

**www.observator.com**

|  |
| --- |
|  |
| *Technical note*  OMC-048 Scriptable data logger  Config.txt and recovery.txt templates |
| Version: 20250930  Status: Released  Confidentiality: Not confidential  Date: 30 September 2025  Author: R. Beun |

Preface

The OMC-048 data logger is configured by creating a **config.txt** file. To help you create a new config.txt file, this document contains a table with pieces of text that can be copied to the config.txt file. Once ready, the config.txt file must be copied to the FLASH of your OMC-048 logger.

In case you want to use ‘Blue2Cast Remote Control’, you also need a **recovery.txt** file. At the end of this document, a table is given for creating a new recovery.txt file. This file should also be placed on the FLASH.

For creating the config.txt file please use notepad or notepad++ or another suitable (ASCII) editor to copy the text to.

In the below tables, note that everything to the right of a ‘:’ is just an example value. You need to change that value for your situation. For example, in below tables it may read:

- id: yourchoice

port: 1

Then, obviously, you need to replace *yourchoice* and indicate the correct number of the port. So, for example, you may change it to:

- id: level\_sens1

port: 2

Please refer to the ‘OMC-048 configuration manual’ for more information.

**Beware that the confix.txt file is sensitive to (leading) spaces! Make sure that lines have 0 or 2 leading spaces. No tabs.**

**=========== This document belongs to FW\_1.03B2803 ================**

Table of contents

[1 File header 4](#_Toc210141993)

[2 System 5](#_Toc210141994)

[3 Cellular modem 6](#_Toc210141995)

[4 Iridium modem 7](#_Toc210141996)

[5 Onboard (internal) sensor drivers 8](#_Toc210141997)

[6 Analog (voltage/current) sensor drivers 9](#_Toc210141998)

[7 Digital (pulse/switch) input drivers 10](#_Toc210141999)

[8 Serial sensor drivers 11](#_Toc210142000)

[9 Generic drivers 15](#_Toc210142001)

[10 Power and relay port drivers 18](#_Toc210142002)

[11 Common settings for drivers 19](#_Toc210142003)

[12 Log\_parameters 20](#_Toc210142004)

[13 The recovery.txt file 21](#_Toc210142005)

# File header

It is recommended to always have these blocks at the top of your file, for your own convenience.

|  |  |
| --- | --- |
| Text blocks to copy to config.txt | Comments |
| # \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  # FILE INFORMATION  # \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  #  # Created by : <your name, company>  # Date : <date of creation/release>  # FW version : <e.g. FW\_1.03B2803>  # =========================================================  #  # < Description>  # \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  # CHANGE LOG  # \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  #  # Date of change: <date>  # Changed by : <name, company>  # Description:  # < Reason of making change, who requested the change, and so on>  #  # \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | This whole text block is just comments for your own benefit. The logger ignores all lines starting with ‘#’  Please replace all text between ‘<’ and ‘>’ with your own text. |

# System

|  |  |
| --- | --- |
| These blocks must be present in EVERY config file | Comments |
| # ========================================================  # --------------System settings---------------------------  # ========================================================  Omc048:  usb\_mode: repl  system\_id: Paris\_Nord  application: Turb\_monitor  file\_log\_level: warn  repl\_log\_level: info  utc\_time\_offset\_hours: +1  utc\_time\_offset\_minutes: +0  sensor\_data\_print: True  self\_test: True  Data\_file:  - id: data  create\_interval: "2 10 \* \* \*" | repl/storage/debug, but don't use 'debug'  debug/info/warn/error/fatal  debug/info/warn/error/fatal  All sensor data is printed in the REPL  ‘False’ to speed-up the start-up process.  Opening/closing data files on the SD card.  "second minute hour day week-day" |

