IMU (Inertial Measurement Unit): M-G355QDG0

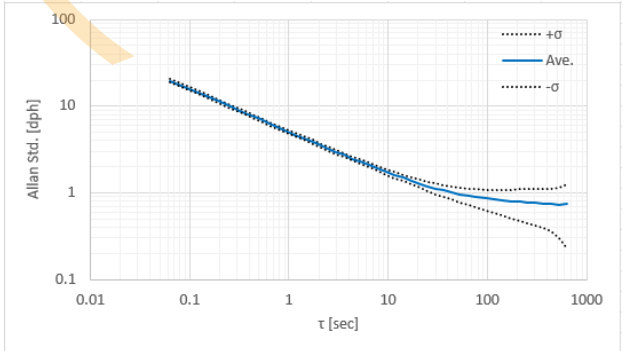
Features

- Small Size, Lightweight : 24 x 24 x 10 mm, 10 grams
- Low-Noise, High-Stability
  - Gyro Bias Instability : 1.2 °/h
  - Angular Random Walk : 0.08 °/$\sqrt{h}$
- Initial Bias Error : 360 °/h (1σ) / 3 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.) : ~400 Sps
- External Trigger Input / External Counter Reset Input
- Calibration Temperature Range : -40 ℃ to +85 ℃
- Operating Temperature Range : -40 ℃ to +85 ℃
- Single Voltage Supply : 3.3 V
- Low Power Consumption : 16 mA (Typ.)
- Standards : IEC 61508-1:2010, IEC 61508-2:2010,
  IEC 61508-3:2010(Conforms to SIL-1)

Application

- Navigation Systems
- Pointing and Tracking Systems
- Autonomous Vehicle

Typical Performance Characteristic



Gyro Allan Variance Characteristic

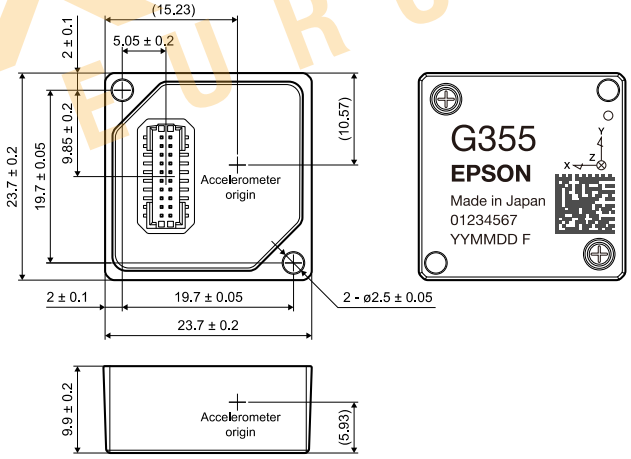
Product Name and Number
M-G355QDG0: X2G000231000100

Description

The M-G355QDG0 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-G355QDG0 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.

Outline Dimensions

Unit: mm



Block Diagram

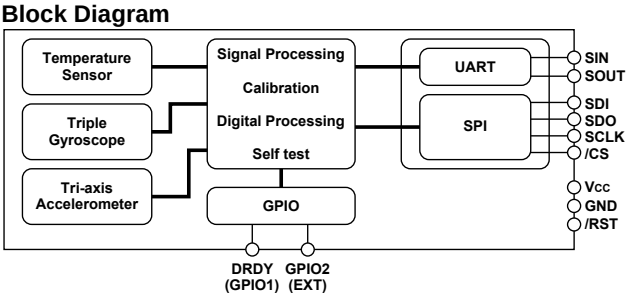


Table of Contents

Revision History 7
Ordering Information 8
Evaluation tools 8
Symbols 9
Notice of the Document 10

1. Product Specifications 11
1.1 Absolute Maximum Ratings 11
1.2 Recommended Operating Condition 11
1.3 Legal Regulations and Applicable Standards (Functional Safety) 11
1.4 Characteristics and Electrical Specifications 12
1.5 Timing Specifications 14
1.6 Connector Pin Layout and Functions 16

2. Mechanical Dimensions 17
2.1 Outline Dimensions 17
2.2 Connector Dimensions 18

3. Typical Performance Characteristics 19

4. Basic Operation 20
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 23
4.6 Sampling Counter 23
4.7 GPIO 24
4.8 Diagnostic Functions 24
4.8.1 Self Test 24
4.8.2 On-board Diagnostics Function 24
4.9 External Trigger Input 24
4.10 External Counter Reset Input 26
4.11 Checksum 27
4.12 Automatic Start (For UART Auto Mode Only) 28
4.13 Filter 28
4.14 Range Over Function 30

5. Digital Interface 31
5.1 SPI Interface 32
5.1.1 SPI Read Timing (Normal Mode) 32
5.1.2 SPI Write Timing (Normal Mode) 33
5.1.3 SPI Read Timing (Burst Mode) 33
5.2 UART Interface 34
5.2.1 UART Read Timing (Normal Mode) 35
5.2.2 UART Read Timing (Burst Mode) 35
5.2.3 UART Write Timing 36
5.2.4 UART Auto Mode Operation 36
5.3 Data Packet Format 37

6. User Registers 40
6.1 BURST Register (Window 0) 42

6.2 MODE_CTRL Register (Window 0) 42
6.3 DIAG_STAT Register (Window 0) 42
6.4 FLAG (ND / EA) Register (Window 0) 43
6.5 GPIO Register (Window 0) 44
6.6 COUNT Register (Window 0) 44
6.7 RANGE_OVER Register (Window 0) 45
6.8 TEMP Register (Window 0) 45
6.9 GYRO Register (Window 0) 45
6.10 ACCL Register (Window 0) 46
6.11 OB_DIAG_STAT Register (Window 0) 46
6.12 ID Register (Window 0) 47
6.13 SIG_CTRL Register (Window 1) 47
6.14 MSC_CTRL Register (Window 1) 47
6.15 SMPL_CTRL Register (Window 1) 48
6.16 FILTER_CTRL Register (Window 1) 49
6.17 UART_CTRL Register (Window 1) 50
6.18 GLOB_CMD Register (Window 1) 50
6.19 BURST_CTRL1 Register (Window 1) 51
6.20 BURST_CTRL2 Register (Window 1) 52
6.21 POL_CTRL Register (Window 1) 52
6.22 GLOB_CMD3 Register (Window 1) 52
6.23 PROD_ID Register (Window 1) 53
6.24 VERSION Register (Window 1) 53
6.25 SERIAL_NUM Register (Window 1) 53
6.26 WIN_CTRL Register (Window 0,1) 53

7. Sample Program Sequence 54

7.1 SPI Sequence 54
7.1.1 Power-on Sequence (SPI) 54
7.1.2 Register Read and Write (SPI) 54
7.1.3 Sampling Data (SPI) 55
7.1.4 Self Test (SPI) 57
7.1.5 Software Reset (SPI) 58
7.1.6 Flash Test (SPI) 58
7.1.7 Flash Backup (SPI) 59
7.1.8 Filter Setting (SPI) 59
7.2 UART Sequence 59
7.2.1 Power-on Sequence (UART) 59
7.2.2 Register Read and Write (UART) 60
7.2.3 Sampling Data (UART) 60
7.2.4 Self Test (UART) 62
7.2.5 Software Reset (UART) 63
7.2.6 Flash Test (UART) 63
7.2.7 Flash Backup (UART) 63
7.2.8 Initial Backup (UART) 64
7.2.9 Filter Setting (UART) 64
7.2.10 Auto Start (UART only) 65

8. Handling Notes 66

8.1 Cautions for Use 66
8.2 Cautions for Storage 66
8.3 Other Cautions 67

8.4 Limited Warranty 67

9. Contact 68

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-G355QDG0 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 ····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 400 Sps, FIR Filter TAP N = 8 ····23
Figure 4.6 Data Ready Signal Timing ····23
Figure 4.7 External Trigger Input (UART / SPI Manual Mode) ····25
Figure 4.8 External Trigger input Functional Block Diagram ····26
Figure 4.9 External Counter Reset Input ····27
Figure 4.10 Checksum ····27
Figure 4.11 Moving Average Filter Characteristics ····28
Figure 4.12 FIR Kaiser Filter Typical Characteristic 1 ····29
Figure 4.13 FIR Kaiser Filter Typical Characteristic 2 ····29
Figure 5.1 SPI Read Timing (Normal Mode) ····32
Figure 5.2 SPI Write Timing (Normal Mode) ····33
Figure 5.3 SPI Timing (Burst Mode) ····33
Figure 5.4 UART Bit Format ····34
Figure 5.5 UART Read Timing (Normal Mode) ····35
Figure 5.6 UART Read Timing (Burst Mode) ····36
Figure 5.7 UART Write Timing ····36
Figure 5.8 UART Auto Mode Operation ····36

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 26
Table 4.3 Transient Response in Number of Samples Based on Output Data Rate vs Filter Tap 30
Table 5.1 SPI Communication Settings 32
Table 5.2 SPI Timing (Normal Mode) 32
Table 5.3 Command Format (Read) 32
Table 5.4 Response Format (Read) 32
Table 5.5 Command Format (Write) 33
Table 5.6 SPI Timing (Burst Mode) 33
Table 5.7 UART Communication Settings 34
Table 5.8 UART Timing 34
Table 5.9 UART Timing ($t_{READRATE}$ Requirements for Burst Mode) 35
Table 5.10 Command Format (Read) 35
Table 5.11 Response Format (Read) 35
Table 5.12 Command Format (Burst Mode) 36
Table 5.13 Command Format (Write) 36
Table 5.14 UART Data Packet Format (UART Burst / Auto Mode) Example: 16-bit Output 37
Table 5.15 UART Data Packet Format (UART Burst / Auto Mode) Example: 32-bit Output 38
Table 5.16 Data Packet Format (SPI Burst Mode) Example: 16-bit Output 39
Table 5.17 Data Packet Format (SPI Burst Mode) Example: 32-bit Output 39
Table 6.1 Register Map 41
Table 6.2 SET_ERR Flag Output Rate and Filter Setting Table Evaluation Table 49

Revision History

Rev. No.	Date	Page	Description
Rev1.0	4/1/2025	ALL	New release.

