UM1744
User manual

STEVAL- MKI121V1 communication protocol

Introduction

The scope of this user manual is to present the communication protocol used between the STEVAL-MKI121V1 product evaluation board (Discovery-M1) and the iNEMO SDK (Software Development Kit). This communication protocol runs upon a physical communication channel based on USB virtual COM, which represents the physical channel used in the communication between the STEVAL-MKI121V1 and the PC.

The first chapter explains the general frame format and main rules used in the protocol.

The second chapter explains all the frames used in the actual release of the embedded firmware and Software Development Kit (SDK).

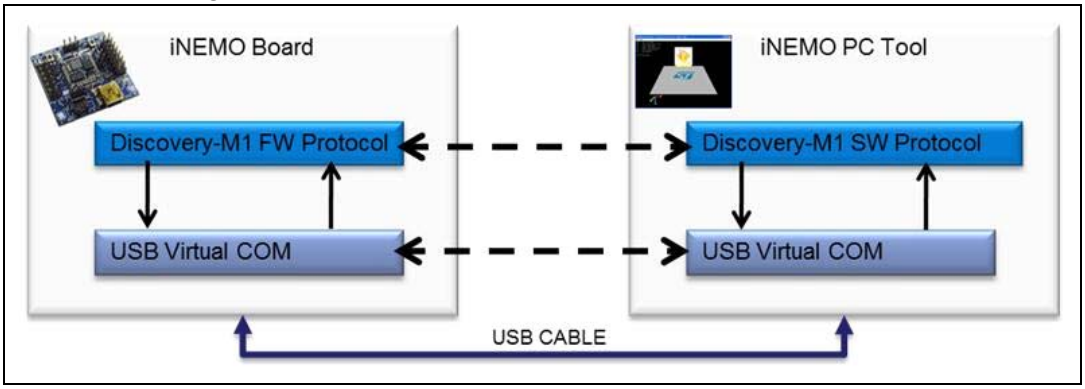
Contents

1 General frame format and protocol rules

1.1 Frame format

This paragraph explains the format of the frame used in the STEVAL-MKI121V1 communication protocol. Because, the STEVAL-MKI121V1 exchanges data and commands with the PC GUI through a physical communication channel based on a USB Virtual COM, each frame, described below, represents the payload of a USB frame.

Figure 1. STEVAL-MKI121V1 communication architecture



The frames are described as a sequence of fields in a specific order. All frame formats are depicted in the order in which they are passed to the USB driver, from left to right. Bits within each field are numbered from k-1 (leftmost and most significant) to 0 (rightmost and least significant), where the length of the field is k bits.

The frame format is composed of a header and an optional payload. The general frame shall be formatted as illustrated in *Figure 2*. The header is composed of three mandatory (M) fields, each of which is 1 byte in length, while the payload is an optional field whose maximum length is 61 bytes. See LF/MF field in the following section to overcome this limit.

Figure 2. General frame format

Bytes: 1	1	1	≤ 61
Frame Control (M)	Length (M)	Message ID (M)	Payload (O)

1.1.1 Frame control field

The frame control field is 1 byte in length and contains information defining the frame type and other control flags. The frame control field shall be formatted as illustrated in *Figure 3*.

Figure 3. Frame control field

Bit: 7	6	5	4	3	2	1	0
Frame Type		Ack	LF/MF	Frame Version		QoS	

The frame type subfield is 2 bits in length and shall be set to one of the values listed in *Table 1*.

Table 1. Frame type list

Value	Frame type
00	CONTROL
01	DATA
10	ACK
11	NACK

The Ack subfield is 1 bit in length and specifies whether an acknowledgment is required from the recipient on receipt of a DATA or CONTROL frame. If this subfield is set to one, the recipient shall send an acknowledgment frame only if, upon reception, the frame passes all the needed levels of filtering. If this subfield is set to zero, the recipient device shall not send an acknowledgment frame. It is possible to embed a payload in an acknowledgment frame (piggybacking) to send useful information to the transmitter and avoiding further transactions. When the Ack field is set to one and upon reception the frame doesn't pass the needed level of filtering, the recipient shall send a not-acknowledgment frame (NACK), whose payload is an error code (e.g. unsupported command, value out of range,…). In the ACK and/or NACK frames the Ack field shall be set to zero and ignored on reception.

The LF/MF (Last Fragment / More Fragment) subfield is 1 bit in length and it is used for fragmentation and reassembling. This field is set to zero to indicate a single frame or the last frame of a multiple-frame transaction. This field is set to 1 to indicate that other frames will follow all those belonging to the same transaction. In the ACK and NACK frames (with or without payload) fragmentation is not supported and this subfield shall be set to zero in transmission of ACK and NACK frames and ignored on reception.

The frame version subfield is 2 bits in length and shall be set to the non-reserved for future use (RFU) value listed in *Table 2*.

Table 2. Frame version list

Value	Frame version
00	Version 1
01	RFU
10	
11	

The QoS (Quality of Service) subfield is 2 bits in length and shall be set to one of the values listed in *Table 3*. This subfield allows the application to exchange and process data and control frames with different priorities.

Table 3. QoS list

Value	Frame version
00	Normal Priority
01	Medium Priority
10	High Priority
11	RFU

1.1.2 Length field

The length field is 1 byte in length and contains the number of bytes that follow the length field. Admitted values are in the range 1 ÷ 62.

1.1.3 Message ID field

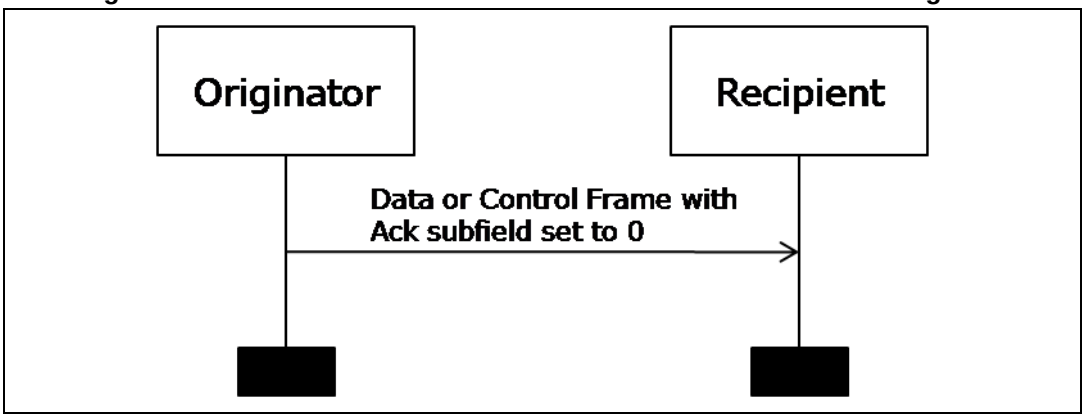
The message ID is 1 byte in length and contains an identifier of the user application messages. See *Section 2.2* and the following sections for further details.

1.2 Protocol rules

There are two types of transactions: acknowledgment or non-acknowledgment of the DATA or CONTROL frame.

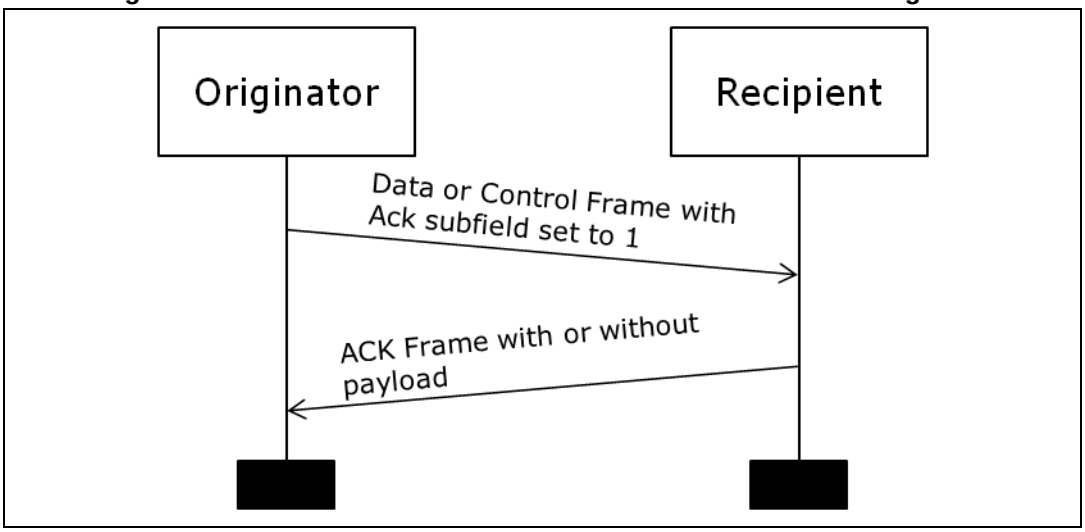
A DATA or CONTROL frame with the Ack subfield of its frame control field set to zero shall not be acknowledged by its intended recipient. The originating device (PC or Discovery-M1 board) shall assume that the transmission of the frame was successful. The message sequence chart in *Figure 4* shows the scenario for transmitting a single DATA or CONTROL frame from an originator to a recipient without requiring an acknowledgment.

Figure 4. Data or control frame transmission without an acknowledgment



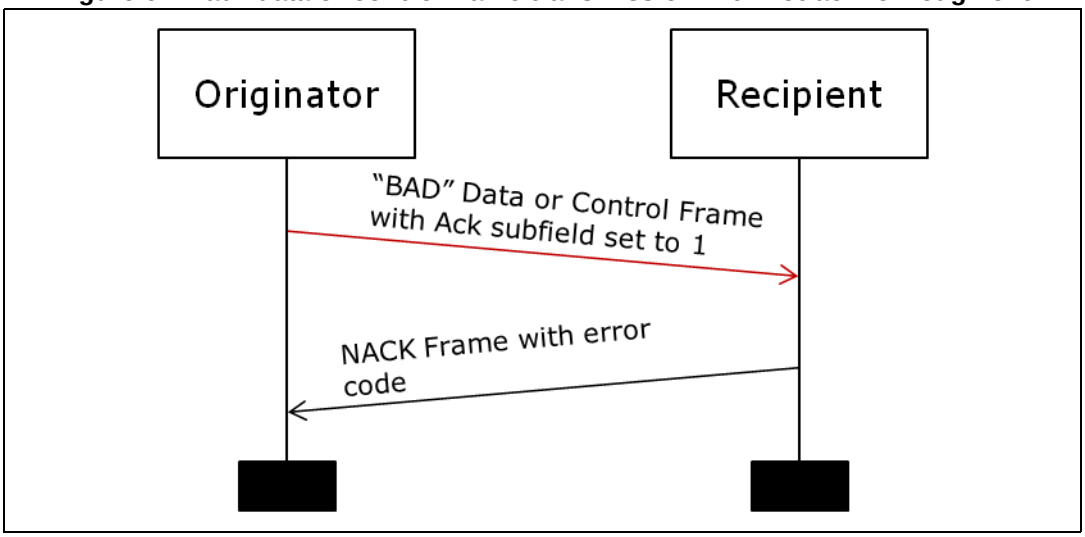
A DATA or CONTROL frame transmitted with the Ack subfield of its frame control field set to one shall be acknowledged by the recipient. If the intended recipient correctly receives the frame, it shall generate and send an ACK frame containing the same message ID from the DATA or CONTROL frame that is being acknowledged. It is possible also to include a payload in the ACK frame to transfer useful data from the recipient to the originator. The message sequence chart in *Figure 5* shows the scenario for transmitting a single DATA or CONTROL frame from an originator to a recipient with an acknowledgment.

