MASSA

GENERATIONS AHEAD IN SONAR & ULTRASONIC TECHNOLOGY

Quick Guide on Getting Started SonAire M3 Sensor & Digi Gateway

Introduction

The Massa Model SonAire M3 or M3is Wireless Ultrasonic Sensors used in combination with a Digi ConnectPort X4 or X2 Gateway is a solution to non-contact wireless tank level monitoring. This document will guide you to quickly get the devices integrated and evaluated in your application. For detailed information, see the SonAire M3 Sensor Users Guide.
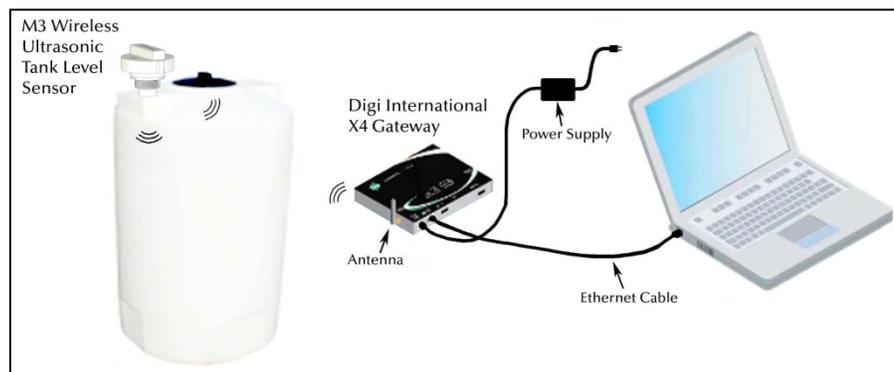
Normal Operating Mode

The basic operating mode is for the SonAire M3 Sensor to obtain range (to liquid level surface), temperature and other information and pack it in a status message then send this message (upon waking up) automatically to the gateway. The gateway will then push the message to the iDigi Cloud server. When evaluating the application for the first time, it may be set up to save the status information to a file in the gateway without a cloud server so you can validate tank level reports.

Overview of Sensor

Sensors supplied from the factory will allow association to any gateway, but with all its settings disabled for automated messaging. The minimum settings required for tank level measurement and automatic outgoing messaging to be enabled includes; the sleep time, the awake time (defaulted to 31 sec), and the outgoing message type. Another noteworthy sensor feature is its ability to obtain diagnostic data (ultrasonic waveforms) that can be used to evaluate the mounting location (and any issues) and general performance. It is strongly recommended that each new application (tank model) obtain diagnostic waveforms in an empty or near empty tank. Sensor settings could require custom sensitivity values and these waveforms will allow you to determine what is best required for your specific tank. Once an application has been established, it is not necessary to obtain diagnostic waveforms for the same tank (and mounting location). In general, normal settings will work in most applications. Typical sensor settings for long battery life have *Sleep* setting between 1 to 12 hours with the *Awake* setting at 30 seconds.

Overview of Gateway

The Digi ConnectPort X4 or X2 Gateway details in this guide will only address the use of the Ethernet port and not a cellular configured X4 gateway (option). You may be required to upgrade gateway firmware and possibly gateway XBee settings. Massa Products provides python code to allow the sensor's settings to be easily configured, store sensor status information within the gateway, store diagnostic waveform data within the gateway, and sending automated messages to iDigi Cloud server. Up to 10 sensors can be associated to a single gateway but additional sensors can be added with a Digi Wall Router. For detailed information, see the SonAire M3 Sensor Users Guide.

M3 Wireless Ultrasonic Tank Level Sensor

Digi International X4 Gateway

Power Supply

Antenna

Ethernet Cable

Getting started

This guide is used to get you quickly started on setting up your application. For more information, see the SonAire M3 User's Guide. Here are the basic steps (don't power sensor just yet):

1) Gateway: Obtain a gateway and connect it to your laptop's Ethernet port
2) Gateway: Gain access by using Digi's Device Discovery utility (40002265_G.exe)
3) Gateway: Verify the gateway's firmware is up to date (see Digi's website).
4) Gateway: Verify the gateway's XBee firmware is at least version 21A0
5) Gateway: Verify/set the gateway's XBee registers as follows: Scan Channels (SC)=0x1ffe, Peripheral sleep count (SN)=2880, Cyclic sleep period (SP): 1000
6) Gateway: Manually set the date and time under the System Configuration UI.
7) Gateway: Obtain latest python files from Massa's website and load dia.zip, dia.py, python.zip and zigbee.py into the gateway's Python Files section. The dia.yml will need to be edited.
8) Gateway: In the Python Auto-start Settings UI, enable command line dia.py dia.yml and keep Action on Exit as *None*
9) EDIT: Using a text editor, open the dia.yml file and adjust the following line items show below. Note that enabling a setting is done by removing the # symbol at the beginning of the line.

