV)	ND (YDLTV)	ND (ZDLTV)	–	EA
4	TEMP_HIGH_H	TEMP_HIGH[15:8]							
5	TEMP_HIGH_L	TEMP_HIGH[7:0]							
6	XGYRO_HIGH_H	XGYRO_HIGH[15:8]							
7	XGYRO_HIGH_L	XGYRO_HIGH[7:0]							
8	YGYRO_HIGH_H	YGYRO_HIGH[15:8]							
9	YGYRO_HIGH_L	YGYRO_HIGH[7:0]							
10	ZGYRO_HIGH_H	ZGYRO_HIGH[15:8]							
11	ZGYRO_HIGH_L	ZGYRO_HIGH[7:0]							
12	XACCL_HIGH_H	XACCL_HIGH[15:8]							
13	XACCL_HIGH_L	XACCL_HIGH[7:0]							
14	YACCL_HIGH_H	YACCL_HIGH[15:8]							
15	YACCL_HIGH_L	YACCL_HIGH[7:0]							
16	ZACCL_HIGH_H	ZACCL_HIGH[15:8]							
17	ZACCL_HIGH_L	ZACCL_HIGH[7:0]							
18	GPIO_H	–	–	–	–	–	–	GPIO _DATA2	GPIO _DATA1
19	GPIO_L	–	–	–	–	–	–	GPIO _DIR2	GPIO _DIR1
20	COUNT_H	COUNT[15:8]							
21	COUNT_L	COUNT[7:0]							
22	CHECKSUM_H	CHECKSUM[15:8]							
23	CHECKSUM_L	CHECKSUM[7:0]							
24	CR	0x0D							

Table 5.15 UART Data Packet Format (UART Burst / Auto Mode) Example: 32-bit Output

BURST\_CTRL1[0x0C (W1)] = 0xF007 / BURST\_CTRL2[0x0E (W1)] = 0x7000

Byte No.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	ADDRESS	0x80							
2	ND	ND (Temp)	ND (XGyro)	ND (YGyro)	ND (ZGyro)	ND (XACCL)	ND (YACCL)	ND (ZACCL)	RO
3	EA	ND (XDLTA)	ND (YDLTA)	ND (ZDLTA)	ND (XDLTV)	ND (YDLTV)	ND (ZDLTV)	–	EA
4	TEMP_HIGH_H	TEMP_HIGH[15:8]							
5	TEMP_HIGH_L	TEMP_HIGH[7:0]							
6	TEMP_LOW_H	TEMP_LOW[15:8]							
7	TEMP_LOW_L	TEMP_LOW[7:0]							
8	XGYRO_HIGH_H	XGYRO_HIGH[15:8]							
9	XGYRO_HIGH_L	XGYRO_HIGH[7:0]							
10	XGYRO_LOW_H	XGYRO_LOW[15:8]							
11	XGYRO_LOW_L	XGYRO_LOW[7:0]							
12	YGYRO_HIGH_H	YGYRO_HIGH[15:8]							
13	YGYRO_HIGH_L	YGYRO_HIGH[7:0]							
14	YGYRO_LOW_H	YGYRO_LOW[15:8]							
15	YGYRO_LOW_L	YGYRO_LOW[7:0]							
16	ZGYRO_HIGH_H	ZGYRO_HIGH[15:8]							
17	ZGYRO_HIGH_L	ZGYRO_HIGH[7:0]							
18	ZGYRO_LOW_H	ZGYRO_LOW[15:8]							
19	ZGYRO_LOW_L	ZGYRO_LOW[7:0]							
20	XACCL_HIGH_H	XACCL_HIGH[15:8]							
21	XACCL_HIGH_L	XACCL_HIGH[7:0]							
22	XACCL_LOW_H	XACCL_LOW[15:8]							
23	XACCL_LOW_L	XACCL_LOW[7:0]							
24	YACCL_HIGH_H	YACCL_HIGH[15:8]							
25	YACCL_HIGH_L	YACCL_HIGH[7:0]							
26	YACCL_LOW_H	YACCL_LOW[15:8]							
27	YACCL_LOW_L	YACCL_LOW[7:0]							
28	ZACCL_HIGH_H	ZACCL_HIGH[15:8]							
29	ZACCL_HIGH_L	ZACCL_HIGH[7:0]							
30	ZACCL_LOW_H	ZACCL_LOW[15:8]							
31	ZACCL_LOW_L	ZACCL_LOW[7:0]							
32	GPIO_H	–	–	–	–	–	–	GPIO _DATA2	GPIO _DATA1
33	GPIO_L	–	–	–	–	–	–	GPIO _DIR2	GPIO _DIR1
34	COUNT_H	COUNT[15:8]							
35	COUNT_L	COUNT[7:0]							
36	CHECKSUM_H	CHECKSUM[15:8]							
37	CHECKSUM_L	CHECKSUM[7:0]							
38	CR	0x0D							

**Table 5.16 Data Packet Format (SPI Burst Mode) Example: 16-bit Output**

BURST\_CTRL1[0x0C (W1)] = 0xF007 / BURST\_CTRL2[0x0E (W1)] = 0x0000

Word No.	Bit 15	• • •	Bit 0
1		FLAG (ND / EA)	
2		TEMP_HIGH	
3		XGYRO_HIGH	
4		YGYRO_HIGH	
5		ZGYRO_HIGH	
6		XACCL_HIGH	
7		YACCL_HIGH	
8		ZACCL_HIGH	
9		GPIO	
10		COUNT	
11		CHECKSUM	

**Table 5.17 Data Packet Format (SPI Burst Mode) Example: 32-bit Output**

BURST\_CTRL1[0x0C (W1)] = 0xF007 / BURST\_CTRL2[0x0E (W1)] = 0x7000

Word No.	Bit 15	• • •	Bit 0
1		FLAG (ND / EA)	
2		TEMP_HIGH	
3		TEMP_LOW	
4		XGYRO_HIGH	
5		XGYRO_LOW	
6		YGYRO_HIGH	
7		YGYRO_LOW	
8		ZGYRO_HIGH	
9		ZGYRO_LOW	
10		XACCL_HIGH	
11		XACCL_LOW	
12		YACCL_HIGH	
13		YACCL_LOW	
14		ZACCL_HIGH	
15		ZACCL_LOW	
16		GPIO	
17		COUNT	
18		CHECKSUM	

## 6. User Registers

A host device (for example, a microcontroller) can control the IMU by accessing the control registers inside the device.

The registers are accessed in this device using a WINDOW method. The prescribed window number is first written to **WINDOW\_ID** of WIN\_CTRL[0x7E (W0/W1)] bit[7:0], then the desired register address can be accessed. The WIN\_CTRL[0x7E (W0/W1)] register can always be accessed without needing to set the window number.

During the Power-On Start-Up Time or the Reset Recovery Time specified in Table 1.4 Interface Specifications, all the register values are undefined because internal initialization is in progress. Ensure the IMU registers are only accessed after the Power-On Start-Up Time is over.

For information about the initial values of the control registers after internal initialization is finished, see the “Default” column in Table 6.1. The control registers with O mark in the “Flash Backup” column can be saved to the non-volatile memory by the user, and the initial values after the power on will be the values read from the non-volatile memory. If the read out from the non-volatile memory fails, the **FLASH\_ERR** (DIAG\_STAT[0x04 (W0)] bit[2]) is set to “1” (error).

Please ensure that the IMU is in the Configuration mode before writing to registers. In the Sampling Mode, writing to registers is ignored except for the following cases.

- Writing to **MODE\_CMD** (MODE\_CTRL[0x02 (W0)] bit[9:8])
- Writing to **GPIO\_DATA** (GPIO[0x08 (W0)] bit[9:8])
- Writing to **SOFT\_RST** (GLOB\_CMD[0x0A (W1)] bit[7])
- Writing to **WINDOW\_ID** (WIN\_CTRL[0x7E (W0/W1)] bit[7:0])

While in the UART Auto mode and Sampling mode is active, register read access is not supported. Otherwise, the sampling data transmitted in the UART Auto mode will be corrupted by the response data from the register read.

Each register is 16-bit wide and one address is assigned to every 8 bits. Registers are read in 16-bit words and are written in 8-bit words. The byte order of each 16-bit register is little endian, but the byte order of the 16-bit data transferred over the digital interface is big endian

Table 6.1 shows the register map, and Section 6.1 through Section 6.30 describes the registers in detail.

The “-” sign in the register assignment table in Section 6.1 through Section 6.30 means “reserved”.

Write “0” to reserved bits during a write operation.

During a read operation, a reserved bit can return either “0” or “1” (don’t care).

Writing to a read-only register is prohibited.

NOTE: The explanation of the register notation, for example, MODE\_CTRL[0x02 (W0)] bit[9:8] is as follows:

- **MODE\_CTRL**: Register name
- **[0x02 (W0)]**: First number is the Register Address, (W0) means Window Number “0”
- **bit[9:8]**: Bits 9 to 8

Table 6.1 Register Map

Name	Window ID	Address	R/W	Flash Backup	Default	Function
BURST	0	0x00	W		-	Burst mode
MODE_CTRL	0	0x03, 0x02	R/W		0x0400	Operation mode control
DIAG_STAT	0	0x04	R		0x0000	Diagnostic result
FLAG	0	0x06	R		0x0000	ND flag / EA flag
GPIO	0	0x09, 0x08	R/W		0x0200	GPIO
COUNT	0	0x0A	R		0x0000	Sampling count value
RANGE_OVER	0	0x0C	R		0x0000	Range Over
TEMP_HIGH	0	0x0E	R		0xFFFF	Temperature sensor value High
TEMP_LOW	0	0x10	R		0xFFFF	Temperature sensor value Low
XGYRO_HIGH	0	0x12	R		0xFFFF	X gyroscope sensor value High
XGYRO_LOW	0	0x14	R		0xFFFF	X gyroscope sensor value Low
YGYRO_HIGH	0	0x16	R		0xFFFF	Y gyroscope sensor value High
YGYRO_LOW	0	0x18	R		0xFFFF	Y gyroscope sensor value Low
ZGYRO_HIGH	0	0x1A	R		0xFFFF	Z gyroscope sensor value High
ZGYRO_LOW	0	0x1C	R		0xFFFF	Z gyroscope sensor value Low
XACCL_HIGH	0	0x1E	R		0xFFFF	X acceleration sensor value High
XACCL_LOW	0	0x20	R		0xFFFF	X acceleration sensor value Low
YACCL_HIGH	0	0x22	R		0xFFFF	Y acceleration sensor value High
YACCL_LOW	0	0x24	R		0xFFFF	Y acceleration sensor value Low
ZACCL_HIGH	0	0x26	R		0xFFFF	Z acceleration sensor value High
ZACCL_LOW	0	0x28	R		0xFFFF	Z acceleration sensor value Low
ID	0	0x4C	R		0x5345	ID read function
XDLTA_HIGH	0	0x64	R		0x0000	X delta angle value High
XDLTA_LOW	0	0x66	R		0x0000	X delta angle value Low
YDLTA_HIGH	0	0x68	R		0x0000	Y delta angle value High
YDLTA_LOW	0	0x6A	R		0x0000	Y delta angle value Low
ZDLTA_HIGH	0	0x6C	R		0x0000	Z delta angle value High
ZDLTA_LOW	0	0x6E	R		0x0000	Z delta angle value Low
XDLTV_HIGH	0	0x70	R		0x0000	X delta velocity value High
XDLTV_LOW	0	0x72	R		0x0000	X delta velocity value Low
YDLTV_HIGH	0	0x74	R		0x0000	Y delta velocity value High
YDLTV_LOW	0	0x76	R		0x0000	Y delta velocity value Low
ZDLTV_HIGH	0	0x78	R		0x0000	Z delta velocity value High
ZDLTV_LOW	0	0x7A	R		0x0000	Z delta velocity value Low
SIG_CTRL	1	0x01, 0x00	R/W	○	0xFE00	Data Ready signal & polarity control
MSC_CTRL	1	0x03, 0x02	R/W	○	0x0006	Other control
SMPL_CTRL	1	0x05, 0x04	R/W	○	0x0103	Sampling control
FILTER_CTRL	1	0x07, 0x06	R/W	○	0x0001	Filter control
UART_CTRL	1	0x09, 0x08	R/W	○	0x0000	UART control
GLOB_CMD	1	0x0B, 0x0A	R/W	○	0x0000	System control
BURST_CTRL1	1	0x0D, 0x0C	R/W	○	0xF006	Burst control 1
BURST_CTRL2	1	0x0F, 0x0E	R/W	○	0x0000	Burst control 2
POL_CTRL	1	0x11, 0x10	R/W	○	0x0000	Polarity control
GLOB_CMD3	1	0x13, 0x12	R/W	○	0x00CC	Acceleration range / Delta control
ATTI_CTRL	1	0x15, 0x14	R/W	○	0x0000	Delta Angle / Delta Velocity output control
GLOB_CMD2	1	0x17, 0x16	R/W		0x0000	System control2
R_MATRIX_M11 *1	1	0x39, 0x38	R/W	◎	0x4000	Gyro R_Matrix coefficient
R_MATRIX_M12 *1	1	0x3B, 0x3A	R/W	◎	0x0000	Gyro R_Matrix coefficient
R_MATRIX_M13 *1	1	0x3D, 0x3C	R/W	◎	0x0000	Gyro R_Matrix coefficient
R_MATRIX_M21 *1	1	0x3F, 0x3E	R/W	◎	0x0000	Gyro R_Matrix coefficient
R_MATRIX_M22 *1	1	0x41, 0x40	R/W	◎	0x4000	Gyro R_Matrix coefficient
R_MATRIX_M23 *1	1	0x43, 0x42	R/W	◎	0x0000	Gyro R_Matrix coefficient
R_MATRIX_M31 *1	1	0x45, 0x44	R/W	◎	0x0000	Gyro R_Matrix coefficient

