

# Inline Adapter ILA210

## Setup Instructions

## Table of Contents

Revision History .....	3
Getting Started .....	4
Device Connection .....	4
Software Setup .....	5
ILA210 Utility .....	5
Modbus RTU .....	5
Device Initial Setup and Editing Settings .....	6
Device Connect RS-485 Settings: .....	6
ILA Settings: .....	6
Measurement Setting: .....	7
RS-485 Setting: .....	8
LED Indicator Definitions: .....	8
ILA Utility Instructions .....	8
Connecting to a Device .....	8
Searching for a Device .....	9
Viewing the Data .....	10
Modbus Instructions .....	12
Modbus RTU Format .....	12
Modbus Functions .....	12
0X03(03) Function Code .....	13
0X04(04) Function Code .....	17
0X06(06) Function Code .....	26
0X10(16) Function Code .....	30
0X2B(43) Function Code .....	33
Error Code Definition .....	37

## Revision History

Version	Date	Comments
1.0	2 Jan 2026	First version
<b>2.0</b>	30 April 2026	Rev A - release version

## Getting Started

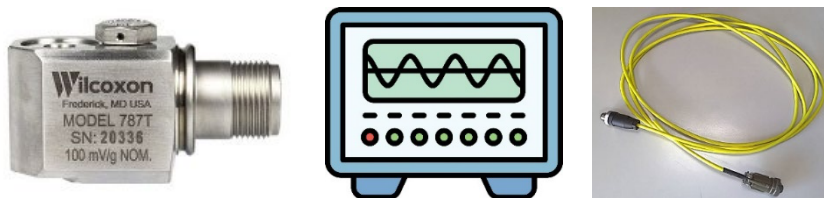
### Device Connection

Assemble the following:

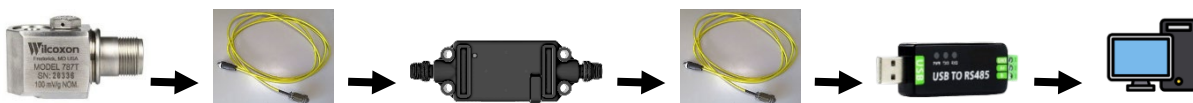
- an ILA device
- Windows OS PC with USB connection
- an M12 to USB / RS485 adapter cable with power cord



The ILA device can operate regardless of whether there is a signal input. However, if a signal source is available, such as a sensor or a function generator, connect it to the ILA device.



Connect all equipment with the cables per the below diagram and instructions.

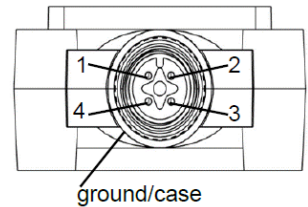


- Connect the sensor or signal generator to the ILA input side (optional)
- Connect the M12 to USB/RS485 cable to ILA output side and PC
- Connect DC power to the M12 to USB/RS485 cable

Pinout connections for the ILA device are as follows.

<b>Sensor input connections</b>	
<i>Connection</i>	<i>Function</i>
pin 1	accelerometer power/signal
pin 2	accelerometer and temp sensor common
pin 3	temp sensor signal (if applicable)
pin 4	NC
ground/case	shield

<b>Modbus RTU output connections</b>	
<i>Connection</i>	<i>Function</i>
pin 1	power, 12-24V
pin 2	common
pin 3	A
pin 4	B
ground/case	shield



## Software Setup

### ILA210 Utility

Download, install, and launch the ILA Utility application to configure and read data from the ILA device.

- Click [here](#) to access the latest ILA Utility.
- Download the utility file.
- Execute the .EXE file and follow the steps to finish the installation.
- Launch the ILA Utility application on PC to configure and read data from the ILA device.

### Modbus RTU

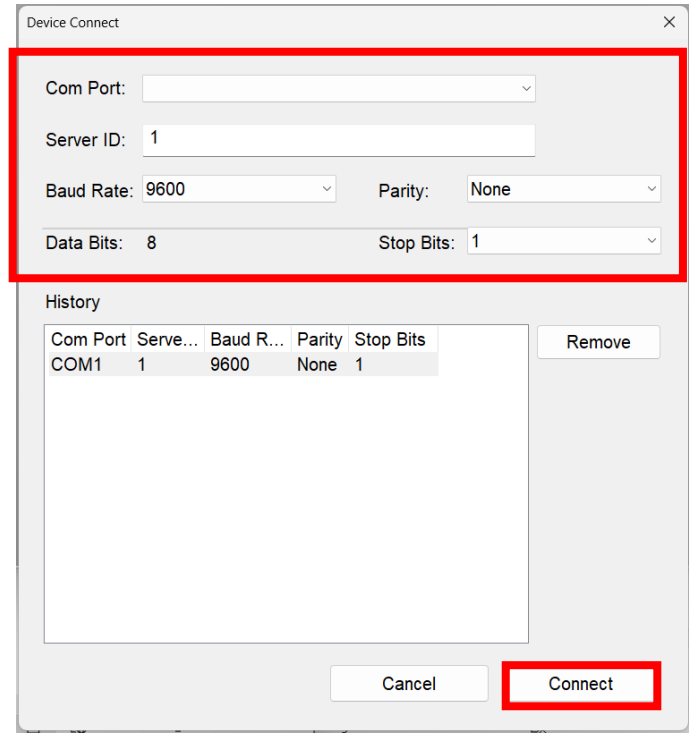
Instead of using the ILA Utility application, the Modbus RTU functionality is fully implemented, allowing direct use of Modbus RTU for configuration and data reading. Please refer to the [Modbus Instructions](#) in this document.

## Device Initial Setup and Editing Settings

### Device Connect RS-485 Settings:

Use the default RS485 settings to connect to a new device.

- ◆ Com Port: [dropdown] [ **USB Serial Port (COM XX)** ] select the serial port that the adapter cable is connected to
- ◆ Server ID: 1
- ◆ Baud rate: **9600**
- ◆ Parity: **None**
- ◆ Data bits: **8**
- ◆ Stop bits: **1**
- ◆ Click Connect



Device Connect

Com Port: [dropdown]  
 Server ID: 1  
 Baud Rate: 9600 Parity: None  
 Data Bits: 8 Stop Bits: 1

