



**MS**  
**MCIP**

**COMMUNICATION  
INTERFACE  
PROTOCOL**

DOC00419  
REVISION M

## CONTENTS

1.0	Overview .....	4
2.0	Communications Structure .....	4
2.1	Overview .....	4
2.2	Header .....	5
2.3	Payload .....	5
2.4	Checksum .....	5
Table 1 - Ping Message Checksum example .....		6
2.5	Error Codes .....	6
Table 2 – Error Codes .....		6
2.6	Implementation Considerations .....	6
3.0	Messages .....	7
3.1	Base Messages .....	7
Ping Command 0x0102 .....		7
Table 3 - Ping Message .....		7
Table 4 – Ping Reply Message, Ack .....		7
Get Device Messages 0x0103 .....		8
Table 5 – Get Device Messages .....		8
Table 5 –Device Messages Reply .....		8
Device Reset 0x0104 .....		9
Table 6 –Device Reset Command .....		9
Table 7 –Device Reset Reply .....		9
Get Device Model 0x0105 .....		10
Table 8 –Get Device Model .....		10
Table 9 –Get Device Model Reply .....		11
Get Device SN 0x0106 .....		12
Table 10 –Get Device SN .....		12
Table 11 –Get Device SN Reply .....		13
Get Device FW 0x0107 .....		14
Table 12 –Get Device FW .....		14
Table 13 –Get Device FW Reply .....		15
Get Device Cal 0x0108 .....		16
Table 14 –Get Device Cal .....		16
Table 15 –Get Device Cal Reply .....		17
Correlate GPS Time 0x0109 .....		18
3.2	IMU Configuration Messages 0x02 .....	19
UART Baud Rate Configure 0x0201 .....		19
Configure Filter 0x0203 .....		21
IMU Sample Rate Configure 0x0204 .....		23
Select Sensors 0x0205 .....		25
Select Sensors errata .....		25
Get IMU Internal Sample Rate 0x0206 .....		27
Table 34 – Get IMU Internal Sample Rate Reply .....		27
Config Accel Range 0x0207 .....		28
Configure Gyro Range 0x0208 .....		30
Configure All 0x0209 .....		32
Data On/Off 0x020A .....		33
XTRIG On/Off 0x020B .....		35
Select Sensors Revision B 0x020C .....		37
Config Aux Accel Range 0x020D .....		39
3.3	Data Messages .....	41

IMU Data Message 0xA2 .....	41
Table 62 - IMU Data Message 0xA2.....	41
Scaled Acceleration Vector Data Message 0xA281 .....	42
Table 63 – Scaled Acceleration Vector Data Message 0xA281.....	42
Scaled Angular Rate Vector Data Message 0xA282 .....	43
Table 64 – Scaled Angular Rate Vector Data Message 0xA282 .....	43
Scaled Magnetic Field Vector Data Message 0xA283 .....	44
Table 65 – Scaled Magnetic Field Vector Data Message 0xA283 .....	44
Scaled Delta Theta Vector Data Message 0xA284 .....	45
Table 66 – Scaled Delta Theta Vector Data Message 0xA284 .....	45
Scaled Delta Velocity Vector Data Message 0xA285.....	46
Table 67 – Scaled Delta Velocity Vector Data Message 0xA285.....	46
Scaled Pressure Data Message 0xA286.....	47
Table 68 – Scaled Pressure Data Message 0xA286.....	47
Scaled Temperature Data Message 0xA287.....	48
Table 69 – Scaled Temperature Data Message 0xA287 .....	48
GPS Correlated Time Data Message 0xA288.....	49
Table 70 – GPS Correlated Time Flag Bit Descriptions .....	49
Table 71 – GPS Correlated Time Data Message 0xA288 .....	50
Scaled Aux Accel Vector Data Message 0xA289.....	51
Table 72 – Scaled Aux Accel Vector Data Message 0xA281 .....	51
3.1 Error Codes.....	<b>Error! Bookmark not defined.</b>
IMU Data Message 0xA2 .....	<b>Error! Bookmark not defined.</b>
Revision History .....	52

## 1.0 OVERVIEW

The Memsense Communication Interface Protocol (MS CIP) is implemented as a simple architecture to communicate information to and from the measurement device. The protocol is intended to be flexible in allowing customers to configure various features of the device achieving optimized communication modes for various application requirements. In cases where further functionality is required to achieve an optimal configuration please contact Memsense to initiate the process of requesting further functionality.

## 2.0 COMMUNICATIONS STRUCTURE

### 2.1 OVERVIEW

The Memsense Communication Interface Protocol utilizes messages to pass information between the host system and the measurement device. There are three types of messages including command, reply, and data messages. Commands are transmitted to the device to affect a change in the configuration or request information from the device. Replies are sent from the device in response to a command received. Data messages are sent from the device to transmit measurement data.

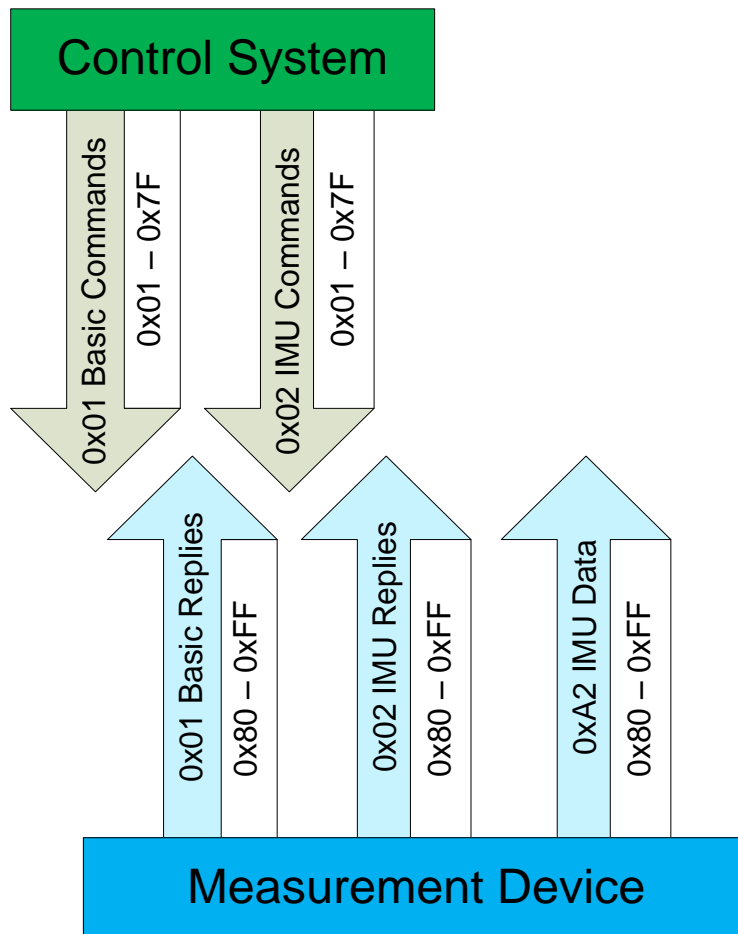
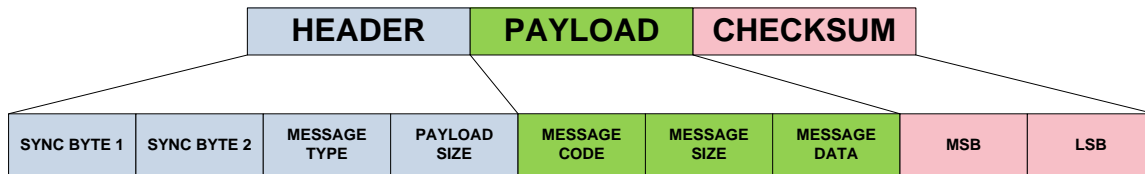


Figure 1 – MS CIP Message Type Diagram

All messages consist of header, payload and checksum sections enabling reception, decoding and verification of the data communicated. The following describes the basic parts of messages common to all device communications.