Figure 5. Data or control frame transmission with an acknowledgment



If the frame received does not conform to all the required filtering rules, the recipient shall generate and send a NACK frame containing the same message ID from the DATA or CONTROL frame that is being acknowledged and containing the error code. The message sequence chart in *Figure 6* shows the scenario for transmitting a single "bad" DATA or CONTROL frame from an originator to a recipient with a not-acknowledgment.

Figure 6. "Bad" data or control frame transmission with not-acknowledgment



2 STEVAL-MKI121V1 frames

2.1 STEVAL-MKI121V1 frame types

The frames used in the STEVAL-MKI121V1 are classified in five types:

1. Communication control frames
2. Board information frames
3. Sensor setting frames
4. Acquisition sensor data frames

2.2 Communication control frames

Communication control frames are frames originated by the software PC (SDK or GUI) and used to send specific commands to the Discovery-M1 board. All the communication control frames are listed in *Table 4*.

Table 4. Communication control frames

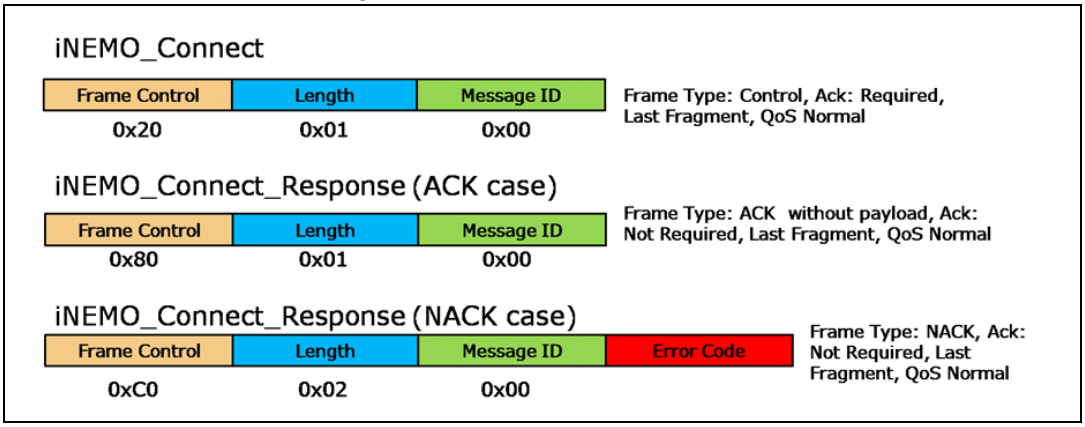
Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
iNEMO _Connect	CONTROL	Y	0x00	N	0		PC
iNEMO _Connect_Response	ACK	N	0x00	N	0		Discovery-M1
	NACK	N	0x00	N	1	Error Code	
iNEMO _Disconnect	CONTROL	Y	0x01	N	0		PC
iNEMO_Disconnect_Response	ACK	N	0x01	N	0		Discovery-M1
	NACK	N	0x01	N	1	Error Code	
iNEMO_Reset_Board	CONTROL	Y	0x02	N	0		PC
iNEMO_Reset_Board_Response	ACK	N	0x02	N	0		Discovery-M1
	NACK	N	0x02	N	1	Error Code	
iNEMO_Enter_DFU_Mode	CONTROL	Y	0x03	N	0		PC
iNEMO_Enter_DFU_Mode_ Response	ACK	N	0x03	N	0		Discovery-M1
	NACK	N	0x03	N	1	Error Code	
iNEMO_Trace	CONTROL	Y	0x07	N	0		PC
iNEMO_Trace_Response	ACK	N	0x07	N	0		Discovery-M1
	NACK	N	0x07	N	1	Error Code	
iNEMO_Trace_Data	DATA	N	0x07	M	Variable	String for debug purpose	
iNEMO_Led_Control	CONTROL	Y	0x08	N	1	0x00 OFF 0x01 ON	PC
iNEMO_Led_Control_Response	ACK	N	0x08	N	0		Discovery-M1
	NACK	N	0x08	N	1	Error Code	

2.2.1 iNEMO_Connect

The iNEMO_Connect command shall be the first command sent from the GUI or SDK to the Discovery-M1 board. Any other command sent before the iNEMO_Connect will not be processed by Discovery-M1. It works like a "ping" and opens the communication between the GUI or SDK and the Discovery-M1 board at the application level.

Figure 7 shows the frames involved in the iNEMO_Connect transaction.

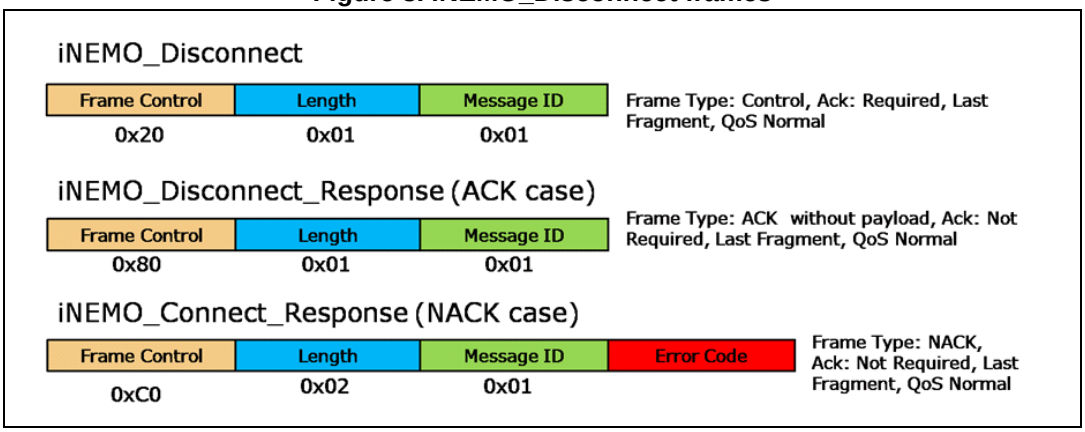
Figure 7. iNEMO_Connect frames



2.2.2 iNEMO_Disconnect

The iNEMO_Disconnect command closes the communication between the PC and the Discovery-M1 board. *Figure 8* shows the frames involved in the iNEMO_Disconnect transaction.

Figure 8. iNEMO_Disconnect frames



The GUI (or SDK), after receiving the ACK frame, shall close the USB Virtual Com. To re-open the communication only the iNEMO_Connect command shall be used.

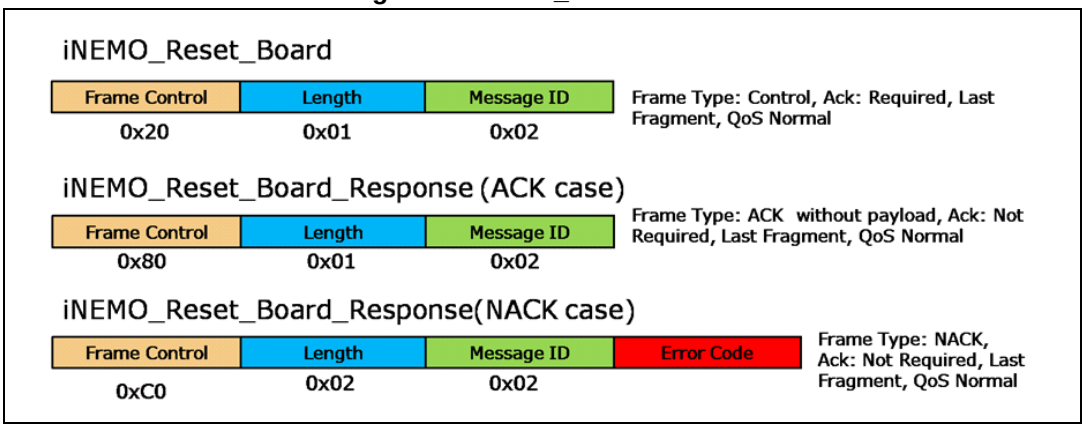
2.2.3 iNEMO_Reset

The iNEMO_Reset command implies a software reset of the Discovery-M1 board. After receiving the iNEMO_Reset command, the Discovery-M1 board replies with the ACK frame; then waits for 5 seconds before disconnecting the USB cable in software mode and invokes a software reset. The GUI (or SDK), after receiving the ACK frame, shall close the USB

Virtual Com. To re-open the communication only the iNEMO_Connect command shall be used.

Figure 9 shows the frames involved in the iNEMO_Reset transaction.

Figure 9. iNEMO_Reset frames

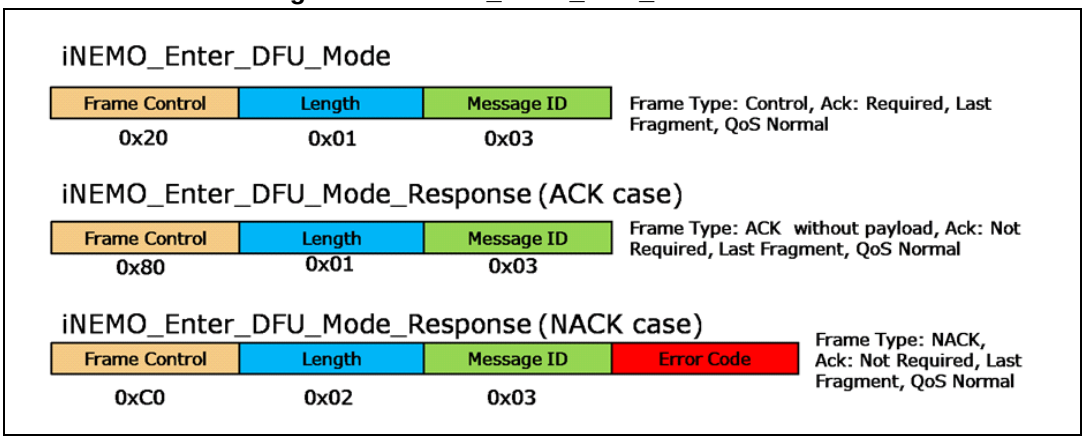


2.2.4 iNEMO_Enter_DFU_Mode

The iNEMO_Enter_DFU_Mode command allows the Discovery-M1 board to enter in DFU mode in software mode. After receiving the iNEMO_Enter_DFU_Mode command, the Discovery-M1 board replies with an ACK frame. Then it will set the Option Byte Data0 (at address 0x1FFFF804) to one, will disconnect the USB cable in software mode, and it will invoke a software reset. After reset, the Discovery-M1 will enter in DFU mode. After entering in DFU mode in software, the Discovery-M1 will change the Option Byte Data0 to zero. The user can leave the DFU mode in two ways: by unplugging and plugging in the USB cable (hardware mode), or by using the Leave_DFU_Mode command available in the DfuSe Demo PC application or in the GUI or SDK. The GUI (or SDK) shall close the USB Virtual Com after receiving the ACK frame.