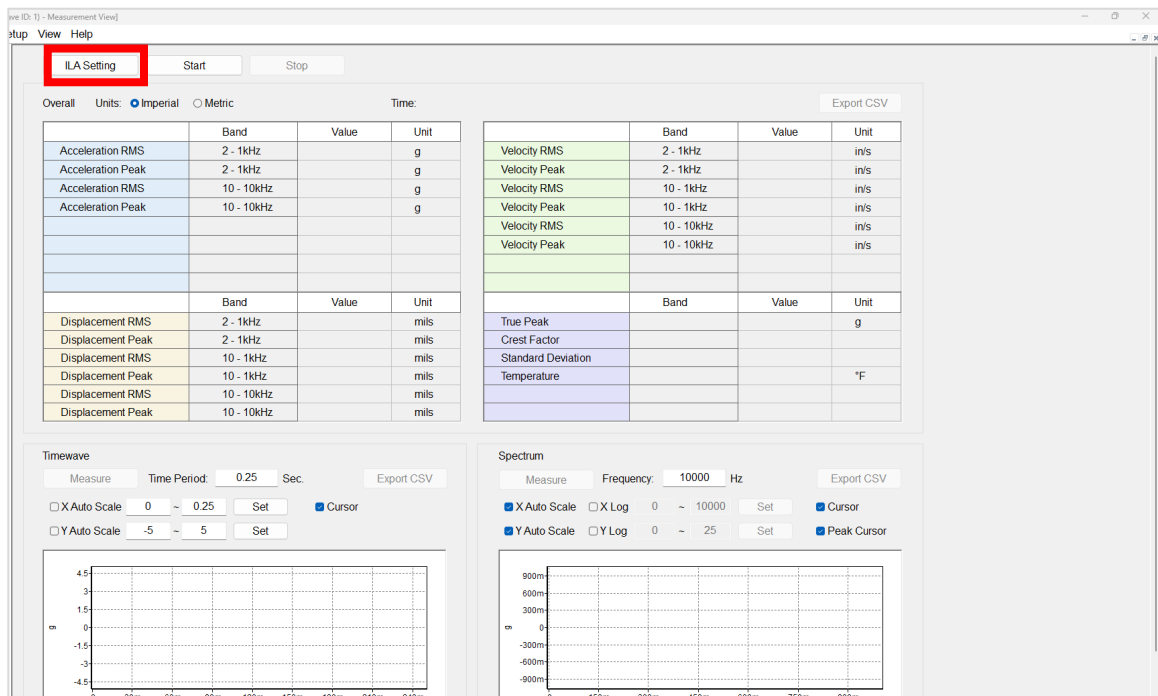
History

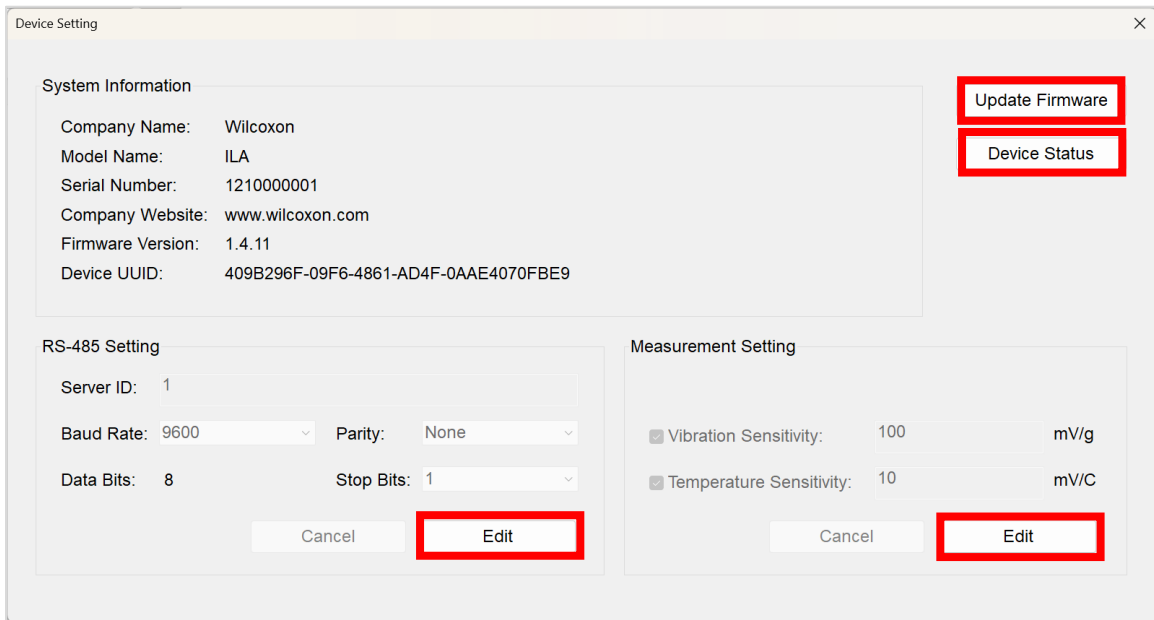
Com Port	Serve...	Baud R...	Parity	Stop Bits	Remove
COM1	1	9600	None	1	

Cancel Connect

### ILA Settings:

After the device is connected, the Measurement View will pop up. Click ILA Setting.

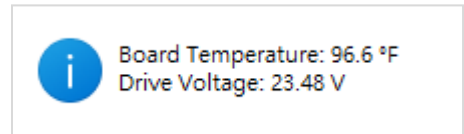




**Device Setting:**

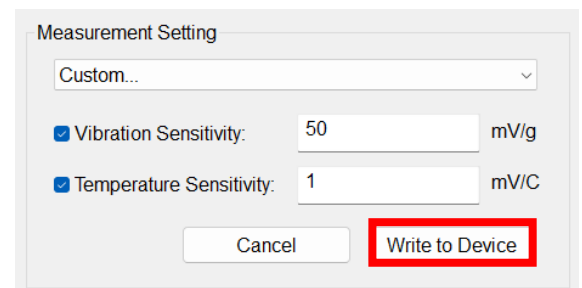
From this dialog box, you can:

- edit the Measurement Settings
- edit the RS485 Settings
- update the firmware
- read the Device Status, “Driver Voltage: IEPE Bias / DC 5V, depend on ILA models”



**Measurement Setting:**

- ◆ Click Edit
- ◆ Vibration: [check box] check to enable
- ◆ Vibration Sensitivity: [ 1-1000 ] mV/g
- ◆ Temperature: [check box] check to enable, if the ILA is being used with a dual output sensor
- ◆ Temperature sensitivity: [ 0-50 ] mV/°C (Temperature voltage)
- ◆ Or select a preset sensor model from the Custom dropdown menu
- ◆ Click Write to Device



RS-485 Setting:

- ◆ Click Edit
- ◆ Server ID: [ 1-247 ]
- ◆ Baud rage: [dropdown] [ 1200 ] [ 2400 ] [ 4800 ] [ 9600 ] [ 19200 ] [ 38400 ] [ 57600 ] [ 115200 ] [ 230400 ] [ 460800 ] [ 921600 ]
- ◆ Parity: [dropdown] [ None ] [ Even ] [ Odd ]
- ◆ Stop Bits: [dropdown] [ 1 ] [ 2 ]
- ◆ Click Write to Device

LED Indicator Definitions:

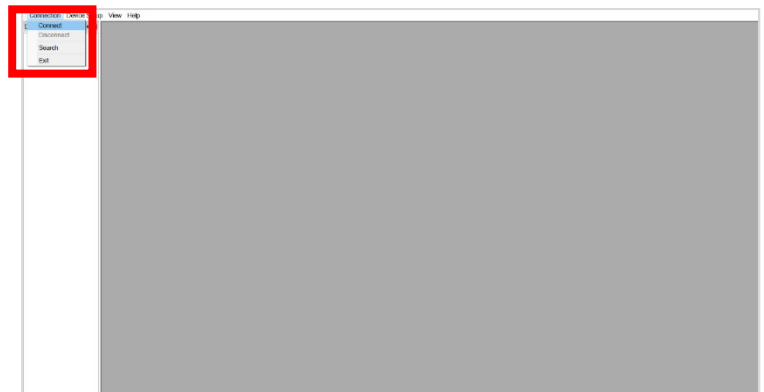
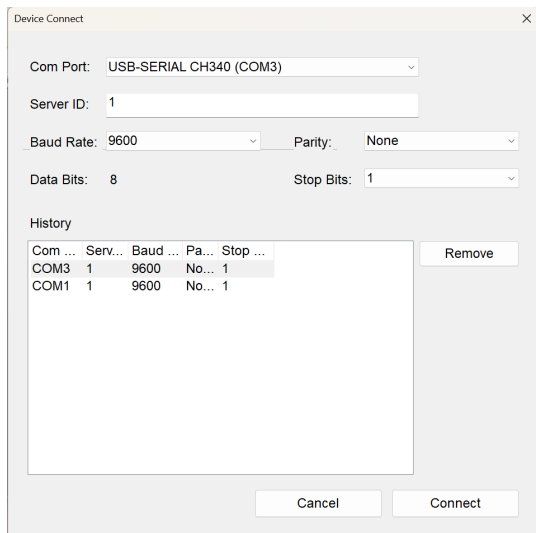
- White light: device initializing
- Blue light: device operating
- Green light: data transmitting
- Red light: firmware updating

## ILA Utility Instructions

### Connecting to a Device

When the ILA Utility program starts, the connect dialog will pop up.

The connect dialog can also be opened from the connection section of the menu.



Select a device in the History and click the Connect button or double click a device in the History list. A message will indicate if the connection was successful.



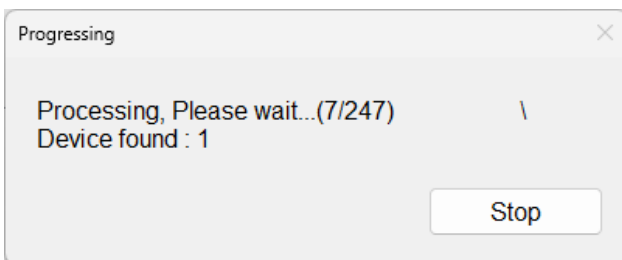
## Searching for a Device

The Search dialog can be opened from the Connection section of the menu.

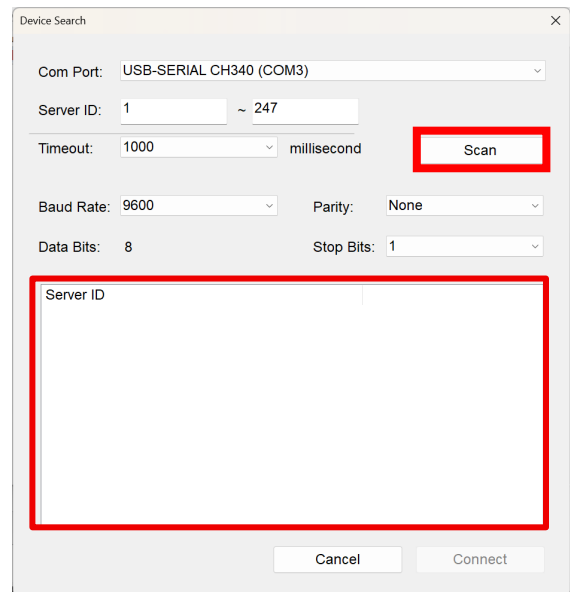
Select the Com Port in the drop down (USB COMXX.) Enter the RS485 settings and the ID range, then click the Scan button to search for the device on the selected Com Port.