**Figure 2 - Memsense Communications Interface Protocol Structure**

## 2.2 HEADER

The header resides in the first 4 bytes and provides information required to parse, identify message type and define the message length. The first 2 bytes in the header section are synchronization bytes (*Sync Byte 1* and *Sync Byte 2*) which facilitate parsing the beginning of each transmitted packet. Next, the *Structure Type* is identified by a unique byte indicating the type of data being sent. The *Payload Length* follows, indicating the length of the payload section in bytes.

## 2.3 PAYLOAD

The payload may contain one or more fields each starting with the *Message Code*. The *Message Code* indicates a code associated with the specific message being sent. The combination of *Message Type* and *Message Code* identifies a unique message. For example, Ping has a *Message Type* of 0x01 and a *Message Code* of 0x02 so the 0x0102 uniquely identifies a ping message. The *Message Size* indicates the size of the message data in bytes and may be zero. Use of the *Message Size* allows the determination of the beginning of the next field to parse, if any exist. The *Message Data* follows and in some basic messages is not included.

## 2.4 CHECKSUM

The last 2 bytes of each sample contain checksum data used in the validation of the data received. The first checksum byte (*Checksum 1*) is the MSB of a Fletcher-16 checksum while the second byte is (*Checksum 2*) and the LSB of the Fletcher-16 checksum. The Ping command in Table 1 below provides an example on the implementation. The value of each byte in decimal is shown to the right of the hex value. The Fletcher Checksum is basically a sequential accumulator with the first value accumulating the current byte and the second byte accumulating the first checksum byte's accumulation.

The first Fletcher checksum byte (F1) and second Fletcher checksum byte (F2) start at zero. The first step in calculating the checksum is adding the Decimal Value to the F1 value followed by the F1 value being added to the F2 value. With each sequential byte the F1 accumulates the new byte value while F2 accumulates F1 until the last byte is accumulated. After the last byte is accumulated the remainder of a divide by 256 results in the F1 and F2 checksum values. The F1

checksum is then multiplied by 256 to shift it to the MSB and the F2 checksum becomes the LSB.

**Table 1 - Ping Message Checksum example**

Byte	Byte Name	Hex Value	Decimal Value	Decimal F1 Value	Decimal F2 Value
0	Sync1	0xA5	165	165	165
1	Sync2	0xA5	165	330	495
2	Message Type	0x01	1	331	826
3	Payload Size	0x02	2	333	1159
4	Message Code	0x02	2	333	1493
5	Message Size	0x00	0	335	1829
6	Checksum 1	0x4F		79	
7	Checksum 2	0x25			37

## 2.5 ERROR CODES

Replies to messages include an error code in byte 7. A 0 in the reply's error code byte indicates the message was received without error. An error code of 1 indicates that the checksum sent with the previous message does not calculate to the correct value. An error code of 2 indicates the previous message contained an invalid message type. An error code of 3 indicates the previous message had an invalid message code. Error code 4 indicates that a parameter in the previous message is not valid. Table 2 below summarizes the error codes.

**Table 2 – Error Codes**

Error Code	Error
0	OK
1	Checksum Error
2	Invalid Message Type
3	Invalid Message Code
4	Invalid Parameter

## 2.6 IMPLEMENTATION CONSIDERATIONS

When communicating with the device consideration should be given to the timing required to receive an acknowledgement from the command sent to the IMU prior to sending the next command. Successfully receiving acknowledgements before sending the next command will ensure message collisions and lost communications do not occur.

The configuration of any of the IMU's bandwidth, sample rate, select sensors or baud rate should be approached considering the relationship between the settings. A reliable approach to utilize is to start with a determination of the required bandwidth for the application. The sample rate should then be configured to at a minimum support the Nyquist sampling theorem. The select sensors configuration should also be considered as it increases or decreases the data being transmitted with each sample. Lastly, the baud rate should be set to support the amount of data being transmitted at the sample rate selected. This method will ensure good communications integrity is maintained.

## 3.0 MESSAGES

### 3.1 BASE MESSAGES

#### PING COMMAND 0x0102

The *Ping Command* provides a simple method of determining if the measurement device is responsive. When the measurement device receives a Ping it responds with an acknowledge or ACK.

**Table 3 - Ping Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x02	Ping identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x4F	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x25	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5010202004F25			

**Table 4 – Ping Reply Message, Ack**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x02	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xD3	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xCF	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5010480020200D3CF			

## GET DEVICE MESSAGES 0x0103

The *Get Device Messages*, command queries the measurement device for the messages supported. The device responds with 2 bytes for each message supported indicating the Message Type in the MSB and the Message ID Code in the LSB.

**Table 5 – Get Device Messages**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x03	Get Device Commands identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x50	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x27	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5010203005027			

**Table 6 –Device Messages Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x0F	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x03	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Message Code	0x83	Device Commands
9	Message Size	0x08	2 * number of commands in bytes.
10	Message 1 MSB	0x01	Supported command 1 MSB.
11	Message 1 LSB	0x02	Supported command 1 LSB.
12	Message 2 MSB	0x01	Supported command 2 MSB.
13	Message 2 LSB	0x03	Supported command 2 LSB.
14	Message 3 MSB	0x01	Supported command 3 MSB.
15	Message 3 LSB	0x04	Supported command 3 LSB.
16	Message 4 MSB	0x01	Supported command 4 MSB.
17	Message 4 LSB	0x05	Supported command 4 LSB.
18	Checksum 1	0x7D	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x73	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5010F80030300830801020103010401057D73			



## DEVICE RESET 0x0104

The *Device Reset* command resets the measurement device. The reset occurs immediately after the reply is sent.

**Table 7 –Device Reset Command**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x04	Device Reset identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x51	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x29	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5010204005129			

**Table 8 –Device Reset Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x04	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xD5	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xD3	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5010480020400D5D3			

## GET DEVICE MODEL 0x0105

The *Get Device Model* message returns the device model number in a 16-character string. The 16-byte string that is returned is right justified with unused characters filled with space characters.

**Table 9 –Get Device Model**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x05	Get Device Model identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x52	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x2B	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A501020500522B			

**Table 10 –Get Device Model Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Command type identification code.
3	Payload Size	0x16	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Data Size	0x02	Data Size in bytes.
6	Data	0x05	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Message Code	0x85	Device Model
9	Data Size	0x10	Data Size in bytes.
10	<sp>	0x20	Device Model String S15, Char <sp>
11	<sp>	0x20	Device Model String S14, Char <sp>
12	<sp>	0x20	Device Model String S13, Char <sp>
13	<sp>	0x20	Device Model String S12, Char <sp>
14	<sp>	0x20	Device Model String S11, Char <sp>
15	<sp>	0x20	Device Model String S10, Char <sp>
16	M	0x4D	Device Model String S9, Char M
17	S	0x53	Device Model String S8, Char S
18	_	0x5F	Device Model String S7, Char _
19	I	0x49	Device Model String S6, Char I
20	M	0x4D	Device Model String S5, Char M
21	U	0x55	Device Model String S4, Char U
22	3	0x33	Device Model String S3, Char 3
23	0	0x30	Device Model String S2, Char 0
24	2	0x32	Device Model String S1, Char 2
25	0	0x30	Device Model String S0, Char 0
26	Checksum 1	0xEC	Fletcher-16 checksum block 1 MSB
27	Checksum 2	0x54	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A501168002050085102020202020204D535F494D5533303230EC54			

## GET DEVICE SN 0x0106

The *Get Device SN* returns the device serial number in a 16-character string. The 16-byte string that is returned is right justified with unused characters filled with space characters.