Ordering Information

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

Product Model Number	Product Name	Comments
X2G000231000100	M-G355QDG0	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-G355QDG0 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 dose not assume any responsibility or liability of any kind of damage and/or fire coursed 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 BEFOR 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

1. Product Specifications

1.1 Absolute Maximum Ratings

Table 1.1 Absolute Maximum Ratings

Parameter	Min.	Typ.	Max.	Unit
V_{CC} to GND	−0.3	—	4.8	V
Digital Input Voltage to GND	−0.3	—	V_{CC} + 0.3	V
Digital Output Voltage to GND	−0.3	—	V_{CC} + 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_{CC} to GND		3.15	3.3	3.45	V
Digital Input Voltage to GND		GND	—	V_{CC}	V
Digital Output Voltage to GND		−0.3	—	V_{CC} + 0.3	V
Calibration Temperature Range	Performance parameters are applicable	−40	—	85	°C
Operating Temperature Range		-40	—	85	°C

1.3 Legal Regulations and Applicable Standards (Functional Safety)

Compliant with the following standards:
• IEC 61508-1: 2010
• IEC 61508-2: 2010
• IEC 61508-3: 2010

1.4 Characteristics and Electrical Specifications

Table 1.3 Sensor Specifications

T_A = 25 °C, V_{CC} = 3.3 V, angular rate = 0 °/s, ≤ ±1 G, unless otherwise noted

Parameter	Test Conditions / Comments	Min.	Typ.	Max.	Unit
GYRO SENSOR					
Sensitivity					
Output Range		—	±450	—	°/s
Scale Factor	16 bits	-0.2%	66	+0.2%	LSB/(°/s)
	32 bits	-0.2%	66 x (2^{16})	+0.2%	
Nonlinearity (Best fit straight line)	1 σ	—	0.05	—	% of FS
Misalignment	1 σ, Axis-to-axis, Δ = 90° ideal	—	0.01	—	°
Bias					
Initial Error	1 σ, −40 °C ≤ T_A ≤ +85 °C	—	360	—	°/h
Repeatability	1 σ, Turn-on to turn-on *3	—	36	—	°/h
Bias Instability	Average	—	1.2	—	°/h
Angular Random Walk	Average	—	0.08	—	°/√h
Linear Acceleration Effect	Average	—	18	—	(°/h)/G
Noise Density	f = 10 Hz to 20 Hz	—	6.9	—	(°/h)/√Hz, rms
Frequency Property					
3 dB Bandwidth		—	189	—	Hz
ACCELEROMETERS					
Sensitivity					
Output Range		—	±8/±16*4	—	G
Scale Factor	16 bits, Output Range ±8	-0.1%	4	+0.1%	LSB/mG
	32 bits, Output Range ±8	-0.1%	4 x (2^{16})	+0.1%	
	16 bits, Output Range ±16	-0.1%	2	+0.1%	
	32 bits, Output Range ±16	-0.1%	2 x (2^{16})	+0.1%	
Nonlinearity (Best fit straight line)	1 σ, < 1 G	—	0.1	—	% of FS
Misalignment	1 σ, Axis-to-axis, Δ = 90 ° ideal	—	0.01	—	°
Bias					
Initial Error	1 σ, −40 °C ≤ T_A ≤ +85 °C	—	3	—	mG
Repeatability	1 σ, Turn-on to turn-on *3	—	3	—	mG
Bias Instability	Average	—	24	—	µG
Velocity Random Walk	Average	—	0.02	—	(m/s)/√h
Noise Density	f = 10 Hz to 20 Hz	—	50	—	µG/√Hz, rms
Frequency Property					
3 dB Bandwidth		—	148	—	Hz
TEMPERATURE SENSOR					
Scale Factor *1*2	Output = 0 @ +25 °C	—	0.00390625	—	°C/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 1σ 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 °C, V_{CC} = 3.3 V, unless otherwise noted

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
LOGIC INPUTS*1					
Positive Trigger Voltage	LVCMOS Schmitt	1.2	—	2.52	V
Negative Trigger Voltage	LVCMOS Schmitt	0.75	—	1.98	V
Hysteresis Voltage	LVCMOS Schmitt	0.3	—	—	V
Logic 1 Input Current, I_{INH}	V_{IH} = 3.3 V	—	0.1	—	µA
Logic 0 Input Current, I_{INL}	V_{IL} = 0 V	—	0.1	—	µA
Input Capacitance, C_{IN}		—	8	—	pF
RST Voltage Range		0	—	V_{CC} + 0.3	V
RST High-level Input Voltage, V_{IH}		0.8 x V_{CC}	—	—	V
RST Low-level Input Voltage, V_{IL}		—	—	0.2 x V_{CC}	V
RST Low Pulse Width		100	—	—	ms
Pull-up Resistor		32	80	224	kΩ
DIGITAL OUTPUTS*1					
Output High Voltage, V_{OH}	I_{SOURCE} = 1.4 mA LVCMOS	V_{CC} - 0.4	—	—	V
Output Low Voltage, V_{OL}	I_{SINK} = 1.4 mA LVCMOS	—	—	0.4	V
FUNCTIONAL TIMES*2	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		—	—	300	ms
Self Test Time		—	—	80	ms
Filter Setting Time		—	—	1	ms
Attitude_Motion_Profile Setting Time		—	—	1	ms
Data Output Rate	DOUT_RATE = 0x08	—	—	400	Sps
Clock Accuracy		—	—	±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.

1.5 Timing Specifications

Table 1.5 Timing Specifications

T_A = 25 °C, V_{CC} = 3.3 V, unless otherwise noted

Parameter	Description	Min.	Typ.	Max.	Unit
NORMAL MODE					
f_{SCLK}		0.01	—	2.0	MHz
t_{STALL}	Stall period between data	20	—	—	µs
$t_{WRITERATE}$	Write rate	40	—	—	µs
$t_{READRATE}$	Read rate	40	—	—	µs
BURST MODE					
f_{SCLK}		0.01	—	1.0	MHz
t_{STALL1}	Stall period between data	45	—	—	µs
t_{STALL2}	Stall period between data	4	—	—	µs
$t_{READRATE2}$	Read rate	32	—	—	µs
COMMON					
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.

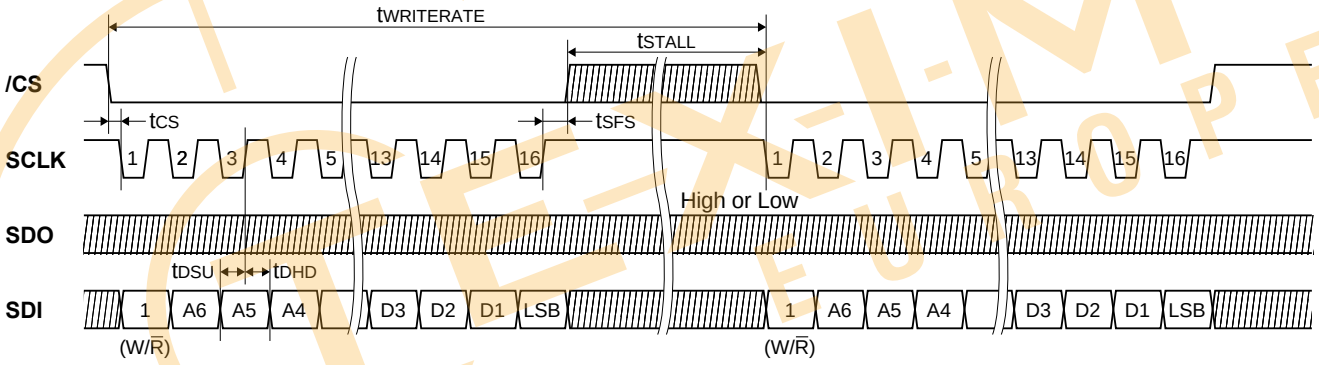


Figure 1.1 SPI Write Timing and Sequence

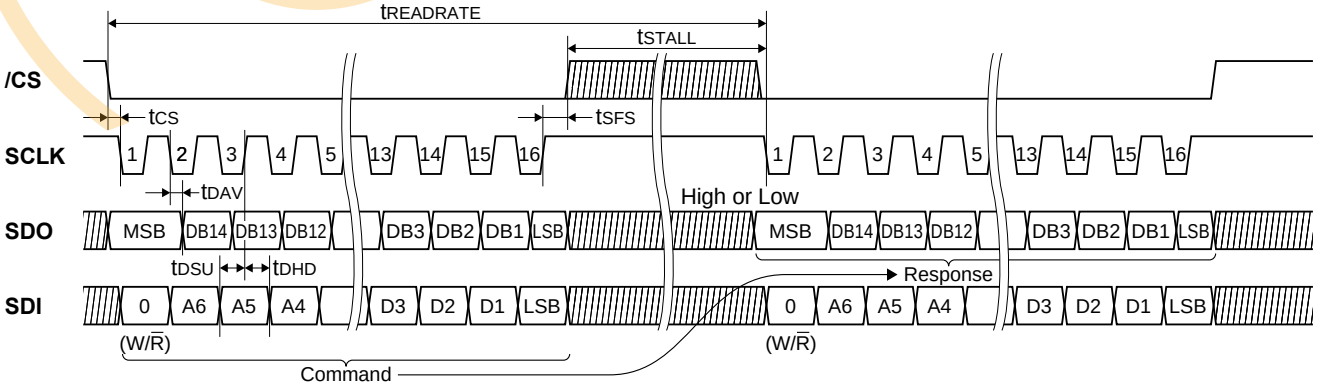


Figure 1.2 SPI Read Timing and Sequence

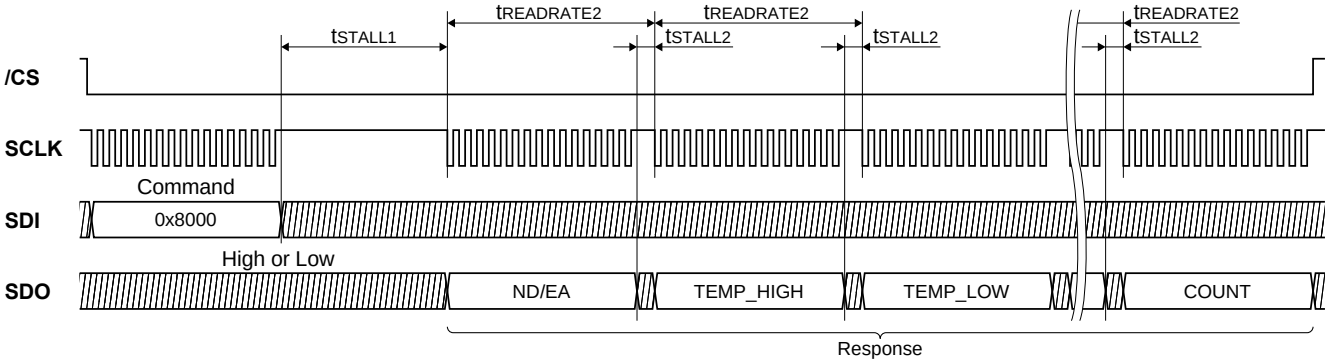


Figure 1.3 SPI Read Timing and Sequence (BURST MODE)