Name	Window ID	Address	R/W	Flash Backup	Default	Function
R_MATRIX_M32 * <sup>1</sup>	1	0x47, 0x46	R/W	◎	0x0000	Gyro R_Matrix coefficient
R_MATRIX_M33 * <sup>1</sup>	1	0x49, 0x48	R/W	◎	0x4000	Gyro R_Matrix coefficient
PROD_ID1	1	0x6A	R		* <sup>3</sup>	Product ID
PROD_ID2	1	0x6C	R		* <sup>3</sup>	Product ID
PROD_ID3	1	0x6E	R		* <sup>3</sup>	Product ID
PROD_ID4	1	0x70	R		* <sup>3</sup>	Product ID
VERSION * <sup>2</sup>	1	0x72	R		* <sup>3</sup>	Version
SERIAL_NUM1	1	0x74	R		* <sup>3</sup>	Serial number
SERIAL_NUM2	1	0x76	R		* <sup>3</sup>	Serial number
SERIAL_NUM3	1	0x78	R		* <sup>3</sup>	Serial number
SERIAL_NUM4	1	0x7A	R		* <sup>3</sup>	Serial number
WIN_CTRL	0,1	0x7F, 0x7E	R/W		0x0000	Register window control

\*1 The R\_MATRIX\_\*\_ coefficient values are stored in non-volatile memory using GLOB\_CMD2[0x17 (W1)] bit[8].

\*2 Version is subject to change without notice.

\*3 Refer to each register for the default value.

## 6.1 BURST Register (Window 0)

Addr (Hex)	Bit 15	· · ·	Bit 0	R/W
0x01	-			-

Addr (Hex)	Bit 7	· · ·	Bit 0	R/W
0x00	BURST_CMD			W

### bit[7:0] BURST\_CMD

A burst mode read operation is initiated by writing 0x00 in **BURST\_CMD** of this register.

NOTE: The data transmission format is described in 5.1.3 *SPI Read Timing (Burst Mode)* and 5.2.2 *UART Read Timing (Burst Mode)*. Also refer to 5.3 *Data Packet Format*. The output data can be selected by setting BURST\_CTRL1[0x0C (W1)] and BURST\_CTRL2[0x0E (W1)].

## 6.2 MODE\_CTRL Register (Window 0)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x03	-					MODE_STAT	MODE_CMD		R/W *1

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x02	-								-

\*1) Only **MODE\_STAT** is read-only.

### bit[10] MODE\_STAT

This read-only status bit shows the current operation mode.

- 1: Configuration mode
- 0: Sampling mode

### bit[9:8] MODE\_CMD

Executes commands related to the operation mode.

- 01: Go to Sampling mode. After the mode transition is completed, the bits automatically go back to "00".
- 10: Go to Configuration mode. After the mode transition is completed, the bits automatically go back to "00".
- 11: (Not used)
- 00: (Not used)

## 6.3 DIAG\_STAT Register (Window 0)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x05	-	ST_ERR (XGyro)	ST_ERR (YGyro)	ST_ERR (ZGyro)	ST_ERR (ACCL)	SET_ERR	DLTA_OVF	DLTV_OVF	R

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x04	-	HARD_ERR		SPI_OVF	UART_OVF	FLASH_ERR	ST_ERR_ALL	FLASH_BU_ERR	R

NOTE: When the host reads the diagnosis result, all the results (including the EA flag in the FLAG register) will be cleared to 0.

### bit[14:11] ST\_ERR (SelfTest ERROR)

Shows the result of **SELF\_TEST** (internal self test) of MSC\_CTRL[0x02 (W1)] bit[10].

- 1: Error occurred
- 0: No error

### bit[10] SET\_ERR (SET ERROR)

Shows that a SET Error condition has occurred.

- 1: Error occurred
- 0: No error

SET Error condition occurs when Delta Angle / Delta Velocity is disabled and an invalid combination of output rate setting (SMPL\_CTRL[0x05 (W1)] bit[11:8]) and filter setting (FILTER\_CTRL[0x06 (W1)] bit[4:0]) is detected.

- bit[9] DLTA\_OVF (DeLTa Angle Over Flow)**  
Shows an overflow error condition of the Delta Angle.  
1: Error occurred  
0: No error
- bit[8] DLTv\_OVF (DeLTa V Over Flow)**  
Shows an overflow error condition of the Delta Velocity.  
1: Error occurred  
0: No error
- bit[6:5] HARD\_ERR (HARD ERRor)**  
Shows the result of the hardware check at startup.  
Other than 00: Error occurred  
00: No error  
When this error occurs, it indicates the IMU is faulty.
- bit[4] SPI\_OVF (SPI Over Flow)**  
Shows an error occurred if the device received too many commands from the SPI interface in a short period of time.  
1: Error occurred  
0: No error  
When this error occurs, review the SPI command transmission interval and the SPI clock setting.
- bit[3] UART\_OVF (UART Over Flow)**  
Shows an error occurred if the data transmission rate is faster than the UART baud rate.  
1: Error occurred  
0: No error  
When this error occurs, review the settings for baud rate, data output rate, UART Burst / Auto mode in combination. Refer to **BAUD\_RATE** of UART\_CTRL[0x08 (W1)] bit[9:8], **DOUT\_RATE** of SMPL\_CTRL[0x04 (W1)] bit[15:8]. If using Burst mode with UART Auto mode, also review BURST\_CTRL1[0x0C (W1)] and BURST\_CTRL2[0x0E (W1)] settings.
- bit[2] FLASH\_ERR (FLASH ERRor)**  
Shows the result of **FLASH\_TEST** of MSC\_CTRL[0x02 (W1)] bit[11].  
1: Error occurred  
0: No error  
This error indicates a failure occurred when reading data out from the non-volatile memory.
- bit[1] ST\_ERR\_ALL (SelfTest ERRor All)**  
Shows the logical sum of bit[14:11] of this register.  
1: Error occurred  
0: No error
- bit[0] FLASH\_BU\_ERR (FLASH BackUp ERRor)**  
Shows the result of **FLASH\_BACKUP** of GLOB\_CMD[0x0A (W1)] bit[3] or **FLASH\_ROTATION\_BACKUP** of GLOB\_CMD2[0x017 (W1)] bit[8].  
1: Error occurred  
0: No error



## 6.4 FLAG (ND / EA) Register (Window 0)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x07	ND (Temp)	ND (XGyro)	ND (YGyro)	ND (ZGyro)	ND (XACCL)	ND (YACCL)	ND (ZACCL)	RO	R
Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x06	ND (XDLTA)	ND (YDLTA)	ND (ZDLTA)	ND (XDLTV)	ND (YDLTV)	ND (ZDLTV)	-	EA	R

### bit[15:9] ND (New Data) flag (Temperature, Gyroscope, Acceleration)

When a new measuring data is set in each register of temperature (TEMP\_HIGH), gyroscope (XGYRO\_HIGH, YGYRO\_HIGH, ZGYRO\_HIGH), and acceleration (XACCL\_HIGH, YACCL\_HIGH, ZACCL\_HIGH), the corresponding ND flag is set to "1". When the measurement output is read from the corresponding register, the flag is reset to "0".

### bit[8] RO (Range Over) flag

When at least one over range condition is detected in RANGE\_OVER[0x0C (W0)], this flag is set to "1".

### bit[7:2] ND (New Data) flag (Delta Angle, Delta Velocity)

When a new measuring data is set in each register of Delta Angle (XDLTA\_HIGH, YDLTA\_HIGH, ZDLTA\_HIGH), and Delta Velocity (XDLTV\_HIGH, YDLTV\_HIGH, ZDLTV\_HIGH), the corresponding ND flag is set to "1". When the measurement output is read from the corresponding register, the flag is reset to "0".

### bit[0] EA (All Error) flag

When at least one failure is found in the diagnostic result (DIAG\_STAT[0x04 (W0)]), the flag is set to "1" (failure occurred).

## 6.5 GPIO Register (Window 0)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x09	-						GPIO_DATA2	GPIO_DATA1	R/W
Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x08	-						GPIO_DIR2	GPIO_DIR1	R/W

### bit[9:8] GPIO\_DATA

If the corresponding GPIO\_DIR bit is set to "output", the value set in the GPIO\_DATA is output to the GPIO port.  
If the corresponding GPIO\_DIR bit is set to "input", the input level of the GPIO port is returned by reading the GPIO\_DATA.

1: High Level  
0: Low Level

### bit[1:0] GPIO\_DIR

Each bit controls the bitwise direction of the GPIO port.

1: Output  
0: Input

- NOTE:
- GPIO1 is shared with the Data Ready signal function on the same pin. The selection between GPIO1 and the Data Ready signal is controlled with **DRDY\_ON** of MSC\_CTRL[0x02 (W1)] bit[2]. The pin functions as GPIO1 when **DRDY\_ON** is "0" (disabled).
  - GPIO2 is shared with the EXT signal input function (External Trigger Input and External Counter Reset Input) on the same pin. The selection between GPIO2 and the EXT signal input is controlled with **EXT\_SEL** of MSC\_CTRL[0x02 (W1)] bit[7:6]. The pin functions as GPIO2 when **EXT\_SEL** is "00" (GPIO2).

## 6.6 COUNT Register (Window 0)

Addr (Hex)	bit 15	...	bit 0	R/W
0x0A	COUNT			R

### bit[15:0] COUNT

The value returned by this register depends on whether the External Counter Reset Input function is enabled or not. The External Counter Reset Input is enabled when **EXT\_SEL** of MSC\_CTRL[0x02 (W1)] bit[7:6] = "01".

When the External Counter Reset Input function is disabled, this register returns the sampling count value of the internal A/D converter.

NOTE: The time unit of the sampling counter value represents 500 μs/count.

Example: If the data output rate equals 1000 Sps, the counter value sequence is 0, 2, 4, 6, ... , 0xFFFE, 0, 2, ...

When the External Counter Reset Input function is enabled, this register returns the timer counter value used by the External Counter Reset Input function.

## 6.7 RANGE\_OVER Register (Window 0)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x0D	-		RO (XGyro)	RO (YGyro)	RO (ZGyro)	RO (XACCL)	RO (YACCL)	RO (ZACCL)	R

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x0C	-								R

### bit[13:8] RO (Range Over) Flag (Gyroscope / Acceleration)

The specified gyroscope or acceleration sensor axis RO flag is set to "1" when the output value exceeds the sensing range. The flags are reset to "0" after reading this register.

## 6.8 TEMP Register (Window 0)

Addr (Hex)	bit 15	...	bit 0	R/W
0x0E	TEMP_HIGH			R
0x10	TEMP_LOW			R

### bit[15:0] Temperature sensor output data

The internal temperature sensor value can be read from this register.

The output data format is 32-bit two's complement format. For 16-bit usage, treat the data as 16-bit two's complement using the upper 16 bits (**TEMP\_HIGH**).

Please refer to the below formula for conversion to temperature in centigrade. Please refer to Table 1.3 Sensor Specifications for the scale factor value.

For 32-bit usage:  $T [^{\circ}\text{C}] = (\text{SF} / 65536) \times A + 25$

For 16-bit usage:  $T [^{\circ}\text{C}] = \text{SF} \times A + 25$

SF: Scale Factor

A: Temperature sensor output data (decimal)

NOTE: The reference value in this register is for the temperature correction. There is no guarantee that the value provides the absolute value of the internal temperature.

## 6.9 GYRO Register (Window 0)

Addr (Hex)	bit 15	...	bit 0	R/W
0x12		XGYRO_HIGH		R
0x14		XGYRO_LOW		R
0x16		YGYRO_HIGH		R
0x18		YGYRO_LOW		R
0x1A		ZGYRO_HIGH		R
0x1C		ZGYRO_LOW		R

### bit[15:0] Gyroscope output data

Returns the 3-axis gyroscope data for X, Y, and Z as referenced in Figure 2.1 M-G370PDG0 Outline Dimensions.

The output data format is 32-bit two's complement. For 16-bit usage, treat the data as 16-bit two's complement using the upper 16 bits.

Please refer to Table 1.3 Sensor Specifications for the scale factor value.