**Table 11 –Get Device SN**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x06	Get Device SN identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x53	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x2D	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A501020600532D			



## GET DEVICE FW 0x0107

The *Get Device FW* returns the device firmware revision in a 16-character string. The 16-byte string that is returned is right justified with unused characters filled with space characters (0x20).

**Table 13 –Get Device FW**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x07	Get Device FW identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x54	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x2F	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A501020700542F			

**Table 14 –Get Device FW Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x16	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x07	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Message Code	0x87	Device FW
9	Message Size	0x10	Message Size in bytes.
10	<sp>	0x20	Device FW String S15, Char <sp>
11	<sp>	0x20	Device FW String S14, Char <sp>
12	<sp>	0x20	Device FW String S13, Char <sp>
13	<sp>	0x20	Device FW String S12, Char <sp>
14	<sp>	0x20	Device FW String S11, Char <sp>
15	<sp>	0x20	Device FW String S10, Char <sp>
16	<sp>	0x20	Device FW String S9, Char <sp>
17	<sp>	0x20	Device FW String S8, Char <sp>
18	<sp>	0x20	Device FW String S7, Char <sp>
19	R	0x52	Device FW String S6, Char R
20	_	0x5F	Device FW String S5, Char _
21	1	0x31	Device FW String S4, Char 1
22	_	0x5F	Device FW String S3, Char _
23	2	0x32	Device FW String S2, Char 2
24	_	0x5F	Device FW String S1, Char _
25	3	0x33	Device FW String S0, Char 3
26	Checksum 1	0xA6	Fletcher-16 checksum block 1 MSB
27	Checksum 2	0x25	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A50116800207008710202020202020202020202020525F315F325F33A625			

## GET DEVICE CAL 0x0108

The *Get Device Cal* returns the calibration date of the device in a 16-character string. Unused portions of the 16-byte string will use a space character (0x20). The date format will be right justified MM-DD-YYYY.

**Table 15 –Get Device Cal**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x08	Get Device Cal identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x55	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x31	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5010208005531			



**Table 16 –Get Device Cal Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Message type identification code.
3	Payload Size	0x16	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x08	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Message Code	0x88	Device Cal Date
9	Message Size	0x10	Message Size in bytes.
10	<sp>	0x20	Device Cal Date String S15, Char <sp>
11	<sp>	0x20	Device Cal Date String S14, Char <sp>
12	<sp>	0x20	Device Cal Date String S13, Char <sp>
13	<sp>	0x20	Device Cal Date String S12, Char <sp>
14	<sp>	0x20	Device Cal Date String S11, Char <sp>
15	<sp>	0x20	Device Cal Date String S10, Char <sp>
16	0	0x30	Device Cal Date String S9, Char 0
17	5	0x35	Device Cal Date String S8, Char 5
18	-	0x2D	Device Cal Date String S7, Char -
19	0	0x30	Device Cal Date String S6, Char 0
20	8	0x38	Device Cal Date String S5, Char 8
21	-	0x2D	Device Cal Date String S4, Char -
22	2	0x32	Device Cal Date String S3, Char 2
23	0	0x30	Device Cal Date String S2, Char 0
24	1	0x31	Device Cal Date String S1, Char 1
25	5	0x35	Device Cal Date String S0, Char 5
26	Checksum 1	0x32	Fletcher-16 checksum block 1 MSB
27	Checksum 2	0x10	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A50116800208008810202020202030352D30382D323031353210			

## CORRELATE GPS TIME 0x0109

The *Correlate GPS Time* command provides a means for synchronizing a timestamp with an external GPS time. The GPS 1-PPS signal is applied on a pin of the IMU interface connector (see *the IMU interface documentation for specific pin number*). The Correlate GPS Time command sets the GPS week number and GPS time of week in seconds. This message should be sent immediately after receiving the 1-PPS from the GPS receiver. Any processor-based time represented in the GPS time format may be substituted for the GPS time.

**Table 17 - Correlate GPS Time Command**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Command type identification code.
3	Payload Size	0x08	Byte length of the payload.
4	Message Code	0x09	Correlate GPS Time identification code.
5	Data Size	0x06	Data Size in bytes.
6	GPS WN MSB	0x07	GPS Week for 4-7-2015 MSB of U16
7	GPS WN LSB	0x2F	GPS Week for 4-7-2015 LSB of U16
8	New Time Value MSB	0x00	Hex MSB of new GPS time in seconds. U32
9	New Time Value Byte 2	0x00	Hex Byte 2 of new GPS time in seconds. U32
10	New Time Value Byte 1	0x02	Hex Byte 1 of new GPS time in seconds. U32
11	New Time Value LSB	0xFF	Hex LSB of new GPS time in seconds. U32
12	Checksum 1	0x99	Fletcher-16 checksum block 1 MSB
13	Checksum 2	0xAF	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A501080906072F000002FF99AF			

**Table 18 - Correlate GPS Time Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x01	Command type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Data Size	0x02	Data Size in bytes.
6	Data	0x09	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xDA	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xDD	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5010480020900DADD			

### 3.2 IMU CONFIGURATION MESSAGES 0x02

#### UART BAUD RATE CONFIGURE 0x0201

The *UART Baud Rate* message provides a means to configure, read back, save, load and reset the baud rate based on the function code communicated. The tables below detail the available functions and baud rates. When the Baud Rate is changed messages following the reply will be at the new settings. When the 0x03 Save current settings as startup settings function is used the reply may take longer as the settings will be saved to flash prior to replying to the command.

**Table 19 – UART Baud Rate Function Codes**

Code	Baud Rate Function
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

**Table 20 – Available UART Baud Rates**

Decimal Baud Rate	Hex Baud Rate
9600	0x00002580
19200	0x00004B00
115200	0x0001C200
230400	0x00038400
460800	0x00070800
921600	0x000E1000

**Table 21 – UART Baud Rate Configure**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x07	Byte length of the payload.
4	Message Code	0x01	UART Baud Rate Config identification code.
5	Message Size	0x05	Message Size in bytes.
6	Message	0x01	Use new settings.
7	Baud Rate MSB	0x00	MSB of UART hex value 0x0001C200 (115200)
8	Baud Rate Byte 2	0x01	Byte 2 of UART hex value 0x0001C200 (115200)
9	Baud Rate Byte 1	0xC2	Byte 1 of UART hex value 0x0001C200 (115200)
10	Baud Rate LSB	0x00	LSB of UART hex value 0x0001C200 (115200)
11	Checksum 1	0x1D	Fletcher-16 checksum block 1 MSB
12	Checksum 2	0x84	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A502070105010001C2001D84			

**Table 22 –UART Baud Rate Config Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x01	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xD3	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xD3	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020100D3D3			

## CONFIGURE FILTER 0x0203

The *Configure Filter* message provides a means for configuring and saving internal digital filtering options.

The *Configure Filter Function* allows the configuration to be used, queried, saved, loaded from startup settings, and reset to defaults. Table 23 details the associated codes and functions.

**Table 23 –Filter Function Codes**

Code	Filter Function Codes
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

The *Filter Bandwidth Control* codes allow filtering to be disabled or enabled in Infinite Impulse Response (IIR) mode. Care in selecting a filter bandwidth value that supports the Nyquist Sampling Theorem is suggested. The filter cutoff options are IMU specific. Refer to the specific IMU Product Specification and User Guide for supported filter cutoff options. For example, the IIR option allows the user to select filter cutoffs. Cutoff options from the MS-IMU3020 are listed in Table 24.