# Cellular modem

|  |  |
| --- | --- |
| Copy to config.txt if you use the cellular modem | Comments |
| # ========================================================  # --------------Cellular Modem-Settings-------------------  # ========================================================  Modem:  id: modem  port: modem  sim\_username: sim\_usr  sim\_password: sim\_pwd  apn: sim\_apn  network\_technology: GSM + LTE  iot\_technology: CAT-M1  cellular\_diagnostics: True  cron\_offset: False  Connection\_settings:  data\_protocol: FTP  ftp\_url: ftp.omc-data-online.com  ftp\_port: 21  ftp\_username: <username>  ftp\_password: <password>  transmit\_interval: "0 10 \* \* \*"  utc\_time\_sync: True  utc\_time\_server: europe.pool.ntp.org  utc\_time\_server\_port: 123  webserver\_access: False  webserver\_url: blue2cast.com  webserver\_port: 5000  webserver\_password: <Contact Observator> | Get these details from your SIM card provider  'GSM + LTE' or 'GSM' or 'LTE'  ‘CAT-M1’ or ‘CAT-M1 + NB-IoT’ or ‘NB-IoT’  For internet connection through the cellular modem.  FTP/SSL/None (e.g. for OMC-DOL/Blue2Cast/Nothing)  Only relevant if ‘data\_protocol: FTP’  for FTP/SSL. Also used for webserver\_acces, if selected.  Sync to an NTP server. Alternatively, see 'GPS'  Blue2Cast remote control.  Leave as-is; Blue2Cast will fill-in the password. |

# Iridium modem

|  |  |
| --- | --- |
| Copy to config.txt if you use the external Iridium modem OMC-IEM. | Comments |
| # ========================================================  # --------------Iridium Modem-Settings--------------------  # ========================================================  #  # IMPORTANT: This driver uses a REFERENCE TABLE.  # so you MUST customize the iridium\_reference\_tables file !  Iridium\_modem:  id: iridium\_modem  port: serial4  supply\_port: 4  Iridium:  - id: data\_iridium  system\_id: Z  transmit\_interval: "10 0 \* \* \*"  add\_timeout\_field: False  Iridium\_file:  - id: iridium  create\_interval: "0 0 \* \* \*" | Don’t forget this!  Baudrate and so on are fixed by the driver.  Select a small id (like one character) to save on data transport cost.  "second minute hour day week-day"  Opening/closing data files for transmission over Iridium |

# Onboard (internal) sensor drivers

|  |  |
| --- | --- |
| Copy to config.txt if needed | Comments |
| Onboard:  - id: onboard  sample\_interval: "30 8,18,28,38,48,58 \* \* \*" | The onboard sensors are 'Int. Temperature', 'Int. Humidity', 'Int. Coin cell voltage'.  For models with SN>x0200 also 'Supply voltage' |
| Input\_power\_monitor:  - id: power\_mon  hysteresis\_high: 12.5  hysteresis\_low: 11.5  sample\_interval: "30 0 \* \* \*" | The power monitor allows you to let the logger shut down gracefully below a certain low voltage and restart above a high voltage.  Loggers with SN>48000200 only. |

# Analog (voltage/current) sensor drivers

|  |  |
| --- | --- |
| Copy to config.txt for each analog input driver you need | Comments |
| Analog\_voltage:  - id: AV1  port: 1  sample\_interval: "0 0 \* \* \*"  min\_in: 0  max\_in: 24000  min\_out: 0  max\_out: 24  log\_name: Ext\_voltage  log\_unit: V  log\_tag: VOLT1 | port 1 = pin 18 with range 0-24V, port 2 = pin 19 with range 0-5V |
| Analog\_current:  - id: AC2  port: 2  sample\_interval: "0 0 \* \* \*"  min\_in: 0  max\_in: 24  min\_out: 0  max\_out: 100  log\_name: Current1  log\_unit: "%"  log\_tag: CURR1 | port 1 to 4 correspond to pin 23 to 26  Special characters like % should be between quotes. |

# Digital (pulse/switch) input drivers

|  |  |
| --- | --- |
| Copy to config.txt for each digital input driver you need | Comments |
| rain\_pulse:  - id: cnt  port: 1  mm\_per\_pulse: 0.2  reset\_interval: None  sample\_interval: "0 0 \* \* \*" | Driver for a tipping-bucket rain gauge Also suitable for counting pulses from other sensors.  Port 1 to 2 correspond to pin 51 to 52  or eg. reset\_interval: "0 0 0 \* \*" To reset the counter at midnight.  To sample the counter every hour. |
| input\_state:  - id: switch1  port: 1  state\_invert: False  trigger\_on\_event: True  sample\_interval: "0,10,20,30,40,50 \* \* \* \*" | To sample the state of the input pins.  Port 1 to 2 correspond to pin 51 to 52  Or: sample\_interval: False |