Figure 10 shows the frames involved in the iNEMO_Enter_DFU_Mode transaction

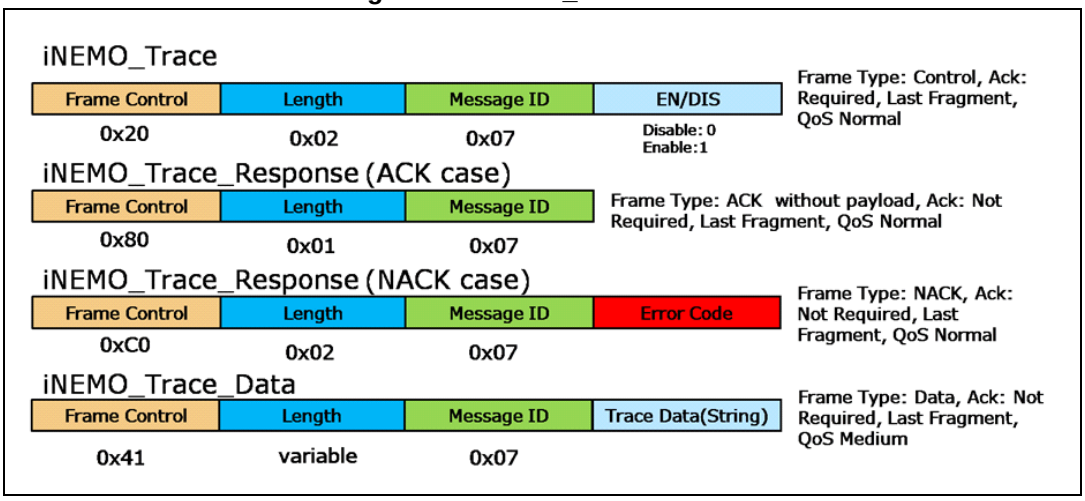
Figure 10. iNEMO_Enter_DFU_Mode frames



2.2.5 iNEMO_Trace

The iNEMO_Trace command allows the user to enable or disable "trace data". Trace data are used for debugging purposes and they will be string displayed in a debug window. The frames are asynchronous and shall have medium priority (QoS sub-field of frame control field). *Figure 11* shows the frames involved in the iNEMO_Trace transaction

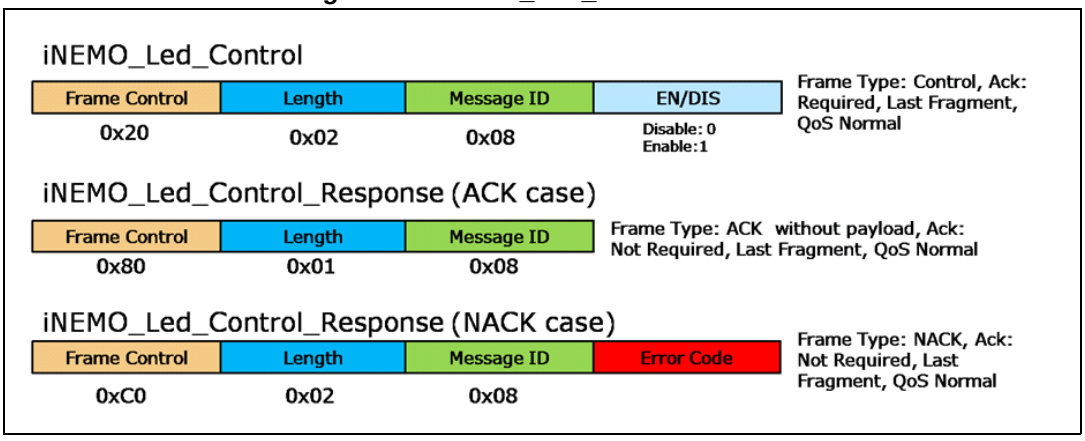
Figure 11. iNEMO_Trace frames



2.2.6 iNEMO_Led_Control

The iNEMO_Led_Control command allows turning on and off the LED available on the iNEMO board. *Figure 12* shows the frames involved in the iNEMO_Led_Control transaction.

Figure 12. iNEMO_Led_Control frames



2.3 Board information frames

Board information frames are frames originated by the software PC (SDK or GUI) and used to retrieve information about firmware and hardware features of the Discovery-M1 board. All the board information frames are listed in *Table 5*.

Table 5. Board information frames

Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
iNEMO_Get_Device_Mode	CONTROL	Y	0x10	N	0		PC
iNEMO_Get_Device_Mode_ Response	ACK	N	0x10	N	1	0x00 Sensor Mode 0x01 Master Mode	Discovery-M1
	NACK	N	0x10	N	1	Error Code	
iNEMO_Get_MCU_ID	CONTROL	Y	0x12	N	0		PC
iNEMO_Get_MCU_ID_Response	ACK	N	0x12	N	12	Unique Device ID	Discovery-M1
	NACK	N	0x12	N	1	Error Code	
iNEMO_Get_FW_Version	CONTROL	Y	0x13	N	0		PC
iNEMO_Get_FW_Version_ Response	ACK	N	0x13	N	Variable	String Firmware Version	Discovery-M1
	NACK	N	0x13	N	1	Error Code	
iNEMO_Get_HW_Version	CONTROL	Y	0x14	N	0	Date, Time	PC
iNEMO_Get_HW_Version_ Response	ACK	N	0x14	N	Variable	String Hardware Version	Discovery-M1
	NACK	N	0x14	N	1	Error Code	
iNEMO_Identify	CONTROL	Y	0x15	N	0		PC
iNEMO_Identify_Response	ACK	N	0x15	N	12	Unique Device ID	Discovery-M1
	NACK	N	0x15	N	1	Error Code	
iNEMO_Get_AHRS_Library	CONTROL	Y	0x17	N	0		PC
iNEMO_Get_AHRS_Library_ Response	ACK	N	0x17	N	Variable	AHRS enable/disable string	Discovery-M1
	NACK	N	0x17	N	1	Error Code	
iNEMO_Get_Libraries	CONTROL	Y	0x18	N	0		PC
iNEMO_Get_Libraries_Response	ACK	N	0x18	N	0	List of supported libraries	Discovery-M1
	NACK	N	0x18	N	1	Error Code	
iNEMO_Get_Available_Sensors	CONTROL	Y	0x19	N	0		PC

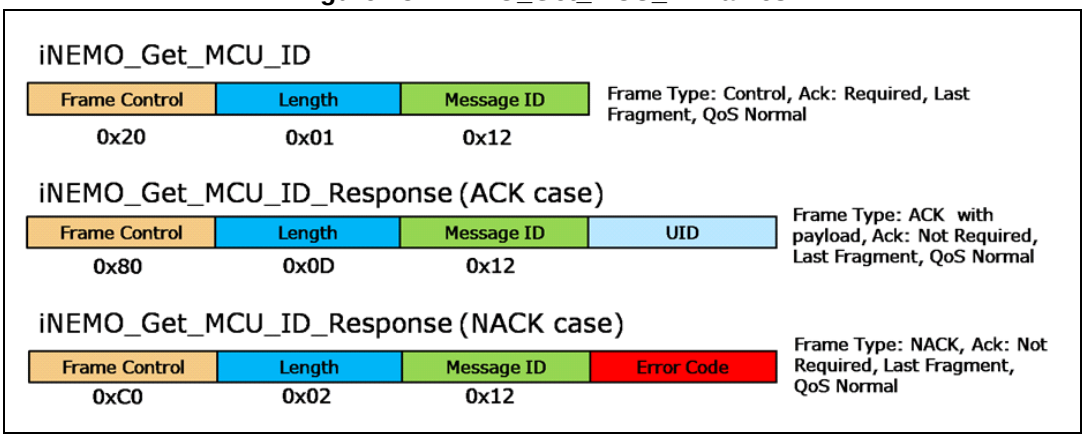
Table 5. Board information frames (continued)

Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
iNEMO_Get_Available_Sensors_ Response	ACK	N	0x19	N	1	List of available sensors	Discovery-M1
	NACK	N	0x19	N	1	Error Code	

2.3.1 iNEMO_Get_MCU_ID

The iNEMO_Get_MCU_ID command allows retrieving from the Discovery-M1 board the 96-bit unique device identifier of the STM32F103 microcontroller. *Figure 13* shows the frames involved in the iNEMO_Get_MCU_ID transaction.

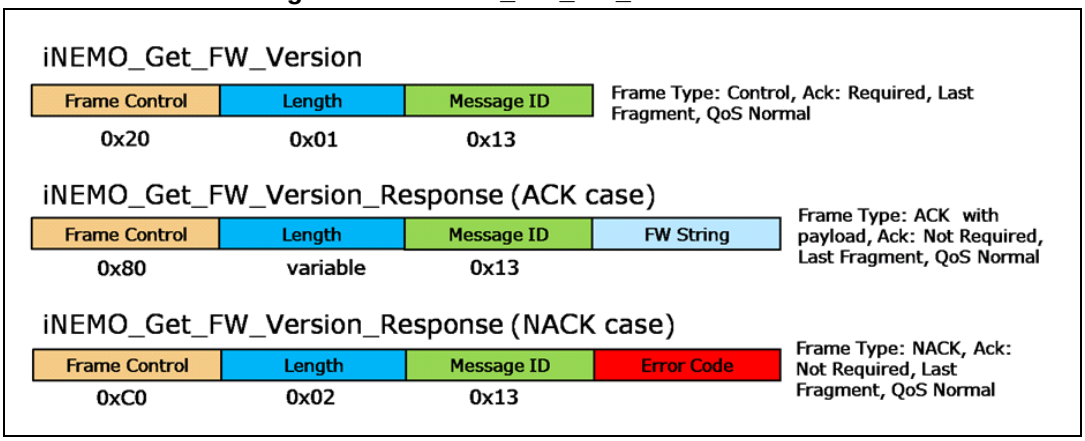
Figure 13. iNEMO_Get_MCU_ID frames



2.3.2 iNEMO_Get_FW_Version

The iNEMO_Get_FW_Version command allows retrieving the board firmware version. *Figure 14* shows the frames involved in the iNEMO_Get_FW_Version transaction

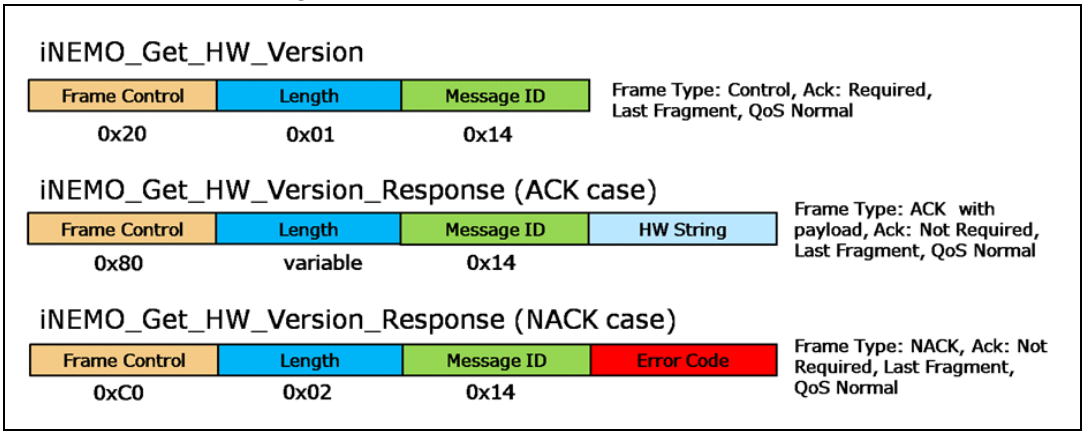
Figure 14. : iNEMO_Get_FW_Version frames



2.3.3 iNEMO_Get_HW_Version

The iNEMO_Get_HW_Version command allows retrieving the board hardware version. *Figure 15* shows the frames involved in the iNEMO_Get_HW_Version transaction.