**Table 24 – Filter Bandwidth Control Codes**

Code	Filter Bandwidth Control Codes
0x00	Disable Filter
0x01	IIR Filter -3 dB at <b>25Hz</b>
0x02	IIR Filter -3 dB at <b>50Hz</b>
0x03	IIR Filter -3 dB at <b>75Hz</b>
0x04	IIR Filter -3 dB at <b>100Hz</b>
0x05	IIR Filter -3 dB at <b>10Hz</b>
0x06	IIR Filter -3 dB at <b>150Hz</b>
0x07	IIR Filter -3 dB at <b>200Hz</b>

***See the IMU Product User Guide for the device specific codes.***

Table 25 provides an example *Config Filter* message while Table 26 depicts a *Config Filter* reply from the IMU.

**Table 25 – Config Filter Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x03	Config Filter identification code.
5	Message Size	0x02	Message Size in bytes.
6	Filter Function	0x01	Use new settings.
7	Bandwidth Control	0x02	Optimized IIR -3 dB at 50Hz
8	Checksum 1	0x58	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xE1	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A502040302010258E1			

**Table 26 –Low Pass Filter Reply Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x03	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xD5	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xD7	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020300D5D7			

## IMU SAMPLE RATE CONFIGURE 0x0204

The *IMU Sample Rate Configure* provides a means to configure and save the rate at which all IMU data messages are transmitted. *IMU Message Config* is used to control individual measurements.

The *IMU Sample Rate Configure* function codes define the function to be performed on the device's inertial measurements. The associated codes and functions are listed in Table 27 below.

**Table 27 – IMU Sample Rate Function Codes**

Code	IMU Sample Rate Function
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

A 16-bit decimation value must be provided and is used to divide the internal sample rate to the desired output sample rate. For example, with an internal sample rate of 800Hz and a decimation value of 0x0008 (8 decimal), then the output sample rate is configured to 100Hz. *See the product user guide for the device specific internal sample rate.*

**Table 28 - IMU Sample Rate Configure**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x05	Byte length of the payload.
4	Message Code	0x04	Sample Rate Config identification code.
5	Message Size	0x03	Message Size in bytes.
6	Function	0x01	Use new settings.
7	Decimation MSB	0x00	MSB of Sample rate decimation value in hex.
	Decimation LSB	0x12	LSB of Sample rate decimation value in hex.
9	Checksum 1	0x6B	Fletcher-16 checksum block 1 MSB
10	Checksum 2	0x56	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5020504030100126B56			

**Table 29 - IMU Sample Rate Configure Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x04	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xD6	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xD9	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020400D6D9			



---

## SELECT SENSORS 0x0205

The *Select Sensors* message provides a means to configure and save the contents of and the rate at which all IMU data messages transmitted.

The *Select Sensors* function codes define the function to be performed on the device's message format. The associated codes and functions are listed in Table 30 below.

**Table 30 – Select Sensors Function Codes**

Code	Select Sensors Function
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

Table 31 lists the codes for the available measurements to be selected.

**Table 31 – Select Sensors Options**

Code	Select Sensors Options
0x81	Scaled Acceleration Vector in <i>g</i>
0x82	Scaled Angular Rate Vector in deg/sec
0x83	Scaled Magnetic Field Vector in gauss
0x84	Delta Theta Vector in Radians
0x85	Delta Velocity Vector in m/s
0x86	Scaled Pressure in milliBar
0x87	Scaled Temperature in Celsius
0x88	GPS Correlated Time

For each measurement selected the associated data code must be provided. An example configuration message is provided in Table 32 – Select Sensors Message with the message reply in

Table 33.

---

## SELECT SENSORS ERRATA

Byte 5 of the Select Sensors Message incorrectly calculates message size to one byte less than the actual size. This issue will be corrected in the MS-IMU3020 revision B which is included firmware revision R\_1\_1\_0 with a new Select Sensors message using a new message code. The current Select Sensors message will be maintained with the same message code but renamed Select Sensors Rev A.

***See the IMU Product User Guide for the device specific codes.***

**Table 32 – Select Sensors Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x06	Byte length of the payload.
4	Message Code	0x05	Select Sensors identification code.
5	Message Size	0x03	Message Size minus 1 in bytes.
6	Message Function	0x01	Use new settings.
7	Reserved	0x00	Reserved for future use.
8	Sensor Select Code	0x81	Acceleration Vector
9	Sensor Select Code	0x82	Angular Rate Vector
10	Checksum 1	0x5E	Fletcher-16 checksum block 1 MSB
11	Checksum 2	0x2E	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A502060503010081825E2E			

**Table 33 – Select Sensors Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x05	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xD7	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xDB	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020500D7DB			

## GET IMU INTERNAL SAMPLE RATE 0x0206

The *Get IMU Internal Sample Rate* command provides the user with the internal sample rate of the IMU that is used in decimation to arrive at the output sample rate.

The response to the command includes the standard Ack followed by a 16-bit hex value of the internal sample rate in hertz.

**Table 34 –Get IMU Internal Sample Rate**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x02	Byte length of the payload.
4	Message Code	0x06	Get IMU Internal Sample Rate identification code.
5	Message Size	0x00	Message Size in bytes.
6	Checksum 1	0x54	Fletcher-16 checksum block 1 MSB
7	Checksum 2	0x31	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5020206005431			

**Table 35 – Get IMU Internal Sample Rate Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x08	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Data Size	0x02	Message Size in bytes.
6	Data	0x06	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Message Code	0x86	IMU internal sample rate response identification code.
9	Data Size	0x02	Data Size in bytes.
10	Internal Sample Rate MSB	0x03	Internal Sample Rate in hex MSB 0x0320 (800Hz)
11	Internal Sample Rate LSB	0x20	Internal Sample Rate in hex LSB 0x0320 (800Hz)
12	Checksum 1	0x87	Fletcher-16 checksum block 1 MSB
13	Checksum 2	0xA5	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A50208800206008602032087A5			

## CONFIG ACCEL RANGE 0x0207

The *Config Accel Range* message provides a means for configuring and saving the triaxial accelerometer dynamic range options.

The *Config Accel Range Function* allows the configuration to be used, queried, saved, loaded from startup settings, and reset to defaults. Table 36 details the associated codes and functions.

**Table 36 –Config Accel Range Function Codes**

Code	Configure Accel Range Function Codes
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

The *Accel Range* codes allow the dynamic range of the accelerometer to be changed to one of the 4 supported ranges and effect all three axes of the sensor. The options for the accelerometer dynamic range are controlled in the *Accel Range Codes* listed in Table 37.

**Table 37 – Accel Range Codes**

Code	Accel Range Codes
0x00	Accelerometer range <b>±2 g</b>
0x01	Accelerometer range <b>±4 g</b>
0x02	Accelerometer range <b>±8 g</b>
0x03	Accelerometer range <b>±10 g</b>
0x04	Accelerometer range <b>±15 g</b>
0x05	Accelerometer range <b>±20 g</b>
0x06	Accelerometer range <b>±40 g</b>

***See the IMU Product User Guide for the device specific codes.***

Table 38 provides an example *Config Accel Range* message while Table 39 depicts a *Config Accel Range* reply from the IMU.

**Table 38 – Config Accel Range Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x07	Config Accel Range identification code.
5	Message Size	0x02	Message Size in bytes.
6	Filter Function	0x01	Use new settings.
7	Accel Range Control	0x02	Accelerometer range $\pm 4$ g
8	Checksum 1	0x5C	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xF1	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A50204070201025CF1			

**Table 39 –Config Accel Range Reply Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x07	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xD9	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xDF	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020700D9DF			

## CONFIGURE GYRO RANGE 0x0208

The *Config Gyro Range* message provides a means for configuring and saving the triaxial gyroscope dynamic range options.