You need to have the correct baud rate or it won't show. Record baud rate. If you cannot find it, reset the device using the button.

While scanning the Com Port, a message will show the progress and the number of devices found.

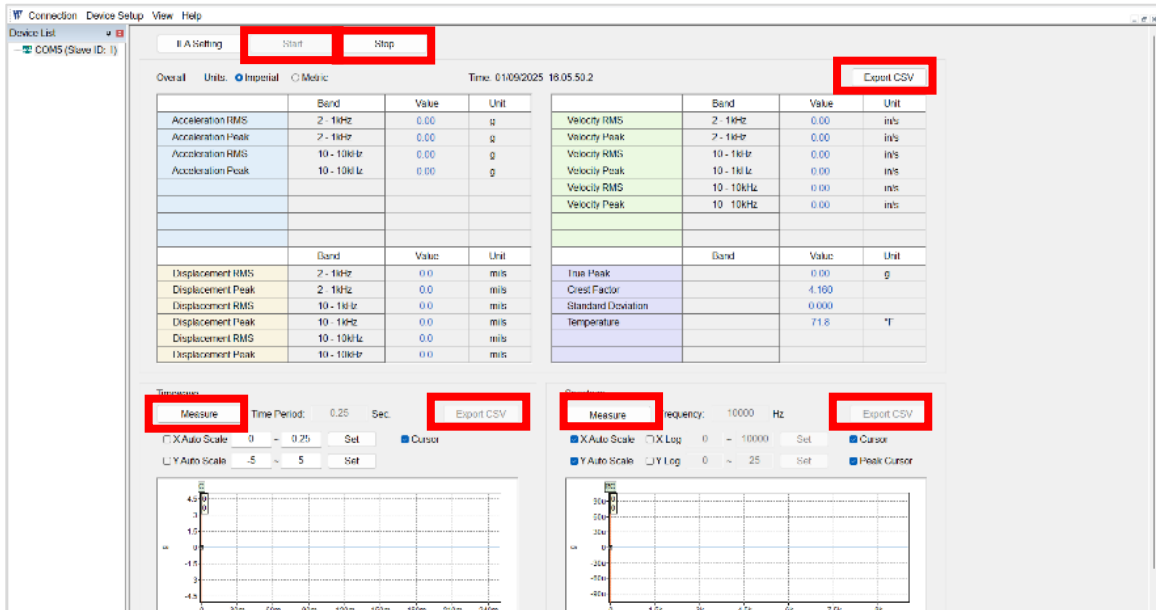


If the device has been found, you can click the Stop button to stop scanning. The device(s) will be shown in the list with its ID.



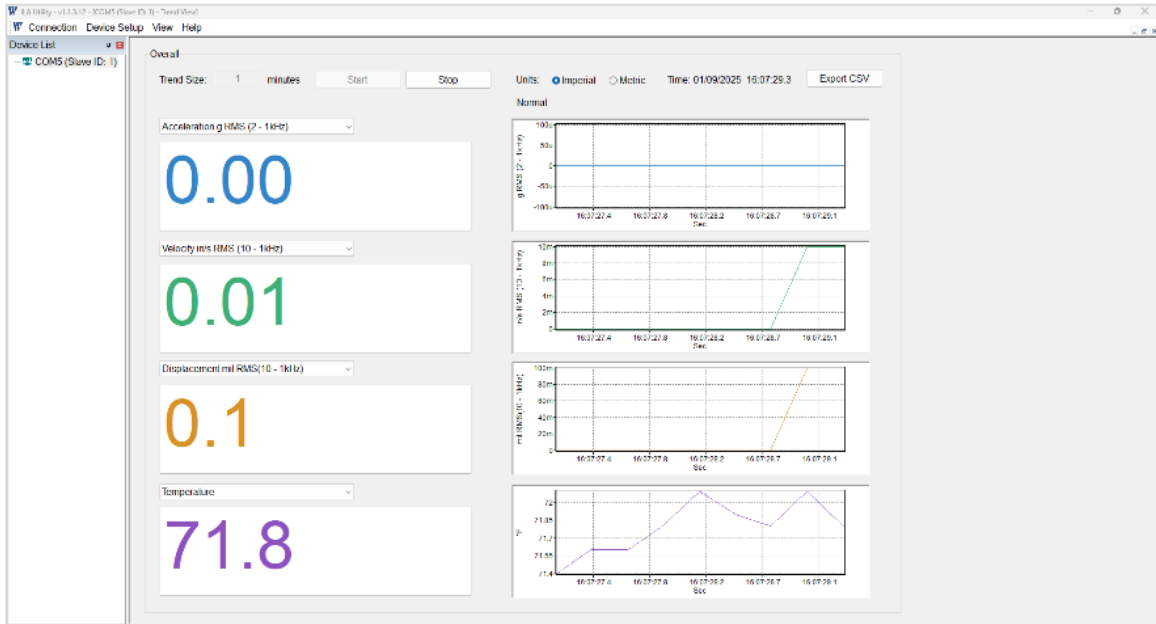
## Viewing the Data

The Measurement View and Trend View can be opened from the View section of the menu.



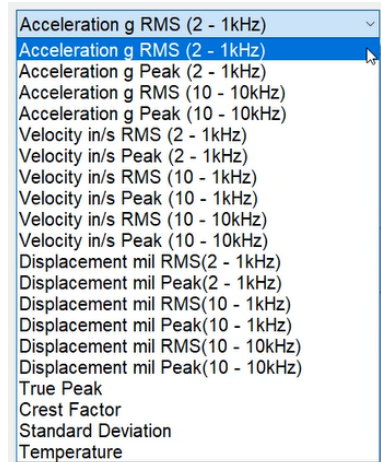
### Measurement View

- Click the Start button to start collecting Overall data from the device.
- Click Stop to end the data collection period.
- Click Measure to view Timewave or Spectrum.
  - The Timewave and Spectrum each have a unique Measure button.
- Click the Export CSV button to export the data to a CSV file.
  - The Overall, Timewave, and Spectrum each have a unique Export CSV button.



### Trend View

- Select the overall metric to trend from the dropdown.
- Click the Start button to start collecting data from the device.
- Click Stop to end the trend period.
- Click the Export CSV button to export the data to a CSV file.



## Modbus Instructions

### Modbus RTU Format

Server ID	Function Code	Data	CRC
1 byte	1 byte	0 up to 252 bytes	2 bytes

### Modbus Functions

code (Hex)	Modbus definition	Feature
0x03	Read Holding Table	Read Device Setting
0x04	Read Input Table	Read Device Data
0x06	Write Single Register	Write Single Setting
0x10	Write Multiple Registers	Write Multiple Setting
0x2B	Read Device ID	Read System Information
0x80	Exception Responses	Error Occurred

## 0X03(03) Function Code

Use 0x03 function to read settings from ILA devices.

### ILA Settings Table

Address (Hex)	Data Format	Default (Hex)	Description
0x0000	unsigned int16	0x0009	Baud rate: 0x0006:1200 bps 0x0007:2400 bps 0x0008:4800 bps 0x0009:9600 bps 0x000A:19200 bps 0x000B:38400 bps 0x000C:57600 bps 0x000D:115200 bps 0x000E:230400 bps 0x000F: 460800 bps 0x0010: 921600 bps
0x0001	unsigned int16	0x0008	Data bits: 0x0008: 8 bits
0x0002	unsigned int16	0x0000	Data parity: 0x0000: None 0x0001: Even 0x0002: Odd
0x0003	unsigned int16	0x0001	Stop bits: 0x0001: 1 bit 0x0002: 2 bits
0x0004	unsigned int16	0x0001	Modbus Server ID: 0x0001~0x00F7: 1 ~ 247
0x0005	unsigned int16	0x0001	Acceleration Item: 0x0000: Disable 0x0001: Enable

Address (Hex)	Data Format	Default (Hex)	Description
0x0006	unsigned int16	0x03E8	Acceleration sensitivity: Unit: 0.1 mV/g
0x0007	unsigned int16	0x0001	Temperature Item: 0x0000: Disable 0x0001: Enable
0x0008	signed int16	0x0064	Acceleration sensitivity Unit: 0.1 mV/C
0x0009	signed int16	0x0000	Temperature voltage at 0°C: Unit: 0.001 mV
...			Reserved
0x0200	unsigned int16	0xEEEE	Table Data Ready for Read  When register value=0x0000, use the function code 0x04 to get corresponding data.