# Serial sensor drivers

|  |  |
| --- | --- |
| Copy to config.txt for each serial driver you need | Comments |
| EXO:  - id: EX1  port: serial1  mode: RS232  baudrate: 9600  sample\_interval: "0 0,10,20,30,40,50 \* \* \*"  supply\_port: 1  supply\_port\_always\_on: False  wiper\_interval: "0, 0,20,40 \* \* \*"  delay\_after\_wipe: 25 | This driver is specific for Xylem EXO sondes, connected by RS232 via a DCP.  Make sure the EXO and DCP are configured properly before using this driver.  Must coincide with the sample time points, but there can be less. |
| manta:  - id: Manta1  port: serial1  baudrate: 19200  sample\_interval: "0 0,10,20,30,40,50 \* \* \*"  response\_timeout: 60  supply\_port: 1  supply\_port\_always\_on: False  wiper\_interval: "0 0,20,40 \* \* \*"  delay\_after\_wipe: 10 | This driver is specific for Eureka Manta sondes, connected by RS232.  Make sure the sonde is configured properly before using this driver.  Must coincide with the sample time points, but there can be less. |
| NEP5000:  - id: NEP2  port: serial2  mode: RS485  baudrate: 9600  sample\_interval: "0 \* \* \* \*"  supply\_port: 2  supply\_port\_always\_on: False | This driver is specific for the NEP-5000 turbidity sensor in streaming mode. Make sure the sensor is configured properly before using this driver. |
| navilock\_md6:  - id: gps  port: serial2  mode: RS232  baudrate: 9600  sample\_interval: "0 0 \* \* \*"  supply\_port: 2  supply\_port\_always\_on: False  startup\_time: 60  response\_timeout: 300  utc\_time\_sync: True | This driver is specific for the OMC-ANT-G09 (Navilock MD6) GNSS (GPS) receiver.  You can use Gps: as an alias for navilock\_md6:  Sync to the GPS clock. You can also sync to NTP. Don't enable both. |
| # IMPORTANT: This driver uses the GMX501 table in  # serial\_input\_reference\_tables!  gmx\_501:  - id: gmx  port: serial1  mode: RS422  baudrate: 19200  sample\_interval: "\* \* \* \* \*"  supply\_port: 1  supply\_port\_always\_on: True  response\_timeout: 2  utc\_time\_sync: True | This driver is for the Maximet GMX501 with GPS from Gill.  Don’t forget to check the GMX501 table against your Maximet settings.  Sync to the GPS clock of the GMX. |
| nortek\_cp:  - id: ncp  port: 3  baudrate: 19200  rxbuf: 1024  sample\_interval: "15 0,30 \* \* \*"  response\_timeout: 1860 | This driver is specific for Nortek Aquadopp and AWAC sensors. It supports only *current profiles* in binary mode.  Check the size of the binary output file |
| # IMPORTANT: This driver uses the indicated table in  # modbus\_reference\_tables!  modbus\_opus:  - id: opus  sample\_interval: "0 0,30 \* \* \*"  port: serial1  baudrate: 9600  mode: RS485  rxbuf: 1024  protocol: RTU  reference\_table: TriOS\_OPUS  measurement\_wait\_time: 25  response\_timeout: 120 | This driver is specific for the TriOS OPUS sensor. It uses the generic modbus driver ‘modbus’. Thus the reference table must be customized to fit the sensor settings!  Check the size data transfer (set of registers) in bytes |
| modbus\_spectrolyser:  - id: s  port: serial2  baudrate: 38400  mode: RS485  rxbuf: 1024  protocol: RTU  relay\_port: 1  relay\_port\_always\_on: True  startup\_time: 70  sample\_interval: "35 9,19,29,39,49,59 \* \* \*"  measurement\_wait\_time: 25  reference\_table: P,A,B,C,D  spect\_sleep\_duration: 555 # 600-25-10-10 =  response\_timeout: 10 | This driver is specific for the s::can Spectrolyser V3 sensor. It uses the generic modbus driver ‘modbus’. Thus the reference table ‘modbus\_reference\_tables.py’ must be customized to fit the sensor settings! |
| modbus\_seametrics:  - id: seam1  protocol: RTU  register\_print: True  sample\_interval: "0 \* \* \* \*"  supply\_port: 1  startup\_time: 1  port: serial1  mode: RS485  baudrate: 9600  reference\_table: PT12,CT2X | This driver is specific for the Seametrics smart sensors sensor. It uses the generic modbus driver ‘modbus’. Thus the reference table must be customized to fit the sensor settings! |
| modbus\_wimo:  - id: wimo  address: 0x80  wiper\_interval: "0 \* \* \* \*"  sample\_interval: "0,10,20,30,40,50 \* \* \* \*"  port: serial1  mode: RS232  baudrate: 9600  protocol: RTU  register\_print: True  supply\_port: 1  supply\_port\_always\_on: True | This driver is specific for the NKE WiMo multi-parameter sondes.  It requires the presence of the “modbus\_reference\_tables.py” file in the script/modules folder! No table is needed. |