The *Config Gyro Range Function* allows the configuration to be used, queried, saved, loaded from startup settings, and reset to defaults. Table 40 details the associated codes and functions.

**Table 40 – Config Gyro Range Function Codes**

Code	Configure Gyro Range Function Codes
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

The *Gyro Range* codes allow the dynamic range of the gyroscope to be changed to 1 of the 5 supported ranges and affect all 3 axes of the sensor. Example options for the *Gyro Range Codes* are listed in Table 41.

**Table 41 – Gyro Range Codes**

Code	Gyro Range Codes
0x01	Gyroscope range $\pm 120$ °/s
0x02	Gyroscope range $\pm 240$ °/s
0x03	Gyroscope range $\pm 480$ °/s
0x04	Gyroscope range $\pm 960$ °/s
0x05	Gyroscope range $\pm 1920$ °/s
0x06	Gyroscope range $\pm 75$ °/s
0x07	Gyroscope range $\pm 200$ °/s

*See the IMU Product User Guide for the device specific codes.*

Table 42 provides an example *Config Gyro Range* message while Table 43 depicts a *Config Gyro Range* reply from the IMU.

**Table 42 – Config Gyro Range Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x08	Config Gyro Range identification code.
5	Message Size	0x02	Message Size in bytes.
6	Function	0x01	Use new settings.
7	Gyro Range Control	0x02	Gyroscope range $\pm 240$ °/s
8	Checksum 1	0x5D	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xF5	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A50204080201025DF5			

**Table 43 – Config Gyro Range Reply Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Message	0x08	Message echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xDA	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xE1	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020800DAE1			

## CONFIGURE ALL 0x0209

The *Config All* message provides a means to save, load or reset to factory defaults all configurable settings within the device.

**Table 44 – Config All Command Codes**

Code	Configure All Commands
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

Table 45 provides an example *Config All* message while Table 46 depicts a *Config All* reply from the IMU.

**Table 45 – Config All Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x03	Byte length of the payload.
4	Message Code	0x09	Config All identification code.
5	Message Size	0x01	Message Size in bytes.
6	Config All Message	0x03	Save current settings as startup settings.
7	Checksum 1	0x5C	Fletcher-16 checksum block 1 MSB
8	Checksum 2	0x97	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A502030901035C97			

**Table 46 – Config All Reply Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Data Size	0x02	Message Size in bytes.
6	Data	0x09	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xDB	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xE3	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020900DBE3			



## DATA ON/OFF 0x020A

The *Data On/Off* message provides a means to enable or disable the data output of the IMU. The configuration function codes for *Data On/Off* are listed in Table 47.

**Table 47 – Config Function Codes**

Code	Config Function
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

The *Data On/Off* control codes are provided in Table 48.

**Table 48 – Data On/Off Command Codes**

Code	Configure All Commands
0x00	Data Off
0x01	Data On

Table 49 provides an example *Data On* message while Table 50 depicts a *Data On* reply from the IMU.

**Table 49 – Data On Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x0A	Data On/Off identification code.
5	Message Size	0x02	Message Size in bytes.
6	Function Code	0x01	Use new settings.
7	Data On/Off Code	0x01	Data On
8	Checksum 1	0x5E	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xFC	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A502040A0201015EFC			

**Table 50 – Data On Reply Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Data Size	0x02	Message Size in bytes.
6	Data	0x0A	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xDC	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xE5	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020A00DCE5			

## XTRIG ON/OFF 0x020B

The *EXTRIG On/Off* message provides a means to enable or disable the external trigger input of the IMU. The configuration function codes for *EXTRIG On/Off* are listed in Table 51.

**Table 51 – Config Function Codes**

Code	Config Function
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

The *EXTRIG On/Off* control codes are provided in Table 52.

**Table 52 – EXTRIG On/Off Command Codes**

Code	Configure All Commands
0x00	<i>EXTRIG</i> Off
0x01	<i>EXTRIG</i> On

Table 53 provides an example *EXTRIG On* message while Table 54 depicts an *EXTRIG On* reply from the IMU.

**Table 53 – EXTRIG On Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x0B	XTRIG On/Off identification code.
5	Message Size	0x02	Message Size in bytes.
6	Function Code	0x01	Use new settings.
7	XTRIG On/Off Control	0x01	XTRIG On
8	Checksum 1	0x5F	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0x00	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A502040B0201015F00			

**Table 54 – EXTRIG On Reply Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Data Size	0x02	Message Size in bytes.
6	Data	0x0B	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xDD	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xE7	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020B00DDE7			

## SELECT SENSORS REVISION B 0x020C

The *Select Sensors* message provides a means to configure and save the contents of and the rate at which all IMU data messages transmitted.

The *Select Sensors* function codes define the function to be performed on the device's message format. The associated codes and functions are listed in Table 55 below.

**Table 55 – Select Sensors Function Codes**

Code	Select Sensors Function
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

Table 56 lists the codes for the available measurements to be selected.

**Table 56 – Select Sensors Options**

Code	Select Sensors Options
0x81	Scaled Acceleration Vector in <i>g</i>
0x82	Scaled Angular Rate Vector in deg/sec
0x83	Scaled Magnetic Field Vector in gauss
0x84	Delta Theta Vector in Radians
0x85	Delta Velocity Vector in m/s
0x86	Scaled Pressure in milliBar
0x87	Scaled Temperature in Celsius
0x88	GPS Correlated Time
0x89	Scaled Auxiliary Acceleration Vector in <i>g</i>

For each measurement selected the associated data code must be provided. An example configuration message is provided in Table 57 with the message reply in Table 58.

***See the IMU Product User Guide for the device specific codes.***

**Table 57 – Select Sensors Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x05	Byte length of the payload.
4	Message Code	0x0C	Select Sensors Rev B identification code.
5	Message Size	0x03	Message Size in bytes.
6	Message Function	0x01	Use new settings.
7	Sensor Select Code	0x81	Acceleration Vector
8	Sensor Select Code	0x82	Angular Rate Vector
9	Checksum 1	0x64	Fletcher-16 checksum block 1 MSB
10	Checksum 2	0xF0	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A502050C0301818264F0			

**Table 58 – Select Sensors Reply**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Data	0x0C	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xDE	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xE9	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020C00DEE9			

## CONFIG AUX ACCEL RANGE 0x020D

The *Config Aux Accel Range* message provides a means for configuring and saving the triaxial accelerometer dynamic range options for the auxiliary accelerometer.

The *Config Aux Accel Range Function* allows the configuration to be used, queried, saved, loaded from startup settings, and reset to defaults. Table 59 details the associated codes and functions.

**Table 59 –Config Accel Range Function Codes**

Code	Configure Aux Accel Range Function
0X01	Use new settings.
0x02	Request current settings.
0x03	Save current settings as startup settings.
0x04	Load saved startup settings.
0x05	Reset to default settings.

The *Aux Accel Range* codes allow the dynamic range of the accelerometer to be changed to one of the 4 supported ranges and effect all three axes of the sensor. The options for the accelerometer dynamic range are controlled in the *Aux Accel Range Codes* listed in Table 60.

**Table 60 – Accel Range Codes**

Code	Accel Range Codes
0x00	Accelerometer range <b>±2 g</b>
0x01	Accelerometer range <b>±4 g</b>
0x02	Accelerometer range <b>±8 g</b>
0x03	Accelerometer range <b>±10 g</b>
0x04	Accelerometer range <b>±15 g</b>
0x05	Accelerometer range <b>±20 g</b>
0x06	Accelerometer range <b>±40 g</b>

***See the IMU Product User Guide for the device specific codes.***

Table 61 provides an example *Config Aux Accel Range* message while Table 62 depicts a *Config Aux Accel Range* reply from the IMU.