1.6 Connector Pin Layout and Functions

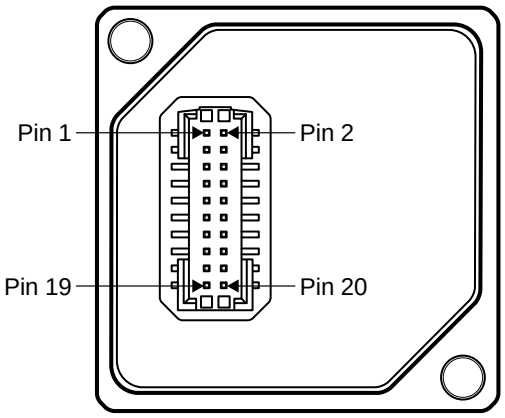


Figure 1.4 Connector Pin Assignment

Table 1.6 Pin Function Descriptions

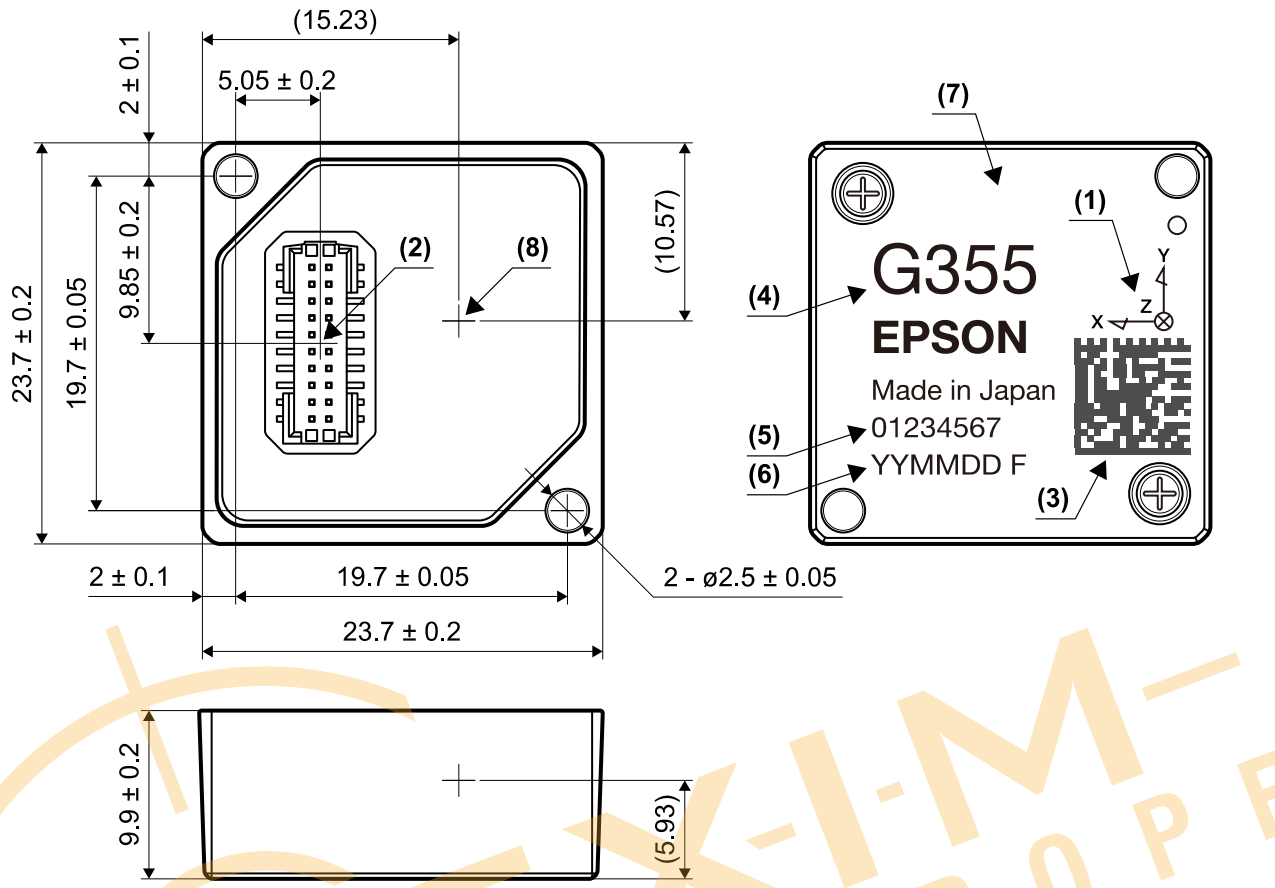
Pin No.	Mnemonic	Type*1	Description
1	SCLK	I	SPI Serial Clock *2
2	SDO	O	SPI Data Output *2
5	SDI	I	SPI Data Input *2
6	/CS	I	SPI Chip Select *2
7	SOUT	O	UART Data Output *2
9	SIN	I	UART Data Input *2
13	DRDY (GPIO1)	I/O	Data Ready *3 (General Purpose I/O1)
14	GPIO2 (EXT)	I/O	General Purpose I/O2 *4 (External Trigger Input or External Counter Reset Input)
16	/RST	I	Reset *5
10, 11, 12	V_{CC}	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_{CC} 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_{CC}) 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-G355QDG0 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.

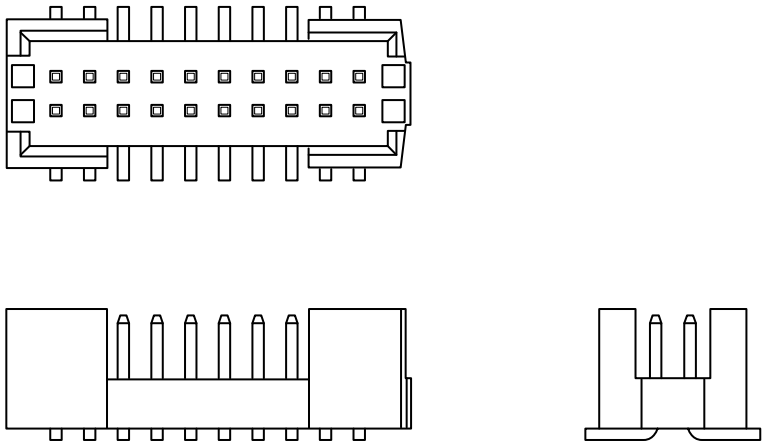


Figure 2.2 Header Pin Dimensions

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

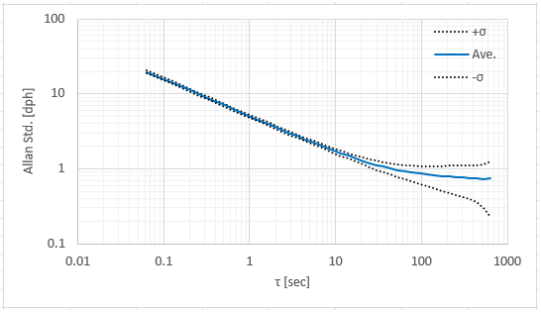


Figure 3.1 Gyro Allan Variance Characteristic

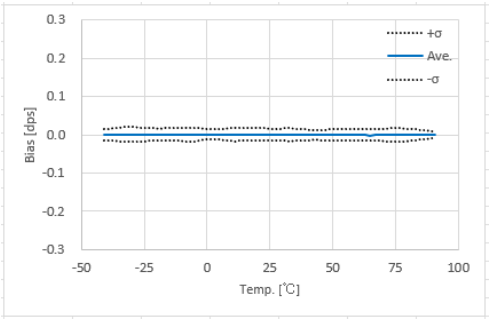


Figure 3.2 Gyro Bias vs. Temperature Characteristic

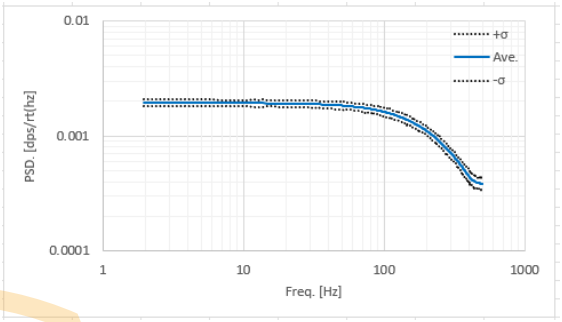


Figure 3.3 Gyro Noise Frequency Characteristic

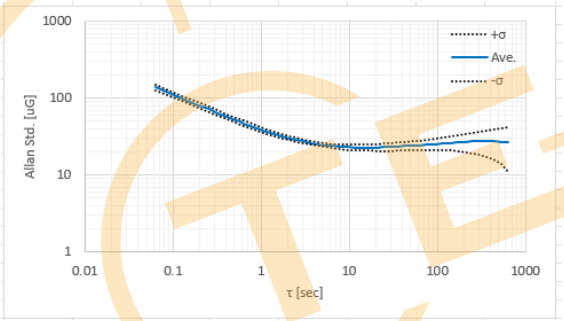


Figure 3.4 Accelerometer Allan Variance Characteristic

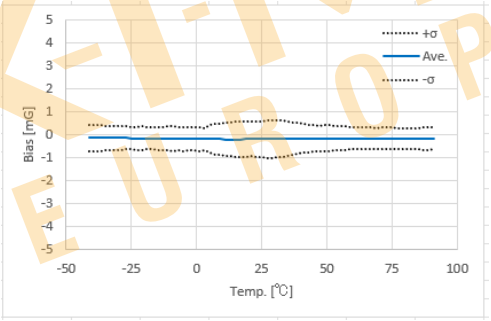


Figure 3.5 Accelerometer Bias vs. Temperature

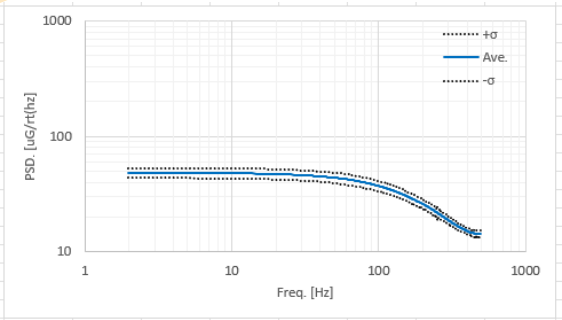


Figure 3.6 Accelerometer Noise Frequency Characteristic

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.