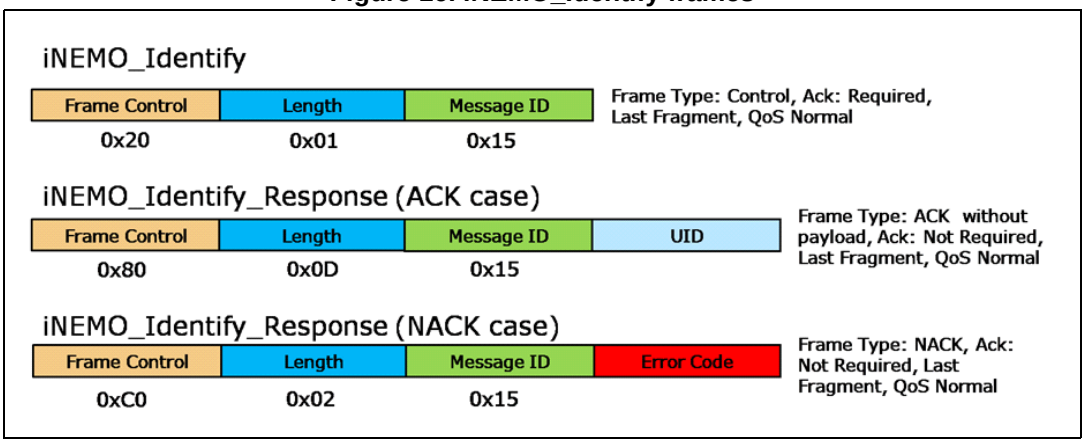
Figure 15. iNEMO_Get_HW_Version frames



2.3.4 iNEMO_Identify

The iNEMO_Identify command can be used to identify a Discovery-M1 board. Upon reception of the iNEMO_Identify command the Discovery-M1 board replies with an ACK containing the MCU Unique Device ID. Then the LED available on the board will blink 3 times. *Figure 16* shows the frames involved in the iNEMO_Identify transaction

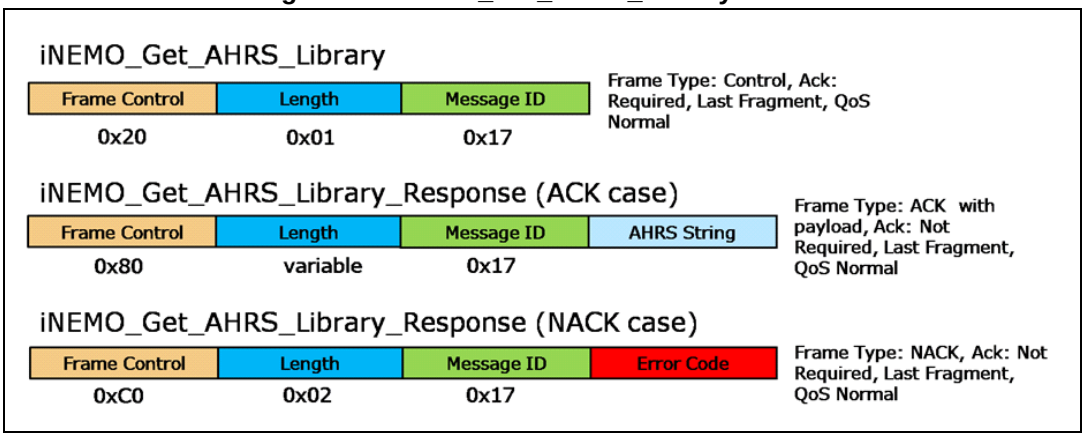
Figure 16. iNEMO_Identify frames



2.3.5 iNEMO_Get_AHRS_Library

The iNEMO_Get_AHRS_Library command allows knowing the version of the Discovery-M1 firmware Attitude Heading Reference System (AHRS) algorithm. The returned value is in string format. *Figure 17* shows the frames involved in the iNEMO_Get_AHRS_Library transaction.

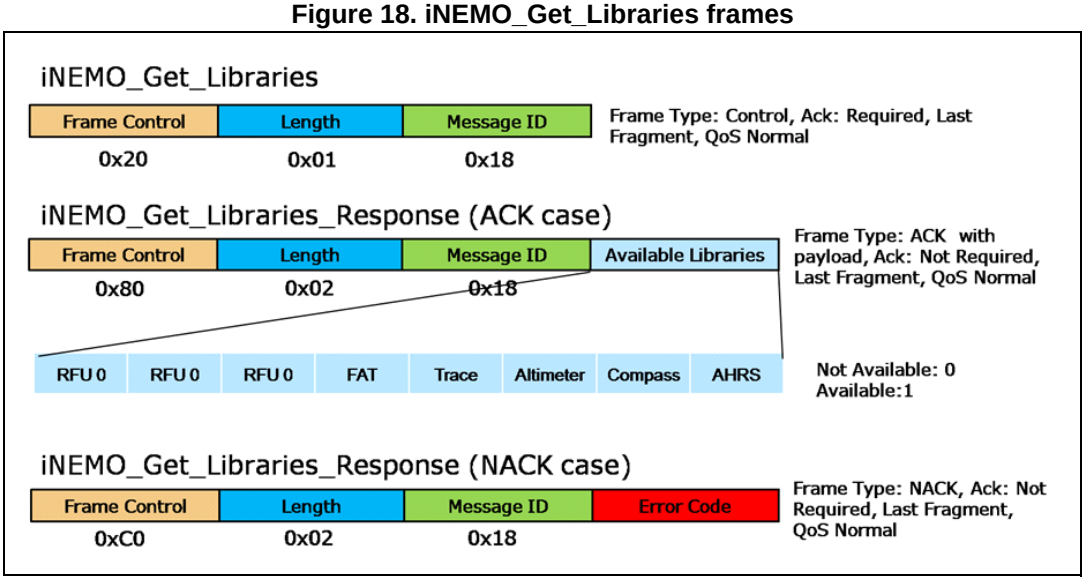
Figure 17. iNEMO_Get_AHRS_Library frames



2.3.6 iNEMO_Get_Libraries

The iNEMO_Get_Libraries command allows knowing which specific libraries are supported by the Discovery-M1 firmware. *Figure 18* shows the frames involved in the iNEMO_Get_Libraries transaction.

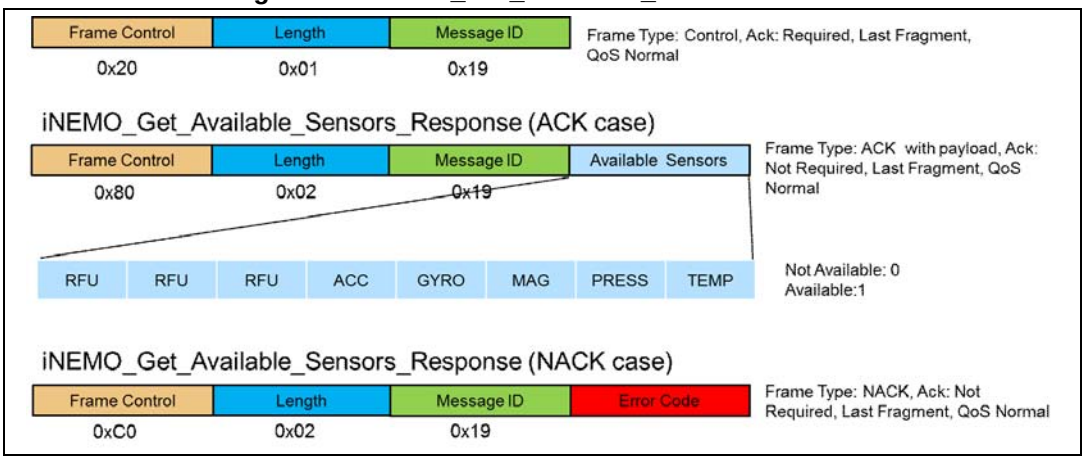
Figure 18. iNEMO_Get_Libraries frames



2.3.7 iNEMO_Get_Available_Sensors

The iNEMO_Get_Available_Sensors command allows knowing which specific sensors are supported by the Discovery-M1 firmware. *Figure 19* shows the frames involved in the iNEMO_Get_Available_Sensors transaction

Figure 19. iNEMO_Get_Available_Sensors frames



2.4 Sensor setting frames

Sensor setting frames are frames originated by the software PC (SDK or GUI) and used to set sensor parameters or to retrieve information about them. All the sensor setting frames are listed in *Table 6*.

Table 6. Sensor setting frames

Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
iNEMO_Set_Sensor_Parameter	CONTROL	Y	0x20	N	variable	Sensor_ Type, Sensor_ Parameter, Parameter_ Value	PC
iNEMO_Set_Sensor_Parameter_ Response	ACK	N	0x20	N	0		Discovery-M1
	NACK	N	0x20	N	1	Error Code	
iNEMO_Get_Sensor_Parameter	CONTROL	Y	0x21	N	2	Sensor_ Type, Sensor_ Parameter,	PC
iNEMO_Get_Sensor_Parameter_ Response	ACK	N	0x21	N	variable	Sensor_ Type, Sensor_ Parameter, Parameter_ Value	Discovery-M1
	NACK	N	0x21	N	1	Error Code	

Table 6. Sensor setting frames (continued)

Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
iNEMO_Restore_Default_ Parameter	CONTROL	Y	0x22	N	2	Sensor_ Type, Sensor_ Parameter	PC
iNEMO_Restore_Default_ Parameter_Response	ACK	N	0x22	N	variable	Sensor_ Type, Sensor_ Parameter, Parameter_ Value	Discovery-M1
	NACK	N	0x22	N	1	Error Code	
iNEMO_Save_to_Flash	CONTROL	Y	0x23	N	0		PC
iNEMO_Save_to_Flash_ Response	ACK	N	0x23	N	0		Discovery-M1
	NACK	N	0x23	N	1	Error Code	
iNEMO_Load_from_Flash	CONTROL	Y	0x24	N			PC
iNEMO_Load_from_Flash_ Response	ACK	N	0x24	N			Discovery-M1
	NACK	N	0x24	N	1	Error Code	

2.4.1 iNEMO_Set_Sensor_Parameter

The iNEMO_Set_Sensor_Parameter command allows setting a specific sensor parameter. *Figure 20* shows the frames involved in the iNEMO_Set_Sensor_Parameter transaction.

Figure 20. iNEMO_Set_Sensor_Parameter frames

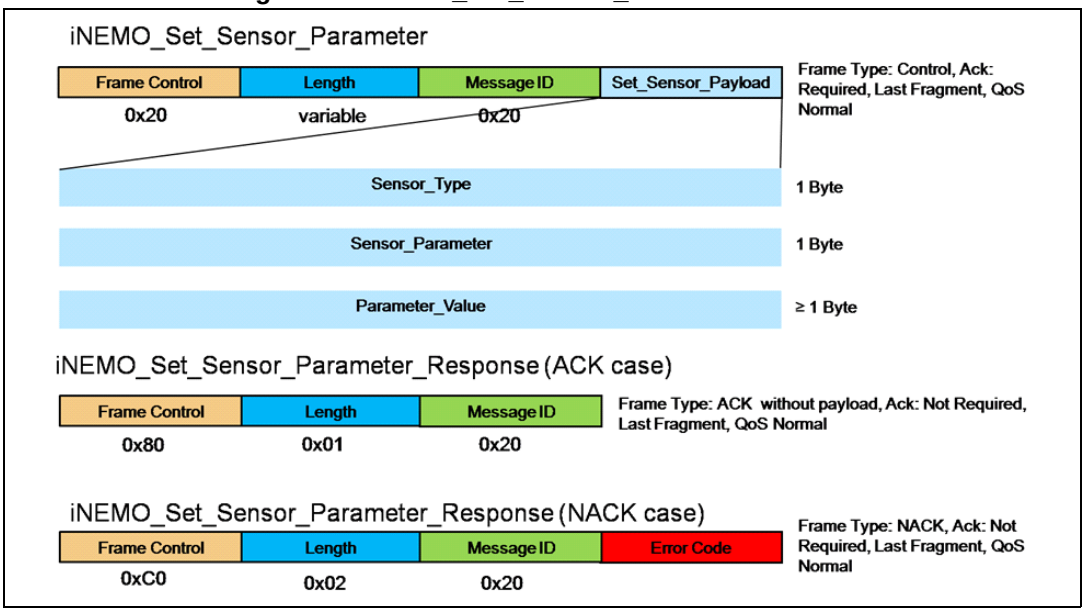


Table 7 describes the "Sensor_Type" field.