**Table 61 – Config Accel Range Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Message Code	0x0D	Config Aux Accel Range identification code.
5	Message Size	0x02	Message Size in bytes.
6	Filter Function	0x01	Use new settings.
7	Accel Range Code	0x05	Accelerometer range $\pm 15$ g
8	Checksum 1	0x65	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0x0C	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A502040D020105650C			

**Table 62 –Config Aux Accel Range Reply Message**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0x02	Message type identification code.
3	Payload Size	0x04	Byte length of the payload.
4	Command Code	0x80	Ack identification code.
5	Message Size	0x02	Message Size in bytes.
6	Data	0x0D	Command echo.
7	Error Code	0x00	0x00 No Errors
8	Checksum 1	0xDF	Fletcher-16 checksum block 1 MSB
9	Checksum 2	0xEB	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Reply</b>			
A5A5020480020D00DFEB			



### 3.3 DATA MESSAGES

#### IMU DATA MESSAGE 0xA2

The IMU Data Message consists of a header, payload and checksum and is the default output configuration on device startup. The IMU Data Message can be configured using Select Sensors Message 0x0205 to include any desired combination of sensor options. The default IMU Data Message is detailed below in Table 63.

**Table 63 - IMU Data Message 0xA2**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x1C	Byte length of the payload.
4	Message Code	0x81	Scaled Acceleration Vector identification code.
5	Message Size	0x0C	Data Size in bytes.
6	X Accel MSB	0x37	X Accel in g. MSB of F32.
7	X Accel Byte 2	0xA7	X Accel in g. Byte 2 of F32.
8	X Accel Byte 1	0xC5	X Accel in g. Byte 1 of F32.
9	X Accel LSB	0xAC	X Accel in g. LSB of F32.
10	Y Accel MSB	0x37	Y Accel in g. MSB of F32.
11	Y Accel Byte 2	0x7B	Y Accel in g. Byte 2 of F32.
12	Y Accel Byte 1	0xA8	Y Accel in g. Byte 1 of F32.
13	Y Accel LSB	0x82	Y Accel in g. LSB of F32.
14	Z Accel MSB	0x3F	Z Accel in g. MSB of F32.
15	Z Accel Byte 2	0x80	Z Accel in g. Byte 2 of F32.
16	Z Accel Byte 1	0x00	Z Accel in g. Byte 1 of F32.
17	Z Accel LSB	0x65	Z Accel in g. LSB of F32.
18	Message Code	0x82	Scaled Angular Rate Vector identification code.
19	Message Size	0x0C	Data Size in bytes.
20	X Gyro MSB	0x37	X Gyro in degrees per second. MSB of F32.
21	X Gyro Byte 2	0xA7	X Gyro in degrees per second. Byte 2 of F32.
22	X Gyro Byte 1	0xC5	X Gyro in degrees per second. Byte 1 of F32.
23	X Gyro LSB	0xAC	X Gyro in degrees per second. LSB of F32.
24	Y Gyro MSB	0x37	Y Gyro in degrees per second. MSB of F32.
25	Y Gyro Byte 2	0x7B	Y Gyro in degrees per second. Byte 2 of F32.
26	Y Gyro Byte 1	0xA8	Y Gyro in degrees per second. Byte 1 of F32.
27	Y Gyro LSB	0x82	Y Gyro in degrees per second. LSB of F32.
28	Z Gyro MSB	0x37	Z Gyro in degrees per second. MSB of F32.
29	Z Gyro Byte 2	0x49	Z Gyro in degrees per second. Byte 2 of F32.
30	Z Gyro Byte 1	0x53	Z Gyro in degrees per second. Byte 1 of F32.
31	Z Gyro LSB	0x9C	Z Gyro in degrees per second. LSB of F32.
32	Checksum 1	0x0C	Fletcher-16 checksum block 1 MSB
33	Checksum 2	0x23	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Command</b>			
A5A5A21C810C37A7C5AC377BA8823F800065820C37A7C5AC377BA8823749539C0C23			

## SCALED ACCELERATION VECTOR DATA MESSAGE 0xA281

The Scaled Acceleration Vector Data Message consists of a header, payload and checksum. The payload has three 32-bit floating point values containing the X, Y, and Z axis acceleration measurements scaled to *gs*. An example message is detailed below in Table 64.

**Table 64 – Scaled Acceleration Vector Data Message 0xA281**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x0E	Byte length of the payload.
4	Message Code	0x81	Scaled Acceleration Vector identification code.
5	Message Size	0x0C	Data Size in bytes.
6	X Accel MSB	0x37	X Accel in g. MSB of F32.
7	X Accel Byte 2	0xA7	X Accel in g. Byte 2 of F32.
8	X Accel Byte 1	0xC5	X Accel in g. Byte 1 of F32.
9	X Accel LSB	0xAC	X Accel in g. LSB of F32.
10	Y Accel MSB	0x37	Y Accel in g. MSB of F32.
11	Y Accel Byte 2	0x7B	Y Accel in g. Byte 2 of F32.
12	Y Accel Byte 1	0xA8	Y Accel in g. Byte 1 of F32.
13	Y Accel LSB	0x82	Y Accel in g. LSB of F32.
14	Z Accel MSB	0x3F	Z Accel in g. MSB of F32.
15	Z Accel Byte 2	0x80	Z Accel in g. Byte 2 of F32.
16	Z Accel Byte 1	0x00	Z Accel in g. Byte 1 of F32.
17	Z Accel LSB	0x65	Z Accel in g. LSB of F32.
18	Checksum 1	0xD6	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x1A	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20E810C37A7C5AC377BA8823F800065D61A			

## SCALED ANGULAR RATE VECTOR DATA MESSAGE 0xA282

The Scaled Angular Rate Vector Data Message consists of a header, payload and checksum. The payload has three 32-bit floating point values containing the X, Y, and Z axis angular rate measurements scaled to degrees per second. An example message is detailed below in Table 65.

**Table 65 – Scaled Angular Rate Vector Data Message 0xA282**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x0E	Byte length of the payload.
4	Message Code	0x82	Scaled Angular Rate Vector identification code.
5	Message Size	0x0C	Data Size in bytes.
6	X Gyro MSB	0x37	X Gyro in degrees per second. MSB of F32.
7	X Gyro Byte 2	0xA7	X Gyro in degrees per second. Byte 2 of F32.
8	X Gyro Byte 1	0xC5	X Gyro in degrees per second. Byte 1 of F32.
9	X Gyro LSB	0xAC	X Gyro in degrees per second. LSB of F32.
10	Y Gyro MSB	0x37	Y Gyro in degrees per second. MSB of F32.
11	Y Gyro Byte 2	0x7B	Y Gyro in degrees per second. Byte 2 of F32.
12	Y Gyro Byte 1	0xA8	Y Gyro in degrees per second. Byte 1 of F32.
13	Y Gyro LSB	0x82	Y Gyro in degrees per second. LSB of F32.
14	Z Gyro MSB	0x37	Z Gyro in degrees per second. MSB of F32.
15	Z Gyro Byte 2	0x49	Z Gyro in degrees per second. Byte 2 of F32.
16	Z Gyro Byte 1	0x53	Z Gyro in degrees per second. Byte 1 of F32.
17	Z Gyro LSB	0x9C	Z Gyro in degrees per second. LSB of F32.
18	Checksum 1	0x22	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x40	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20E820C37A7C5AC377BA8823749539C2240			

## SCALED MAGNETIC FIELD VECTOR DATA MESSAGE 0xA283

The Magnetic Field Vector Data Message consists of a header, payload and checksum. The payload has three 32-bit floating point values containing the X, Y, and Z axis magnetic field measurements scaled to gauss. An example message is detailed below in Table 66.