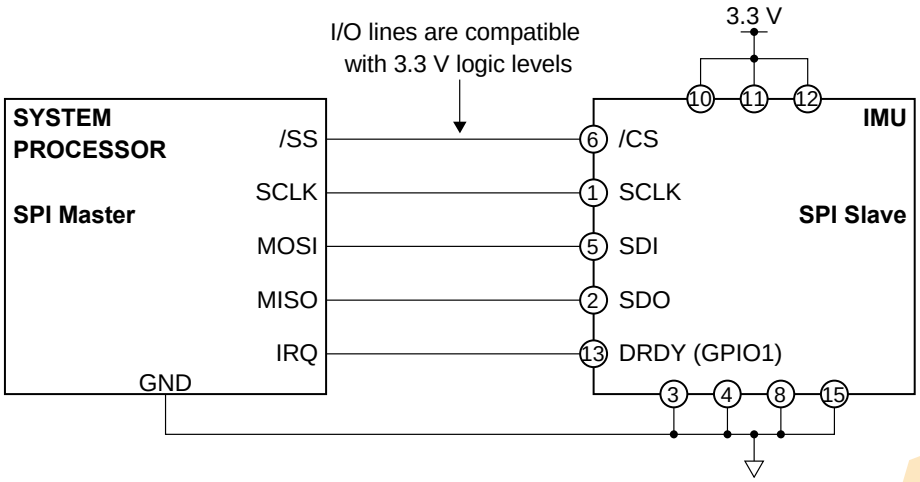


Figure 4.1 SPI Connection

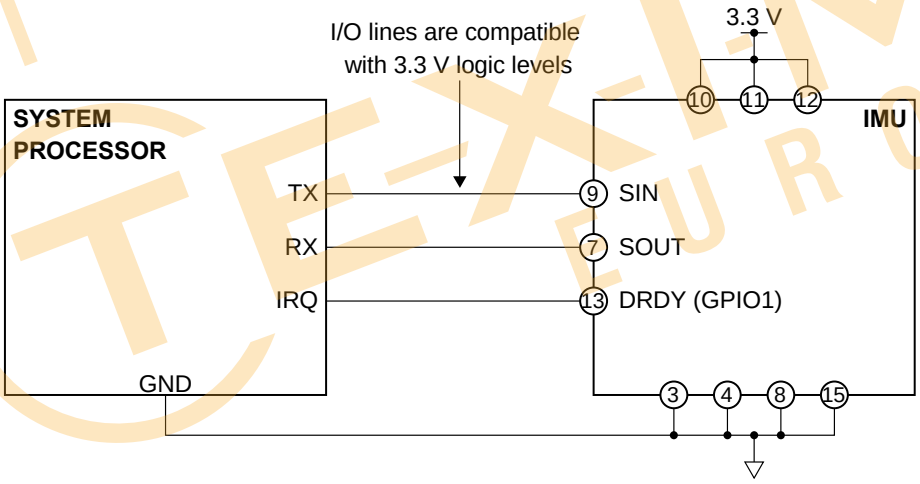


Figure 4.2 UART Connection

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 (*1). 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 (*2).

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.

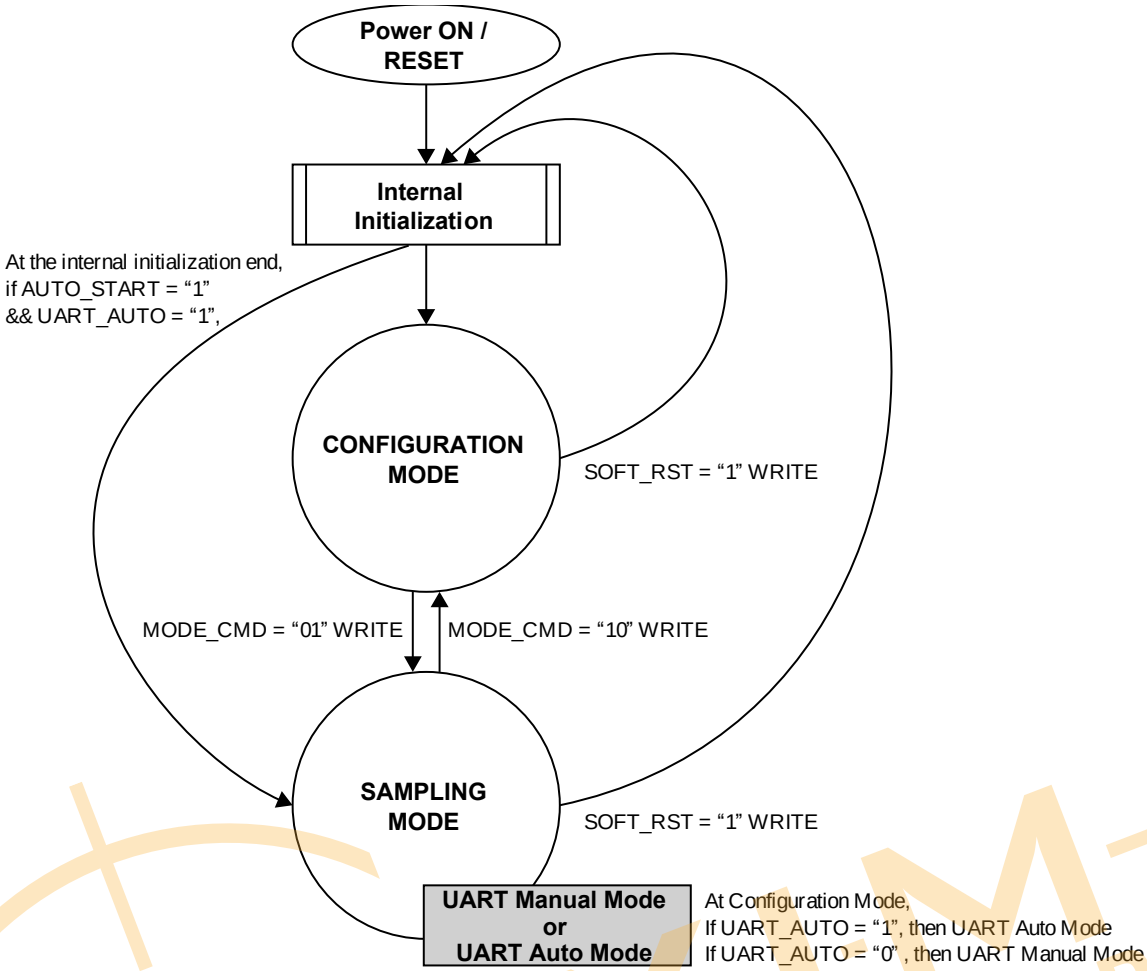


Figure 4.3 Operational State Diagram

4.3 Functional Block Diagram

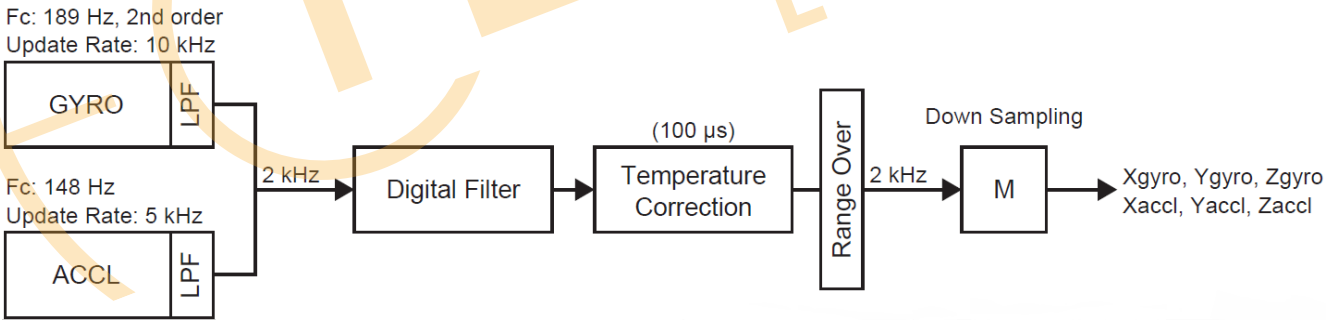


Figure 4.4 Functional Block Diagram

4.4 Data Output Timing

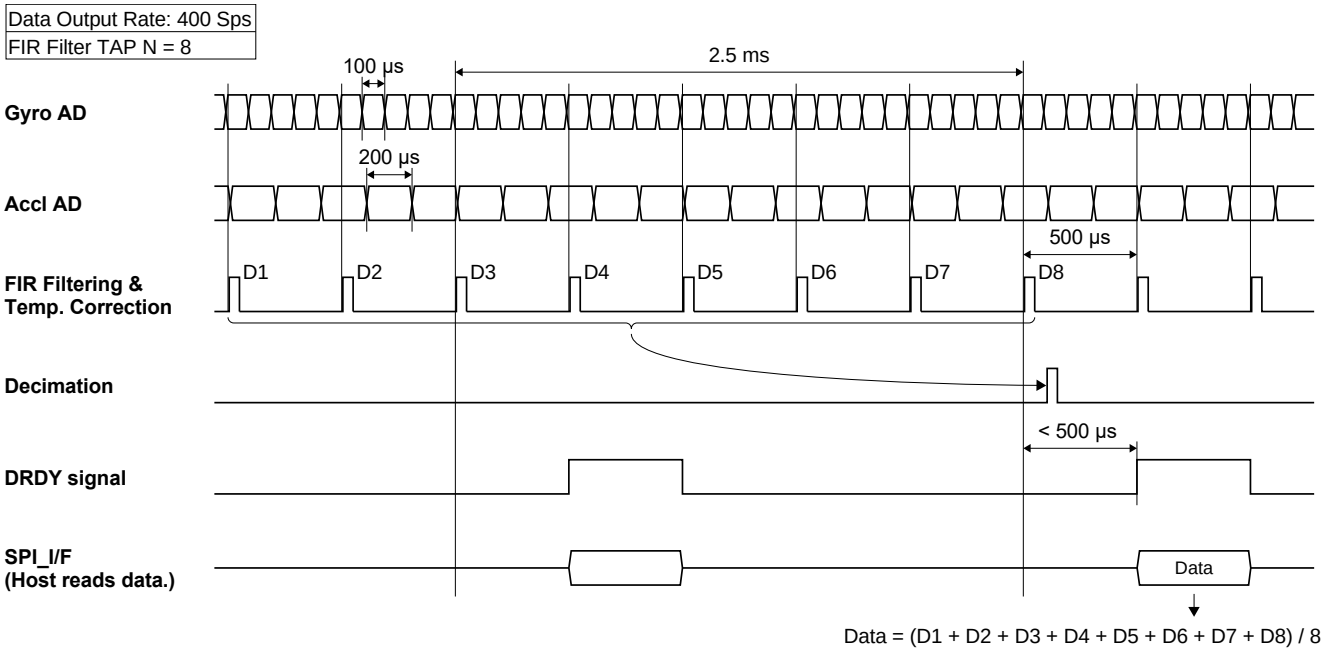


Figure 4.5 Data Output Timing – Data Output Rate 400 Sps, FIR Filter TAP N = 8

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.