2.4.2 iNEMO_Get_Sensor_Parameter

The iNEMO_Get_Sensor_Parameter command allows retrieving from the Discovery-M1 a specific sensor parameter. *Figure 21* shows the frames involved in the iNEMO_Get_Sensor_Parameter transaction.

Figure 21. iNEMO_Get_Sensor_Parameter frames

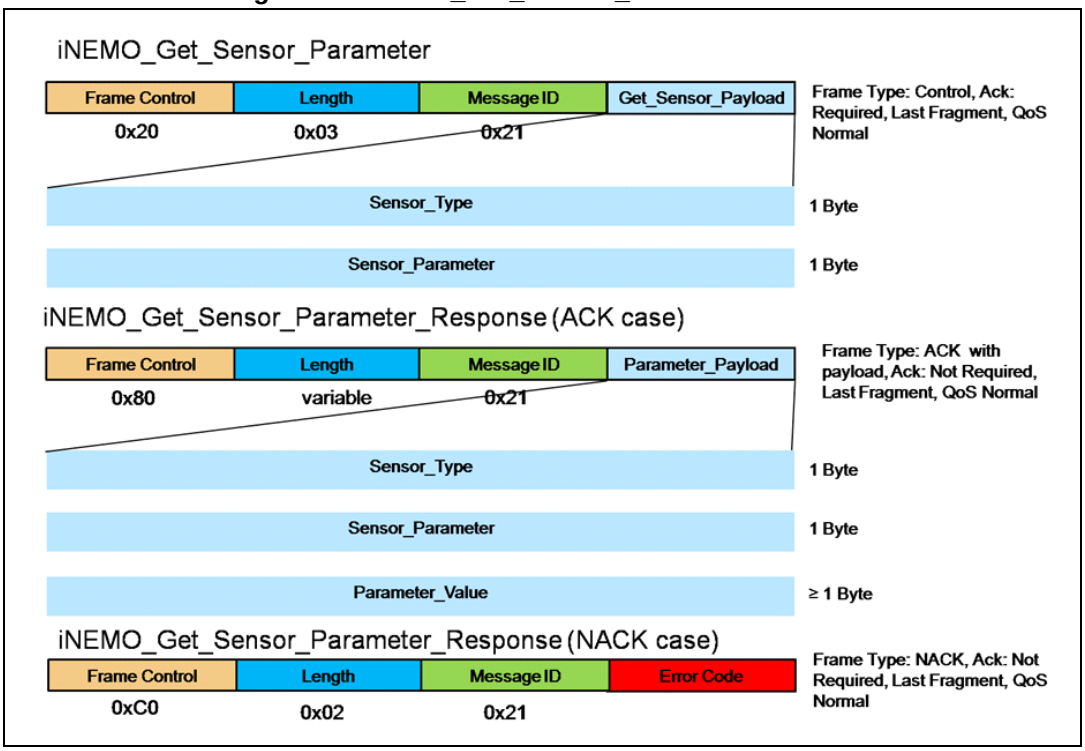


Table 7 describes the "Sensor_Type" field.

2.4.3 iNEMO_Restore_Default_Parameter

The iNEMO_Restore_Default_Parameter command allows restoring a default, specific sensor parameter. *Figure 22* shows the frames involved in the iNEMO_Restore_Default_Parameter transaction.

Figure 22. iNEMO_Restore_Default_Parameter frames

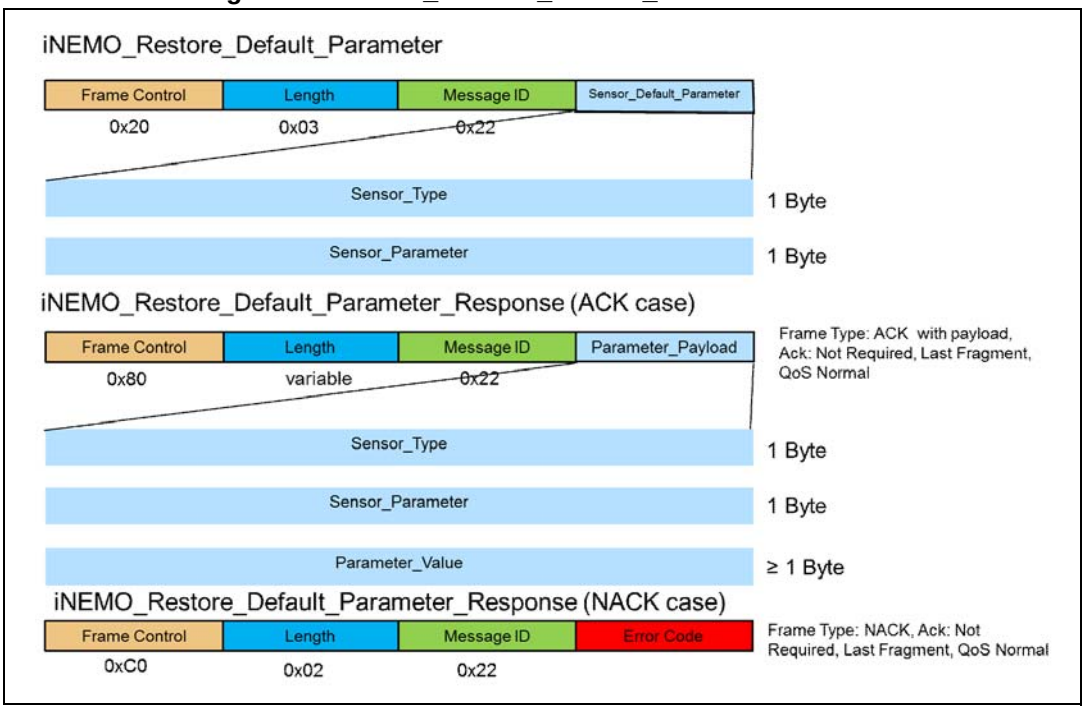


Table 7 describes the "Sensor_Type" list.

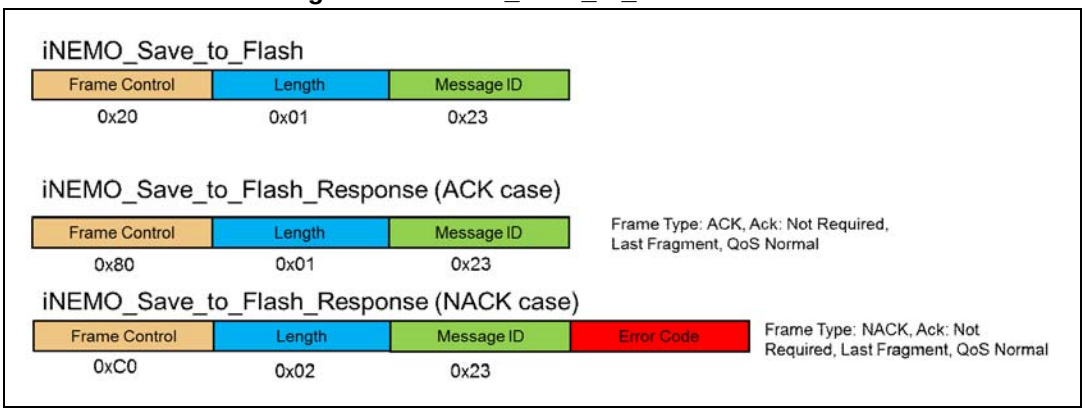
Table 7. Sensor_Type list

Sensor_Type Field	Sensor
0x00	3-axis accelerometer
0x01	3-axis magnetometer
0x02	3-axis gyroscope
0x03	-
0x04	Pressure
0x05	Temperature
0x06 – 0xFF	Reserved for Future Use

2.4.4 iNEMO_Save_to_Flash

The iNEMO_Save_to_Flash command allows storing the settings of the sensor parameters in Discovery-M1 flash. *Figure 23* shows the frames involved in the iNEMO_Save_to_Flash transaction.

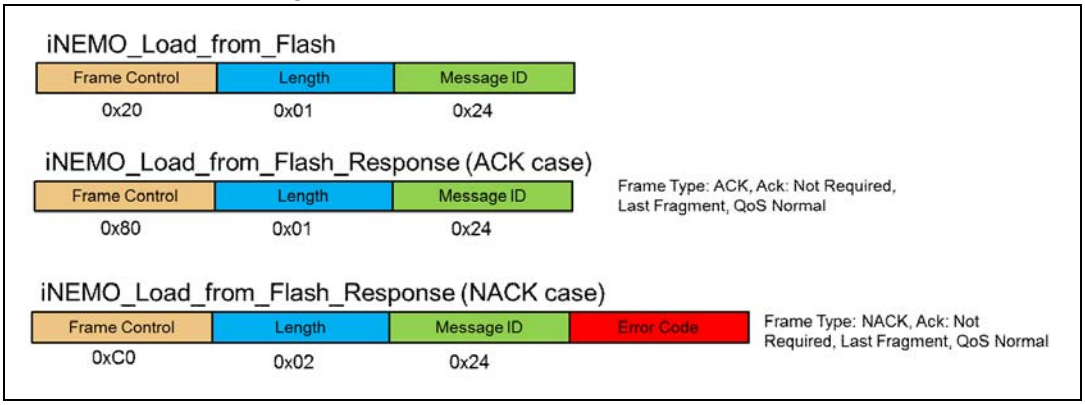
Figure 23. iNEMO_Save_to_Flash frames



2.4.5 iNEMO_Load_from_Flash

The iNEMO_Load_from_Flash command allows loading from Discovery-M1 flash the sensors parameters stored in it. *Figure 24* shows the frames involved in the iNEMO_Load_from_Flash transaction.

Figure 24. iNEMO_Load_from_Flash frames



2.4.6 Accelerometer "Sensor_Parameter" field

Table 8 describes the parameters of the accelerometer sensor and the values of the "Sensor_Parameter" field.

Table 8. Accelerometer Sensor_Parameter list

Sensor_Parameter field	Sensor
0x00	Output data rate
0x01	Full scale
0x02	Acc_HPF
0x03	Offset_X
0x04	Offset_Y
0x05	Offset_Z
0x06	Scale factor X

Table 8. Accelerometer Sensor_Parameter list (continued)

Sensor_Parameter field	Sensor
0x07	Scale factor Y
0x08	Scale factor Z
0xFF	Sensor name (read only)
0x09 – 0xFE	Reserved for Future Use

2.4.7 Accelerometer Output_Data_rate

The "Parameter_Value" field for the output data rate setting is 1 byte in length. *Table 9* describes the supported output data rate for the accelerometer.

Table 9. Accelerometer output data rate field

Parameter_Value field for accelerometer ODR	Output data rate (Hz)
0x00	1
0x01	10
0x02	25
0x03	50
0x04	100
0x05	200
0x06	400
0x07– 0xFF	RFU

2.4.8 Accelerometer full scale

The "Parameter_Value" field for the full-scale setting is 1 byte in length. *Table 10* describes the supported full scale for the accelerometer.

Table 10. Accelerometer full scale field

Parameter_Value field for accelerometer FS	Full scale (g)
0x00	±2g
0x01	±4g
0x02	±8g
0x03	±16g
0x04 – 0xFF	RFU

2.4.9 Accelerometer high-pass filter

The "Parameter_Value" field for the high-pass filter setting is 2 bytes in length as described in *Figure 25*. *Table 11* describes the possible cutoff frequencies.