**Table 66 – Scaled Magnetic Field Vector Data Message 0xA283**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x0E	Byte length of the payload.
4	Message Code	0x83	Scaled Magnetic Field Vector identification code.
5	Message Size	0x0C	Data Size in bytes.
6	X Mag MSB	0x37	X Mag in gauss. MSB of F32.
7	X Mag Byte 2	0xA7	X Mag in gauss. Byte 2 of F32.
8	X Mag Byte 1	0xC5	X Mag in gauss. Byte 1 of F32.
9	X Mag LSB	0xAC	X Mag in gauss. LSB of F32.
10	Y Mag MSB	0x37	Y Mag in gauss. MSB of F32.
11	Y Mag Byte 2	0x7B	Y Mag in gauss. Byte 2 of F32.
12	Y Mag Byte 1	0xA8	Y Mag in gauss. Byte 1 of F32.
13	Y Mag LSB	0x82	Y Mag in gauss. LSB of F32.
14	Z Mag MSB	0x37	Z Mag in gauss. MSB of F32.
15	Z Mag Byte 2	0x49	Z Mag in gauss. Byte 2 of F32.
16	Z Mag Byte 1	0x53	Z Mag in gauss. Byte 1 of F32.
17	Z Mag LSB	0x9C	Z Mag in gauss. LSB of F32.
18	Checksum 1	0x23	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x4E	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20E830C37A7C5AC377BA8823749539C234E			

## SCALED DELTA THETA VECTOR DATA MESSAGE 0xA284

The Delta Theta Vector Data Message consists of a header, payload and checksum. The payload has three 32-bit floating point values containing the X, Y, and Z axis change in angle measurements scaled to radians. An example message is detailed below in Table 67.

**Table 67 – Scaled Delta Theta Vector Data Message 0xA284**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x0E	Byte length of the payload.
4	Message Code	0x84	Delta Theta Vector identification code.
5	Message Size	0x0C	Data Size in bytes.
6	X Delta Theta MSB	0x37	X Delta Theta in radians. MSB of F32.
7	X Delta Theta Byte 2	0xA7	X Delta Theta in radians. Byte 2 of F32.
8	X Delta Theta Byte 1	0xC5	X Delta Theta in radians. Byte 1 of F32.
9	X Delta Theta LSB	0xAC	X Delta Theta in radians. LSB of F32.
10	Y Delta Theta MSB	0x37	Y Delta Theta in radians. MSB of F32.
11	Y Delta Theta Byte 2	0x7B	Y Delta Theta in radians. Byte 2 of F32.
12	Y Delta Theta Byte 1	0xA8	Y Delta Theta in radians. Byte 1 of F32.
13	Y Delta Theta LSB	0x82	Y Delta Theta in radians. LSB of F32.
14	Z Delta Theta MSB	0x37	Z Delta Theta in radians. MSB of F32.
15	Z Delta Theta Byte 2	0x49	Z Delta Theta in radians. Byte 2 of F32.
16	Z Delta Theta Byte 1	0x53	Z Delta Theta in radians. Byte 1 of F32.
17	Z Delta Theta LSB	0x9C	Z Delta Theta in radians. LSB of F32.
18	Checksum 1	0x24	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x5C	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20E840C37A7C5AC377BA8823749539C245C			

## SCALED DELTA VELOCITY VECTOR DATA MESSAGE 0xA285

The Delta Velocity Vector Data Message consists of a header, payload and checksum. The payload has three 32-bit floating point values containing the X, Y, and Z axis change in velocity measurements scaled to meters per second. An example message is detailed below in Table 68.

**Table 68 – Scaled Delta Velocity Vector Data Message 0xA285**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x0E	Byte length of the payload.
4	Message Code	0x85	Delta Velocity Vector identification code.
5	Message Size	0x0C	Data Size in bytes.
6	X Delta Velocity MSB	0x37	X Delta Velocity in m/s. MSB of F32.
7	X Delta Velocity Byte 2	0xA7	X Delta Velocity in m/s. Byte 2 of F32.
8	X Delta Velocity Byte 1	0xC5	X Delta Velocity in m/s. Byte 1 of F32.
9	X Delta Velocity LSB	0xAC	X Delta Velocity in m/s. LSB of F32.
10	Y Delta Velocity MSB	0x37	Y Delta Velocity in m/s. MSB of F32.
11	Y Delta Velocity Byte 2	0x7B	Y Delta Velocity in m/s. Byte 2 of F32.
12	Y Delta Velocity Byte 1	0xA8	Y Delta Velocity in m/s. Byte 1 of F32.
13	Y Delta Velocity LSB	0x82	Y Delta Velocity in m/s. LSB of F32.
14	Z Delta Velocity MSB	0x37	Z Delta Velocity in m/s. MSB of F32.
15	Z Delta Velocity Byte 2	0x49	Z Delta Velocity in m/s. Byte 2 of F32.
16	Z Delta Velocity Byte 1	0x53	Z Delta Velocity in m/s. Byte 1 of F32.
17	Z Delta Velocity LSB	0x9C	Z Delta Velocity in m/s. LSB of F32.
18	Checksum 1	0x25	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x6A	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20E850C37A7C5AC377BA8823749539C256A			

## SCALED PRESSURE DATA MESSAGE 0xA286

The Scaled Pressure Data Message consists of a header, payload and checksum. The payload has one 32-bit floating point value containing the atmospheric pressure scaled to millibar. An example message is detailed below in Table 69.

**Table 69 – Scaled Pressure Data Message 0xA286**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x06	Byte length of the payload.
4	Message Code	0x86	Scaled Pressure identification code.
5	Message Size	0x04	Data Size in bytes.
6	X Pressure MSB	0x00	Pressure in milliBar. MSB of F32.
7	X Pressure Byte 2	0x00	Pressure in milliBar. Byte 2 of F32.
8	X Pressure Byte 1	0x03	Pressure in milliBar. Byte 1 of F32.
9	X Pressure LSB	0xFD	Pressure in milliBar. LSB of F32.
10	Checksum 1	0x7C	Fletcher-16 checksum block 1 MSB
11	Checksum 2	0xB4	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A2068604000003FD7CB4			

## SCALED TEMPERATURE DATA MESSAGE 0xA287

The Scaled Temperature Data Message consists of a header, payload and checksum. The payload has one 32-bit floating point value containing the IMU temperature scaled to degrees Celsius. An example message is detailed below in Table 70.

**Table 70 – Scaled Temperature Data Message 0xA287**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x06	Byte length of the payload.
4	Message Code	0x87	Scaled Temperature identification code.
5	Message Size	0x04	Data Size in bytes.
6	X Temp MSB	0x00	Temperature in Celsius. MSB of F32.
7	X Temp Byte 2	0x00	Temperature in Celsius. Byte 2 of F32.
8	X Temp Byte 1	0x00	Temperature in Celsius. Byte 1 of F32.
9	X Temp LSB	0x19	Temperature in Celsius. LSB of F32.
10	Checksum 1	0x96	Fletcher-16 checksum block 1 MSB
11	Checksum 2	0xD0	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20687040000001996D0			



## GPS CORRELATED TIME DATA MESSAGE 0xA288

The GPS Correlated Time Data Message consists of a header, payload and checksum. The payload has one unsigned 16-bit integer containing the GPS week number, one 64-bit floating point value containing the GPS seconds of the week and an unsigned 8-bit integer containing the GPS Correlated Time Flags. The Flag bit descriptions are provided in Table 71.