### 0x03 Function Usage

#### Request

Function code	1 Byte	0x03
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	1 to 125 (0x7D)

#### Response

Function code	1 Byte	0x03
Byte count	1 Byte	2 x N*
Register value	N* x 2 Bytes	

\*N = Quantity of Registers

#### Error

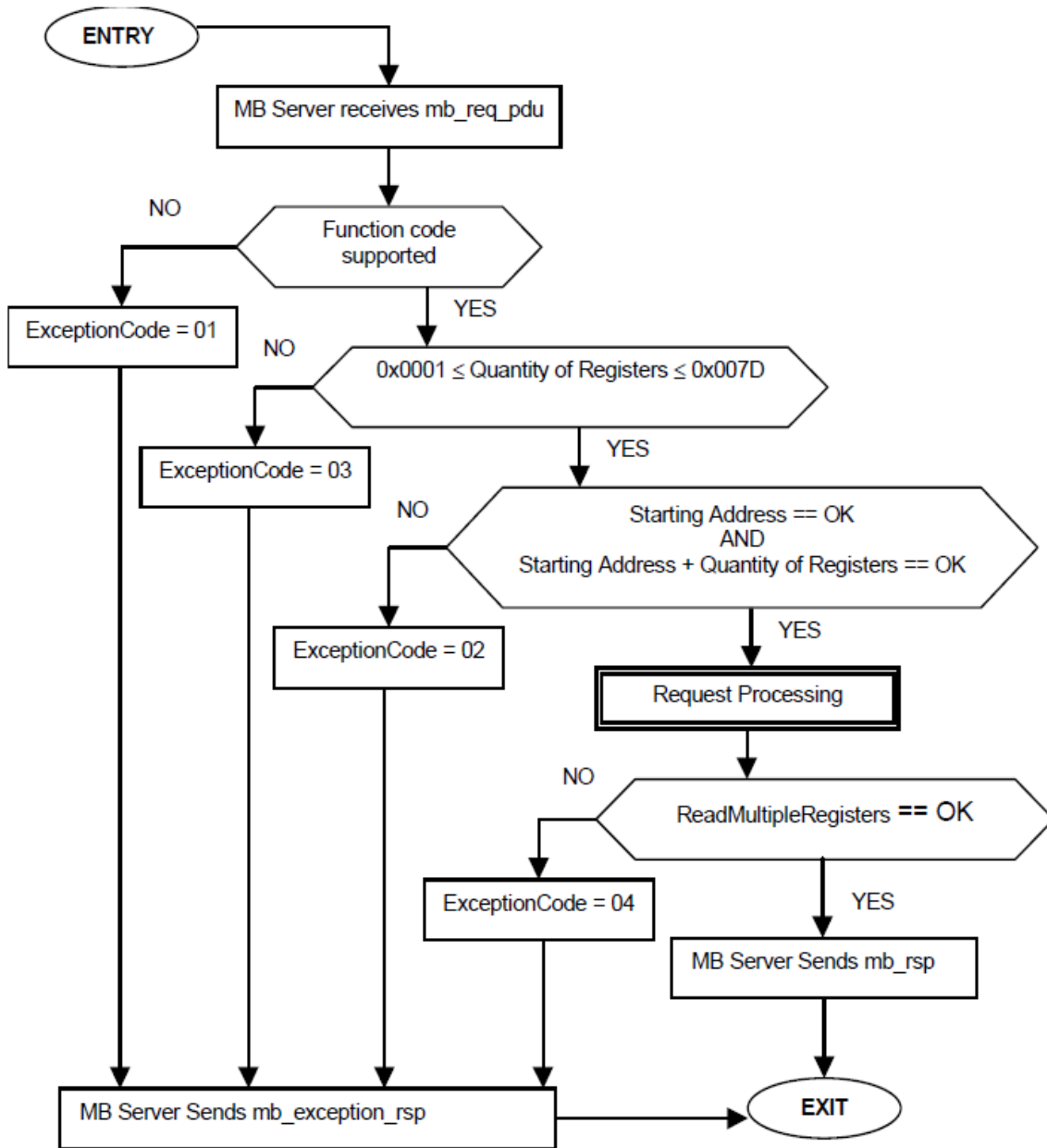
Error code	1 Byte	0x83
Exception code	1 Byte	01 or 02 or 03 or 04

Here is an example of a request to read registers 108 – 110:

Request		Response	
Field Name	(Hex)	Field Name	(Hex)
Function	03	Function	03
Starting Address Hi	00	Byte Count	06
Starting Address Lo	6B	Register value Hi (108)	02
No. of Registers Hi	00	Register value Lo (108)	2B
No. of Registers Lo	03	Register value Hi (109)	00
		Register value Lo (109)	00
		Register value Hi (110)	00
		Register value Lo (110)	64

The contents of register 108 are shown as the two byte values of 02 2B hex, or 555 decimal. The contents of registers 109–110 are 00 00 and 00 64 hex, or 0 and 100 decimal, respectively.

0x03 Function State Diagram



### 0x03 Function Example

Read RS485 settings from ILA device with Server ID = 1

#### **Request(01 03 00 00 00 05 85 C9):**

*Server ID = 0x01*

*Function code = 0x03*

*Starting address = 0x0000*

*Quantity of Registers = 0x0005*

*CRC = 0x85C9*

#### **Response(01 03 0A 00 09 00 08 00 00 00 01 00 01 22 3E):**

*Server ID = 0x01*

*Function code = 0x03*

*Byte count = 0x0A (2\*0x0005)*

*Address 0x0000(baud rate) = 0x0009(9600)*

*Address 0x0001(data bits) = 0x0008(8)*

*Address 0x0002(data parity) = 0x0000(None)*

*Address 0x0003(stop bits) = 0x0001(1)*

*Address 0x0004(Server id) = 0x0001(1)*

*CRC = 0x223E*

## 0X04(04) Function Code

Use 0x04 function to read data from ILA devices.

### ILA Overall Data Table

Measurement Value (Metric)	Address (Hex)	Data Format	Unit	Description
Acc (2Hz-1KHz)	0x0000	unsigned int16	0.01g	Acceleration (2Hz~1kHz) RMS
	...			Reserved
	0x0004	unsigned int16	0.01g	Acceleration (2Hz~1kHz) Peak
	...			Reserved
	0x0008	unsigned int16	0.1 m/s <sup>2</sup>	Acceleration (2Hz~1kHz) RMS
	...			Reserved
	0x000C	unsigned int16	0.1 m/s <sup>2</sup>	Acceleration (2Hz~1kHz) Peak
	...			Reserved
Acc (10Hz-10KHz)	0x0010	unsigned int16	0.01g	Acceleration (10Hz~10kHz) RMS
	...			Reserved
	0x0014	unsigned int16	0.01g	Acceleration (10Hz~10kHz) Peak
	...			Reserved
	0x0018	unsigned int16	0.1 m/s <sup>2</sup>	Acceleration (10Hz~10kHz) RMS
	...			Reserved
	0x001C	unsigned int16	0.1 m/s <sup>2</sup>	Acceleration (10Hz~10kHz) Peak
	...			Reserved
Vel (10Hz-1KHz)	0x0020	unsigned int16	0.1 mm/s	Velocity (10Hz~1kHz) RMS
	...			Reserved
	0x0024	unsigned int16	0.1 mm/s	Velocity (10Hz~1kHz) Peak
	...			Reserved
Vel (10Hz-10KHz)	0x0028	unsigned int16	0.1 mm/s	Velocity (10Hz~10kHz) RMS
	...			Reserved
	0x002C	unsigned int16	0.1 mm/s	Velocity (10Hz~10kHz) Peak
	...			Reserved

Measurement Value (Metric)	Address (Hex)	Data Format	Unit	Description
Displ (2Hz-1KHz)	0x0030	unsigned int16	um	Displacement (2Hz~1kHz) RMS
	...			Reserved
	0x0034	unsigned int16	um	Displacement (2Hz~1kHz) Peak
	...			Reserved
Displ (10Hz-1KHz)	0x0038	unsigned int16	um	Displacement (10Hz~1kHz) RMS
	...			Reserved
	0x003C	unsigned int16	um	Displacement (10Hz~1kHz) Peak
	...			Reserved
True Peak	0x0040	unsigned int16	0.01g	True Peak
	...			Reserved
Crest Factor	0x0044	unsigned int16	0.01	Crest Factor
	...			Reserved
Std. Dev.	0x0048	unsigned int16	0.01	Standard Deviation
	...			Reserved
	0x004C			Reserved
	...			Reserved
Temp	0x0050	unsigned int16	0.1°C	Temperature
	...			Reserved
	0x0054			Reserved
	...			Reserved
	0x0058			Reserved
	...			Reserved
	0x005C			Reserved
	...			Reserved
Drive Volt	0x0060	unsigned int16	0.01V	IEPE Mode: IEPE Volt DC Mode: 5V Volt
	...			Reserved
Vel (2Hz-1KHz)	0x0064	unsigned int16	0.1 mm/s	Velocity (2Hz~1kHz) RMS
	...			Reserved
	0x0068	unsigned int16	0.1 mm/s	Velocity (2Hz~1kHz) Peak
	...			Reserved