   extended_address '00:13:a2:00:40:98:13:a3!' # *enter your sensor's MAC address as seen on label*
   sleep_interval_sec: 120   # example sleep time of 2 minutes
   awake_time_sec: 31   # maintain 31 sec
   outgoing_message_mode: 5   # 5 sets operation to obtain status (level) then wake up & send message
   log_status: 'true'   # record messages in python files section named status_log.txt

   Note: For editing the dia.yml, consider obtaining a free copy of Notepad ++ text editor and configure it for python use.

10) Gateway: Load the edited dia.yml into the gateway's Python Files section
11) Gateway: Reboot and wait 1 minute
12) Sensor: Power sensor and observe the LED to sequence 3 green blinks followed by a steady red.   In approximately 15 seconds the sensor should start blinking red every 2 seconds indicating it is associated with the gateway.   Since the dia.yml control file has been set up to reprogram the sensor, you may notice additional blinks other than the steady 2 seconds.
13) Sensor: After the sensor settings have been programmed, it will go to sleep and wake up every 2 minutes to send status message. Point the sensor to a flat target within the sensor operating sensing range.
14) Gateway: After a few wake-up periods, open the status_log.txt file to view the automated messages that will include the range (distance) to wall, temperature and other information.
15) EDIT: Obtain diagnostic data on an empty tank (or near empty). Using a text editor, open the dia.yml file and enable the following line item show below (by removing the '#' symbol. Load this edited file into the gateway and reboot. After a few minutes, check the Python Manage Files UI for waveform files that will appear as: M3data_0013a200402c163e_201202229075448.M3. For model SonAire M3/150 there will be 2 such files, for other models there will be 4 files. To view these files, obtain a copy of the SonAire M3 Application Software and recall each one. After you have obtained the files, disable this line (with # symbol) and reload the dia.yml to avoid obtaining diagnostic data on every gateway boot.

      obtain_diagnostic_data: 'true'

NOTE: When a sensor is going to be installed for long term testing or in a final installation, it is recommended that the setting "*comm_fail_boot*" be enabled. This enables watchdog operation to reboot sensor in rare cases of a Zigbee communication failure. See sensor manual for information.

These are the minimum requirements of setting up a gateway and sensor for local observation. The next step is to obtain an account at Digi's Device Cloud (by Etherios) that will be used as a server to receive the sensor's automated messages.

Screen shots of minimum requirements to run an application

Figure 1
dia.yml example showing settings enabled (with # symbol removed). Note extended_address contains MAC of your sensor

```
Sensor #1 ###############################################################
- name: M3-1235
  driver: devices.vendors.massa.m3_driver:M3Driver
  settings:
# Main DIA related
    extended_address: '00:13:a2:00:40:8d:12:35!'
    xbee_device_manager: 'xbee_device_manager'
# Sensor Settings
    sleep_interval_sec: 120          # units in seconds
    awake_time_sec: 31               # 20 sec min, 31 suggested
    #data_collection_interval_sec: 0 # units in sec (inactive for outgoing msg modes 5 and 6)
    outgoing_message_mode: 5         # limit 0-6
    #auto_message_length: 8          # limit 1-8
    #sensitivity: 'normal'           # 'normal', 'high', 'very high', 'low', 'very low', 'custom'
    #user_comment: ''                # limit 32 ASCII characters
    #comm_fail_boot: 48              # Reboot sensor on failed Zigbee Ack / Unassoc Limits: 0,4-254
# XBee settings
    # pan_id: 0x0000000000000000     # limits 0x0000000000000000 to 0x000000007fffffff
    # xbee_security: 'disable'       # XBee Encryption enable EE register 'enabled' or 'disabled'
    # xbee_security_key: 0x0000000000000000 # XBee (Link encryption key) KY register, 16 bytes
    # poll_rate: 0                   # XBee PO register, PO=0 for XBee FW 29A0 or greater
    # node_discovery: 1              # XBee NO register 0 or 1, set this to 1
    # pan_conflict_threshold: 3      # XBee PAN conflict threshold register (CR), set to 3
    # device_options: 0             # XBee DO register, 0 first find, 4 for strongest signal assoc.
    #clear_errors: 'true'            # Clear all sensor errors
    # reboot_sensor: 'true'          # Reboot Sensor
# Routine control
    obtain_diagnostic_data: 'true'   # 'true' or 'false'
    # clear_data_history: 'false'    # select 'true' to clear sensor history
    # program_m3_firmware_file: 'M3-95V31-8.bin' #see massa.com for most up to date sensor FW ver.
    # keep_xbee_awake: 'true'        # 'true' to keep Xbee awake 5 min, 'false' to disable keep awake
    log_status: 'true'               # 'true' to output 'log_status' file recording autosend messages
    log_dia: 'true'                  # 'true' to output 'log_dia' file reporting sensor settings
```