Figure 25. "Parameter_Value" fields for accelerometer HPF setting

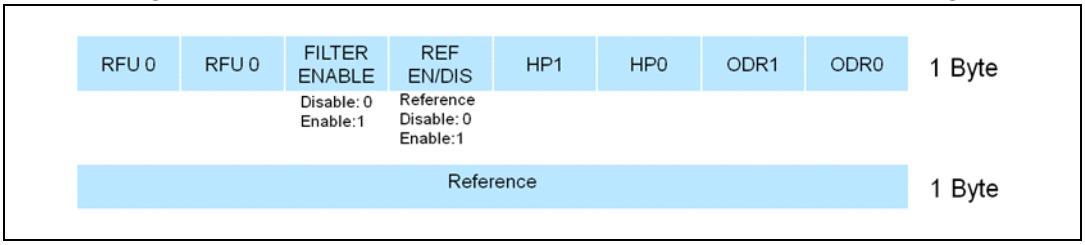


Table 11. Accelerometer high-pass filter setting

HP1	HP0	f_t[HZ] Data rate = 1 Hz	f_t[HZ] Data rate = 10 Hz	f_t[HZ] Data rate = 25 Hz	f_t[HZ] Data rate = 50 Hz	f_t[HZ] Data rate = 100 Hz	f_t[HZ] Data rate = 200 Hz	f_t[HZ] Data rate = 400 Hz
0	0	0.02	0.2	0.052	1.04	2.08	4.16	8.33
0	1	0.01	0.1	0.26	0.52	1.04	2.08	4.16
1	0	0.005	0.05	0.13	0.26	0.52	1.04	2.08
1	1	0.0026	0.026	0.065	0.13	0.26	0.52	1.04

For further details please refer to the LSM303DLHC datasheet .

2.4.10 Accelerometer offset

The "Parameter_Value" field for the offset (X-, Y- or Z-axis) setting is 2 bytes in length and expressed in milli-*g* (thousandth of gravitational force) as signed short (16-bit), with the most significant byte first.

2.4.11 Accelerometer scale factor

The "Parameter_Value" field for the scale factor (X-, Y- or Z-axis) setting is 2 bytes in length (abstract number not mg) as signed short (16-bit) multiplied x1000 and with the most significant byte first. For example, if in the setting view scale factor x is 1.230, it will be multiplied x1000 and sent as 1230 (signed short 16-bit) in this case.

2.4.12 Accelerometer sensor name

The "Parameter_Value" of the sensor name is a read-only field and it returns the name of accelerometer (LSM303DLHC) from Discovery-M1.

2.4.13 Magnetometer "Sensor_Parameter" field

Table 12 describes the parameters of the magnetometer sensor and the values of the "Sensor_Parameter" field.

Table 12. Accelerometer Sensor_Parameter list

Sensor_Parameter field	Parameter
0x00	Output data rate
0x01	Full scale
0x02	Operating mode
0x03	Offset_X
0x04	Offset_Y
0x05	Offset_Z
0x06	Scale factor X
0x07	Scale factor Y
0x08	Scale factor Z
0xFF	Sensor name (read only)
0x09 – 0xFE	RFU

2.4.14 Magnetometer Output_Data_rate

The "Parameter_Value" field for the output data rate setting is 1 byte in length. *Table 13* describes the output data rate supported for the magnetometer.

Table 13. Magnetometer output data rate field

Parameter_Value field for magnetometer ODR	Output data rate (Hz)
0x00	0.75
0x01	1.5
0x02	3
0x03	7.5
0x04	15
0x05	30
0x06	75
0x07	220
0x08 – 0xFF	RFU

2.4.15 Magnetometer full scale

The "Parameter_Value" field for the full-scale setting is 1 byte in length. *Table 14* describes the full scale supported for the magnetometer.

Table 14. Magnetometer full-scale field

Parameter_Value field for magnetometer FS	Full scale (gauss)
0x01	±1.3
0x02	±1.9
0x03	±2.5
0x04	±4.0
0x05	±4.7
0x06	±5.6
0x07	±8.1
0x00, 0x08 – 0xFF	Forbidden-RFU

2.4.16 Magnetometer operating mode

The "Parameter_Value" field for the operating mode setting is 1 byte in length as described in *Figure 26*. *Table 15* describes the possible magnetometer operating modes.

Figure 26. "Parameter_Value" fields for magnetometer operating mode setting

Table 15. Magnetometer operating mode setting

Mode1	Mode0	Magnetic sensor operating mode
0	0	Normal
0	1	Positive bias
1	0	Negative bias
1	1	Forbidden

For further details please refer to the LSM303DLHC datasheet.

2.4.17 Magnetometer offset

The "Parameter_Value" field for the offset (X-, Y- or Z-axis) setting is 2 bytes in length and expressed in milli-gauss (thousandth of gauss) as signed short (16-bit), with the most significant byte first.

2.4.18 Magnetometer scale factor

The "Parameter_Value" field for the scale factor (X-, Y- or Z-axis) setting is 2 bytes in length (abstract number not mgauss) as signed short (16-bit) multiplied x1000.

2.4.19 Magnetometer sensor name

The "Parameter_Value" of sensor name is a read-only field and it returns the name of magnetometer (LSM303DLHC) from Discovery-M1.

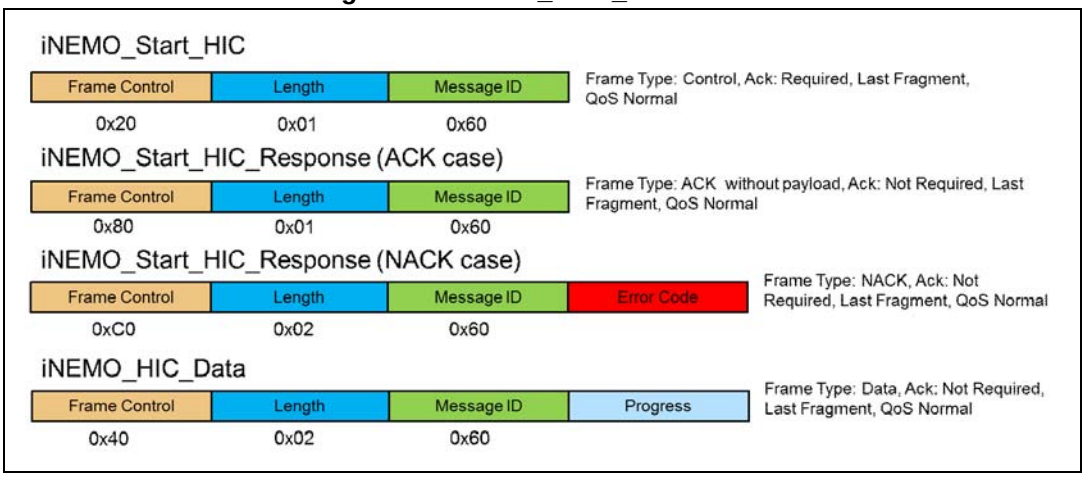
2.4.20 Calibration sensor frames

Calibration sensor frames are frames originated by the software PC (SDK or GUI) used to calibrate sensors. The calibration sensor frames implemented in this version of the protocol are related to the magnetometer hard-iron calibration (HIC) as described below.

2.4.21 iNEMO_Start_HIC

The iNEMO_Start_HIC command is used to start the magnetic sensor HIC procedure. *Table 27* shows the frames involved in the iNEMO_Start_HIC transaction.

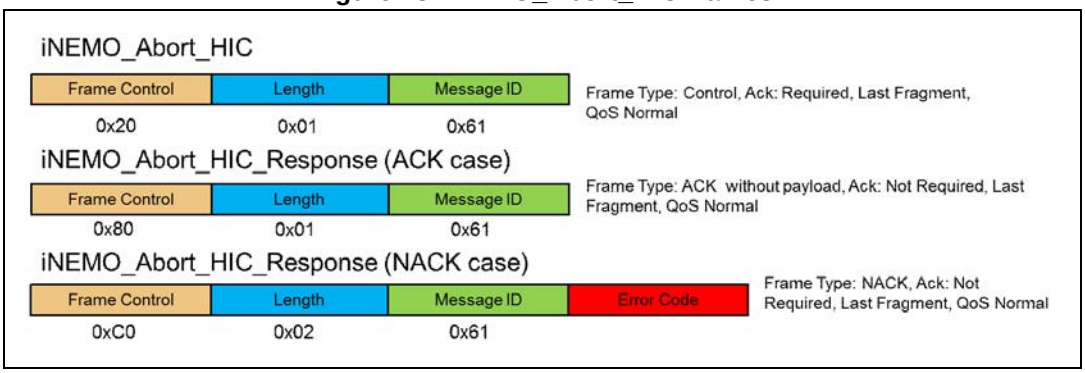
Figure 27. iNEMO_Start_HIC frames



2.4.22 iNEMO_Abort_HIC

The iNEMO_Abort_HIC command is used to abort the magnetic sensor HIC procedure. *Table 28* shows the frames involved in the iNEMO_Abort_HIC transaction.

Figure 28. iNEMO_Abort_HIC frames



2.4.23 Gyroscope "Sensor_Parameter" field

Table 16 describes the parameters of the gyroscope sensor and the values of the "Sensor_Parameter" field.

Table 16. Gyroscope Sensor_Parameter list

Sensor_Parameter field	Parameter
0x00	Output data rate – LPF cutoff
0x01	Full scale
0x02	Gyro_HPF
0x03	Offset X
0x04	Offset Y
0x05	Offset Z
0x06	Scale factor X
0x07	Scale factor Y
0x08	Scale factor Z
0xFF	Sensor name (read only)
0x09 – 0xFE	RFU

2.4.24 Gyroscope output data rate

The "Parameter_Value" field for the output data rate setting is 1 byte in length. *Table 17* describes the output data rate supported for the gyroscope.

Table 17. Gyroscope output data rate

Parameter_Value field for gyroscope ODR	Output data rate (Hz)	LP filter cutoff (Hz)
0x00	95	12.5
0x01		25
0x02		25
0x03		25
0x04	190	12.5
0x05		25
0x06		50
0x07		70
0x08	380	20
0x09		25
0x0A		50
0x0B		110

Table 17. Gyroscope output data rate (continued)

Parameter_Value field for gyroscope ODR	Output data rate (Hz)	LP filter cutoff (Hz)
0x0C	760	30
0x0D		35
0x0E		50
0x0F		110

2.4.25 Gyroscope full scale

The "Parameter_Value" field for the full scale setting is 1 byte in length. *Table 18* describes the full scale supported for the gyroscope.

Table 18. Gyroscope Full Scale field

Parameter_Value field for gyroscope FS	Full scale (dps)
0x00	±250 dps
0x01	±500 dps
0x02	±2000 dps
0x03 – 0xFF	Forbidden - RFU

2.4.26 Gyroscope offset

The "Parameter_Value" field for the Offset (X-,Y- and Z-axis) setting is 2 bytes in length and expressed in dps (degrees per second) as signed short (16-bit) with the most significant byte first.

2.4.27 Gyroscope scale factor

The "Parameter_Value" field for the scale factor (X-, Y- and Z-axis) setting is 2 bytes in length (abstract number not dps) as signed short (16-bit) multiplied x1000.

2.4.28 Gyroscope sensor name

The "Parameter_Value" of the sensor name is a read-only field and it returns the gyroscope part number (L3GD20) present on the Discovery-M1.

2.4.29 Pressure "Sensor_Parameter" field

Table 19 describes the parameters of the pressure sensor and the values of the "Sensor_Parameter" field.

Table 19. Pressure Sensor_Parameter list

Sensor_Parameter field	Parameter
0x00	Output data rate
0x01	Offset
0x02	Scale factor

Table 19. Pressure Sensor_Parameter list (continued)

Sensor_Parameter field	Parameter
0xFF	Sensor name (read only)
0x02 – 0xFF	Forbidden - RFU

2.4.30 Pressure sensor output data rate

The "Parameter_Value" field for the output data rate setting is 1 byte in length. *Table 20* describes the output data rate supported for the pressure sensor.