For 32-bit usage:  $G \text{ [deg/s]} = ((1 / SF) / 65536) \times B$

For 16-bit usage:  $G \text{ [deg/s]} = (1 / SF) \times B$

SF: Scale Factor

B: Gyroscope output data (decimal)

## 6.10 ACCL Register (Window 0)

Addr (Hex)	bit 15	...	bit 0	R/W
0x1E		XACCL_HIGH		R
0x20		XACCL_LOW		R
0x22		YACCL_HIGH		R
0x24		YACCL_LOW		R
0x26		ZACCL_HIGH		R
0x28		ZACCL_LOW		R

### bit[15:0] Acceleration sensor output data

Returns the 3-axis acceleration data for X, Y, and Z as referenced in Figure 2.1 M-G370PDG0 Outline Dimensions.

The output data format is 32-bit two's complement. For 16-bit usage, treat the data as 16-bit two's complement using the upper 16 bits.

Please refer to Table 1.3 Sensor Specifications for the scale factor value.

For 32-bit usage:  $A \text{ [mG]} = ((1 / SF) / 65536) \times C$

For 16-bit usage:  $A \text{ [mG]} = (1 / SF) \times C$

SF: Scale Factor

C: Acceleration sensor output data (decimal)

## 6.11 ID Register (Window 0)

Addr (Hex)	bit 15	...	bit 0	R/W
0x4C		ID		R

### bit[15:0] ID data

This register will return the value "0x5345" when read.

## 6.12 DELTA\_ANGLE Register (Window 0)

Addr (Hex)	bit 15	...	bit 0	R/W
0x64	XDLTA_HIGH			R
0x66	XDLTA_LOW			R
0x68	YDLTA_HIGH			R
0x6A	YDLTA_LOW			R
0x6C	ZDLTA_HIGH			R
0x6E	ZDLTA_LOW			R

### bit[15:0] Delta Angle (Gyroscope) output data

Returns the Delta Angle output format from the 3-axis gyroscope data for X, Y, and Z as referenced in Figure 2.1 M-G370PDG0 Outline Dimensions.

The output data format is 32-bit two's complement. For 16-bit usage, treat the data as 16-bit two's complement using the upper 16 bits.

Please refer to **DLTA\_RANGE\_CTRL** of GLOB\_CMD3[0x12 (W1)] bit[7:4] for the appropriate delta angle scale factor value.

For 32-bit usage:  $\Delta\text{Angle} [\text{deg}] = (\text{SF} / 65536) \times D$

For 16-bit usage:  $\Delta\text{Angle} [\text{deg}] = \text{SF} \times D$

SF: Scale Factor (Refer to **DLTA\_RANGE\_CTRL**)

D: Gyroscope delta angle output data (decimal)

## 6.13 DELTA\_VELOCITY Register (Window 0)

Addr (Hex)	bit 15	...	bit 0	R/W
0x70	XDLTV_HIGH			R
0x72	XDLTV_LOW			R
0x74	YDLTV_HIGH			R
0x76	YDLTV_LOW			R
0x78	ZDLTV_HIGH			R
0x7A	ZDLTV_LOW			R

### bit[15:0] Delta Velocity (Acceleration) output data

Returns the Delta Velocity output format from the 3-axis acceleration data for X, Y, and Z as referenced in Figure 2.1 M-G370PDG0 Outline Dimensions.

The output data format is 32-bit two's complement. For 16-bit usage, treat the data as 16-bit two's complement using the upper 16 bits.

Please refer to **DLTV\_RANGE\_CTRL** of GLOB\_CMD3[0x12 (W1)] bit[3:0] for the appropriate delta velocity scale factor value.

For 32-bit usage:  $\Delta\text{Velocity} [\text{m/s}] = (\text{SF} / 65536) \times E$

For 16-bit usage:  $\Delta\text{Velocity} [\text{m/s}] = \text{SF} \times E$

SF: Scale Factor (Refer to **DLTV\_RANGE\_CTRL**)

E: Acceleration delta velocity output data (decimal)

## 6.14 SIG\_CTRL Register (Window 1)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x01	ND_EN (Temp)	ND_EN (XGyro)	ND_EN (YGyro)	ND_EN (ZGyro)	ND_EN (XACCL)	ND_EN (YACCL)	ND_EN (ZACCL)	-	R/W

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x00	ND_EN (XDLTA)	ND_EN (YDLTA)	ND_EN (ZDLTA)	ND_EN (XDLTV)	ND_EN (YDLTV)	ND_EN (ZDLTV)	-	-	R/W

**bit[15:9] ND\_EN (Temperature, Gyroscope, Acceleration)**

Enables or disables the ND flags in FLAG[0x06 (W0)] bit[15:9].

- 1: Enable
- 0: Disable

**bit[7:2] ND\_EN (Delta Angle, Delta Velocity)**

Enables or disables the ND flags in FLAG[0x06 (W0)] bit[7:2].

- 1: Enable
- 0: Disable

NOTE: **ND\_EN** settings are ignored and have no effect when UART Auto mode is active.

**6.15 MSC\_CTRL Register (Window 1)**

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x03	-				FLASH_TEST	SELF_TEST	-		R/W

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x02	EXT_SEL		-			DRDY_ON	DRDY_POL	-	R/W

NOTE: The FLASH\_TEST, and SELF\_TEST functions can not be executed at the same time. When executing them in succession, confirm the execution of the previous command is finished by waiting until the bit changes from “1” to “0” and then execute the next command.

**bit[11] FLASH\_TEST**

Write “1” to execute the data consistency test for the non-volatile memory. The read value of the bit is “1” during the test and “0” after the test is completed. After writing “1” to this bit, wait until this bit goes back to “0” and then read the **FLASH\_ERR** of DIAG\_STAT[0x04 (W0)] bit[2] to check the result.

**bit[10] SELF\_TEST**

Write “1” to execute the self test to check if the gyroscope and the accelerometer are working properly. The read value of the bit is “1” during the test and “0” after the test is completed. After writing “1” to this bit, wait until this bit goes back to “0” and then read the **ST\_ERR\_ALL** of DIAG\_STAT[0x04 (W0)] bit[1] to check the results.

NOTE: When executing the self-test, be sure to disable the external trigger function.

MSC\_CTRL[0x02 (W1)] bit[7:6] = “00” or “01”

**bit[7:6] EXT\_SEL**

These bits select the function of GPIO2 pin to be GPIO2, External Counter Reset Input, or External Trigger Input.

- 00: GPIO2
- 01: External Counter Reset Input
- 10: Invalid
- 11: External Trigger Input

**bit[2] DRDY\_ON**

Selects the function of the GPIO1 pin for either GPIO1 or Data Ready.

- 1: Data Ready Signal
- 0: GPIO1

**bit[1] DRDY\_POL**

Selects the polarity of the Data Ready signal when selected in **DRDY\_ON** above.

- 1: Active High
- 0: Active Low

## 6.16 SMPL\_CTRL Register (Window 1)

Addr (Hex)	bit 15	...	bit 8	R/W
0x05	DOUT_RATE			R/W
Addr (Hex)	bit 7	...	bit 0	R/W
0x04	-			-

**bit[15:8] DOUT\_RATE**

Specifies the data output rate.

The following lists the data output rate option with the recommended number of filter taps when using the moving average filter.

0x00: 2000 Sps	TAP ≥ 0
0x01: 1000 Sps	TAP ≥ 2
0x02: 500 Sps	TAP ≥ 4
0x03: 250 Sps	TAP ≥ 8
0x04: 125 Sps	TAP ≥ 16
0x05: 62.5 Sps	TAP ≥ 32
0x06: 31.25 Sps	TAP ≥ 64
0x07: 15.625 Sps	TAP = 128
0x08: 400 Sps	TAP ≥ 8
0x09: 200 Sps	TAP ≥ 16
0x0A: 100 Sps	TAP ≥ 32
0x0B: 80 Sps	TAP ≥ 32
0x0C: 50 Sps	TAP ≥ 64
0x0D: 40 Sps	TAP ≥ 64
0x0E: 25 Sps	TAP = 128
0x0F: 20 Sps	TAP = 128

- NOTE:
- When Delta Angle / Delta Velocity is enabled, there is no restriction (SET\_ERR event is disabled) on the combination in output rate setting and filter setting.
  - The **SET\_ERR** bit in DIAG\_STAT[0x05 (W0)] bit[10] will indicate an error if an invalid combination (denoted by “-”) in output rate setting (SMPL\_CTRL[0x05 (W1)] bit[11:8]) and filter setting (FILTER\_CTRL[0x06 (W1)] bit[4:0]) is selected as outlined in Table 6.2.  
During a detected SET\_ERR condition, the output registers TEMP\_HIGH–TEMP\_LOW / X\_GYRO\_HIGH–Z\_GYRO\_LOW / X\_ACCL\_HIGH–Z\_ACCL\_LOW will output a fixed value 0x7EF0.

Table 6.2 SET\_ERR Flag Output Rate and Filter Setting Table Evaluation Table

DOUT_RATE[3:0] (SMPL_CTRL[0x05 (W1)] bit[11:8])		FILTER_SEL (Filter_CTRL[0x06 (W1)] bit[4:0])																			
		0	2	4	8	16	32	64	128	32 fc50	32 fc100	32 fc200	32 fc400	64 fc50	64 fc100	64 fc200	64 fc400	128 fc50	128 fc100	128 fc200	128 fc400
Setting	Sps	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK
0x00	2000	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK
0x01	1000	–	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK
0x02	500	–	–	OK	OK	OK	OK	OK	OK	OK	OK	OK	–	OK	OK	OK	–	OK	OK	OK	–
0x08	400	–	–	–	OK	OK	OK	OK	OK	OK	OK	OK	–	OK	OK	OK	–	OK	OK	OK	–
0x03	250	–	–	–	OK	OK	OK	OK	OK	OK	OK	–	–	OK	OK	–	–	OK	OK	–	–
0x09	200	–	–	–	–	OK	OK	OK	OK	OK	OK	–	–	OK	OK	–	–	OK	OK	–	–
0x04	125	–	–	–	–	OK	OK	OK	OK	OK	–	–	–	OK	–	–	–	OK	–	–	–
0x0A	100	–	–	–	–	–	OK	OK	OK	OK	–	–	–	OK	–	–	–	OK	–	–	–
0x0B	80	–	–	–	–	–	OK	OK	OK	–	–	–	–	–	–	–	–	–	–	–	–
0x05	62.5	–	–	–	–	–	OK	OK	OK	–	–	–	–	–	–	–	–	–	–	–	–
0x0C	50	–	–	–	–	–	–	OK	OK	–	–	–	–	–	–	–	–	–	–	–	–
0x0D	40	–	–	–	–	–	–	OK	OK	–	–	–	–	–	–	–	–	–	–	–	–
0x06	31.25	–	–	–	–	–	–	OK	OK	–	–	–	–	–	–	–	–	–	–	–	–
0x0E	25	–	–	–	–	–	–	–	OK	–	–	–	–	–	–	–	–	–	–	–	–
0x0F	20	–	–	–	–	–	–	–	OK	–	–	–	–	–	–	–	–	–	–	–	–
0x07	15.625	–	–	–	–	–	–	–	OK	–	–	–	–	–	–	–	–	–	–	–	–

OK = Supported, - = Invalid

NOTE: There are limitations on the supported output features depending on the output rate as shown in the following table.

Table 6.3 Output Rate Limitations

Output Rate (Sps)	Gyro [0x12–0x1D (W0)] Accl [0x1C–0x29 (W0)]	Rotation Matrix [0x38–0x5B (W0)]	Delta [0x64–0x7B (W0)]	Range Over [0x0C, 0x0D (W0)]
2000	OK	OK	–	OK
1000	OK	OK	OK	OK
500	OK	OK	OK	OK
400	OK	OK	OK	OK
250	OK	OK	OK	OK
200	OK	OK	OK	OK
125	OK	OK	OK	OK
100	OK	OK	OK	OK
80	OK	OK	OK	OK
62.5	OK	OK	OK	OK
50	OK	OK	OK	OK
40	OK	OK	OK	OK
31.25	OK	OK	OK	OK
25	OK	OK	OK	OK
20	OK	OK	OK	OK
15.625	OK	OK	OK	OK

OK = Supported, – = Invalid

## 6.17 FILTER\_CTRL Register (Window 1)

Addr (Hex)	bit 15	...	bit 8	R/W
0x07	-			-

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x06	-		<b>FILTER_STAT</b>	<b>FILTER_SEL</b>					R/W *1

\*1) Only **FILTER\_STAT** is read-only.

### bit[5] **FILTER\_STAT**

This read-only status bit shows the completion status of the filter selection. After setting the **FILTER\_SEL** in bit[4:0], this status bit will be set to "1". After completion of the filter setting operation, this bit will return to "0".

- 1: Filter setting is busy.
- 0: Filter setting is completed.

### bit[4:0] **FILTER\_SEL**

Specifies the type of filter (moving average filter and FIR Kaiser filter) and TAP setting.  
For the FIR Kaiser filter, these bits also select the cutoff frequency  $f_c$  in Hz.