Figure 2
Python files required: python.zip, dia.zip, dia.py, dia.yml, and zigbee.py. Also shown here are the diagnostic files that were created with the settings enabled in the dia.yml as seen on the left. This includes: dia_log.txt, status_log.txt and 4 waveform files named M3data….M3

Manage Files

Action	File Name	Size
☐	M3data_0013a200408d1235_20130522124242.M3	7127 bytes
☐	M3data_0013a200408d1235_20130522124226.M3	7121 bytes
☐	M3data_0013a200408d1235_20130522124209.M3	6965 bytes
☐	M3data_0013a200408d1235_20130522124153.M3	6772 bytes
☐	python.zip	144321 bytes
☐	dia.zip	486324 bytes
☐	dia.py	12485 bytes
☐	dia.yml	5168 bytes
☐	dia_log.txt	7276 bytes
☐	status_log.txt	464740 bytes
☐	zigbee.py	1147 bytes

Delete

▶ Auto-start Settings

Figure 3
Autostart settings command line set to run python code.

Python Configuration
▶ Python Files
▼ Auto-start Settings
Specify python programs to be run when the device boots.

Enable	Action On Exit	Auto-start Command Line (specify program filename to ex
☑	None	dia.py dia.yml
☐	None	
☐	None	
☐	None	

Apply Cancel

Figure 4
Connections Management UI shows python properly running. The dia.py will not be displayed here if an error occurs in not properly editing the dia.yml file. Such errors include not lining up settings in the same column.

Connections Management

Active System Connections

Action	Connected From	Connected To	Protocol	Sessions
☐			Python: dia.py	0
☐			Python thread	0
☐			Python thread	0
☐			Python thread	0
☐			Python thread	0
☐			Python thread	0
☐			Python thread	0

Refresh Disconnect

Figure 6
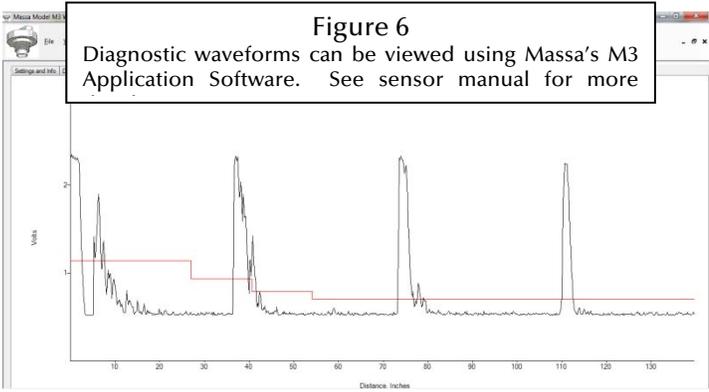Diagnostic waveforms can be viewed using Massa's M3 Application Software. See sensor manual for more

Figure 7
dia_log.txt file reports driver version and sensor settings

Sensor 00:13:a2:00:40:30:09:4b!: M3 driver started, version 1.02. Dia version 1.4.14. 2013-05-20 12:14:44
Sensor 00:13:a2:00:40:30:09:4b!: Sensor Firmware Version = 31.8 2013-05-20 12:28:47
Sensor 00:13:a2:00:40:30:09:4b!: Serial Number = 2873 2013-05-20 12:28:47
Sensor 00:13:a2:00:40:30:09:4b!: Sensor Model = M-3/95 2013-05-20 12:28:47
Sensor 00:13:a2:00:40:30:09:4b!: data_collection_interval_sec = 0 2013-05-20 12:28:50
Sensor 00:13:a2:00:40:30:09:4b!: comm_fail_boot = 0 2013-05-20 12:28:50
Sensor 00:13:a2:00:40:30:09:4b!: sleep_interval_sec = 120.0 2013-05-20 12:28:50
Sensor 00:13:a2:00:40:30:09:4b!: awake_time_sec = 30.72 2013-05-20 12:28:50
Sensor 00:13:a2:00:40:30:09:4b!: outgoing_message_mode = 5 2013-05-20 12:28:50
Sensor 00:13:a2:00:40:30:09:4b!: user_comment = 2013-05-20 12:28:52
Sensor 00:13:a2:00:40:30:09:4b!: error code = 0 2013-05-20 12:28:52
Sensor 00:13:a2:00:40:30:09:4b!: auto_message_length = 8 2013-05-20 12:28:55
Sensor 00:13:a2:00:40:30:09:4b!: sensitivity = Normal 2013-05-20 12:28:55