# Generic drivers

These blocks must only be present if you use the corresponding generic driver.

|  |  |
| --- | --- |
| Copy to config.txt for each generic driver you need | Comments |
| # This driver uses the serial\_input\_reference\_tables.py file!  generic\_serial\_input:  - id: gsi1  port: serial1  mode: RS485  baudrate: 9600  reference\_table: table\_gsi  sample\_interval: "0 \* \* \* \*"  supply\_port: 3  supply\_port\_always\_on: False  startup\_time: 10  response\_timeout: 60 | You must customize this file!  Check this table with your sensor output.  "second minute hour day week-day" |
| # This driver uses the nmea\_reference\_tables.py file.  generic\_nmea:  - id: gnmea  port: nmea  mode: RS422  baudrate: 4800  reference\_table: VTG,GGA,RMC  sample\_interval: "0 \* \* \* \*"  supply\_port: 3  supply\_port\_always\_on: False  startup\_time: 10  response\_timeout: 60 | You must customize this file!  The NMEA protocol (this driver) can be used on any of the serial ports, including the nmea port  Check these table(s) with your sensor output. |
| # This driver uses the sdi12\_reference\_tables.py file.  generic\_sdi12:  - id: sens1  port: sdi12  # For SDI-12, you must indicate 1200 Bd, 7 bits, parity 0  baudrate: 1200  bits: 7  parity: 0  reference\_table: sdi\_sens1  sample\_interval: "0 \* \* \* \*"  supply\_port: 1  supply\_port\_always\_on: False  startup\_time: 1 | You must customize this file!  Must be these values!  ,,  ,,  You must customize this table! |
| # This driver uses the modbus\_reference\_tables.py file.  modbus:  - id: modbus\_1  protocol: RTU  register\_print: True  port: serial1  mode: RS485  rxbuf: 1024  baudrate: 9600  reference\_table: mt1,mt2 | You must customize this file!  Change to ‘False’ once the configuration is redy for deployment  Default no parity |
| # This driver uses the serial\_output\_reference\_tables.py file.  generic\_serial\_output:  - id: gso  port: serial4  mode: RS232  baudrate: 9600  reference\_table: table\_gso | You must customize this file!  You must customize this table! |
| Data\_serial\_out:  - id: data\_serial  create\_interval: "0,10,20,30,40,50 \* \* \* \*"  port: serial1  baudrate: 115200  mode: RS232  txbuf: 1024  reference\_table: mt | The table is optional. If present, customize the ‘serial\_output\_reference\_tables.py file’ |

# Power and relay port drivers

The power and relay ports are mostly controlled from sensor drivers. This is done using the “supply\_port” or “relay\_port” settings. However, if a port is not used by any of the sensors drivers, it can be controlled from below driver.

|  |  |
| --- | --- |
| Copy to config.txt for each driver you need | Comments |
| pulse\_driver:  - id: wiper  sample\_interval: "0,20,40 \* \* \* \*"  supply\_pulse\_port: 3  relay\_pulse\_port: 2  pulse\_duration: 5  normally\_off: True | Generates a pulse of the indicated duration and polarity on a power and/or relay port. |

# Common settings for drivers

You can add a selection of below lines to any of the sensor (input) drivers. Don’t forget that these lines need 2 leading spaces!

|  |  |
| --- | --- |
| Text lines to use in any sensor driver | Comments |
| startup\_time: 10  response\_timeout: 60  cooldown\_time: 0  sample\_interval: "0 \* \* \* \*"  cron\_offset: 00:00:12  average\_interval: "30 \* \* \* \*"  moving\_average\_window: 16  minimum\_required\_samples: 12  supply\_port: 1  supply\_port\_always\_on: False  relay\_port: 2  relay\_port\_always\_on: True | Timing  ,,  ,,  ,,  ,,  Averaging. Note that averaging over many samples uses a lot of internal memory!  ,,  ,,  Power/switching  ,,  ,,  ,, |

You can add a selection of below lines to (almost) any of the drivers **that use a serial port**. Don’t forget that these lines need 2 leading spaces!

|  |  |
| --- | --- |
| Text lines to use in (input/output) drivers that use serial ports | Comments |
| port: serial1  mode: RS485  baudrate: 9600  bits: 8  parity: None  stop: 1  rxbuf: 128  txbuf: 128 | serial1/serial/2/serial3/serial4/nmea/SDI\_12  RS232/RS422/RS485 settings  7/8  0/1/None (0=even, 1 =odd, None = no parity)  Number of stop bits  A data burst from the sensor should fit in the RX buffer (number of bytes)  For output drivers. |

# Log\_parameters

This block must only be present if you want to include only a selection of the parameters from specific sensors.