Measurement Value (Metric)	Address (Hex)	Data Format	Unit	Description
Displ (10Hz- 1KHz)	0x006C	unsigned int16	um	Displacement (10Hz~1kHz) RMS
	...			Reserved
	0x0070	unsigned int16	um	Displacement (10Hz~1kHz) Peak
	...			Reserved
	0x0074			Reserved
	...			Reserved
	0x0078			Reserved
	...			Reserved
Board Temp	0x007C	unsigned int16	0.1°C	Board Temperature
	...			Reserved
...				Reserved
	0x00FF			Reserved

Measurement Value (Imperial)	Address (Hex)	Data Format	Unit	Description
Acc (2Hz- 1KHz)	0x0100	unsigned int16	0.01g	Acceleration (2Hz~1kHz) RMS
	...			Reserved
	0x0104	unsigned int16	0.01g	Acceleration (2Hz~1kHz) Peak
	...			Reserved
	0x0108			Reserved
	...			Reserved
	0x010C			Reserved
	...			Reserved
Acc (10Hz- 10KHz)	0x0110	unsigned int16	0.01g	Acceleration (10Hz~10kHz) RMS
	...			Reserved
	0x0114	unsigned int16	0.01g	Acceleration (10Hz~10kHz) Peak
	...			Reserved

Measurement Value (Imperial)	Address (Hex)	Data Format	Unit	Description
	0x0118			Reserved
	...			Reserved
	0x011C			Reserved
	...			Reserved
Vel (10Hz-1KHz)	0x0120	unsigned int16	0.01 in/s	Velocity (10Hz~1kHz) RMS
	...			Reserved
	0x0124	unsigned int16	0.01 in/s	Velocity (10Hz~1kHz) Peak
	...			Reserved
Vel (10Hz-10KHz)	0x0128	unsigned int16	0.01 in/s	Velocity (10Hz~10kHz) RMS
	...			Reserved
	0x012C	unsigned int16	0.01 in/s	Velocity (10Hz~10kHz) Peak
	...			Reserved
Displ (2Hz-1KHz)	0x0130	unsigned int16	0.1 mil	Displacement (2Hz~1kHz) RMS
	...			Reserved
	0x0134	unsigned int16	0.1 mil	Displacement (2Hz~1kHz) Peak
	...			Reserved
Displ (10Hz-1KHz)	0x0138	unsigned int16	0.1 mil	Displacement (10Hz~1kHz) RMS
	...			Reserved
	0x013C	unsigned int16	0.1 mil	Displacement (10Hz~1kHz) Peak
	...			Reserved
True Peak	0x0140	unsigned int16	0.01g	True Peak
	...			Reserved
Crest Factor	0x0144	unsigned int16	0.01	Crest Factor
	...			Reserved
Std. Dev.	0x0148	unsigned int16	0.01	Standard Deviation
	...			Reserved
	0x014C			Reserved
	...			Reserved

Measurement Value (Imperial)	Address (Hex)	Data Format	Unit	Description
Temp	0x0150	unsigned int16	0.1°F	Temperature
	...			Reserved
	0x0154			Reserved
	...			Reserved
	0x0158			Reserved
	...			Reserved
	0x015C			Reserved
	...			Reserved
Drive Volt	0x0160	unsigned int16	0.01V	IEPE Mode: IEPE Volt DC Mode: 5V Volt
	...			Reserved
Vel (2Hz-1KHz)	0x0164	unsigned int16	0.01 in/s	Velocity (2Hz~1kHz) RMS
	...			Reserved
	0x0168	unsigned int16	0.01 in/s	Velocity (2Hz~1kHz) Peak
	...			Reserved
Displ (10Hz-1KHz)	0x016C	unsigned int16	0.1 mil	Displacement (10Hz~1kHz) RMS
	...			Reserved
	0x0170	unsigned int16	0.1 mil	Displacement (10Hz~1kHz) Peak
	...			Reserved
	0x0174			Reserved
	...			Reserved
	0x0178			Reserved
	...			Reserved
Board Temp	0x017C	unsigned int16	0.1°F	Board Temperature
	...			Reserved
...			Reserved	
	0x01FF			Reserved

ILA Dynamic Data Table

	Size	Span	Interval
Spectrum	6401	0 ~ 10000 Hz	1.5625Hz
Timewave	25600	0 ~ 1 sec	0.000039 sec

Spectrum			
Address (Hex)	Data Format	Unit	Description
0x0600	unsigned int16	0.001g	Spectrum data 0Hz (delta F Index 0)
0x0601	unsigned int16	0.001g	Spectrum data (delta F Index 1)
0x0602	unsigned int16	0.001g	Spectrum data (delta F Index 2)
...			...
0x1F00	unsigned int16	0.001g	Spectrum data (delta F Index 6400)
Timewave			
Address (Hex)	Data Format	Unit	Description
0x1F01	signed int16	0.001g	Timewave data (delta T Index 0)
0x1F02	signed int16	0.001g	Timewave data (delta T Index 1)
0x1F03	signed int16	0.001g	Timewave data (delta T Index 2)
...			...
0x8300	signed int16	0.001g	Timewave data (delta T Index 25599)

### 0x04 Function Usage

#### Request

Function code	1 Byte	0x04
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Input Registers	2 Bytes	0x0001 to 0x007D

#### Response

Function code	1 Byte	0x04
Byte count	1 Byte	2 x N*
Input Registers	N* x 2 Bytes	

\*N = Quantity of Input Registers

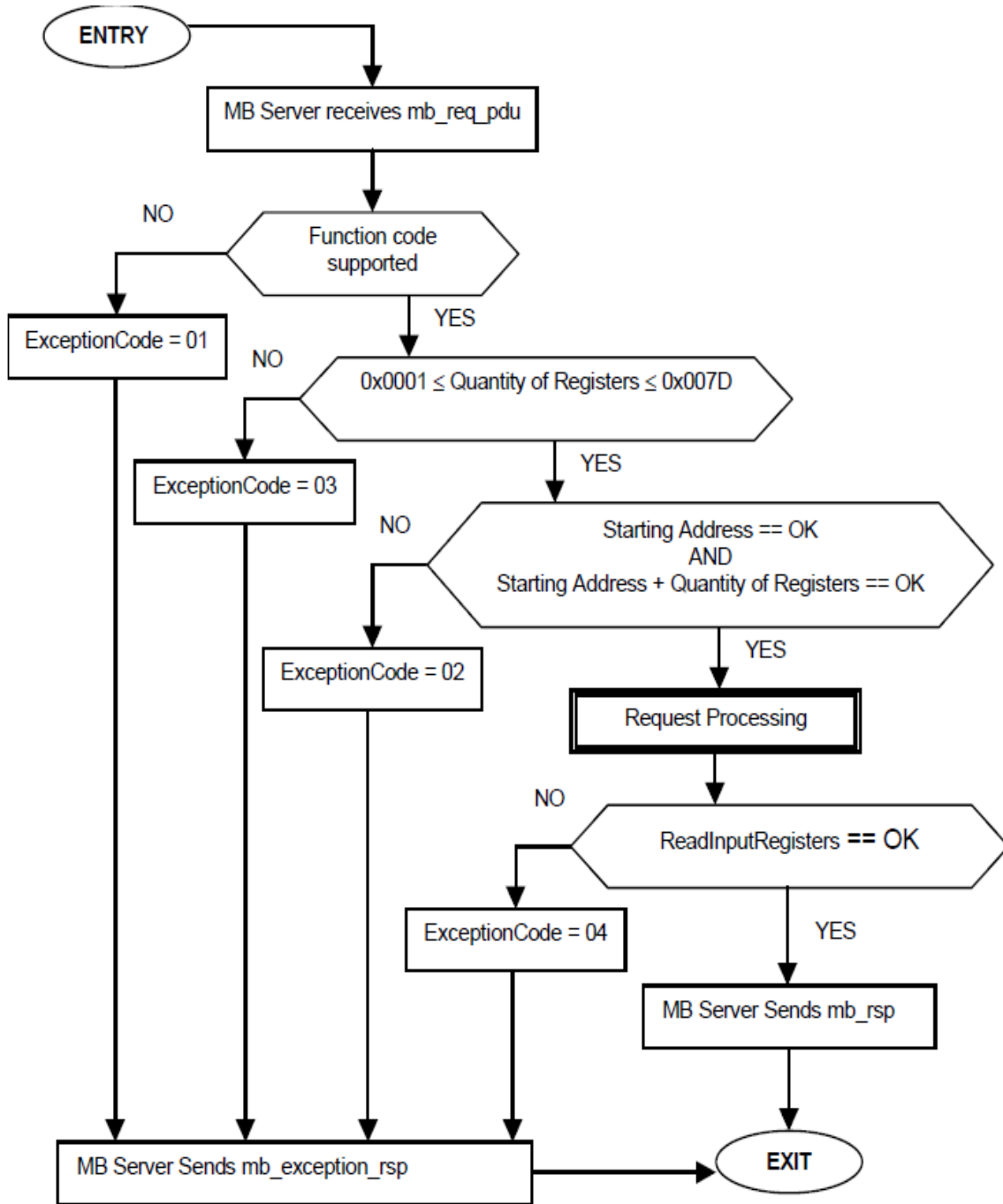
#### Error

Error code	1 Byte	0x84
Exception code	1 Byte	01 or 02 or 03 or 04

Here is an example of a request to read input register 9:

Request		Response	
Field Name	(Hex)	Field Name	(Hex)
Function	04	Function	04
Starting Address Hi	00	Byte Count	02
Starting Address Lo	08	Input Reg. 9 Hi	00
Quantity of Input Reg. Hi	00	Input Reg. 9 Lo	0A
Quantity of Input Reg. Lo	01		

0x04 Function State Diagram



### 0x04 Function Example

Read Imperial Acc Data from ILA device with Server ID = 1

**Request (01 04 01 00 00 18 + CRC):**

*Server ID = 0x01*

*Function code = 0x04*

*Starting address = 0x0100*

*Quantity of Registers = 0x0018*

*CRC*

**Response (01 04 30 00 05 00 00 ... 00 04 ... + CRC):**

*Server ID = 0x01*

*Function code = 0x04*

*Byte count = 0x30 (2\*0x0018)*

*Address 0x0100(acc 2~1kHz RMS) = 0x0005(0.05g)*

*Address 0x0101(reserved) = 0x0000*

...

*Address 0x0110(acc 10~10kHz RMS) = 0x0004(0.05g)*

...

*CRC*

## 0X06(06) Function Code

Use 0x06 function to write a setting to ILA devices.

### ILA Settings Table

Address (Hex)	Data Format	Default (Hex)	Description
0x0000	unsigned int16	0x0009	Baud rate: 0x0006:1200 bps 0x0007:2400 bps 0x0008:4800 bps 0x0009:9600 bps 0x000A:19200 bps 0x000B:38400 bps 0x000C:57600 bps 0x000D:115200 bps 0x000E:230400 bps 0x000F: 460800 bps 0x0010: 921600 bps
0x0001	unsigned int16	0x0008	Data bits: 0x0008: 8 bits
0x0002	unsigned int16	0x0000	Data parity: 0x0000: None 0x0001: Even 0x0002: Odd
0x0003	unsigned int16	0x0001	Stop bits: 0x0001: 1 bit 0x0002: 2 bits
0x0004	unsigned int16	0x0001	Modbus Server ID: 0x0001~0x00F7: 1 ~ 247
0x0005	unsigned int16	0x0001	Acceleration Item: 0x0000: Disable 0x0001: Enable

Address (Hex)	Data Format	Default (Hex)	Description
0x0006	unsigned int16	0x03E8	Acceleration sensitivity: Unit: 0.1 mV/g
0x0007	unsigned int16	0x0001	Temperature Item: 0x0000: Disable 0x0001: Enable
0x0008	Signed int16	0x0064	Acceleration sensitivity Unit: 0.1 mV/C
0x0009	Signed int16	0x0000	Temperature voltage at 0°C: Unit: 0.001 mV
...			Reserved
0x0200	Q16.0	0xEEEE	Table Data Ready for Read 0xEEEE: Empty data at initial 0xFFFF: Start query data 0x0000: Data ready

### 0x06 Function Usage

#### Request

Function code	1 Byte	0x06
Register Address	2 Bytes	0x0000 to 0xFFFF
Register Value	2 Bytes	0x0000 to 0xFFFF

#### Response

Function code	1 Byte	0x06
Register Address	2 Bytes	0x0000 to 0xFFFF
Register Value	2 Bytes	0x0000 to 0xFFFF

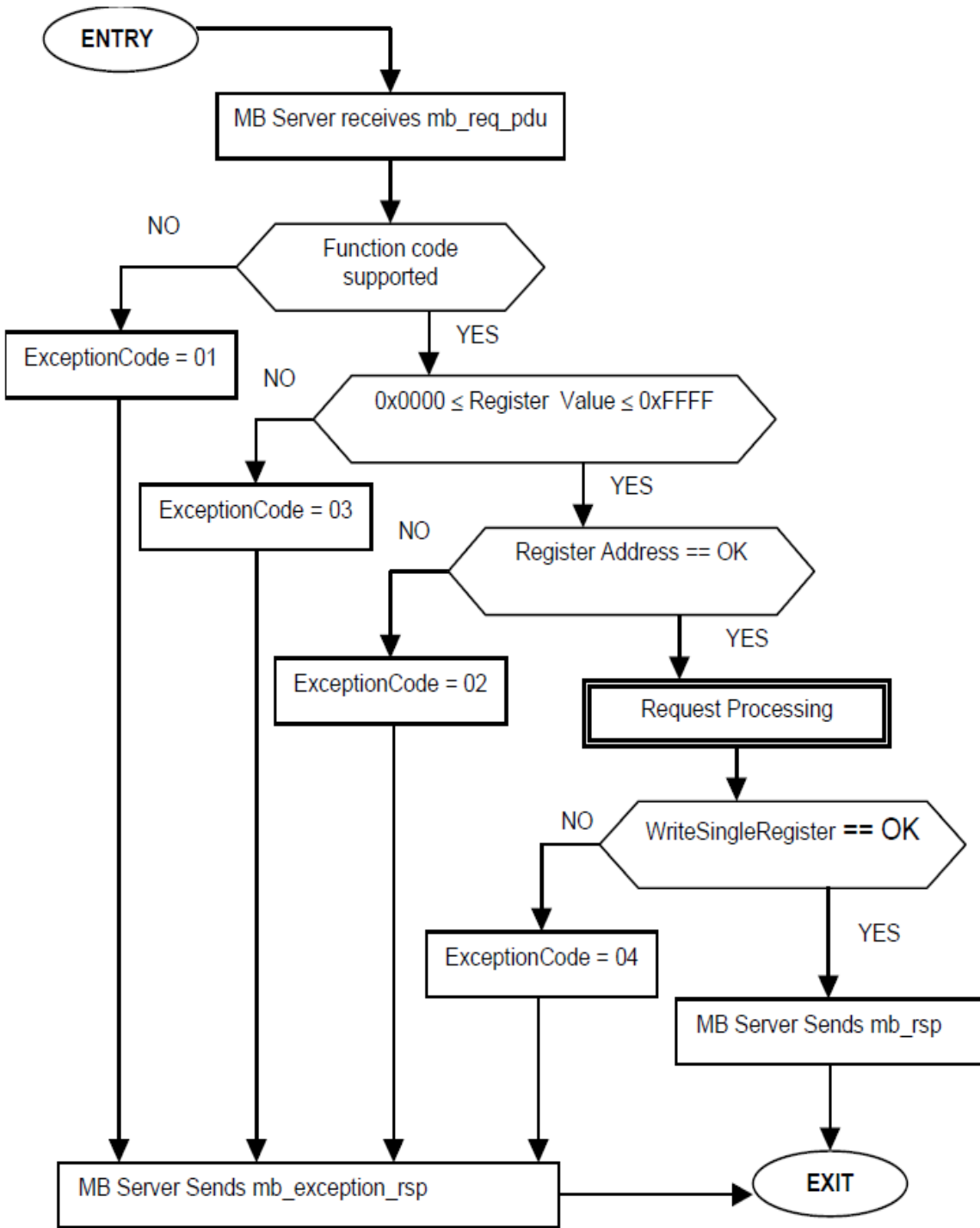
#### Error

Error code	1 Byte	0x86
Exception code	1 Byte	01 or 02 or 03 or 04

Here is an example of a request to write register 2 to 00 03 hex:

Request		Response	
Field Name	(Hex)	Field Name	(Hex)
Function	06	Function	06
Register Address Hi	00	Register Address Hi	00
Register Address Lo	01	Register Address Lo	01
Register Value Hi	00	Register Value Hi	00
Register Value Lo	03	Register Value Lo	03