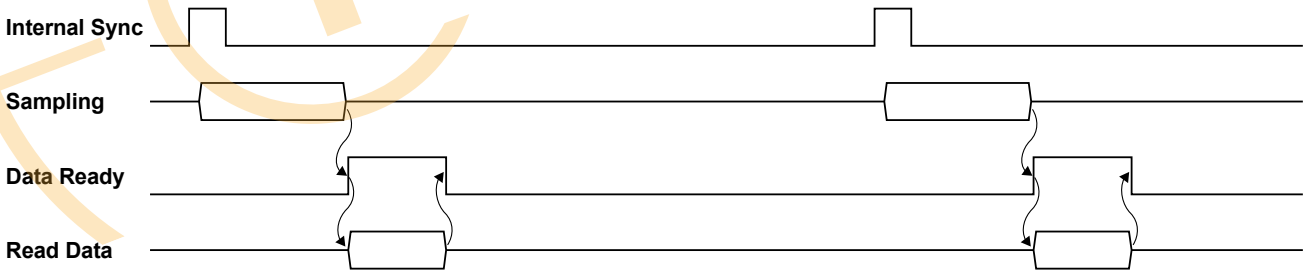


Figure 4.6 Data Ready Signal Timing

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 µsec/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 Diagnostic Functions

4.8.1 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”

4.8.2 On-board Diagnostics Function

The onboard diagnostic function checks for abnormalities in this sensor during startup and sampling. If it finds an abnormality, the **ODB** flag (FLAG[0x06 (W0)] bit[1]) is set. Diagnostic results can be read from the OB_DIAG_STAT[0x2A (W0)] register.

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** = “00”), 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.

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 4.7, and Figure 4.8.

Table 4.1 External Trigger Input Timing Requirements

Parameter	Description	Min.	Max.	Unit
t_{ETW}	External Trigger Input width	100	–	ns
t_{ETC}*1	External Trigger Input cycle	2.5	1000	ms
t_{ETA2T_G}	Time from Gyro ADC's completion to External Trigger Input (This is a precision of External Sync.)	0	600	µs
t_{ETA2T_A}	Time from Accl ADC's completion to External Trigger Input (This is a precision of External Sync.)	0	700	µs
t_{ETD1}*2	Delay time from External Trigger Input to DRDY asserted	–	300	µs

*1) For normal angular rate and linear acceleration output (ATTI_ON = “00”), 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.

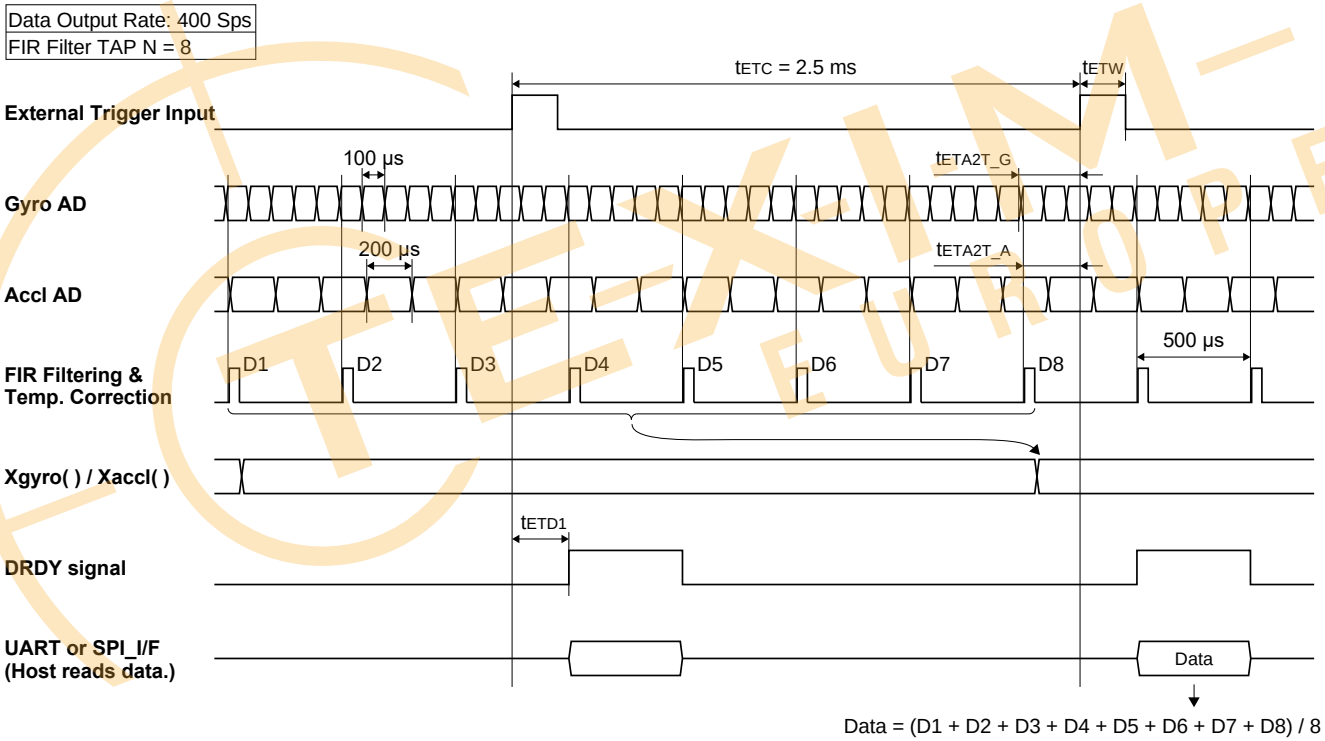


Figure 4.7 External Trigger Input (UART / SPI Manual Mode)

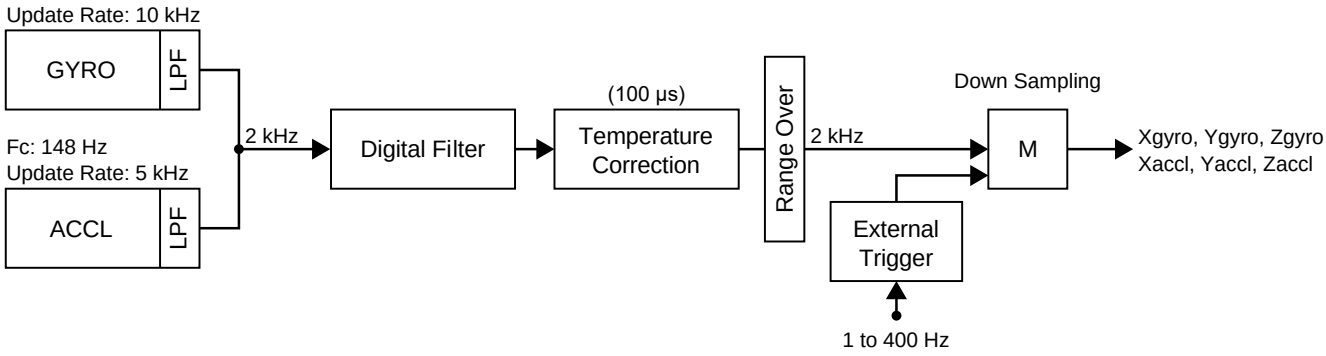


Figure 4.8 External Trigger input Functional Block Diagram

4.10 External Counter Reset Input

The External Counter Reset Input feature allows you to measure the time offset between the next sampling instant from an externally supplied input trigger on the GPIO2 (EXT) pin.

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 (*1) when Sampling mode begins. The counting resolution is 16 µs.
- 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 sampling timing 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 sampling timing by reading the sampling data with the counter value when Data Ready signal is asserted.
- The counter stops counting (*2) 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 4.9.

Table 4.2 External Counter Reset Input Timing

Parameter	Description	Min.	Max.	Unit
t_{ERW}	External Reset Input Width	100	–	ns
t_{ERC}	External Reset Input Cycle	5	1000	ms
t_{ER2S}	Time from External Reset Input to Sampling	(count*1 x16) + Δ_{ER2S}		µs
Δ_{ER2S}	Precision of t_{ER2S}	-150	150	µs

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

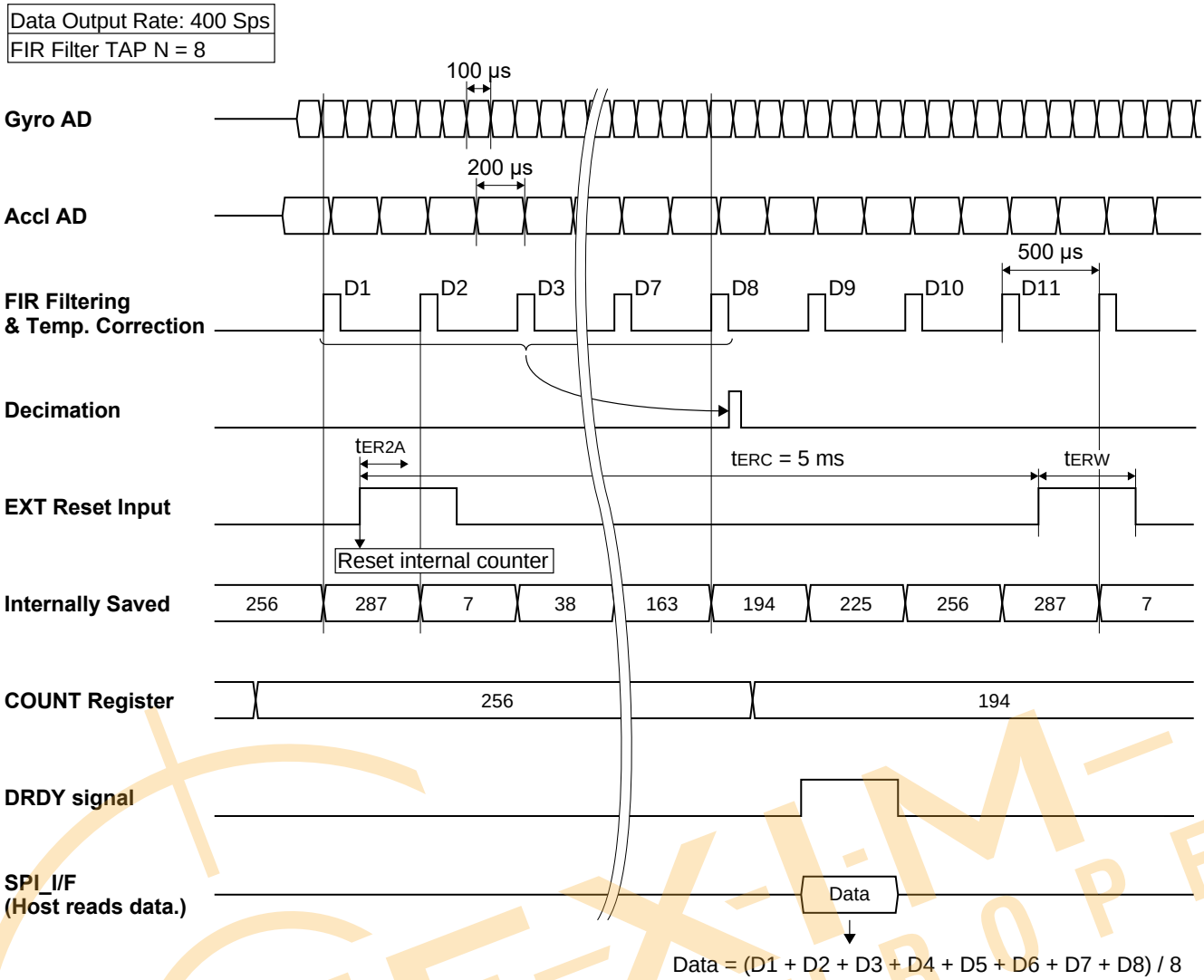


Figure 4.9 External Counter Reset Input

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”