Figure 8
status_log.txt file records automated status messages that will include time stamp, distance, temperature, battery voltage, etc.

Sensor 00:13:a2:00:40:30:09:4b!: Dist=47.7" Temp=18.1C Batt=4.9V UltrasonicStrength=50% RadioSignal=VeryStrong Event=4062 2013-01-08 04:37:20
Sensor 00:13:a2:00:40:30:09:4b!: Dist=47.7" Temp=18.1C Batt=4.9V UltrasonicStrength=50% RadioSignal=VeryStrong Event=4062 2013-01-08 04:37:40
Sensor 00:13:a2:00:40:91:1c:94!: Dist=54.8" Temp=15.2C Batt=5.3V UltrasonicStrength=75% RadioSignal=VeryStrong Event=1819 2013-01-08 05:24:45
Sensor 00:13:a2:00:40:91:1c:94!: Dist=54.8" Temp=15.2C Batt=5.3V UltrasonicStrength=75% RadioSignal=VeryStrong Event=1819 2013-01-08 05:25:02

Massa Products Corporation assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using Massa's sensors.

Troubleshooting

When the gateway's XBee settings have been adjusted, power the sensor. It will blink green 3 times and then turn on RED and typically within 18 seconds, it will associate with the gateway. Once associated, the sensor will blink RED every 2 seconds.

Unassociated indication: If the sensor does not associate within 18 seconds (LED solid red), then it will continue to search for a gateway to associate with (up to 50 seconds). After this power up sequence, sensor with sleep setting = 0 (factory default) will default to a 1 minute sleep cycle to conserve battery power. Once associated, then it will blink every 2 seconds.

Reasons a sensor will not associate with a gateway:

a) Gateway's "Scan Channel (SC)" register not set to "0x1ffe".

b) Sensor PAN ID is non-zero (previously changed by user) and does not match gateway's PAN ID.

c) Encryption is enabled (EE=1) with non-matching gateway and sensor "Link encryption key (KY)".

d) Gateway has more than 10 sensors associated with it. If *Number of remaining children (NC)* register (see *Xbee Configuration – Device*) is 0, then sensor will not associate. You may manually reset this by changing Scan Channel (SC) register to 0x1 and then return it to 0x1ffe which will reset the NC register to 10 allowing sensors to rejoin. Also, another option is to reboot the gateway if using Massa python code version 1.02 which will reset the NC register (upon boot). To expedite sensor rejoining the gateway, you can re-power the sensor (press commission button after battery removal to discharge any internal voltage).

e) Sensor is out of RF range with the gateway. Avoid placing a gateway in an all metal building with the sensor outdoors. In such cases, place gateway by window facing sensor.

f) If gateway was power cycled with sensor that has s Deep Sleep setting = 0 (factory default), then it may take sensor up to 1 hour to re-associate with gateway. To expedite sensor rejoining the gateway, you can re-power the sensor (press commission button after battery removal to discharge any internal voltage).

g) If sensor is sleeping longer than the gateway's poll timeout setting (0.03 x SN x SP in seconds), then it will have to rejoin gateway. If the intention is for the gateway to maintain association with sensor, program the gateway's poll timeout to at least 3-4 times longer than the sensor's sleep setting. If the gateway had dropped a sensor from its association list, the sensor will have to rejoin the gateway which causes additional battery power usage and should be avoided in most applications. The gateway maintains individual poll timeout registers for each sensor which, when associated, will reset for a full poll timeout period.

h) Antennas for both gateway and sensor may not be vertical.

First step is to have the sensor associated with the gateway before you proceed with adjusting any sensor settings (with sensor's red LED blinking every 2 seconds).

For additional troubleshooting information, see the SonAire M3 Sensor Users Guide Appendix H.