0x06 Function State Diagram



### 0x06 Function Example

Read Timewave Data from ILA device with Server ID = 1

**Request start query data (01 06 02 00 FF FF + CRC):**

*Server ID = 0x01*

*Function code = 0x06*

*Register Address = 0x0200*

*Register Value = 0xFFFF*

*CRC*

**Response (01 06 02 00 FF FF + CRC):**

*Response same to confirm setting changed.*

**Request table data ready (01 03 02 00 00 01 + CRC):**

*Server ID = 0x01*

*Function code = 0x03*

*Address = 0x0200*

*Quantity of Registers = 0x0001*

*CRC*

**Response (01 03 02 FF FF + CRC / 01 03 02 00 00+ CRC):**

*Server ID = 0x01*

*Function code = 0x03*

*Byte count = 0x02*

*Address 0x0200(table data ready to read) = 0x0000 (ready) / 0xFFFF (not ready)*

**Request (01 04 1F 01 00 7D + CRC):**

*Server ID = 0x01*

*Function code = 0x04*

*Starting address = 0x1F01*

*Quantity of Registers = 0x007D (max length)*

*CRC*

**Response (01 04 FA ... + CRC):**

*Server ID = 0x01*

*Function code = 0x04*

*Byte count = 0xFA (2\*0x007D)*

*Address Timewave (index 0 ~ index249)*

*CRC*

## 0X10(16) Function Code

Use 0x10 function to write multiple settings to ILA devices.

### ILA Settings Table

Shares [ILA Setting Table](#) with function code 0x06.

### 0x10 Function Usage

#### Request

Function code	1 Byte	0x10
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	0x0001 to 0x007B
Byte Count	1 Byte	2 x N*
Registers Value	N* x 2 Bytes	value

\*N = Quantity of Registers

#### Response

Function code	1 Byte	0x10
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	1 to 123 (0x7B)

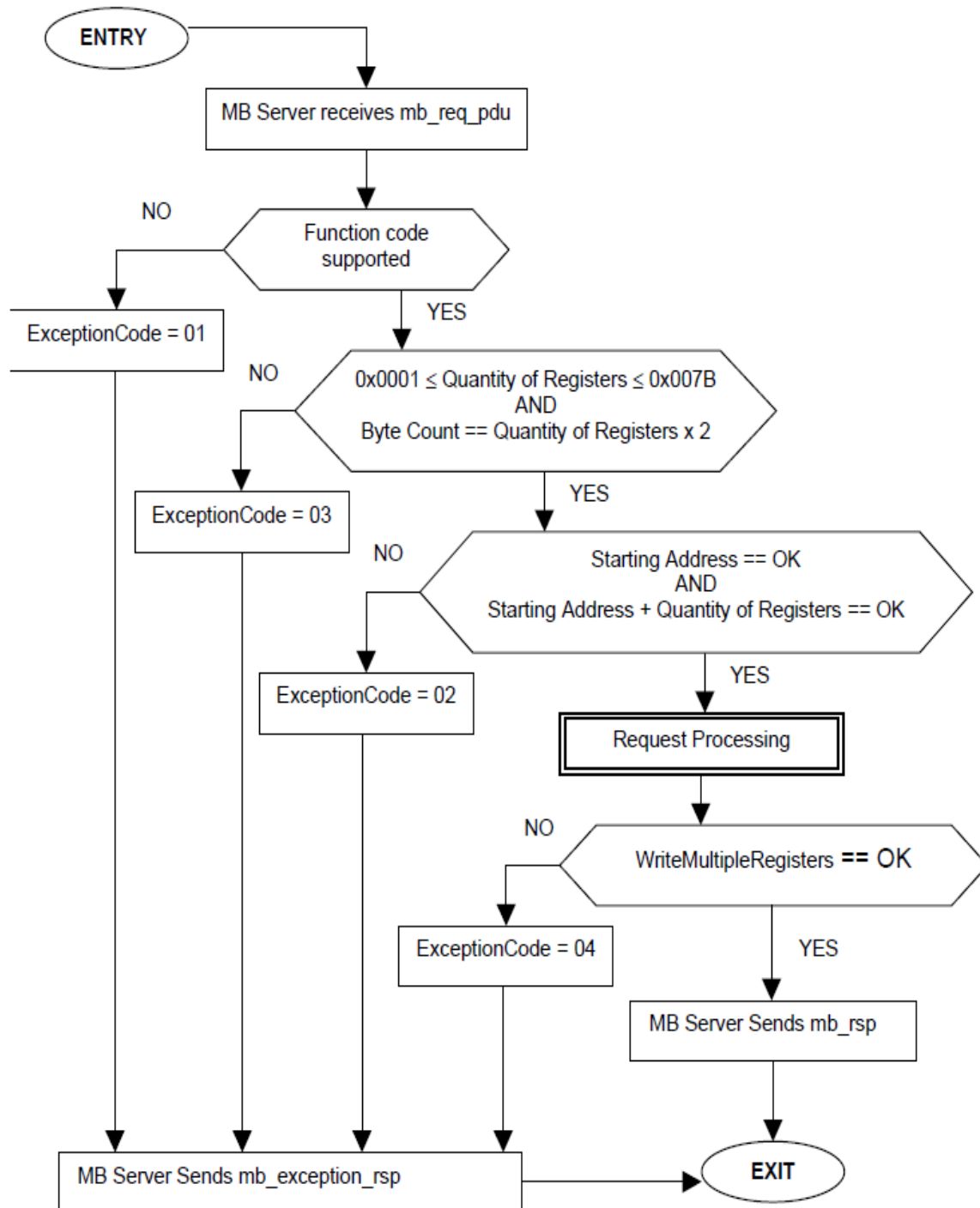
#### Error

Error code	1 Byte	0x90
Exception code	1 Byte	01 or 02 or 03 or 04

Here is an example of a request to write two registers starting at 2 to 00 0A and 01 02 hex:

Request		Response	
Field Name	(Hex)	Field Name	(Hex)
Function	10	Function	10
Starting Address Hi	00	Starting Address Hi	00
Starting Address Lo	01	Starting Address Lo	01
Quantity of Registers Hi	00	Quantity of Registers Hi	00
Quantity of Registers Lo	02	Quantity of Registers Lo	02
Byte Count	04		
Registers Value Hi	00		
Registers Value Lo	0A		
Registers Value Hi	01		
Registers Value Lo	02		

### 0x10 Function State Diagram



0x10 Function Example

Write RS485 settings to ILA device with Server ID = 1

**Request (01 10 00 00 00 05 0A 00 09 00 08 00 00 00 01 00 01+ CRC):**

*Server ID = 0x01*

*Function code = 0x10*

*Starting address = 0x0000*

*Quantity of Registers = 0x0005*

*Byte count = 0x0A ( 2 \* 0x0005 )*

*Address 0x0000 (baud rate) Value = 0x0009 (9600)*

*Address 0x0001 (data bits) Value = 0x0008 (8)*

*Address 0x0002 (data parity) Value = 0x0000 (None)*

*Address 0x0003 (stop bits) Value = 0x0001 (1)*

*Address 0x0004 (Server id) Value = 0x0001 (1)*

*CRC*

**Response (01 10 00 00 00 05 + CRC):**

*Server ID = 0x01*

*Function code = 0x10*

*Starting address = 0x0000*

*Quantity of Registers = 0x0005*

*CRC*

## 0X2B(43) Function Code

Use 0x2B function to read system information from ILA devices.