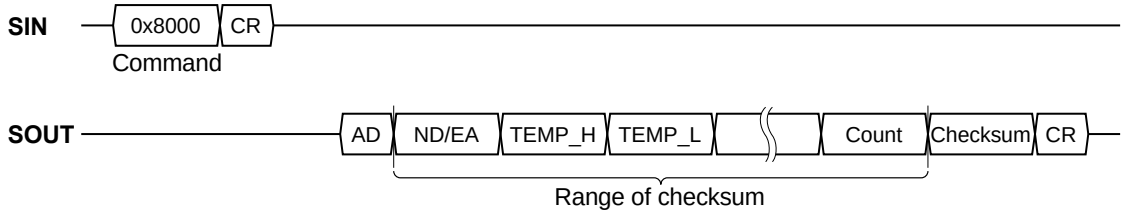


Figure 4.10 Checksum

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 = 8, 16, 32, 64, or 128.
Figure 4.11 shows the characteristics of this filter.

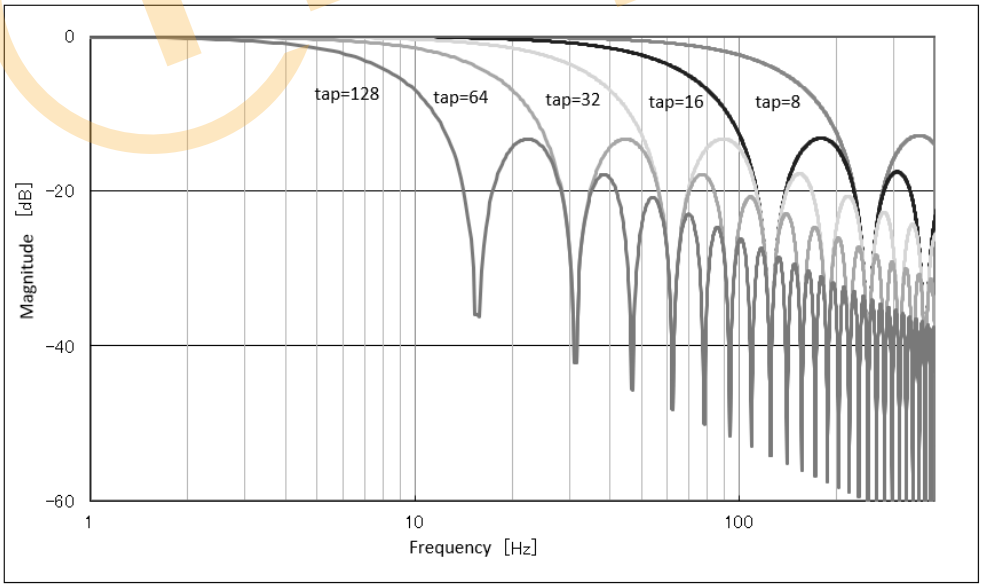


Figure 4.11 Moving Average Filter Characteristics

(2) FIR Kaiser Filter

Uses Kaiser Window (parameter = 8).
TAP setting can be N = 32, 64, or 128 with cutoff frequency fc = 50, 100, or 200 Hz.
Figure 4.12 and Figure 4.13 show the typical characteristic of this filter.

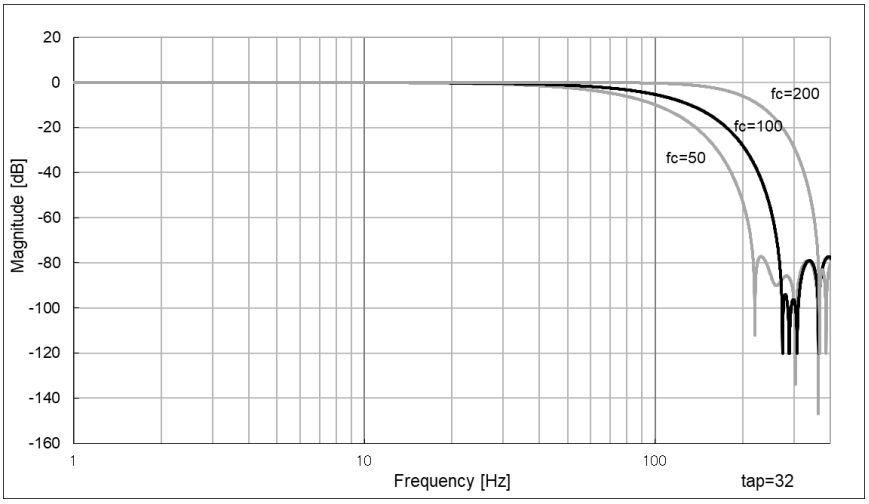


Figure 4.12 FIR Kaiser Filter Typical Characteristic 1

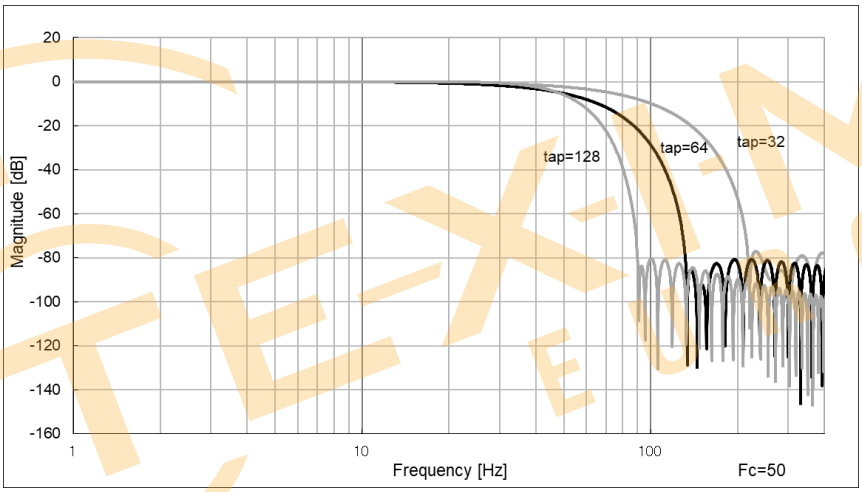


Figure 4.13 FIR Kaiser Filter Typical Characteristic 2

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

	TAP8	TAP16	TAP32	TAP64	TAP128
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	TAP64 Fc50	TAP64 Fc100	TAP64 Fc200	TAP128 Fc50	TAP128 Fc100	TAP128 Fc200
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 Range Over Function

This device supports the notification when a range over condition is detected in the sensor data. The range over thresholds are as follows.

Gyro Sensor: ±450 [deg/s]

Accelerometer: ±7.5 [G] for Output Range ±8 G (**A_RANGE_CTRL** of GLOB_CMD3[0x13 (W1)] bit[8] = “0”)
±15 [G] for Output Range ±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.

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, 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:
- **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]) = 0x08: 400 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 MHz and t_{STALL} = 24 µs for normal mode
- **DOUT_RATE** (SMPL_CTRL[0x04 (W1)] bit[15:8]) = 0x08: 400 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]) = 0x08: 400 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 MHz and t_{STALL} = 24 µs for normal mode
- **DOUT_RATE** (SMPL_CTRL[0x04 (W1)] bit[15:8]) = 0x08: 400 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	–	µs
$t_{WRITERATE}$	40	–	µs
$t_{READRATE}$	40	–	µs

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).

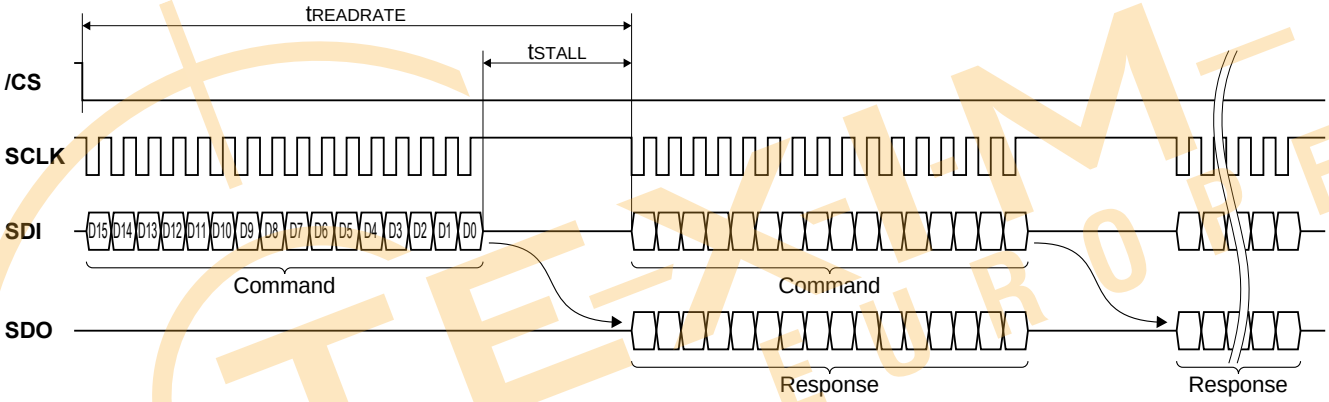


Figure 5.1 SPI Read Timing (Normal Mode)

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.