- 00000: Moving average filter TAP = 0
- 00001: Moving average filter TAP = 2
- 00010: Moving average filter TAP = 4
- 00011: Moving average filter TAP = 8
- 00100: Moving average filter TAP = 16
- 00101: Moving average filter TAP = 32
- 00110: Moving average filter TAP = 64
- 00111: Moving average filter TAP = 128
- 01000: FIR Kaiser filter (parameter = 8) TAP = 32 and  $f_c$  = 50
- 01001: FIR Kaiser filter (parameter = 8) TAP = 32 and  $f_c$  = 100
- 01010: FIR Kaiser filter (parameter = 8) TAP = 32 and  $f_c$  = 200
- 01011: FIR Kaiser filter (parameter = 8) TAP = 32 and  $f_c$  = 400
- 01100: FIR Kaiser filter (parameter = 8) TAP = 64 and  $f_c$  = 50
- 01101: FIR Kaiser filter (parameter = 8) TAP = 64 and  $f_c$  = 100
- 01110: FIR Kaiser filter (parameter = 8) TAP = 64 and  $f_c$  = 200
- 01111: FIR Kaiser filter (parameter = 8) TAP = 64 and  $f_c$  = 400

10000: FIR Kaiser filter (parameter = 8) TAP = 128 and fc = 50  
 10001: FIR Kaiser filter (parameter = 8) TAP = 128 and fc = 100  
 10010: FIR Kaiser filter (parameter = 8) TAP = 128 and fc = 200  
 10011: FIR Kaiser filter (parameter = 8) TAP = 128 and fc = 400  
 10100-11111: Unused

After setting the filter with these bits, the completion of the operation requires the Filter Setting Time specified in Table 1.4 to elapse or confirming completion by checking **FILTER\_STAT** (bit[5]).

NOTE: Refer to 4.13 *Filter* for description of filter transient response from sampling start.

## 6.18 UART\_CTRL Register (Window 1)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x09	-						<b>BAUD_RATE</b>		<b>R/W</b>
Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x08	-						<b>AUTO_START</b>	<b>UART_AUTO</b>	<b>R/W</b>

### bit[9:8] **BAUD\_RATE**

These bits specify the Baud Rate of UART interface.

00: 460.8 kbps  
 01: 230.4 kbps  
 10: 921.6 kbps

NOTE: The baud rate change using these **BAUD\_RATE** bits becomes effective immediately after write access completes.

### bit[1] **AUTO\_START (Only valid for UART Auto mode)**

Enables or disables the Auto Start function.

1: Automatic Start is enabled.  
 0: Automatic Start is disabled.

When Auto Start is enabled, the device enters sampling mode and sends sampling data automatically after completing internal initialization when IMU is powered on or reset.

Write "1" to this **AUTO\_START** bit and the **UART\_AUTO** bit of this register to enable this function. Then execute **FLASH\_BACKUP** of GLOB\_CMD[0x0A (W1)] bit[3] to preserve the current register settings.

### bit[0] **UART\_AUTO**

Enables or disables the UART Auto mode function.

1: UART Auto mode is selected.  
 0: UART Manual mode is selected.

If UART Auto mode is active, burst read register values such as FLAG, temperature, angle rate (XGYRO, YGYRO, ZGYRO), accelerations (XACCL, YACCL, ZACCL), and GPIO are continuously transmitted automatically according to the data output rate set by the SMPL\_CTRL[0x04 (W1)] register.

In UART Manual mode, register data is transmitted as a response to a register read command.

NOTE: • For more information on UART Auto mode, refer to 5.2.4 *UART Auto Mode Operation* and 5.3 *Data Packet Format*. The burst output data is configured by register setting in BURST\_CTRL1[0x0C (W1)] and BURST\_CTRL2[0x0E (W1)].

• This register bit must be set to "0" when using the SPI interface.



## 6.19 GLOB\_CMD Register (Window 1)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x0B	-					NOT_READY	-		R

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x0A	SOFT_RST	-		INITIAL_BACKUP	FLASH_BACKUP	-			R/W

### bit[10] NOT\_READY

Indicates whether the IMU is currently ready. Immediately after power on, this bit is "1" and becomes "0" when the IMU is ready. After the power on, wait until the Power-On Start-Up Time has elapsed and then wait until this bit becomes "0" before starting sensor measurement. This bit is read-only.

1: Not ready  
0: Ready


### bit[7] SOFT\_RST

Write "1" to execute software reset. After the software reset is completed, the bit automatically goes back to "0".

### bit[4] INITIAL\_BACKUP

Write "1" to set the non-volatile memory for the registers to the factory default value. After the execution is completed, the bit automatically goes back to "0". After confirming this bit goes back to "0", check the result in **FLASH\_BU\_ERR** of **DIAG\_STAT[0x04 (W0)]** bit[0]. The values saved in the non-volatile memory are reflected in the "Flash Backup" target registers after power on or a software reset.

### bit[3] FLASH\_BACKUP

Write "1" to save the current values of the control registers with the  mark in the "Flash Backup" column of Table 6.1 to the non-volatile memory. After the execution is completed, the bit automatically goes back to "0". After confirming this bit goes back to "0", check the result in **FLASH\_BU\_ERR** of **DIAG\_STAT[0x04 (W0)]** bit[0].

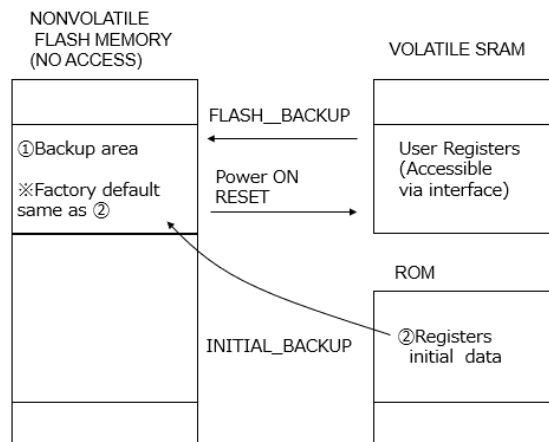


Figure 6.1 SRAM and Flash Memory Diagram

## 6.20 BURST\_CTRL1 Register (Window 1)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x0D	FLAG_OUT	TEMP_OUT	GYRO_OUT	ACCL_OUT	DLTA_OUT	DLTV_OUT	-		R/W

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x0C	-					GPIO_OUT	COUNT_OUT	CHKSM_OUT	R/W

These bits enable/disable the content in the output data for Burst mode and UART Auto mode.

### bit[15] FLAG\_OUT

Controls the output of FLAG status.

- 1: Enables output.  
0: Disables output.

**bit[14] TEMP\_OUT**  
Controls the output of temperature sensor.

- 1: Enables output.  
0: Disables output.

**bit[13] GYRO\_OUT**  
Controls the output of gyroscope sensor.

- 1: Enables output.  
0: Disables output.

**bit[12] ACCL\_OUT**  
Controls the output of acceleration sensor.

- 1: Enables output.  
0: Disables output.

**bit[11] DLTA\_OUT**  
Controls the output of delta angle.

- 1: Enables output.  
0: Disables output.

**bit[10] DLTV\_OUT**  
Controls the output of delta velocity.

- 1: Enables output.  
0: Disables output.

**bit[2] GPIO\_OUT**  
Controls the output of GPIO status.

- 1: Enables output.  
0: Disables output.

**bit[1] COUNT\_OUT**  
Controls the output of counter value.

- 1: Enables output.  
0: Disables output.

**bit[0] CHKSM\_OUT**  
Controls the output of checksum.

- 1: Enables output.  
0: Disables output.

## 6.21 BURST\_CTRL2 Register (Window 1)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x0F	-	TEMP_BIT	GYRO_BIT	ACCL_BIT	DLTA_BIT	DLTV_BIT	-		R/W
Addr (Hex)	bit 7	...						bit 0	R/W
0x0E		-							-

These bits select the output bit length of output data for Burst mode and UART Auto mode.

**bit[14] TEMP\_BIT**  
Selects the bit length of the temperature output.

- 1: 32 bits  
0: 16 bits

- bit[13] GYRO\_BIT**  
Selects the bit length of the gyroscope output.
- 1: 32 bits  
0: 16 bits
- bit[12] ACCL\_BIT**  
Selects the bit length of the acceleration output.
- 1: 32 bits  
0: 16 bits
- bit[11] DLTA\_BIT**  
Selects the bit length of the delta angle output.
- 1: 32 bits  
0: 16 bits
- bit[10] DLTV\_BIT**  
Selects the bit length of the delta velocity output.
- 1: 32 bits  
0: 16 bits

## 6.22 POL\_CTRL Register (Window 1)

Addr (Hex)	bit 15	...						bit 8	R/W
0x11	-								-

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x10	-	POL_CTRL (XGyro)	POL_CTRL (YGyro)	POL_CTRL (ZGyro)	POL_CTRL (XACCL)	POL_CTRL (YACCL)	POL_CTRL (ZACCL)	-	-

- bit[6:1] POL\_CTRL**  
Specifies whether to bitwise invert the output value of the following registers: angular rate (XGYRO, YGYRO, ZGYRO) and acceleration (XACCL, YACCL, ZACCL). This bitwise inversion will also have effect in the internal processing for the delta angle (XDLTA, YDLTA, ZDLTA), and delta velocity (XDLTV, YDLTV, ZDLTV).
- 1: Inverted  
0: Not inverted

## 6.23 GLOB\_CMD3 Register (Window 1)

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x13	-							A_RANGE_CTRL	R/W

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x12	DLTA_RANGE_CTRL				DLTV_RANGE_CTRL				R/W

- bit[8] A\_RANGE\_CTRL**  
This bit sets the output range of the accelerometer.
- 1:  $\pm 16$  G  
0:  $\pm 8$  G
- bit[7:4] DLTA\_RANGE\_CTRL**  
These bits specify the scale factor and range of Delta Angle output.
- NOTE: The user must carefully select the desired scale factor to optimize Delta Angle resolution but avoid an overflow condition in the internal Delta Angle processing.
- bit[3:0] DLTV\_RANGE\_CTRL**  
These bits specify the scale factor and range of Delta Velocity output.

NOTE: The user must carefully select the desired scale factor to optimize Delta Velocity resolution but avoid an overflow condition in the internal Delta Velocity processing.

**Table 6.4 Delta Angle: Scale Factor & Range**

<b>DLTA_RANGE_CTRL (bit[7:4])</b>	<b>Scale Factor [deg/LSB]</b>	<b>Range [± deg]</b>
0000	7.576.E-06	2.482E-01
0001	1.515.E-05	4.965E-01
0010	3.030.E-05	9.930E-01
0011	6.061.E-05	1.986E+00
0100	1.212.E-04	3.972E+00
0101	2.424.E-04	7.944E+00
0110	4.848.E-04	1.589E+01
0111	9.697.E-04	3.178E+01
1000	1.939.E-03	6.355E+01
1001	3.879.E-03	1.271E+02
1010	7.758.E-03	2.542E+02
1011	1.552.E-02	5.084E+02
1100	3.103.E-02	1.017E+03
1101	6.206.E-02	2.034E+03
1110	1.241.E-01	4.067E+03
1111	2.482.E-01	8.134E+03

**Table 6.5 Delta Velocity: Scale Factor & Range**

OutputRange ±8 G, **A\_RANGE\_CTRL** of  
GLOB\_CMD3[0x13 (W1)] bit[8] = "0"

<b>DLTV_RANGE_CTRL (bit[3:0])</b>	<b>Scale Factor [(m/s)/LSB]</b>	<b>Range [± (m/s)]</b>
0000	1.226.E-06	4.017E-02
0001	2.452.E-06	8.034E-02
0010	4.903.E-06	1.607E-01
0011	9.807.E-06	3.213E-01
0100	1.961.E-05	6.427E-01
0101	3.923.E-05	1.285E+00
0110	7.845.E-05	2.571E+00
0111	1.569.E-04	5.142E+00
1000	3.138.E-04	1.028E+01
1001	6.276.E-04	2.057E+01
1010	1.255.E-03	4.113E+01
1011	2.511.E-03	8.226E+01
1100	5.021.E-03	1.645E+02
1101	1.004.E-02	3.291E+02
1110	2.008.E-02	6.581E+02
1111	4.017.E-02	1.316E+03

OutputRange ±16 G, **A\_RANGE\_CTRL** of  
GLOB\_CMD3[0x13 (W1)] bit[8] = "1"

<b>DLTV_RANGE_CTRL (bit[3:0])</b>	<b>Scale Factor [(m/s)/LSB]</b>	<b>Range [± (m/s)]</b>
0000	2.452.E-06	8.034E-02
0001	4.903.E-06	1.607E-01
0010	9.807.E-06	3.213E-01
0011	1.961.E-05	6.427E-01
0100	3.923.E-05	1.285E+00
0101	7.845.E-05	2.571E+00
0110	1.569.E-04	5.142E+00
0111	3.138.E-04	1.028E+01
1000	6.276.E-04	2.057E+01
1001	1.255.E-03	4.113E+01
1010	2.511.E-03	8.226E+01
1011	5.021.E-03	1.645E+02
1100	1.004.E-02	3.291E+02
1101	2.008.E-02	6.581E+02
1110	4.017.E-02	1.316E+03
1111	8.034.E-02	2.632E+03

**6.24 ATTI\_CTRL Register (Window 1)**

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x15	-						ATTI_ON	-	R/W

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x14	-								R

**bit[9] ATTI\_ON**

This bit enables or disables the Delta Angle / Delta Velocity output.