Table 20. Pressure sensor output data rate field

“Parameter_Value” field for pressure sensor ODR	Output data rate (Hz)
0x00	1
0x01	7
0x02	12.5
0x03	25
0x04 – 0xFF	Forbidden - RFU

2.4.31 Pressure sensor offset

The "Parameter_Value" field for the offset setting is 2 bytes in length and expressed in millibar as signed short (16-bit) with the most significant byte first.

2.4.32 Pressure scale factor

The "Parameter_Value" field for the scale factor setting is 2 bytes in length as signed short (16-bit) multiplied x1000.

2.4.33 Pressure sensor name

The "Parameter_Value" of the sensor name is a read-only field and it returns the pressure sensor part number (LPS331AP) present on the Discovery-M1.

2.4.34 Temperature "Sensor_Parameter" field

Table 21 describes the parameters of the temperature sensor and the values of the "Sensor_Parameter" field.

Table 21. Temperature Sensor_Parameter list

Sensor_Parameter field	Parameter
0x00	Offset
0x01	Scale factor
0xFF	Sensor name (read only)
0x02 – 0xFE	RFU

2.4.35 Temperature sensor offset

The "Parameter_Value" field for the offset setting is 2 bytes in length and expressed in dºC (tenth of Celsius degrees) as signed short (16-bit) with the most significant byte first.

2.4.36 Temperature sensor scale factor

The "Parameter_Value" field for the scale factor setting is 2 bytes in length (abstract number) as signed short (16-bit) multiplied x1000.

2.4.37 Temperature sensor name

The "Parameter_Value" of the sensor name is a read-only field and it returns the part number sensor, which is the source of the temperature value, present in Discovery-M1.

2.5 Acquisition sensor data frames

Acquisition sensor data frames are frames originated by the software PC (SDK or GUI) to set how to retrieve sensor data from Discovery-M1 and data frames originated by Discovery-M1 to send sensor data. Acquisition sensor data frames are listed in *Table 22*.

Table 22. Acquisition sensor data frames

Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
iNEMO_Set_Output_Mode	CONTROL	Y	0x50	N	4	Sensors, acquisition frequency, output type, number of samples	PC
iNEMO_Set_Output_Mode_ Response	ACK	N	0x50	N	0		Discovery-M1
	NACK	N	0x50	N	1	Error code	
iNEMO_Get_Output_Mode	CONTROL	Y	0x51	N	0		PC
iNEMO_Get_Output_Mode_ Response	ACK	N	0x51	N	4	Sensors, acquisition frequency, output type, number of samples	Discovery-M1
	NACK	N	0x51	N	1	Error code	
iNEMO_Start_Acquisition	CONTROL	Y	0x52	N	0		PC
iNEMO_Start_Acquisition_Respo nse	ACK	N	0x52	N	0		Discovery-M1
	NACK	N	0x52	N	1	Error Code	
iNEMO_Acquisition_Data	DATA	N	0x52	N	variable	Sensor Data	

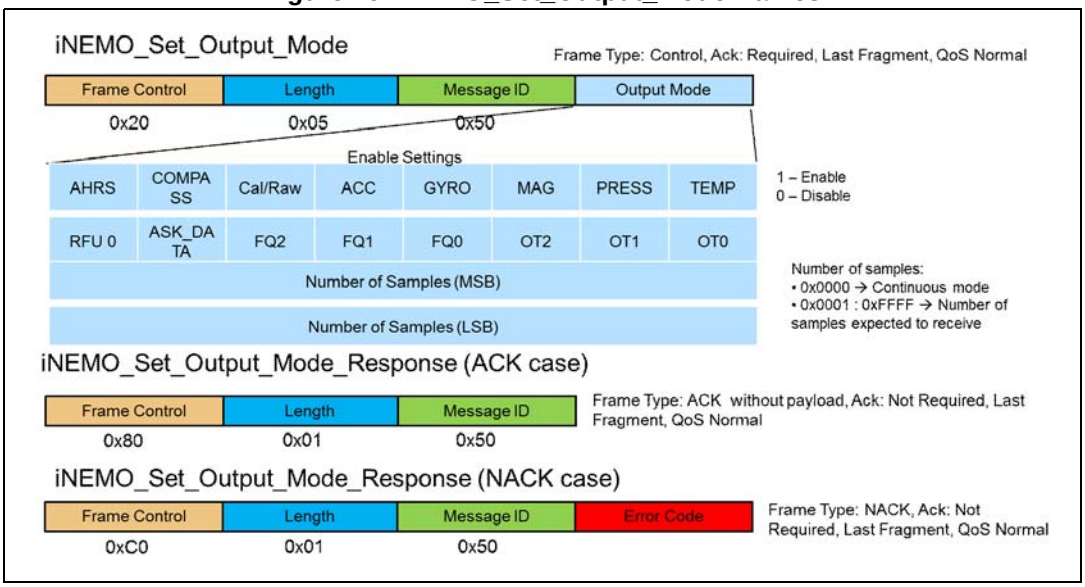
Table 22. Acquisition sensor data frames (continued)

Commands	Frame Type	Ack required	Message ID	QoS	Payload Length (in bytes)	Payload	Originator
iNEMO_Stop_Acquisition	CONTROL	Y	0x53	N	0		PC
iNEMO_Stop_Acquisition_Response	ACK	N	0x53	N	0		Discovery-M1
	NACK	N	0x53	N	1	Error Code	
iNEMO_Get_Acq_Data	CONTROL	Y	0x54	N	0		PC
iNEMO_Get_Acq_Data_Response	ACK	N	0x54	N	0		Discovery-M1
	NACK	N	0x54	N	1	Error Code	

2.5.1 iNEMO_Set_Output_Mode

The iNEMO_Set_Output_Mode command allows setting which sensors shall be enabled, in which format the data sensor shall be sent from Discovery-M1 to SDK, and other parameters. *Figure 29* shows the frames involved in the iNEMO_Set_Output_Mode transaction.

Figure 29. iNEMO_Set_Output_Mode frames



When the "AHRS" bit is set to "1", the Discovery-M1 will run the AHRS algorithm, and the AHRS data (quaternion and roll, pitch, yaw angles) will be sent in the data frame.

When the "COMPASS" field is set to "1", the Discovery-M1 will runs the compass demo, and the compass data (heading, roll and pitch angles) will be sent in the data frames.

The "Cal/Raw" field is used to define how to send sensor data, according to *Table 23*.

Table 23. Calibrated and raw fields

Cal/Raw field	Output Data Type
0	Calibrated Data
1	Raw data [LSB]

The "ACC", "GYRO", "MAG", "PRESS" and "TEMP" fields are used to enable or disable the acquisition of the respective sensors.

The "ASK_DATA" field enables/disables the ask data mode. If this field is set to 1, data will not be sent if not requested by the iNEMO_Get_Acq_Data frame.

The FQx fields are used to set the acquisition rate, according to *Table 24*.

Table 24. Acquisition rate

FQ2	FQ1	FQ0	Acquisition rate (Hz)
0	0	0	1
0	0	1	10
0	1	0	25
0	1	1	50
1	0	0	30
1	0	1	100
1	1	0	400
1	1	1	Synchronized to sensor

If the selected frequency is one of the above values from 1 Hz to 400 Hz, the data acquisition is based on a timer running at the selected frequency and it is not synchronized with any sensors. Otherwise, if the "Acquisition Rate" field is "Synchronized to sensor" (FQ[2:0] set to ‘111’), the acquisition frequency is synchronized with the ODR (output data rate) of one sensor. In this last case only the selected sensor will run and the other ones will be automatically disabled.

The OTx fields are used to set the interface through which the Discovery-M1 shall send the data. The only interface supported by the actual version of Discovery-M1 is the USB interface, as shown in *Table 25*.

Table 25. Output interface

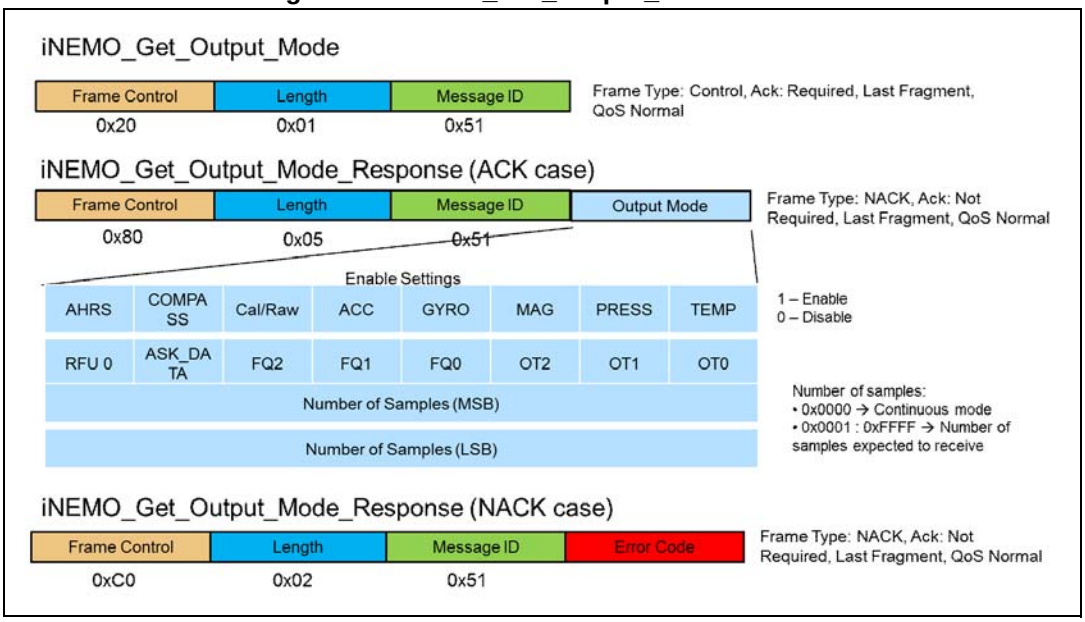
OT2	OT1	OT0	Output Interface
0	0	0	USB
0	0	1	RFU
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

The "Number of Samples" bytes specify how many sensor data samples shall be acquired. When set to zero (continuous mode), the Discovery-M1 will acquire and send sensor data to the PC until it receives the "iNEMO_Stop_Acquisition" command.

2.5.2 iNEMO_Get_Output_Mode

The iNEMO_Get_Output_Mode command allows retrieving information from Discovery-M1 about its acquisition settings. *Figure 30* shows the frames involved in the iNEMO_Get_Output_Mode transaction.