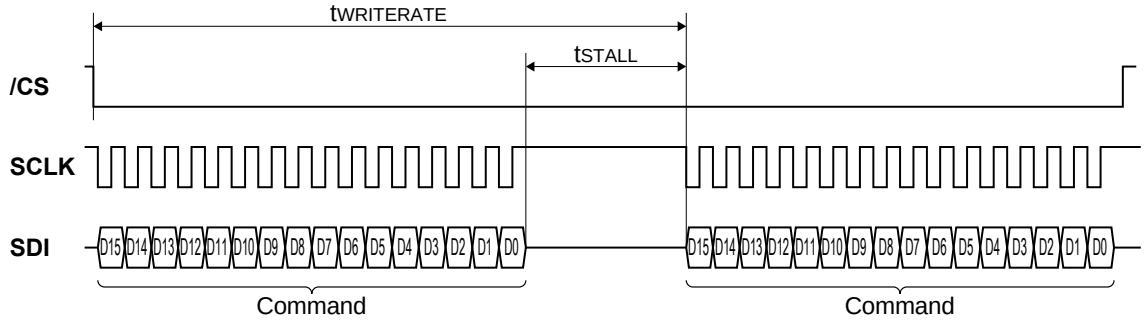


Figure 5.2 SPI Write Timing (Normal Mode)

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
f_{SCLK}	0.01	1.0	MHz
t_{STALL1}	45	–	µs
t_{STALL2}	4	–	µs
$t_{READRATE2}$	32	–	µs

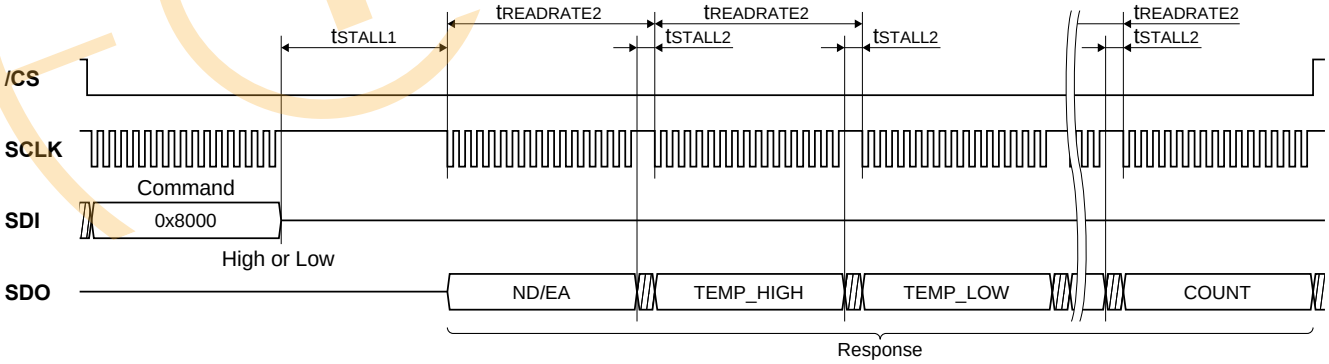


Figure 5.3 SPI Timing (Burst Mode)

NOTE: SPI Burst Read mode can support t_{STALL2} = 0 µs 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 µs.
- t_{STALL2}: 0 µs
- $t_{READRATE2}$: Minimum 32 µs.
- 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)

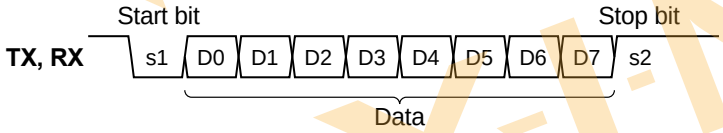


Figure 5.4 UART Bit Format

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.	
t_{STALL}	(230.4 kbps)	–	25	–	70	–	–*2	µs
t_{STALL}	(460.8 kbps)	–	25	–	70	–	–*2	µs
t_{STALL}	(921.6 kbps)	–	25	–	70	–	–*2	µs
$t_{WRITERATE}$	(230.4 kbps)	350	–	–	–	350	–	µs
$t_{WRITERATE}$	(460.8 kbps)	200	–	–	–	200	–	µs
$t_{WRITERATE}$	(921.6 kbps)	150	–	–	–	150	–	µs
$t_{READRATE}$	(230.4 kbps)	350	–	*1	–	–*2	–	µs
$t_{READRATE}$	(460.8 kbps)	200	–	*1	–	–*2	–	µs
$t_{READRATE}$	(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 ($t_{READRATE}$ Requirements for Burst Mode)

Parameter	Burst Mode (Min.)	Unit
$t_{READRATE}$ (230.4kbps)	300 + (43.4 x B)	µs
$t_{READRATE}$ (460.8kbps)	200 + (21.7 x B)	µs
$t_{READRATE}$ (921.6kbps)	150 + (10.9 x B)	µs

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

Example $t_{READRATE}$ Calculation:

BURST_CTRL1[0x0C (W1)]: Set value 0xF006
BURST_CTRL2[0x0E (W1)]: Set value 0x7000
B = 34 bytes for the above stated register setting
$t_{READRATE}$ (460.8 kbps) = 200 + (21.7 x 34) = 937.8 (µs)

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.

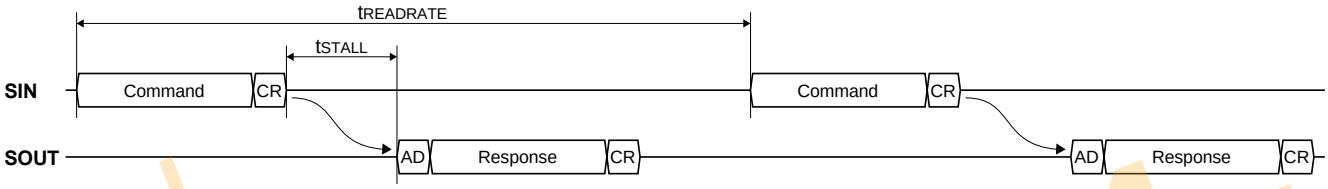


Figure 5.5 UART Read Timing (Normal Mode)

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.

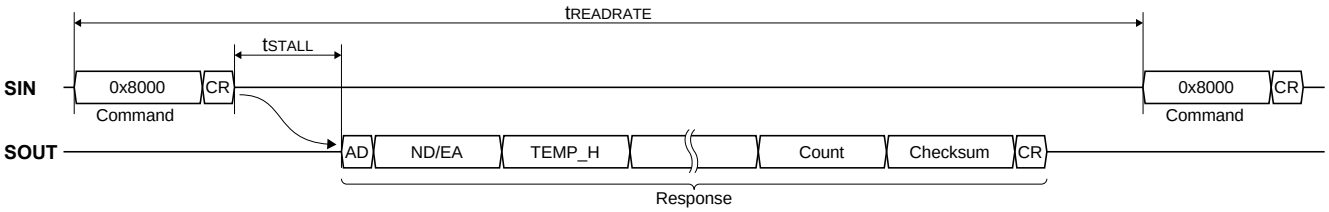


Figure 5.6 UART Read Timing (Burst Mode)

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.

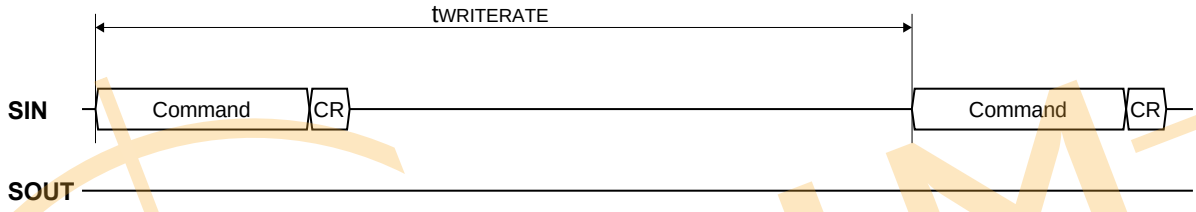


Figure 5.7 UART Write Timing

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)].

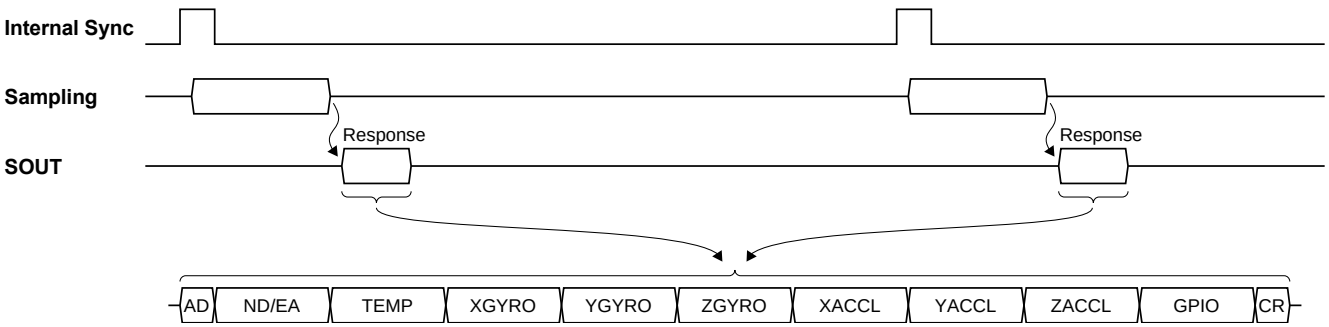


Figure 5.8 UART Auto Mode Operation

5.3 Data Packet Format

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)] =0 xF007 / 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 the 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 the Table 6.1. The control registers with ○ 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 6.26 describes the registers in detail.