0: Disable

1: Delta Angle / Delta Velocity Output

**6.25 GLOB\_CMD2 Register (Window 1)**

Addr (Hex)	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	R/W
0x17	-						INITIAL_ROTATION_BACKUP	FLASH_ROTATION_BACKUP	R/W

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x16	-								R/W

**bit[9] INITIAL\_ROTATION\_BACKUP**

Write "1" to set the non-volatile memory for the R\_MATRIX registers with ☉ in the "Flash Backup" column in Table 6.1 to the factory default value. This bit automatically returns to "0" after execution is completed. The factory default value is reflected in the registers after power on or after a hardware or software reset.

**bit[8] FLASH\_ROTATION\_BACKUP**

Write "1" to save the current values of the R\_MATRIX registers with the ☉ mark in the "Flash Backup" column of Table 6.1 to the non-volatile memory. After the execution is completed, the bit automatically goes back to "0". After confirming this bit goes back to "0", check the result in **FLASH\_BU\_ERR** of **DIAG\_STAT[0x04 (W0)] bit[0]**.

**6.26 R\_MATRIX Register (Window 1)**

Addr (Hex)	bit 15	...	bit 0	R/W
0x38	R_MATRIX_M11			R/W
0x3A	R_MATRIX_M12			R/W
0x3C	R_MATRIX_M13			R/W
0x3E	R_MATRIX_M21			R/W
0x40	R_MATRIX_M22			R/W
0x42	R_MATRIX_M23			R/W
0x44	R_MATRIX_M31			R/W
0x46	R_MATRIX_M32			R/W
0x48	R_MATRIX_M33			R/W

**bit[15:0] Rotation Matrix Coefficient**

The frame alignment of gyroscope triad and acceleration sensor triad can be corrected by using the R\_MATRIX function. Refer to 4.16 *Frame Alignment Correction* for more details.

**6.27 PROD\_ID Register (Window 1)**

Addr (Hex)	bit 15	...	bit 0	R/W
0x6A	PROD_ID1			R
0x6C	PROD_ID2			R
0x6E	PROD_ID3			R
0x70	PROD_ID4			R

**bit[15:0] Product ID**

These registers return the product model number represented in ASCII code.

Product ID return value depends on the model.

PROD_ID	M-G370PDG0
PROD_ID1	0x3347
PROD_ID2	0x3037
PROD_ID3	0x4450
PROD_ID4	0x3047

**6.28 VERSION Register (Window 1)**

Addr (Hex)	bit 15	...	bit 0	R/W
0x72	VERSION			R

**bit[15:0] Version**

This register returns the Firmware Version.

**6.29 SERIAL\_NUM Register (Window 1)**

Addr (Hex)	bit 15	...	bit 0	R/W
0x74	SERIAL_NUM1			R
0x76	SERIAL_NUM2			R
0x78	SERIAL_NUM3			R
0x7A	SERIAL_NUM4			R

**bit[15:0] Serial Number**

These registers return the serial number represented in ASCII code.

For example, if the Serial Number is 01234567 then the return value is:

SERIAL\_NUM1: 0x3130

SERIAL\_NUM2: 0x3332

SERIAL\_NUM3: 0x3534

SERIAL\_NUM4: 0x3736

**6.30 WIN\_CTRL Register (Window 0,1)**

Addr (Hex)	bit 15	...	bit 8	R/W
0x7F	-			-

Addr (Hex)	bit 7	...	bit 0	R/W
0x7E	WINDOW_ID			R/W

**bit[7:0] WINDOW\_ID**

Select the desired register window by writing the window number to this register.

0x00: Window 0

0x01: Window 1

0x02–0xFF: Unused

## 7. Sample Program Sequence

The following describes the recommended procedures for operating this device.

### 7.1 SPI Sequence

#### 7.1.1 Power-on Sequence (SPI)

The following shows a power-on sequence:

- (1) Power-on.
- (2) Wait 800 ms.
- (3) Wait until the NOT\_READY bit goes to 0. NOT\_READY is GLOB\_CMD[0x0A (W1)]'s bit[10].  
 TXdata = {0xFE01} / RXdata = {0x----} /\* WINDOW = 1 \*/  
 TXdata = {0x0A00} / RXdata = {0x----} /\* GLOB\_CMD read command \*/  
 TXdata = {0x0000} / RXdata = {GLOB\_CMD} /\* get response \*/  
 Confirm the NOT\_READY bit.  
 When NOT\_READY becomes 0, this step ends. Otherwise, please repeat (3).
- (4) Confirm HARD\_ERR bits. HARD\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[6:5].  
 TXdata = {0xFE00} / RXdata = {0x----} /\* WINDOW\_ID write command. (WINDOW = 0) \*/  
 TXdata = {0x0400} / RXdata = {0x----} /\* DIAG\_STAT read command \*/  
 TXdata = {0x0000} / RXdata = {DIAG\_STAT} /\* get response \*/  
 Confirm if HARD\_ERR is 00.  
 If HARD\_ERR is 00, the IMU is operating normally. Otherwise, the IMU is faulty.

-: don't care

#### 7.1.2 Register Read and Write (SPI)

##### [Read Example]

To read a 16-bit data from a register (addr = 0x02 / WINDOW = 0):

```
TXdata = {0xFE00} / RXdata = {0x----} /* WINDOW = 0 */
TXdata = {0x0200} / RXdata = {0x----} /* read command */
TXdata = {0x----} / RXdata = {0x0400} /* get response*/
```

-: don't care

0x04 in the upper byte of RXdata indicates the device is in Configuration mode.

0x00 in the lower byte of RXdata is Reserved.

Please note that read data unit is 16 bits, and Most Significant Bit first in 16-bit SPI.

##### [Write Example]

To write an 8-bit data into a register (addr = 0x03 / WINDOW = 0):

```
TXdata = {0xFE00} / RXdata = {0x----} /* WINDOW = 0 */
TXdata = {0x8301} / RXdata = {0x----} /* write command */
```

There is no response at Write.

-: don't care

By sending this command, the IMU enters Sampling mode.

Please note that write data unit is 8 bits.

#### 7.1.3 Sampling Data (SPI)

##### [Sample Flow 1 (SPI Normal mode)]

- (1) Power-on sequence. Please refer to *Section 7.1.1*.
- (2) Filter setting sequence. Please refer to *Section 7.1.8*.
- (3) Configure Sampling mode.  
 TXdata = {0xFE01} / RXdata = {0x----} /\* WINDOW = 1 \*/

TXdata = {0x8504} / RXdata = {0x----}	/* 125 Sps */
TXdata = {0x8800} / RXdata = {0x----}	/* disable UART Auto mode, just in case. */
TXdata = {0xFE00} / RXdata = {0x----}	/* WINDOW = 0 */
TXdata = {0x8301} / RXdata = {0x----}	/* set to Sampling mode */

## (4) Receive sampling data.

(a) Wait until the Data Ready signal is asserted.

(b) Read data.

TXdata = {0x0600} / RXdata = {0x----}	/* FLAG read command */
TXdata = {0x0E00} / RXdata = {FLAG}	/* TEMP_HIGH read command */
TXdata = {0x1000} / RXdata = {TEMP_HIGH}	/* TEMP_LOW read command */
TXdata = {0x1200} / RXdata = {TEMP_LOW}	/* XGYRO_HIGH read command */
TXdata = {0x1400} / RXdata = {XGYRO_HIGH}	/* XGYRO_LOW read command */
TXdata = {0x1600} / RXdata = {XGYRO_LOW}	/* YGYRO_HIGH read command */
TXdata = {0x1800} / RXdata = {YGYRO_HIGH}	/* YGYRO_LOW read command */
TXdata = {0x1A00} / RXdata = {YGYRO_LOW}	/* ZGYRO_HIGH read command */
TXdata = {0x1C00} / RXdata = {ZGYRO_HIGH}	/* ZGYRO_LOW read command */
TXdata = {0x1E00} / RXdata = {ZGYRO_LOW}	/* XACCL_HIGH read command */
TXdata = {0x2000} / RXdata = {XACCL_HIGH}	/* XACCL_LOW read command */
TXdata = {0x2200} / RXdata = {XACCL_LOW}	/* YACCL_HIGH read command */
TXdata = {0x2400} / RXdata = {YACCL_HIGH}	/* YACCL_LOW read command */
TXdata = {0x2600} / RXdata = {YACCL_LOW}	/* ZACCL_HIGH read command */
TXdata = {0x2800} / RXdata = {ZACCL_HIGH}	/* ZACCL_LOW read command */
TXdata = {0x0800} / RXdata = {ZACCL_LOW}	/* GPIO read command */
TXdata = {0x0A00} / RXdata = {GPIO}	/* COUNT read command */
TXdata = {0x----} / RXdata = {COUNT}.	

(c) Repeat from (a) to (b).

## (5) Exit Sampling mode.

TXdata = {0x8302} / RXdata = {0x----}	/* return to Configuration mode */
---------------------------------------	------------------------------------

-: don't care

## Note

Please remember to wait until the Data Ready signal is asserted.

**[Sample Flow 2 (SPI Normal mode)]**

To read upper 16 bits of temperature, gyroscope, and accelerometer data:

(1) Power-on sequence. Please refer to *Section 7.1.1*.(2) Filter setting sequence. Please refer to *Section 7.1.8*.

(3) Configure Sampling mode.

TXdata = {0xFE01} / RXdata = {0x----}	/* WINDOW = 1 */
TXdata = {0x8504} / RXdata = {0x----}	/* 125 Sps */
TXdata = {0x8800} / RXdata = {0x----}	/* disable UART Auto mode, just in case. */
TXdata = {0xFE00} / RXdata = {0x----}	/* WINDOW = 0 */
TXdata = {0x8301} / RXdata = {0x----}	/* set to Sampling mode */

## (4) Receive sampling data.

(a) Wait until the Data Ready signal is asserted.

(b) Read data.

TXdata = {0x0600} / RXdata = {0x----}	/* FLAG read command */
TXdata = {0x0E00} / RXdata = {FLAG}	/* TEMP_HIGH read command */
TXdata = {0x1200} / RXdata = {TEMP_HIGH}	/* XGYRO_HIGH read command */
TXdata = {0x1600} / RXdata = {XGYRO_HIGH}	/* YGYRO_HIGH read command */
TXdata = {0x1A00} / RXdata = {YGYRO_HIGH}	/* ZGYRO_HIGH read command */
TXdata = {0x1E00} / RXdata = {ZGYRO_HIGH}	/* XACCL_HIGH read command */
TXdata = {0x2200} / RXdata = {XACCL_HIGH}	/* YACCL_HIGH read command */
TXdata = {0x2600} / RXdata = {YACCL_HIGH}	/* ZACCL_HIGH read command */



```

TXdata = {0x0800} / RXdata = {ZACCL_HIGH}    /* GPIO read command */
TXdata = {0x0A00} / RXdata = {GPIO}           /* COUNT read command */
TXdata = {0x----} / RXdata = {COUNT}

```

(c) Repeat from (a) to (b).

(5) Exit Sampling mode.

```
TXdata = {0x8302} / RXdata = {0x----}          /* return to Configuration mode */
```

∴ don't care

#### Note

Please remember to wait until the Data Ready signal is asserted.

### [Sample Flow 3 (SPI Burst mode)]

(1) Power-on sequence. Please refer to *Section 7.1.1*.

(2) Filter setting sequence. Please refer to *Section 7.1.8*.

(3) Configure Sampling mode.

```

TXdata = {0xFE01} / RXdata = {0x----}          /* WINDOW = 1 */
TXdata = {0x8504} / RXdata = {0x----}          /* 125 Sps */
TXdata = {0x8800} / RXdata = {0x----}          /* disable UART Auto mode, just in case. */
TXdata = {0x8C06} / RXdata = {0x----}          /* GPIO = on, COUNT = on, CheckSum = off */
TXdata = {0x8DF0} / RXdata = {0x----}          /* FLAG = on, TEMP = on, Gyro = on, ACCL = on */
TXdata = {0x8F70} / RXdata = {0x----}          /* TEMP = 32 bits, Gyro = 32 bits, ACCL = 32 bits */
TXdata = {0xFE00} / RXdata = {0x----}          /* WINDOW = 0 */
TXdata = {0x8301} / RXdata = {0x----}          /* set to Sampling mode */

```

(4) Receive sampling data.

(a) Wait until the Data Ready signal is asserted.

(b) Read data.

```

TXdata = {0x8000} / RXdata = {0x----}          /* BURST command */
TXdata = {0x----} / RXdata = {FLAG}.
TXdata = {0x----} / RXdata = {TEMP_HIGH}
TXdata = {0x----} / RXdata = {TEMP_LOW}
TXdata = {0x----} / RXdata = {XGYRO_HIGH}
TXdata = {0x----} / RXdata = {XGYRO_LOW}
TXdata = {0x----} / RXdata = {YGYRO_HIGH}
TXdata = {0x----} / RXdata = {YGYRO_LOW}
TXdata = {0x----} / RXdata = {ZGYRO_HIGH}
TXdata = {0x----} / RXdata = {ZGYRO_LOW}
TXdata = {0x----} / RXdata = {XACCL_HIGH}
TXdata = {0x----} / RXdata = {XACCL_LOW}
TXdata = {0x----} / RXdata = {YACCL_HIGH}
TXdata = {0x----} / RXdata = {YACCL_LOW}
TXdata = {0x----} / RXdata = {ZACCL_HIGH}
TXdata = {0x----} / RXdata = {ZACCL_LOW}
TXdata = {0x----} / RXdata = {GPIO}
TXdata = {0x----} / RXdata = {COUNT}

```

(c) Repeat from (a) to (b).