**Table 71 – GPS Correlated Time Flag Bit Descriptions**

Bit	GPS Correlated Time Flag Bit Description
0	Set to indicate GPS 1 PPS received.
1	Set to indicate GPS Time was set.
2	No GPS Time Set, Using GPS Week 0 and Time 0
3	1 PPS Loss
4-7	Reserved
8-15	Reserved

The *GPS Correlated Time Flags* allow indication to the interface system for reception of the GPS 1 PPS signal, GPS Time successfully set, No GPS time set, and a missing 1 PPS signal after GPS Time is set.

**Bit 0** indicates that the GPS 1 PPS signal was received on the appropriate pin of the IMU being used. Bit 0 will be set HIGH for one sample immediately following the reception of the signal on the interface pin. Bit 0 will only function after GPS time is set on the device.

**Bit 1** indicates the GPS Time was successfully initialized. The bit will be HIGH for a single sample immediately following the message that set the GPS Time. Monitoring Bit 1 after the loss of a 1 PPS signal and the GPS Time is set again provides a positive indication of successful re-initialization.

**Bit 2** indicates that the GPS Time has not been set and remains HIGH until the GPS Time is set. When the GPS Time is not set the GPS Time data starts incrementing from GPS Week 0 and GPS Time 0 using the IMU's internal clock. The monitoring of Bit 2 in a LOW state can be used to verify the GPS Time is successfully set.

**Bit 3** in a HIGH state indicates that any time after the GPS Time was set and a period of greater than 1 second occurred without reception of the GPS 1 PPS signal. The Bit 3 state can be cleared by setting GPS Time and continued reception of the GPS 1 PPS signal. When Bit 3 is HIGH the IMU's internal clock will be used without 1 PPS updates to provide GPS Time which may result in GPS Time error from clock drift.

**GPS Seconds of Week (SOW)** – GPS SOW is transmitted as a double. The precision of the time is associated with the IMU Output Sample Rate. For instance, if the Output Sample Rate is 200 Hz then the GPS SOW will display a 5 ms precision.

A GPS Correlated Time Data Message example message is shown in Table 72.

**Table 72 – GPS Correlated Time Data Message 0xA288**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x0E	Byte length of the payload.
4	Message Code	0x88	GPS Correlation Time
5	Message Size	0x0C	Data Size in bytes.
6	GPS SOW MSB	0x41	GPS Seconds of week for 9:30:00 AM MSB of F64.
7	GPS SOW Byte 6	0x09	GPS Seconds of week for 9:30:00 AM Byte 6 of F64.
8	GPS SOW Byte 5	0x44	GPS Seconds of week for 9:30:00 AM Byte 5 of F64.
9	GPS SOW Byte 4	0xc0	GPS Seconds of week for 9:30:00 AM Byte 4 of F64.
10	GPS SOW Byte 3	0x00	GPS Seconds of week for 9:30:00 AM Byte 3 of F64.
11	GPS SOW Byte 2	0x00	GPS Seconds of week for 9:30:00 AM Byte 2 of F64.
12	GPS SOW Byte 1	0x00	GPS Seconds of week for 9:30:00 AM Byte 1 of F64.
13	GPS SOW LSB	0x00	GPS Seconds of week for 9:30:00 AM LSB of F64
14	GPS WN MSB	0x07	GPS Week for 4-7-2015 MSB of U16
15	GPS WN LSB	0x2F	GPS Week for 4-7-2015 LSB of U16
16	Timestamp Flags	0x00	GPS Timestamp Flags MSB of U16
17	Timestamp Flags	0x08	GPS Timestamp Flags LSB of U16
18	Checksum 1	0x1A	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x32	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20E880C410944c00000000072F00081A32			

## SCALED AUX ACCEL VECTOR DATA MESSAGE 0xA289

The Scaled Acceleration Vector Data Message consists of a header, payload and checksum. The payload has three 32-bit floating point values containing the X, Y, and Z axis acceleration measurements scaled to *gs*. An example message is detailed below in Table 73.

**Table 73 – Scaled Aux Accel Vector Data Message 0xA281**

Byte	Byte Name	Value	Description
0	Sync1	0xA5	First synchronization value used in sample parsing.
1	Sync2	0xA5	Second synchronization value used in sample parsing.
2	Message Type	0xA2	Message type identification code.
3	Payload Size	0x0E	Byte length of the payload.
4	Message Code	0x89	Scaled Aux Accel Vector identification code.
5	Message Size	0x0C	Data Size in bytes.
6	X Accel MSB	0x37	X Accel in g. MSB of F32.
7	X Accel Byte 2	0xA7	X Accel in g. Byte 2 of F32.
8	X Accel Byte 1	0xC5	X Accel in g. Byte 1 of F32.
9	X Accel LSB	0xAC	X Accel in g. LSB of F32.
10	Y Accel MSB	0x37	Y Accel in g. MSB of F32.
11	Y Accel Byte 2	0x7B	Y Accel in g. Byte 2 of F32.
12	Y Accel Byte 1	0xA8	Y Accel in g. Byte 1 of F32.
13	Y Accel LSB	0x82	Y Accel in g. LSB of F32.
14	Z Accel MSB	0x3F	Z Accel in g. MSB of F32.
15	Z Accel Byte 2	0x80	Z Accel in g. Byte 2 of F32.
16	Z Accel Byte 1	0x00	Z Accel in g. Byte 1 of F32.
17	Z Accel LSB	0x65	Z Accel in g. LSB of F32.
18	Checksum 1	0xDE	Fletcher-16 checksum block 1 MSB
19	Checksum 2	0x8A	Fletcher-16 checksum block 2 LSB
<b>Resulting Complete Message</b>			
A5A5A20E890C37A7C5AC377BA8823F800065DE8A			

## REVISION HISTORY

REV	STATUS	DESCRIPTION	DATE
A	Obsolete	Initial Release	12-29-2015
A.1	Obsolete	Added Data On/Off Command	3-9-2016
B	Obsolete	Corrected Ping example in <a href="#">Table 1</a> . Corrected Correlate GPS Time Command <a href="#">Table 16</a> Added <a href="#">Select Sensors Errata</a> paragraph. Updated Select Sensors Command in <a href="#">Table 31</a> Corrected Rate in Config Gyro Command in <a href="#">Table 41</a>	5-26-2016
C	Obsolete	Added XTRIG on/off Command Added Select Sensors Rev B Command Updated table numbers and references.	6-8-2016
D	Obsolete	Removed the characters 'S' and 'N' from get SN response. <a href="#">p.13</a> Updated IMU Sample Rate Config description <a href="#">p.23</a> Removed Magnetometer and Pressure from Select Sensors. <a href="#">p.26</a> Updated Get Internal Sample Rate example Response to 800Hz. <a href="#">p.27</a> Corrected Select Sensors Rev B Command Checksum. <a href="#">p.38</a>	6-15-2016
E	Obsolete	Added 200Hz Filter Bandwidth Control Code <a href="#">Table 23</a> .	9-20-2016
F	Obsolete	Corrected errors in <a href="#">GPS Correlated Time Data Message 0xA288</a> .	12-14-2016
G	Obsolete	Added <a href="#">Aux Accel Configure Range Message</a> . Added <a href="#">Scaled Aux Accel Data Message</a> .	07-25-2017
H	Obsolete	Corrected missing byte errors in several messages. Added detail to the Correlated GPS Time Data Message	10-31-2017
J	Obsolete	Added missing 150Hz filter and corrected acceleration codes.	4-13-2018
K	Obsolete	Corrected inconsistent use of Message and Command. Rewrote Section 2.3 Payload.	9-17-2018
L	Obsolete	Moved revision table to the end of the document. Changed the document number from DOC00381 to DOC00419.	11-7-2018
M	Released	Added error code definition in section 2.5.	8-26-2019