The “-” sign in the register assignment table in Section 6.1 through Section 6.26 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)]: The 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, 0x01	W		0xFFFF	Burst mode
MODE_CTRL	0	0x02, 0x03	R/W		0x0400	Operation mode control
DIAG_STAT	0	0x04, 0x05	R		0x0000	Diagnostic results
FLAG	0	0x06, 0x07	R		0x0000	ND flag / EA flag
GPIO	0	0x08, 0x09	R/W		0x0200	GPIO
COUNT	0	0x0A, 0x0B	R		0x0000	Sampling count value
RANGE_OVER	0	0x0C, 0x0D	R		0x0000	Range Over
TEMP_HIGH	0	0x0E, 0x0F	R		0xFFFF	Temperature sensor value High
TEMP_LOW	0	0x10, 0x11	R		0xFFFF	Temperature sensor value Low
XGYRO_HIGH	0	0x12, 0x13	R		0xFFFF	X gyroscope sensor value High
XGYRO_LOW	0	0x14, 0x15	R		0xFFFF	X gyroscope sensor value Low
YGYRO_HIGH	0	0x16, 0x17	R		0xFFFF	Y gyroscope sensor value High
YGYRO_LOW	0	0x18, 0x19	R		0xFFFF	Y gyroscope sensor value Low
ZGYRO_HIGH	0	0x1A, 0x1B	R		0xFFFF	Z gyroscope sensor value High
ZGYRO_LOW	0	0x1C, 0x1D	R		0xFFFF	Z gyroscope sensor value Low
XACCL_HIGH	0	0x1E, 0x1F	R		0xFFFF	X acceleration sensor value High
XACCL_LOW	0	0x20, 0x21	R		0xFFFF	X acceleration sensor value Low
YACCL_HIGH	0	0x22, 0x23	R		0xFFFF	Y acceleration sensor value High
YACCL_LOW	0	0x24, 0x25	R		0xFFFF	Y acceleration sensor value Low
ZACCL_HIGH	0	0x26, 0x27	R		0xFFFF	Z acceleration sensor value High
ZACCL_LOW	0	0x28, 0x29	R		0xFFFF	Z acceleration sensor value Low
OB_DIAG_STAT	0	0x2A, 0x2B	R		0x0000	On-board diagnostic results
ID	0	0x4C, 0x4D	R		0x5345	ID read function
SIG_CTRL	1	0x00, 0x01	R/W	○	0xFE00	Data Ready signal & polarity control
MSC_CTRL	1	0x02, 0x03	R/W	○	0x0006	Other control
SMPL_CTRL	1	0x04, 0x05	R/W	○	0x0803	Sampling control
FILTER_CTRL	1	0x06, 0x07	R/W	○	0x0003	Filter control
UART_CTRL	1	0x08, 0x09	R/W	○	0x0000	UART control
GLOB_CMD	1	0x0A, 0x0B	R/W	○	0x0000	System control
BURST_CTRL1	1	0x0C, 0x0D	R/W	○	0xF006	Burst control 1
BURST_CTRL2	1	0x0E, 0x0F	R/W	○	0x0000	Burst control 2
POL_CTRL	1	0x10, 0x11	R/W	○	0x0000	Polarity control
GLOB_CMD3	1	0x12, 0x13	R/W	○	0x00CC	Acceleration range
PROD_ID1	1	0x6A, 0x6B	R		*2	Product ID
PROD_ID2	1	0x6C, 0x6D	R		*2	
PROD_ID3	1	0x6E, 0x6F	R		*2	
PROD_ID4	1	0x70, 0x71	R		*2	
VERSION *1	1	0x72, 0x73	R		*2	Version
SERIAL_NUM1	1	0x74, 0x75	R		*2	Serial number
SERIAL_NUM2	1	0x76, 0x77	R		*2	
SERIAL_NUM3	1	0x78, 0x79	R		*2	
SERIAL_NUM4	1	0x7A, 0x7B	R		*2	
WIN_CTRL	0,1	0x7E, 0x7F	R/W		0x0000	Register window control

*1 Version is subject to change without notice.
*2 Refer to each register description 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 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**	**-**		**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 diagnostics 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 if 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 when a samplind starts (see Table 6.2).

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	**-**						**OBD**	**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[1] **OBD (On-Board Diagnostics Error) flag**
When at least one failure is found in the on-board diagnostic results (OB_DIAG_STAT[0x2A (W0)]), the flag is set to “1” (failure occurred).

bit[0] **EA (All Error) flag**
When at least one failure is found in the diagnostic results (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 at the sampling timing.

NOTE: The time unit of the sampling counter value represents 500 µs/count.
Example: If the data output rate equals 400 Sps, the counter value sequence is 5, 10, 15, ... , 65535, 4, 9, ...

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 0	R/W
0x0C	**-**			**-**

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 [℃] = (SF / 65536) x A + 25

For 16-bit usage: T [℃] = SF x 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-G355QDG0 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 [deg/s] = ((1 / SF) / 65536) x B

For 16-bit usage: G [deg/s] = (1 / SF) x 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-G355QDG0 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 [mG] = ((1 / SF) / 65536) x C

For 16-bit usage: A [mG] = (1 / SF) x C

SF: Scale Factor
C: Acceleration sensor output data (decimal)

6.11 OB_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
0x2B	WHO_AM_I_ERR	-	GYRO_ERR (XGyro)	GYRO_ERR (YGyro)	GYRO_ERR (ZGyro)	ACCL_ERR (XACCL)	ACCL_ERR (YACCL)	ACCL_ERR (ZACCL)	R

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x2A	FDSNS_ERR (XGyro)	FDSNS_ERR (YGyro)	FDSNS_ERR (ZGyro)	-			COMPU_ERR	PRG_SEQ_ERR	R

NOTE: When the host reads the diagnostic results, all the results will be cleared to 0.

bit[15] WHO_AM_I_ERR

Shows the results of the WHO_AM_I check for on-board diagnostics during startup and sampling.

1: Abnormal
0: Normal

bit[13:11] GYRO_ERR

Shows the diagnostic results of the X, Y, and Z-axis gyro sensors for on-board diagnostics during startup and sampling.

1: Abnormal
0: Normal

bit[10:8] ACCL_ERR

Shows the diagnostic results of the X, Y, and Z-axis acceleration sensors for on-board diagnostics during startup and sampling.

1: Abnormal
0: Normal

bit[7:5] FDSNS_ERR

Shows the results of the fault diagnosis sensors check for on-board diagnostics at startup.

1: Abnormal
0: Normal

bit[1] **COMPU_ERR**
Shows the results of the arithmetic operation check for on-board diagnostics during startup and sampling.

1: Abnormal
0: Normal

bit[0] **PRG_SEQ_ERR**
Shows the results of the program sequence monitor for on-board diagnostics during sampling.

1: Abnormal
0: Normal

6.12 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.

6.13 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 0	R/W
0x00	**-**			**-**

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

6.14 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

6.15 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.

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: 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])													
Setting	Sps	8	16	32	64	128	32 fc50	32 fc100	32 fc200	64 fc50	64 fc100	64 fc200	128 fc50	128 fc100	128 fc200
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

6.16 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	-		**FILTER_STAT**	**FILTER_SEL**					**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 fc in Hz.

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 fc = 50
01001: FIR Kaiser filter (parameter = 8) TAP = 32 and fc = 100
01010: FIR Kaiser filter (parameter = 8) TAP = 32 and fc = 200
01100: FIR Kaiser filter (parameter = 8) TAP = 64 and fc = 50
01101: FIR Kaiser filter (parameter = 8) TAP = 64 and fc = 100
01110: FIR Kaiser filter (parameter = 8) TAP = 64 and fc = 200
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
00000–00010, 01011, 01111, 10011–11111: 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.

6.17 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	**-**						**BAUD _RATE**		**R/W**

Addr (Hex)	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	R/W
0x08	**-**						**AUTO _START**	**UART _AUTO**	**R/W**

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 infomation 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.

6.18 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].

6.19 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**	**-**				**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[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.

6.20 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**	**-**				**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

6.21 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).

1: Inverted
0: Not inverted

6.22 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 0	R/W
0x12	**-**			**-**

bit[8] A_RANGE_CTRL
This bit sets the output range of the accelerometer.

1: ±16 G
0: ±8 G

6.23 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-G355QDG0
PROD_ID1	0x3347
PROD_ID2	0x3535
PROD_ID3	0x4451
PROD_ID4	0x3047

6.24 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.

6.25 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

6.26 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).

7.2 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.
- If a diagnostic error is set in the on-board fault diagnosis result, restart this product. For details about diagnostic errors, refer to *4.8.2 On-board Diagnostics Function*.

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 accidently 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, USA
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.cm

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.
JR Shinjuku Miraina Tower, 4-1-6 Shinjuku, Shinjuku-ku, Tokyo, 160-8801, Japan
Phone: +81-3-6682-4322 FAX: +81-3-6682-5016

Product Information on www server
https://global.epson.com/products_and_drivers/sensing_system/

Disclaimer

ALL PRODUCTS, PRODUCT SPECIFICATIONS AND DATA ARE SUBJECT TO CHANGE WITHOUT NOTICE TO IMPROVE RELIABILITY, FUNCTION OR DESIGN OR OTHERWISE.

Texim Europe B.V. its affiliates, agents, and employees, and all persons acting on its or their behalf (collectively, “Texim”), disclaim any and all liability for any errors, inaccuracies or incompleteness contained in any datasheet or in any other disclosure relating to any product.

Texim makes no warranty, representation or guarantee regarding the suitability of the products for any particular purpose or the continuing production of any product.

It is the customer’s responsibility to validate that a particular product with the properties described in the product specification is suitable for use in a particular application.

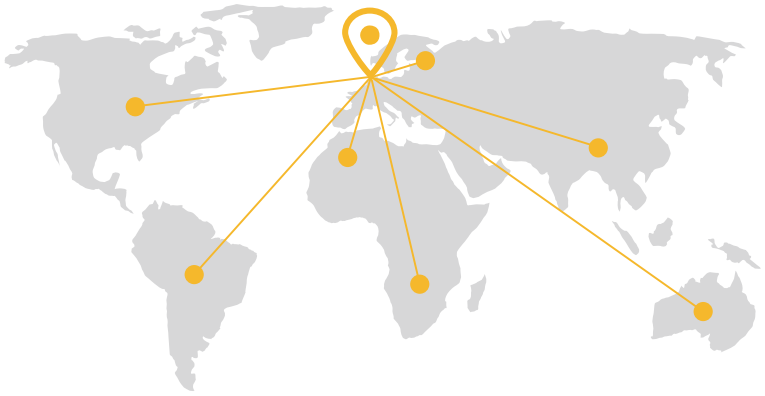
Parameters provided in datasheets and / or specifications may vary in different applications and performance may vary over time.

All operating parameters, including typical parameters, must be validated for each customer application by the customer’s technical experts.

Please contact us if you have any questions about the contents of the datasheet.

This may not be the latest version of the datasheet. Please check with us if a later version is available.

Texim Europe - contact details



Headquarters & Warehouse

Elektrostraat 17
NL-7483 PG Haaksbergen
The Netherlands

T: +31 (0)53 573 33 33
E: info@texim-europe.com
Homepage: www.texim-europe.com

The Netherlands

Elektrostraat 17
NL-7483 PG Haaksbergen

T: +31 (0)53 573 33 33
E: nl@texim-europe.com

Belgium

Zuiderlaan 14, box 10
B-1731 Zellik

T: +32 (0)2 462 01 00
E: belgium@texim-europe.com

UK & Ireland

St Mary's House, Church Lane
Carlton Le Moorland
Lincoln LN5 9HS

T: +44 (0)1522 789 555
E: uk@texim-europe.com

Germany

Bahnhofstrasse 92
D-25451 Quickborn

T: +49 (0)4106 627 07-0
E: germany@texim-europe.com

Germany

Martin-Kollar-Strasse 9
D-81829 München

T: +49 (0)89 436 086-0
E: muenchen@texim-europe.com

Austria

Warwitzstrasse 9
A-5020 Salzburg

T: +43 (0)662 216 026
E: austria@texim-europe.com

Nordic

Stockholmsgade 45
2100 Copenhagen

T: +45 88 20 26 30
E: nordic@texim-europe.com

Italy

Martin-Kollar-Strasse 9
D-81829 München

T: +49 (0)89 436 086-0
E: italy@texim-europe.com