Figure 30. iNEMO_Get_Output_Mode frames

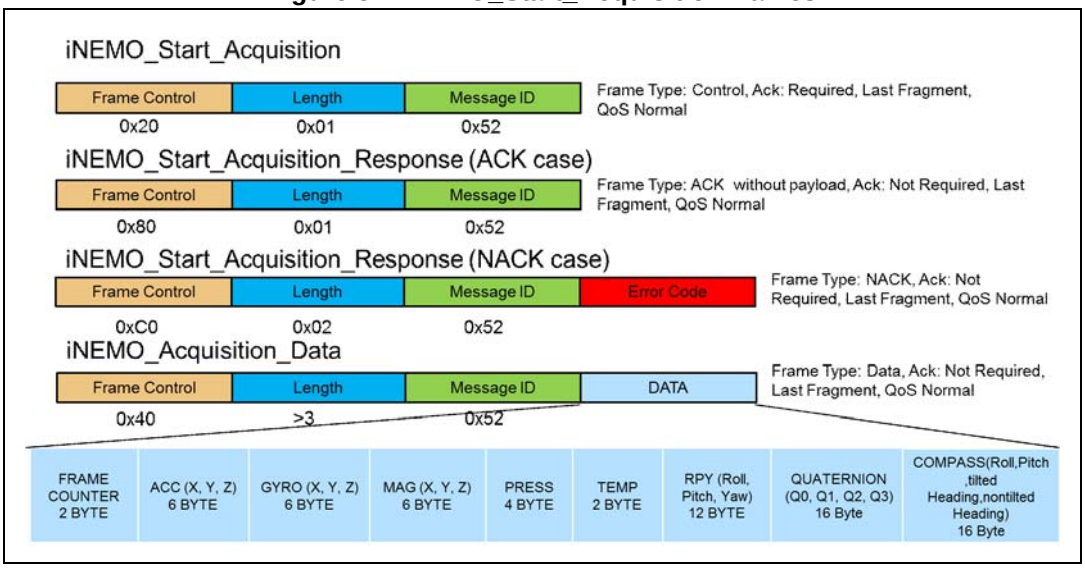


The "Output Mode" fields are described in *Section 2.5.1*.

2.5.3 iNEMO_Start_Acquisition

The iNEMO_Start_Acquisition command allows starting the acquisition of sensor data according to the output settings. *Figure 31* shows the frames involved in the iNEMO_Start_Acquisition transaction.

Figure 31. iNEMO_Start_Acquisition frames



The frame counter is an unsigned int (16-bit) incremented every time a new data frame is sent to the PC.

"ACC(X,Y,Z)" represents the accelerometer data field. This data field will be available only if the accelerometer sensor has been enabled ("ACC" field in the "iNEMO_Set_Output_Mode" frame is set to 1). If the "Raw" option is not enabled ("Raw" field in "iNEMO_Set_Output_Mode" frame is 0), each accelerometer axis value is expressed in m*g* (thousandth of gravitational force) in one of the possible ranges defined in *Table 10*, as a signed short value (2 bytes). If the "Raw" option is enabled, each axis value represents the raw sensor data (LSB value).

"GYRO(X,Y,Z)" represents the gyroscope data field. This data field will be available only if the gyroscope sensor has been enabled ("GYRO" field in the "iNEMO_Set_Output_Mode" frame is set to 1). If the "Raw" option is not enabled, each gyroscope axis value is expressed in dps (degrees per second) in one of the possible ranges defined in *Table 18*, as a signed short value (2 bytes). If the "Raw" option is enabled, each axis represents the raw sensor data (LSB value).

"MAG(X,Y,Z)" represents the magnetometer data field. This data field will be available only if the magnetometer sensor has been enabled ('MAG" field in the "iNEMO_Set_Output_Mode" frame is set to 1). If the "Raw" option is not enabled, each magnetometer axis value is expressed in mG (thousandth of Gauss) in one of the possible ranges defined in *Table 14*, as a signed short value (2 bytes). If the "Raw" option is enabled, each axis value represents the raw sensor data (LSB value).

"PRESS" represents the pressure data field. This data field will be available only if the pressure sensor has been enabled ("PRESS" field in the "iNEMO_Set_Output_Mode" frame is 1). If the "Raw" option is not enabled, the pressure value is expressed in c-mbar (centi-mbar ie. one hundredth of a millibar) in the range [+26000 c-mbar, +126000 c-mbar], as a signed int value (4 bytes). If the "Raw" option is enabled, the pressure data field represents the raw sensor data (LSB value).

"TEMP" represents the temperature data field. This data field will be available only if the temperature sensor has been enabled ("TEMP" field in the "iNEMO_Set_Output_Mode" frame is set to 1). If the "Raw" option is not enabled, the temperature value is expressed in dºC (tenth of Celsius degrees) in the range [-400 dºC, + 1250 dºC], as a signed short value (2 bytes). If the "Raw" option is enabled, the temperature data field represents the raw sensor data (LSB value).

"RPY" represents the roll, pitch, yaw data field. This data field will be available only if the "AHRS" option has been enabled ("AHRS" field in the "iNEMO_Set_Output_Mode" frame is 1).

The Roll data is expressed as a floating point value (4 bytes) in the range of ±180 degrees.

The Pitch data is expressed as a floating point value (4 bytes) in the range of ±90 degrees.

The Yaw data is expressed as a floating point value (4 bytes) in the range of ±180 degrees.

"Quaternion" represents the quaternion data field. This data field will be available only if the "AHRS" option has been enabled ("AHRS" field in the "iNEMO_Set_Output_Mode" frame is 1). Each quaternion data is expressed as a floating point value (4 bytes) in the range ±1. The Q0 field represents the scalar part of the quaternion, while the Q1, Q2 and Q3 fields represent the vector part of the quaternion.

The "COMPASS" data will be available only if the user enables the "Compass" option ("COMPASS" field in the "iNEMO_Set_Output_Mode" frame is set to 1).

The compass Roll data is expressed as a floating point value (4 bytes) in the range ±90 degree.

The compass Pitch data is expressed as a floating point value (4 bytes) in the range ±90 degree.

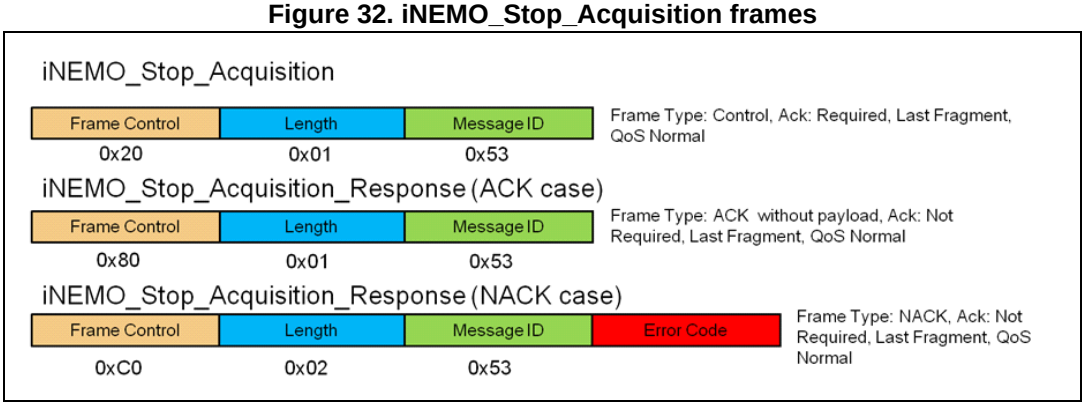
The compass Heading data is expressed as a floating point value (4 bytes) in the range ±180 degree.

During the acquisition and data transmission phase it is not possible to use commands that change the sensor settings or the output mode. It is necessary to stop the acquisition before sending these commands.

2.5.4 iNEMO_Stop_Acquisition

The iNEMO_Stop_Acquisition command stops the acquisition and data transmission. *Figure 32* shows the frames involved in the iNEMO_Stop_Acquisition transaction.

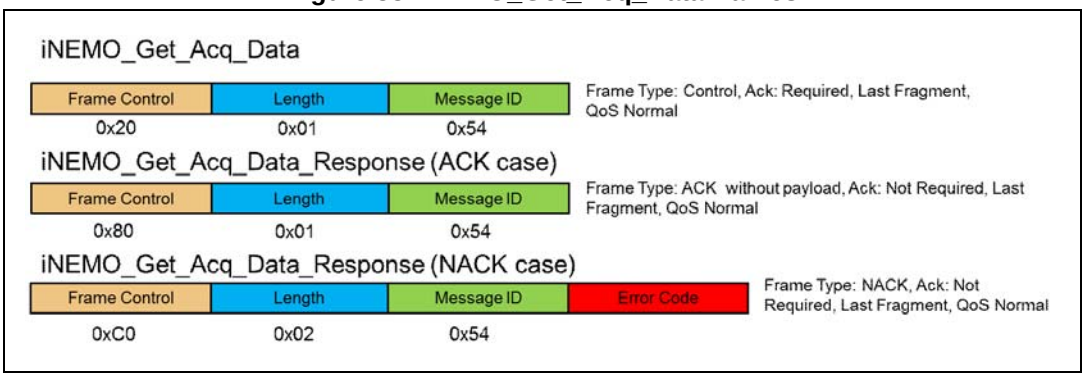
Figure 32. iNEMO_Stop_Acquisition frames



2.5.5 iNEMO_Get_Acquired_Data

The iNEMO_Get_Acquired_Data command is used to send the acquired data. *Figure 33* shows the frames involved in iNEMO_Get_Acquired_Data transaction.

Figure 33. iNEMO_Get_Acq_Data frames



2.6 Error code

All the error codes used in the NACK frames are listed in *Table 26*.

Table 26. Calibrated and raw fields

“Error Code” field	Error
0x00	Forbidden
0x01	Unsupported command
0x02	Out-of-range value
0x03	Not executable command
0x04	Wrong syntax
0x05	Discovery-M1 not connected
0x06 – 0xFF	RFU

3 Revision history

Table 27. Document revision history

Date	Revision	Changes
23-May-2014	1	Initial release.

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

ST PRODUCTS ARE NOT DESIGNED OR AUTHORIZED FOR USE IN: (A) SAFETY CRITICAL APPLICATIONS SUCH AS LIFE SUPPORTING, ACTIVE IMPLANTED DEVICES OR SYSTEMS WITH PRODUCT FUNCTIONAL SAFETY REQUIREMENTS; (B) AERONAUTIC APPLICATIONS; (C) AUTOMOTIVE APPLICATIONS OR ENVIRONMENTS, AND/OR (D) AEROSPACE APPLICATIONS OR ENVIRONMENTS. WHERE ST PRODUCTS ARE NOT DESIGNED FOR SUCH USE, THE PURCHASER SHALL USE PRODUCTS AT PURCHASER’S SOLE RISK, EVEN IF ST HAS BEEN INFORMED IN WRITING OF SUCH USAGE, UNLESS A PRODUCT IS EXPRESSLY DESIGNATED BY ST AS BEING INTENDED FOR “AUTOMOTIVE, AUTOMOTIVE SAFETY OR MEDICAL” INDUSTRY DOMAINS ACCORDING TO ST PRODUCT DESIGN SPECIFICATIONS. PRODUCTS FORMALLY ESCC, QML OR JAN QUALIFIED ARE DEEMED SUITABLE FOR USE IN AEROSPACE BY THE CORRESPONDING GOVERNMENTAL AGENCY.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.
Information in this document supersedes and replaces all information previously supplied.
The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Multiple Function Sensor Development Tools ***category:***

Click to view products by STMicroelectronics ***manufacturer:***

Other Similar products are found below :

MAXWSNENV# STEVAL-MKIT01V1 KT-O2-25%-TB200A-E KT-TVOC-200-TB200A KT-NmHc-200-TB200A SEN0344 PIM520 PIM518 PIM519 PIM502 EVAL-AD7746HDZ AS7022-EVALKIT ALTEHTG2SMIP MAX30101WING# OB1203SD-U-EVK MIKROE-4265 A000070 EV_ICG-20660L GX-F12A-P GX-F15A GX-F6A-P GX-F8AI-P GX-F8B GX-H12A-P GX-H15AI-P GX-H6A-P GX-HL15B-P 1093 MIKROE-2455 MIKROE-2458 MIKROE-2507 MIKROE-2508 MIKROE-2516 MIKROE-2529 1458 MIKROE-1628 176 189 1893 2106 ATQT4-XPRO GP30-DEMO MODULE GX-F12AI-P GX-F15A-P GX-F8A-P GX-FL15B-P GX-H12AI-P GX-H15A-P GX-H6AI-P GX-H8A