(5) Exit Sampling mode.

```
TXdata = {0x8302} / RXdata = {0x----}          /* return to Configuration mode */
```

∴ don't care

#### Note

Please remember to wait until the Data Ready signal is asserted.

### [Sample Flow 4 (SPI Burst mode)]

To read upper 16 bits of temperature, gyroscope, and accelerometer data:

- (1) Power-on sequence. Please refer to *Section 7.1.1*.
- (2) Filter setting sequence. Please refer to *Section 7.1.8*.
- (3) Configure Sampling mode.

TXdata = {0xFE01} / RXdata = {0x----	/* WINDOW = 1 */
TXdata = {0x8504} / RXdata = {0x----	/* 125 Sps */
TXdata = {0x8800} / RXdata = {0x----	/* disable UART Auto mode, just in case. */
TXdata = {0x8C06} / RXdata = {0x----	/* GPIO = on, COUNT = on, CheckSum = off */
TXdata = {0x8DF0} / RXdata = {0x----	/* FLAG = on, TEMP = on, Gyro = on, ACCL = on */
TXdata = {0x8F00} / RXdata = {0x----	/* TEMP = 16 bits, Gyro = 16 bits, ACCL = 16 bits */
TXdata = {0xFE00} / RXdata = {0x----	/* WINDOW = 0 */
TXdata = {0x8301} / RXdata = {0x----	/* set to Sampling mode */

- (4) Receive sampling data.

- (a) Wait until the Data Ready signal is asserted.

- (b) Read data.

TXdata = {0x8000} / RXdata = {0x----	/* BURST command */
TXdata = {0x----} / RXdata = {FLAG}	
TXdata = {0x----} / RXdata = {TEMP_HIGH}	
TXdata = {0x----} / RXdata = {XGYRO_HIGH}	
TXdata = {0x----} / RXdata = {YGYRO_HIGH}	
TXdata = {0x----} / RXdata = {ZGYRO_HIGH}	
TXdata = {0x----} / RXdata = {XACCL_HIGH}	
TXdata = {0x----} / RXdata = {YACCL_HIGH}	
TXdata = {0x----} / RXdata = {ZACCL_HIGH}	
TXdata = {0x----} / RXdata = {GPIO}	
TXdata = {0x----} / RXdata = {COUNT}	

- (c) Repeat from (a) to (b).

- (5) Exit Sampling mode.

TXdata = {0x8302} / RXdata = {0x----	/* return to Configuration mode */
--------------------------------------	------------------------------------

-: don't care

#### Note

Please remember to wait until the Data Ready signal is asserted.

### 7.1.4 Self Test (SPI)

The following shows a self test execution procedure:

- (1) Power-on sequence. Please refer to *Section 7.1.1*.
- (2) Execute the self test.

- (a) Send the self test command.

TXdata = {0xFE01} / RXdata = {0x----	/* WINDOW = 1 */
TXdata = {0x8304} / RXdata = {0x----	/* Self test command */

- (b) Wait until the self test has finished.

Wait until the SELF\_TEST bit goes to 0. SELF\_TEST is MSC\_CTRL[0x02 (W1)]'s bit[10].

TXdata = {0x0200} / RXdata = {0x----	/* MSC_CTRL read command */
TXdata = {0x0000} / RXdata = {MSC_CTRL}	/* get response */

Confirm the SELF\_TEST bit.

When SELF\_TEST becomes 0, this step ends. Otherwise, repeat (b).

- (c) Confirm the result.

Confirm the ST\_ERR bits. ST\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[14:11].

TXdata = {0xFE00} / RXdata = {0x----	/* WINDOW = 0 */
TXdata = {0x0400} / RXdata = {0x----	/* DIAG_STAT read command */
TXdata = {0x0000} / RXdata = {DIAG_STAT}	/* get response */

Confirm if all the ST\_ERR bits are 0.

If all the ST\_ERR bits are 0, the test has finished successfully. Otherwise, an error has occurred.

∴ don't care

### 7.1.5 Software Reset (SPI)

The following shows a software reset execution procedure:

(1) Power-on sequence. Please refer to *Section 7.1.1*.

(2) Issue a software reset.

(a) Send software reset command.

TXdata = {0xFE01} / RXdata = {0x----}

/\* WINDOW = 1 \*/

TXdata = {0x8A80} / RXdata = {0x----}

/\* Software reset command \*/

(b) Wait 800 ms.

∴ don't care

### 7.1.6 Flash Test (SPI)

The following shows a flash test execution procedure:

(1) Power-on sequence. Please refer to *Section 7.1.1*.

(2) Execute the flash test.

(a) Send the flash test command.

TXdata = {0xFE01} / RXdata = {0x----}

/\* WINDOW = 1 \*/

TXdata = {0x8308} / RXdata = {0x----}

/\* Flash test command \*/

(b) Wait until the flash test has finished.

Wait until the FLASH\_TEST bit goes to 0. FLASH\_TEST is MSC\_CTRL[0x02 (W1)]'s bit[11].

TXdata = {0x0200} / RXdata = {0x----}

/\* MSC\_CTRL read command \*/

TXdata = {0x0000} / RXdata = {MSC\_CTRL}

/\* get response \*/

Confirm the FLASH\_TEST bit.

When FLASH\_TEST becomes 0, this step ends. Otherwise, repeat (b).

(c) Confirm the result.

Confirm the FLASH\_ERR bit. FLASH\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[2].

TXdata = {0xFE00} / RXdata = {0x----}

/\* WINDOW = 0 \*/

TXdata = {0x0400} / RXdata = {0x----}

/\* DIAG\_STAT read command \*/

TXdata = {0x0000} / RXdata = {DIAG\_STAT}

/\* get response \*/

Confirm if FLASH\_ERR is 0.

If FLASH\_ERR is 0, the test has finished successfully. Otherwise, an error has occurred.

∴ don't care

### 7.1.7 Flash Backup (SPI)

The following shows a flash backup execution procedure:

(1) Power-on sequence. Please refer to *Section 7.1.1*.

(2) Write the desired settings to the registers that support flash backup shown in Table 6.1. Please refer to *Section 7.1.2*.

(3) Execute the flash backup.

(a) Send the flash backup command.

TXdata = {0xFE01} / RXdata = {0x----}

/\* WINDOW = 1 \*/

TXdata = {0x8A08} / RXdata = {0x----}

/\* Flash backup command \*/

(b) Wait until the flash backup has finished.

Wait until the FLASH\_BACKUP bit goes to 0. FLASH\_BACKUP is GLOB\_CMD[0x0A (W1)]'s bit[3].

TXdata = {0x0A00} / RXdata = {0x----}

/\* GLOB\_CMD read command \*/

TXdata = {0x0000} / RXdata = {GLOB\_CMD}

/\* get response \*/

Confirm the FLASH\_BACKUP bit.

When FLASH\_BACKUP becomes 0, this step ends. Otherwise, repeat (b).

## (c) Confirm the result.

Confirm the FLASH\_BU\_ERR bit. FLASH\_BU\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[0].

TXdata = {0xFE00} / RXdata = {0x----} /\* WINDOW = 0 \*/

TXdata = {0x0400} / RXdata = {0x----} /\* DIAG\_STAT read command \*/

TXdata = {0x0000} / RXdata = {DIAG\_STAT} /\* get response \*/

Confirm if FLASH\_BU\_ERR is 0.

If FLASH\_BU\_ERR is 0, the backup has finished successfully. Otherwise, an error has occurred.

∴ don't care

### 7.1.8 Filter Setting (SPI)

The following shows a filter setting procedure:

(1) Power-on sequence. Please refer to *Section 7.1.1*.

(2) Configure the filter.

(a) Send the filter setting command for the moving average filter and TAP32.

TXdata = {0xFE01} / RXdata = {0x----} /\* WINDOW = 1 \*/

TXdata = {0x8605} / RXdata = {0x----} /\* Filter setting command \*/

(b) Wait until the filter setting has finished.

Wait until the FILTER\_STAT bit goes to 0. FILTER\_STAT is FILTER\_CTRL[0x06 (W1)]'s bit[5].

TXdata = {0x0600} / RXdata = {0x----} /\* FILTER\_CTRL read command \*/

TXdata = {0x0000} / RXdata = {FILTER\_CTRL} /\* get response \*/

Confirm the FILTER\_STAT bit.

When FILTER\_STAT becomes 0, this step ends. Otherwise, repeat (b).

### 7.1.9 Frame Alignment Correction (SPI)

#### [Sample Flow (SPI normal mode)]

Rotate the gyro triad and accelerometer triad -30 degrees around y-axis

Gyro sensor axis / Acceleration sensor axis

TXdata = {0xB86C} / RXdata = {0x----} /\* R\_MATRIX\_G\_M11 = 0.866 \*/

TXdata = {0xB937} / RXdata = {0x----}

TXdata = {0xBA00} / RXdata = {0x----} /\* R\_MATRIX\_G\_M12 = 0 \*/

TXdata = {0xBB00} / RXdata = {0x----}

TXdata = {0xBCFF} / RXdata = {0x----} /\* R\_MATRIX\_G\_M13 = 0.5 \*/

TXdata = {0xBD1F} / RXdata = {0x----}

TXdata = {0xBE00} / RXdata = {0x----} /\* R\_MATRIX\_G\_M21 = 0 \*/

TXdata = {0xBF00} / RXdata = {0x----}

TXdata = {0xC000} / RXdata = {0x----} /\* R\_MATRIX\_G\_M22 = 1 \*/

TXdata = {0xC140} / RXdata = {0x----}

TXdata = {0xC200} / RXdata = {0x----} /\* R\_MATRIX\_G\_M23 = 0 \*/

TXdata = {0xC300} / RXdata = {0x----}

TXdata = {0xC400} / RXdata = {0x----} /\* R\_MATRIX\_G\_M31 = - 0.5 \*/

TXdata = {0xC5E0} / RXdata = {0x----}

TXdata = {0xC600} / RXdata = {0x----} /\* R\_MATRIX\_G\_M32 = 0 \*/

TXdata = {0xC700} / RXdata = {0x----}

TXdata = {0xC86C} / RXdata = {0x----} /\* R\_MATRIX\_G\_M33 = 0.866 \*/

TXdata = {0xC937} / RXdata = {0x----}

## 7.2 UART Sequence

### 7.2.1 Power-on Sequence (UART)

The following shows a power-on sequence:

- (1) Power-on.
- (2) Wait 800 ms.
- (3) Wait until the NOT\_READY bit goes to 0. NOT\_READY is GLOB\_CMD[0x0A (W1)]'s bit[10].  
 TXdata = {0xFE, 0x01, 0x0d} /\* WINDOW = 1 \*/  
 TXdata = {0x0A, 0x00, 0x0d} /\* GLOB\_CMD read command \*/  
 RXdata = {0x0A, MSByte, LSByte, 0x0d} /\* get response \*/  
 Confirm the NOT\_READY bit.  
 When NOT\_READY becomes 0, this step ends. Otherwise, please repeat (3).
- (4) Confirm the HARD\_ERR bits. HARD\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[6:5].  
 TXdata = {0xFE, 0x00, 0x0d} /\* WINDOW = 0 \*/  
 TXdata = {0x04, 0x00, 0x0d} /\* DIAG\_STAT read command \*/  
 RXdata = {0x04, MSByte, LSByte, 0x0d} /\* get response \*/  
 Confirm if HARD\_ERR is 00.  
 If HARD\_ERR is 00, the IMU is operating normally. Otherwise, the IMU is faulty.

### 7.2.2 Register Read and Write (UART)

#### [Read Example]

To read a 16-bit data from a register (addr = 0x02 / WINDOW = 0).

```
TXdata = {0xFE, 0x00, 0x0d} /* WINDOW = 0 */
TXdata = {0x02, 0x00, 0x0d} /* command */
RXdata = {0x02, 0x04, 0x00, 0x0d} /* response */
```

0x04 in 2nd byte of RXdata indicates that the device is in Configuration mode.

0x00 in 3rd byte of RXdata is Reserved.

Please note that read data unit is 16 bits, and Most Significant Byte first.

#### [Write Example]

To write an 8-bit data into a register (addr = 0x03 / WINDOW = 0).

```
TXdata = {0xFE, 0x00, 0x0d} /* WINDOW = 0 */
TXdata = {0x83, 0x01, 0x0d} /* command */
RXdata = w/o response
```

By sending this command, the IMU enters Sampling mode.

Please note that write data unit is 8 bits.