### Read Device Identification Table of MEI Type 0x0E (14)

Object Id	Object Name / Description	Type	M/O	category
0x00	VendorName	ASCII String	<b>Mandatory</b>	<b>Basic</b>
0x01	ProductCode	ASCII String	<b>Mandatory</b>	
0x02	MajorMinorRevision	ASCII String	<b>Mandatory</b>	
0x03	VendorUrl	ASCII String	Optional	<b>Regular</b>
0x04	ProductName	ASCII String	Optional	
0x05	ModelName	ASCII String	Optional	
0x06	UserApplicationName	ASCII String	Optional	
0x07	<i>Reserved</i>		Optional	
...				
0x7F				
0x80	<i>Private objects may be optionally defined.</i>	device dependant	Optional	<b>Extended</b>
...				
0xFF	<i>The range [0x80 - 0xFF] is Product dependant.</i>			

### ILA System Information Table

Object ID	Object Name	Definition	Value	Size
0x00	Vendor Name	Company Name	Wilcoxon	0-63
0x01	Product Code	Serial Number	25010000	8
0x02	Major / Minor / Revision	Firmware Version	1.2.13	0-63
0x03	Vendor URL	Company website	www.wilcoxon.com	0-63
0x04	ProductName	Device UUID	XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX	36
0x05	Model Name	Model Name	ILA	0-63

### Read Device ID Code Definition

Device ID code	Definition	Supported by ILA	Object ID
01	Request to get from Basic category	Yes	0x00-0x02
02	Request to get from Regular category	Yes	0x03-0x7F
03	Request to get from Extended category	No	0x80-0xFF
04	Request to get specific object	Yes	0x00-0xFF

### Conformity Level Definition

Level	Definition	Supported Device ID code
0x01	Identification of Basic category	01
0x02	Identification of Regular category	01-02
0x03	Identification of Extended category	01-03
0x81	Identification of Basic category with specific	01+04
0x82	Identification of Regular category	01-02+04
0x83	Identification of Extended category	01-04

### 0x2B Function Usage

#### Request

Function code	1 Byte	<b>0x2B</b>
MEI Type*	1 Byte	0x0E
Read Device ID code	1 Byte	01 / 02 / 03 / 04
Object Id	1 Byte	0x00 to 0xFF

\* MEI = MODBUS Encapsulated Interface

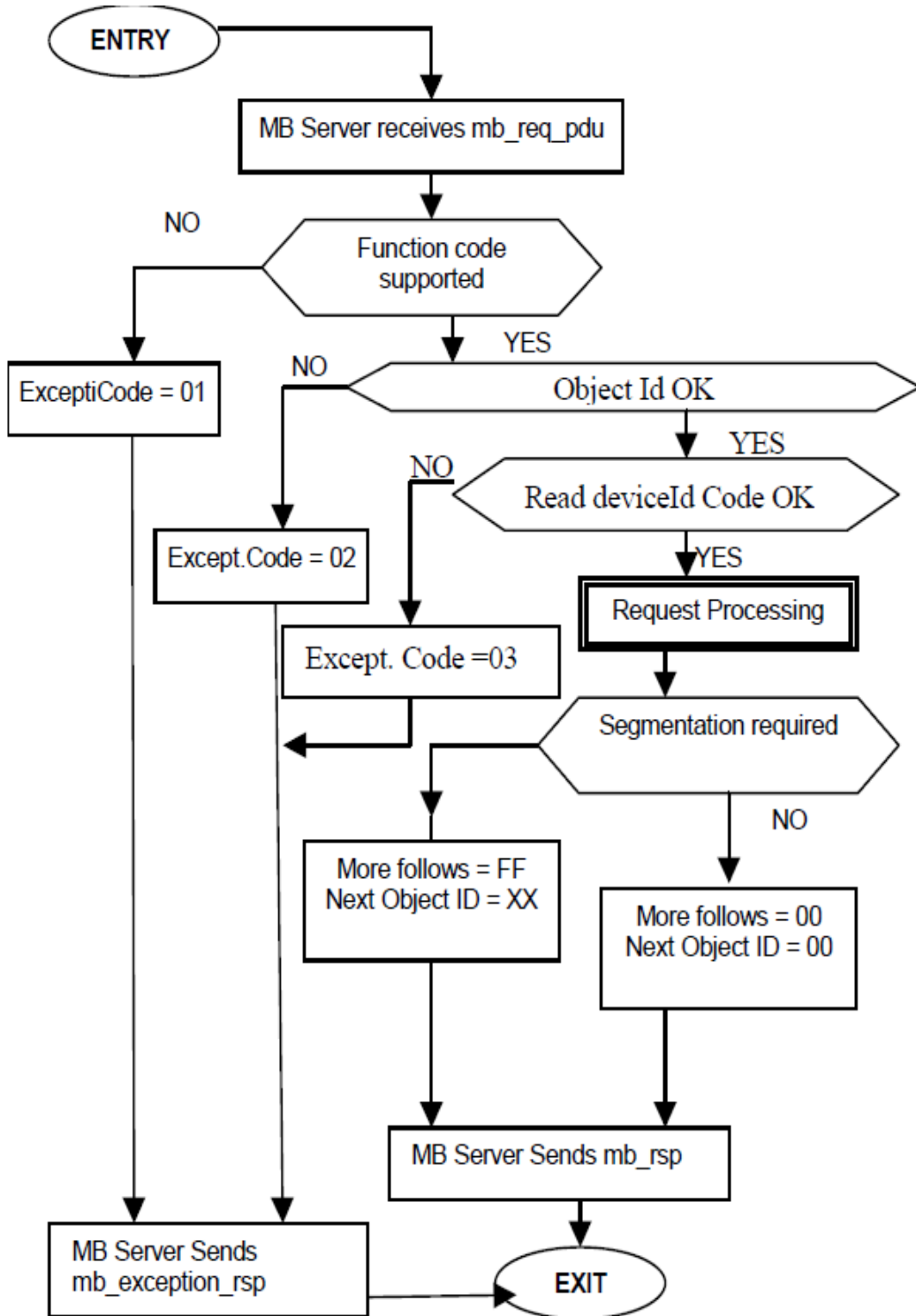
#### Response

Function code	1 Byte	<b>0x2B</b>
MEI Type	1 byte	0x0E
Read Device ID code	1 Byte	01 / 02 / 03 / 04
Conformity level	1 Byte	0x01 or 0x02 or 0x03 or 0x81 or 0x82 or 0x83
More Follows	1 Byte	00 / FF
Next Object Id	1 Byte	Object ID number
Number of objects	1 Byte	
List Of		
Object ID	1 Byte	
Object length	1 Byte	
Object Value	Object length	Depending on the object ID

#### Error

Function code	1 Byte	<b>0xAB :</b> <b>Fc 0x2B + 0x80</b>
Exception code	1 Byte	01 or 02 or 03 or 04

0x2B Function State Diagram



### 0x2B Function Example

Read Basic category from ILA device with Server ID = 1

#### **Request (01 2B 0E 01 00 + CRC):**

*Server ID = 0x01*

*Function code = 0x2B*

*MEI type = 0x0E*

*Read Device ID code = 0x01 (basic category)*

*Object ID = 0x00(start at object ID 0x00)*

*CRC*

#### **Response (01 2B 0E 01 82 00 00 03 00 08 57 69 6C 63 6F 78 6F 6E 01 08 32 35 30 31 30 30 30 30 02 06 31 2E 32 2E 31 33 + CRC):**

*Server ID = 0x01*

*Function code = 0x2B*

*MEI type = 0x0E*

*Read Device ID code = 0x01 (basic category)*

*Conformity level = 0x82 (conformity level of ILA device)*

*More Follows = 0x00 (No follows)*

*Next Object ID = 0x00*

*Number Of Object = 0x03*

Object ID = 0x00 (Company Name)

Object Length = 0x08

Object Value = 0x57696C636F786F6E("Wilcoxon" in ASCII)

Object ID = 0x01 (Serial Name)

Object Length = 0x08

Object Value = 0x3235303130303030 ("25010000" in ASCII)

Object ID = 0x02 (Firmware Version)

Object Length = 0x06

Object Value = 0x312E322E3133 ("1.2.13" in ASCII)

*CRC*

## Error Code Definition

Error code	Code Name	Definition	Type
0	ILA_SUCCESS	No error occurred	
70001	ILA_FAILED	The operation failed	ILA device type error
70002	ILA_OVER_DEVICE_MAX	The device amount has reached the maximum value	
70003	ILA_NOT_CONNECT	The device is not connected	
70004	ILA_CONNECT_CLOSED	The device connection has closed	
70005	ILA_TIMEOUT	The operation time has timed out	
70006	ILA_ID_ERROR	No device ID has been found	
70007	ILA_NOT_RECV_ERROR	The device is not in the receiving status	
70008	ILA_MODEL_ERROR	The device is not the ILA model type	
70010	ILA_DATA_NOT_READY	The data has not ready to be read	
78888	ILA_HW_ERROR	Hardware error occurred	
79999	ILA_UNKNOW_ERROR	Unknown error occurred	
80001	ILA_COM_CAN_NOT_OPENED	Failed to open the COM port	COM port type error
80002	ILA_COM_PORT_CLOSED	The COM port has been closed	
80003	ILA_COM_CAN_NOT_SEND	Error occurred while sending	
80004	ILA_COM_CAN_NOT_RECEIVE	Error occurred while receiving	
80006	ILA_COM_CRC_ERROR	CRC check failed	
80051	ILA_COM_SEND_FUNC_ERROR	COM error occurred while calling send function	
80101	ILA_COM_RECV_LENGTH_ERROR	Received wrong size from COM port	
80201	ILA_COM_RAWQUEUE_OVERFLOW	Catch queue overflow while calling send function	
80202	ILA_COM_OPTION_NOTSUPPORT_ERROR	Catch unsupported error while calling send function	
80210	ILA_COM_NULL_POINT_ERROR	Catch null pointer error while calling send function	