### 7.2.3 Sampling Data (UART)

#### [Sample Flow 1 (UART Auto mode)]

- (1) Power-on sequence. Please refer to *Section 7.2.1*.
- (2) Filter setting sequence. Please refer to *Section 7.2.8*.
- (3) Configure Sampling mode.  
 TXdata = {0xFE, 0x01, 0x0d} /\* WINDOW = 1 \*/  
 TXdata = {0x85, 0x04, 0x0d} /\* 125 Sps \*/  
 TXdata = {0x88, 0x01, 0x0d} /\* UART Auto mode \*/  
 TXdata = {0x8C, 0x06, 0x0d} /\* GPIO = on, COUNT = on, CheckSum = off \*/  
 TXdata = {0x8D, 0xF0, 0x0d} /\* FLAG = on, TEMP = on, Gyro = on, ACCL = on \*/  
 TXdata = {0x8F, 0x70, 0x0d} /\* TEMP = 32 bits, Gyro = 32 bits, ACCL = 32 bits \*/  
 TXdata = {0xFE, 0x00, 0x0d} /\* WINDOW = 0 \*/  
 TXdata = {0x83, 0x01, 0x0d} /\* set to Sampling mode \*/
- (4) Receive sampling data.

## (a) Read data.

```

RXdata = {0x80,  FLAG_Hi,      FLAG_Lo,
                TEMP_HIGH_Hi,  TEMP_HIGH_Lo,  TEMP_LOW_Hi,  TEMP_LOW_Lo,
                XGYRO_HIGH_Hi, XGYRO_HIGH_Lo, XGYRO_LOW_Hi, XGYRO_LOW_Lo,
                YGYRO_HIGH_Hi, YGYRO_HIGH_Lo, YGYRO_LOW_Hi, YGYRO_LOW_Lo,
                ZGYRO_HIGH_Hi, ZGYRO_HIGH_Lo, ZGYRO_LOW_Hi, ZGYRO_LOW_Lo,
                XACCL_HIGH_Hi, XACCL_HIGH_Lo, XACCL_LOW_Hi, XACCL_LOW_Lo,
                YACCL_HIGH_Hi, YACCL_HIGH_Lo, YACCL_LOW_Hi, YACCL_LOW_Lo,
                ZACCL_HIGH_Hi, ZACCL_HIGH_Lo, ZACCL_LOW_Hi, ZACCL_LOW_Lo,
                GPIO_Hi,      GPIO_Lo,
                COUNT_Hi,     COUNT_Lo,      0x0d}

```

## (b) Repeat (a).

## (5) Exit Sampling mode.

```
TXdata = {0x83, 0x02, 0x0d} /* return to Configuration mode */
```

**[Sample Flow 2 (UART Auto mode)]**

To read upper 16 bits of temperature, gyroscope, and accelerometer data.

(1) Power-on sequence. Please refer to *Section 7.2.1*.(2) Filter setting sequence. Please refer to *Section 7.2.8*.

## (3) Configure Sampling mode.

```

TXdata = {0xFE, 0x01, 0x0d} /* WINDOW = 1 */
TXdata = {0x85, 0x04, 0x0d} /* 125 Sps */
TXdata = {0x88, 0x01, 0x0d} /* UART Auto mode */
TXdata = {0x8C, 0x06, 0x0d} /* GPIO = on, COUNT = on, CheckSum = off */
TXdata = {0x8D, 0xF0, 0x0d} /* FLAG = on, TEMP = on, Gyro = on, ACCL = on */
TXdata = {0x8F, 0x00, 0x0d} /* TEMP = 16 bits, Gyro = 16 bits, ACCL = 16 bits */
TXdata = {0xFE, 0x00, 0x0d} /* WINDOW = 0 */
TXdata = {0x83, 0x01, 0x0d} /* set to Sampling mode */

```

## (4) Receive sampling data.

## (a) Read data.

```

RXdata = {0x80,  FLAG_Hi,      FLAG_Lo,
                TEMP_HIGH_Hi,  TEMP_HIGH_Lo,
                XGYRO_HIGH_Hi, XGYRO_HIGH_Lo,
                YGYRO_HIGH_Hi, YGYRO_HIGH_Lo,
                ZGYRO_HIGH_Hi, ZGYRO_HIGH_Lo,
                XACCL_HIGH_Hi, XACCL_HIGH_Lo,
                YACCL_HIGH_Hi, YACCL_HIGH_Lo,
                ZACCL_HIGH_Hi, ZACCL_HIGH_Lo,
                GPIO_Hi,      GPIO_Lo,
                COUNT_Hi,     COUNT_Lo,      0x0d}

```

## (b) Repeat (a).

## (5) Exit Sampling mode.

```
TXdata = {0x83, 0x02, 0x0d} /* return to Configuration mode */
```

**[Sample Flow 3 (UART Burst mode)]**(1) Power-on sequence. Please refer to *Section 7.2.1*.(2) Filter setting sequence. Please refer to *Section 7.2.8*.

## (3) Configure Sampling mode.

```

TXdata = {0xFE, 0x01, 0x0d} /* WINDOW = 1 */
TXdata = {0x85, 0x04, 0x0d} /* 125 Sps */
TXdata = {0x88, 0x00, 0x0d} /* UART Manual mode */
TXdata = {0x8C, 0x06, 0x0d} /* GPIO = on, COUNT = on, CheckSum = off */
TXdata = {0x8D, 0xF0, 0x0d} /* FLAG = on, TEMP = on, Gyro = on, ACCL = on */

```

```

TXdata = {0x8F, 0x70, 0x0d}          /* TEMP = 32 bits, Gyro = 32 bits, ACCL = 32 bits */
TXdata = {0xFE, 0x00, 0x0d}          /* WINDOW = 0 */
TXdata = {0x83, 0x01, 0x0d}          /* set to Sampling mode */

```

## (4) Receive sampling data.

(a) Wait until the Data Ready signal is asserted.

(b) Read data.

```

TXdata = {0x80, 0x00, 0x0d}          /* BURST command */
RXdata = {0x80,  FLAG_Hi,  FLAG_Lo,
          TEMP_HIGH_Hi,  TEMP_HIGH_Lo,  TEMP_LOW_Hi,  TEMP_LOW_Lo,
          XGYRO_HIGH_Hi, XGYRO_HIGH_Lo, XGYRO_LOW_Hi, XGYRO_LOW_Lo,
          YGYRO_HIGH_Hi, YGYRO_HIGH_Lo, YGYRO_LOW_Hi, YGYRO_LOW_Lo,
          ZGYRO_HIGH_Hi, ZGYRO_HIGH_Lo, ZGYRO_LOW_Hi, ZGYRO_LOW_Lo,
          XACCL_HIGH_Hi, XACCL_HIGH_Lo, XACCL_LOW_Hi, XACCL_LOW_Lo,
          YACCL_HIGH_Hi, YACCL_HIGH_Lo, YACCL_LOW_Hi, YACCL_LOW_Lo,
          ZACCL_HIGH_Hi, ZACCL_HIGH_Lo, ZACCL_LOW_Hi, ZACCL_LOW_Lo,
          GPIO_Hi,      GPIO_Lo,
          COUNT_Hi,     COUNT_Lo,     0x0d}

```

(c) Repeat from (a) to (b).

## (5) Exit Sampling mode.

```

TXdata = {0x83, 0x02, 0x0d}          /* return to Configuration mode */

```

## Note

Please remember to wait until the Data Ready signal is asserted.

**[Sample Flow 4 (UART Burst mode)]**

To read upper 16 bits of temperature, gyroscope, and accelerometer data.

(1) Power-on sequence. Please refer to *Section 7.2.1*.(2) Filter setting sequence. Please refer to *Section 7.2.8*.

(3) Configure Sampling mode.

```

TXdata = {0xFE, 0x01, 0x0d}          /* WINDOW = 1 */
TXdata = {0x85, 0x04, 0x0d}          /* 125 Sps */
TXdata = {0x88, 0x00, 0x0d}          /* UART Manual mode */
TXdata = {0x8C, 0x06, 0x0d}          /* GPIO = on, COUNT = on, CheckSum = off */
TXdata = {0x8D, 0xF0, 0x0d}          /* FLAG = on, TEMP = on, Gyro = on, ACCL = on */
TXdata = {0x8F, 0x00, 0x0d}          /* TEMP = 16 bits, Gyro = 16 bits, ACCL = 16 bits */
TXdata = {0xFE, 0x00, 0x0d}          /* WINDOW = 0 */
TXdata = {0x83, 0x01, 0x0d}          /* set to Sampling mode */

```

## (4) Receive sampling data.

(a) Wait until the Data Ready signal is asserted.

(b) Read data.

```

TXdata = {0x80, 0x00, 0x0d}          /* BURST command */
RXdata = {0x80,  FLAG_Hi,  FLAG_Lo,
          TEMP_HIGH_Hi,  TEMP_HIGH_Lo,
          XGYRO_HIGH_Hi, XGYRO_HIGH_Lo,
          YGYRO_HIGH_Hi, YGYRO_HIGH_Lo,
          ZGYRO_HIGH_Hi, ZGYRO_HIGH_Lo,
          XACCL_HIGH_Hi, XACCL_HIGH_Lo,
          YACCL_HIGH_Hi, YACCL_HIGH_Lo,
          ZACCL_HIGH_Hi, ZACCL_HIGH_Lo,
          GPIO_Hi,      GPIO_Lo,
          COUNT_Hi,     COUNT_Lo,     0x0d}

```

(c) Repeat from (a) to (b).

## (5) Exit Sampling mode.

```

TXdata = {0x83, 0x02, 0x0d}          /* return to Configuration mode */

```

## Note

Please remember to wait until the Data Ready signal is asserted.

NOTE:

- Please note that read data unit is 16 bits, and Most Significant Byte first.
- Please note that write data unit is 8 bits.
- XGYRO\_HIGH\_Hi means MSByte of XGYRO\_HIGH data
- XGYRO\_HIGH\_Lo means LSByte of XGYRO\_LOW data

#### 7.2.4 Self Test (UART)

The following shows a self test execution procedure:

(1) Power-on sequence. Please refer to *Section 7.2.1*.

(2) Execute the self test.

(a) Send the self test command.

TXdata = {0xFE, 0x01, 0x0d} /\* WINDOW = 1 \*/

TXdata = {0x83, 0x04, 0x0d} /\* Self test command \*/

(b) Wait until the self test has finished.

Wait until the SELF\_TEST bit goes to 0. SELF\_TEST is MSC\_CTRL[0x02 (W1)]'s bit[10].

TXdata = {0x02, 0x00, 0x0d} /\* MSC\_CTRL read command \*/

RXdata = {0x02, MSByte, LSByte, 0x0d} /\* get response \*/

Confirm the SELF\_TEST bit.

When SELF\_TEST becomes 0, this step ends. Otherwise, repeat (b).

(c) Confirm the result.

Confirm the ST\_ERR bits. ST\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[14:11].

TXdata = {0xFE, 0x00, 0x0d} /\* WINDOW = 0 \*/

TXdata = {0x04, 0x00, 0x0d} /\* DIAG\_STAT read command \*/

RXdata = {0x04, MSByte, LSByte, 0x0d} /\* get response \*/

Confirm if all the ST\_ERR bits are 0.

If all the ST\_ERR bits are 0, the test has finished successfully. Otherwise, an error has occurred.

#### 7.2.5 Software Reset (UART)

The following shows a software reset execution procedure:

(1) Power-on sequence. Please refer to *Section 7.2.1*.

(2) Issue a software reset.

(a) Send software reset command.

TXdata = {0xFE, 0x01, 0x0d} /\* WINDOW = 1 \*/

TXdata = {0x8A, 0x80, 0x0d} /\* Software reset command \*/

(b) Wait 800 ms.

#### 7.2.6 Flash Test (UART)

The following shows a flash test execution procedure:

(1) Power-on sequence. Please refer to *Section 7.2.1*.

(2) Execute the flash test.

(a) Send the flash test command.

TXdata = {0xFE, 0x01, 0x0d} /\* WINDOW = 1 \*/

TXdata = {0x83, 0x08, 0x0d} /\* Flash test command \*/

(b) Wait until the flash test has finished.

Wait until the FLASH\_TEST bit goes to 0. FLASH\_TEST is MSC\_CTRL[0x02 (W1)]'s bit[11].

TXdata = {0x02, 0x00, 0x0d} /\* MSC\_CTRL read command \*/

RXdata = {0x02, MSByte, LSByte, 0x0d} /\* get response \*/

Confirm the FLASH\_TEST bit.

When FLASH\_TEST becomes 0, this step ends. Otherwise, repeat (b).



## (c) Confirm the result.

Confirm the FLASH\_ERR bit. FLASH\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[2].

TXdata = {0xFE, 0x00, 0x0d}

/\* WINDOW = 0 \*/

TXdata = {0x04, 0x00, 0x0d}

/\* DIAG\_STAT read command \*/

RXdata = {0x04, MSByte, LSByte, 0x0d}

/\* get response \*/

Confirm if FLASH\_ERR is 0.

If FLASH\_ERR is 0, the test has finished successfully. Otherwise, an error has occurred.

### 7.2.7 Flash Backup (UART)

The following shows a flash backup execution procedure:

(1) Power-on sequence. Please refer to *Section 7.2.1*.

(2) Write the desired settings to the registers that support flash backup shown in Table 6.1. Please refer to *Section 7.2.2*.

(3) Execute the flash backup.

## (a) Send the flash backup command.

TXdata = {0xFE, 0x01, 0x0d}

/\* WINDOW = 1 \*/

TXdata = {0x8A, 0x08, 0x0d}

/\* Flash backup command \*/

## (b) Wait until the flash backup has finished.

Wait until the FLASH\_BACKUP bit goes to 0. FLASH\_BACKUP is GLOB\_CMD[0x0A (W1)]'s bit[3].

TXdata = {0x0A, 0x00, 0x0d}

/\* GLOB\_CMD read command \*/

RXdata = {0x0A, MSByte, LSByte, 0x0d}

/\* get response \*/

Confirm the FLASH\_BACKUP bit.

When FLASH\_BACKUP becomes 0, this step ends. Otherwise, repeat (b).

## (c) Confirm the result.

Confirm the FLASH\_BU\_ERR bit. FLASH\_BU\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[0].

TXdata = {0xFE, 0x00, 0x0d}

/\* WINDOW = 0 \*/

TXdata = {0x04, 0x00, 0x0d}

/\* DIAG\_STAT read command \*/

RXdata = {0x04, MSByte, LSByte, 0x0d}

/\* get response \*/

Confirm if FLASH\_BU\_ERR is 0.

If FLASH\_BU\_ERR is 0, the backup has finished successfully. Otherwise, an error has occurred.

### 7.2.8 Initial Backup (UART)

The following shows an initial backup execution procedure:

(1) Power-on sequence. Please refer to *Section 7.2.1*.

(2) Execute initial backup.

## (a) Send the initial backup command.

TXdata = {0xFE, 0x01, 0x0d}

/\* WINDOW = 1 \*/

TXdata = {0x8A, 0x10, 0x0d}

/\* Initial backup command \*/

## (b) Wait until the initial backup has finished.

Wait until the INITIAL\_BACKUP bit goes to 0. INITIAL\_BACKUP is GLOB\_CMD[0x0A (W1)]'s bit[4].

TXdata = {0x0A, 0x00, 0x0d}

/\* GLOB\_CMD read command \*/

RXdata = {0x0A, MSByte, LSByte, 0x0d}

/\* get response \*/

Confirm the INITIAL\_BACKUP bit.

When INITIAL\_BACKUP becomes 0, this step ends. Otherwise, repeat (b).

## (c) Confirm the result.

Confirm the FLASH\_BU\_ERR bit. FLASH\_BU\_ERR is DIAG\_STAT[0x04 (W0)]'s bit[0].

TXdata = {0xFE, 0x00, 0x0d}

/\* WINDOW = 0 \*/

TXdata = {0x04, 0x00, 0x0d}

/\* DIAG\_STAT read command \*/

RXdata = {0x04, MSByte, LSByte, 0x0d}

/\* get response \*/

Confirm if FLASH\_BU\_ERR is 0.

If FLASH\_BU\_ERR is 0, the backup has finished successfully. Otherwise, an error has occurred.

(d) Issue a software reset. Please refer to *Section 7.2.5*.

### 7.2.9 Filter Setting (UART)

The following shows a filter setting procedure:

(1) Power-on sequence. Please refer to *Section 7.2.1*.

(2) Configure the filter.

(a) Send the filter setting command for the moving average filter and TAP32.

TXdata = {0xFE, 0x01, 0x0d} /\* WINDOW = 1 \*/

TXdata = {0x86, 0x05, 0x0d} /\* Filter setting command \*/

(b) Wait until the filter setting has finished.

Wait until the FILTER\_STAT bit goes to 0. FILTER\_STAT is FILTER\_CTRL[0x06 (W1)]'s bit[5].

TXdata = {0x06, 0x00, 0x0d} /\* FILTER\_CTRL read command \*/

RXdata = {0x06, MSByte, LSByte, 0x0d} /\* get response \*/

Confirm the FILTER\_STAT bit.

When FILTER\_STAT becomes 0, this step ends. Otherwise, repeat (b).

### 7.2.10 Auto Start (UART only)

The following shows a procedure to configure the Auto Start function and read data:

(1) Power-on sequence. Please refer to *Section 7.2.1*.

(2) Configure the following registers:

TXdata = {0xFE, 0x01, 0x0d} /\* WINDOW = 1 \*/

TXdata = {0x85, 0x04, 0x0d} /\* 125 Sps \*/

TXdata = {0x86, 0x04, 0x0d} /\* TAP = 16 \*/

TXdata = {0x88, 0x03, 0x0d} /\* UART Auto mode, Auto start = on \*/

TXdata = {0x8C, 0x06, 0x0d} /\* GPIO = on, COUNT = on, CheckSum = off \*/

TXdata = {0x8D, 0xF0, 0x0d} /\* FLAG = on, TEMP = on, Gyro = on, ACCL = on \*/

TXdata = {0x8F, 0x70, 0x0d} /\* TEMP = 32 bits, Gyro = 32 bits, ACCL = 32 bits \*/

(2) Execute Flash backup. Please refer to *Section 7.2.7*.

(4) Power-off.

(5) Power-on.

(6) Wait 800 ms.

(7) Receive sampling data.

(a) Read data.

RXdata = {0x80,	FLAG_Hi,	FLAG_Lo,		
	TEMP_HIGH_Hi,	TEMP_HIGH_Lo,	TEMP_LOW_Hi,	TEMP_LOW_Lo,
	XGYRO_HIGH_Hi,	XGYRO_HIGH_Lo,	XGYRO_LOW_Hi,	XGYRO_LOW_Lo,
	YGYRO_HIGH_Hi,	YGYRO_HIGH_Lo,	YGYRO_LOW_Hi,	YGYRO_LOW_Lo,
	ZGYRO_HIGH_Hi,	ZGYRO_HIGH_Lo,	ZGYRO_LOW_Hi,	ZGYRO_LOW_Lo,
	XACCL_HIGH_Hi,	XACCL_HIGH_Lo,	XACCL_LOW_Hi,	XACCL_LOW_Lo,
	YACCL_HIGH_Hi,	YACCL_HIGH_Lo,	YACCL_LOW_Hi,	YACCL_LOW_Lo,
	ZACCL_HIGH_Hi,	ZACCL_HIGH_Lo,	ZACCL_LOW_Hi,	ZACCL_LOW_Lo,
	GPIO_Hi,	GPIO_Lo,		
	COUNT_Hi,	COUNT_Lo,	0x0d}	

(b) Repeat (a).

(8) If you want to stop sampling,

TXdata = {0x83, 0x02, 0x0d} /\* return to Configuration mode \*/

## 8. Handling Notes

### 8.1 Cautions for Use

- When you attach the product to a housing, equipment, jig, or tool, make sure you attach it properly so that no mechanical stress is added to create a distortion such as a warp or twist. In addition, tighten the screws firmly but not too firmly because the mount of the product may break. Use screw locking techniques as necessary.
- When you set up the product, make sure the equipment, jigs, tools, and workers maintain a good ground in order not to generate high voltage leakage. If you add overcurrent or static electricity to the product, the product may be damaged permanently.
- When you install the product, make sure metallic or other conductors do not enter the product. Otherwise, malfunction or damage of the product may result.
- If excessive shock is added to the product when, for example, the product falls, the quality of the product may be degraded. Make sure the product does not fall when you handle it.
- Before you start using the product, test it in the actual equipment under the actual operating environment.
- Since the product has capacitors inside, inrush current will occur during power-on. Evaluate in the actual environment in order to check the effect of the supply voltage drop by inrush current in the system.
- If water enters the product, malfunction or damage of the product may result. If the product can be exposed to water, the system must have a waterproof structure. We do not guarantee the operation of the product when the product is exposed to condensation, dust, oil, corrosive gas (salt, acid, alkaline, or the like), or direct sunlight.
- This product is not designed to be radiation resistant.
- Never use this product if the operating condition is over the absolute maximum rating. If you do, the characteristics of the product may never recover.
- If the product is exposed to excessive exogenous noise or the like, degradation of the precision, malfunction, or damage of the product may result. The system needs to be designed so that the noise itself is suppressed or the system is immune to the noise.
- Mechanical vibration or shock, continuous mechanical stress, rapid temperature change, or the like may cause cracks or disconnections at the various connecting parts.
- Take sufficient safety measure for the equipment this product is built into.
- This product is not intended for general use by the consumer but instead for engineering design. For the customer, please consider it safely with the proper use.
- This product is not designed to be used in the equipment that demands extremely high reliability and where its failure may threaten human life or property (for example, aerospace equipment, submarine repeater, nuclear power control equipment, life support equipment, medical equipment, transportation control equipment, etc.). Therefore, Seiko Epson Corporation will not be liable for any damages caused by the use of the product for those applications.
- Do not alter or disassemble the product.
- The casing of this product is electrically conductive. When the product is connected or mounted to the circuit board, ensure the board substrate or board wiring pattern does not short-circuit or contact to the case.

### 8.2 Cautions for Storage

- Do not add shock or vibration to the packing box. Do not spill water over the packing box. Do not store or use the product in the environment where dew condensation occurs due to rapid temperature change.
- To suppress the characteristic change by prolonged storage, it is recommended to maintain the environment at normal temperature and normal humidity. Normal temperature: +5 °C to +35 °C Normal humidity: 45% RH to 85% RH (JIS Z 8703).
- Do not store the product in a location subject to High Temperature, high humidity, under direct sunlight, corrosive gas or dust.
- Do not put mechanical stress on the product while it is stored.

### 8.3 Other Cautions

- When you connect the socket to the header of this product, make sure you do not insert the header in the reverse orientation. If you do, the IMU may be damaged permanently. In addition, if you attach the product to the equipment, etc. using connection harness, connect the connection harness to the product first, and then attach it to the equipment, etc.
- The gloss marks derived from the adhesive material may have appeared on the casing surface of the product, but it does not affect the function and quality of the product.
- The Parting line as a result of die cast manufacturing process may have appeared on the casing surface of the product, but it is not an abnormality.
- Please take care not to tamper with or accidentally disturb the assembly screw on the surface where the serial number is printed when attaching and detaching the product to the system. We do not guarantee the performance and the quality of the product in case the assembly screw is manipulated.
- Never turn off power while the host communicates the product. Otherwise, malfunction of the product may result.
- Small performance deterioration due to long-term use and aging effects, etc. cannot be detected through the self-test in this product. Discontinue use immediately even when the self-test results in a "pass" when experiencing abnormality in the sensor performance.
- If noise is induced on the external trigger terminal, there is a possibility an invalid measurement process is unintentionally sent to the host. To prevent this, when using an external trigger, take precaution to minimize noise on the external trigger terminal.
- Exercise care and precaution with the packaging and during transport of the equipment that this product is installed on to avoid excessive vibration and or damage from impact.

### 8.4 Limited Warranty

- The product warranty period is one year from the date of shipment.
- If a defect due to a quality failure of the product is found during the warranty period, we will promptly provide a replacement.

## 9. Contact

### AMERICA

---

**EPSON AMERICA, INC.**

Headquarter:  
3131 Katella Ave.  
Los Alamitos, CA 90720, U.S.A  
Phone: +1-800-463-7766

San Jose Office:  
2860 Zanker Road, Suite 204,  
San Jose, CA 95134, U.S.A  
Phone: +1-800-463-7766

### EUROPE

---

**EPSON EUROPE ELECTRONICS GmbH**

Riesstrasse 15, 80992 Munich,  
GERMANY  
Phone: +49-89-14005-0 Fax: +49-89-14005-110

### ASIA

---

**EPSON (CHINA) CO., LTD.**

4F, Tower 1 of China Central Place, 81 Jianguo Street, Chaoyang  
District, Beijing 100025 CHINA  
Phone: +86-400-810-9972 X ext.2  
Mail EPSON\_MSM@ecc.epson.com.cn

**EPSON SINGAPORE PTE. LTD.**

438B Alexandra Road, Block B Alexandra TechnoPark, #04-01/04,  
Singapore 119968  
Phone: +65-6586-5500 Fax: +65-6271-3182

**EPSON TAIWAN TECHNOLOGY & TRADING LTD.**

15F, No.100, Song Ren Road, Taipei 110, TAIWAN  
Phone: +886-2-8786-6688 Fax: +886-2-8786-6660

**EPSON KOREA Co., Ltd.**

10F Posco Tower Yeoksam, Teheranro 134 Gangnam-gu,  
Seoul, 06235 KOREA  
Phone: +82-2-558-4270 Fax: +82-2-3420-6699

### JAPAN

---

**SEIKO EPSON CORPORATION.****MD SALES & MARKETING DEPT.**

[https://global.epson.com/products\\_and\\_drivers/sensing\\_system/contact/](https://global.epson.com/products_and_drivers/sensing_system/contact/)

**Product Information on www server**

[https://global.epson.com/products\\_and\\_drivers/sensing\\_system/](https://global.epson.com/products_and_drivers/sensing_system/)