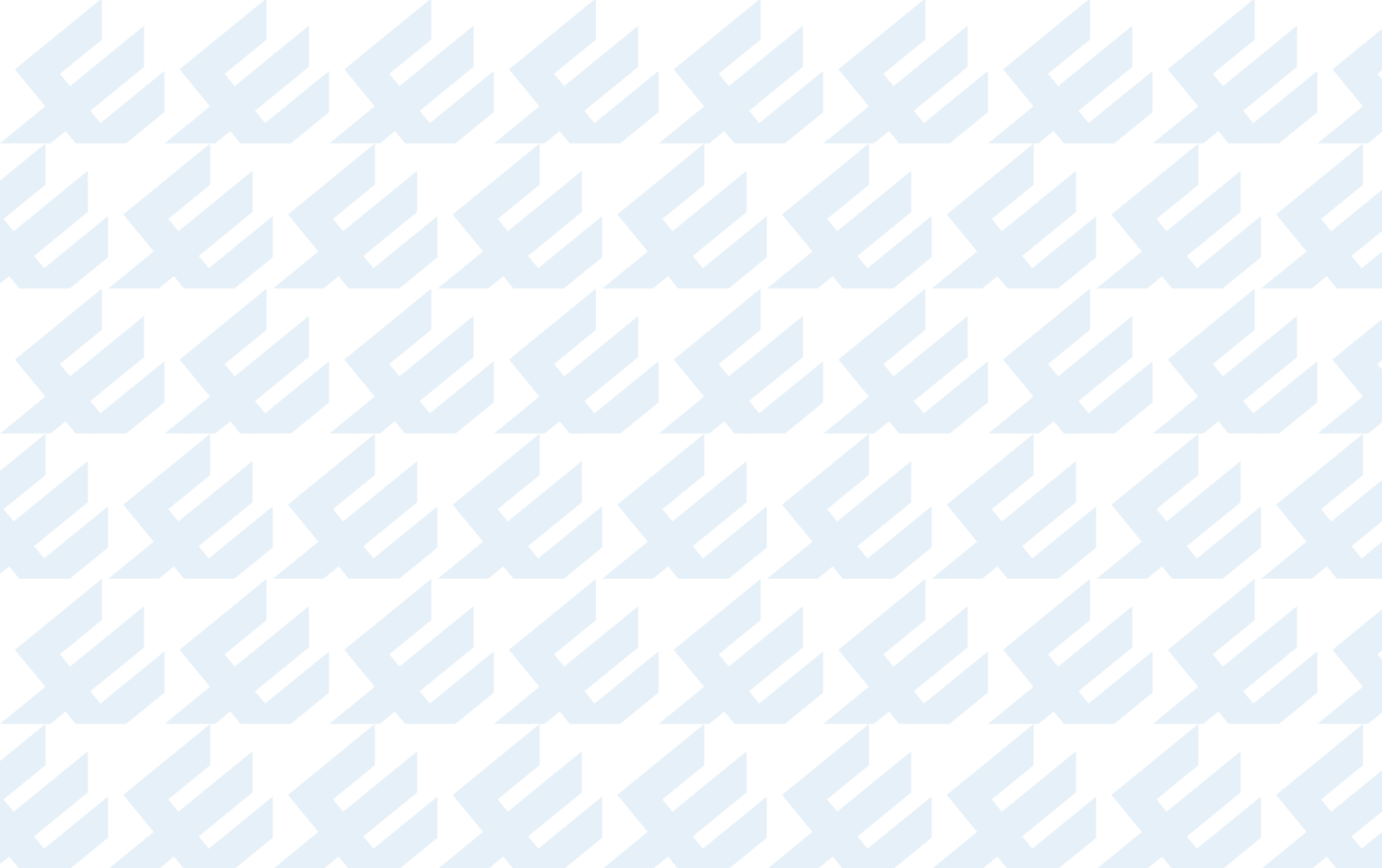
|  |  |
| --- | --- |
| Text blocks to copy to config.txt | Comments |
| Log\_parameters:  gps:  - Latitude  - Longitude  - Satellites  modem:  - Signal strength | id of the sensor of which only the listed parameters should be logged.  You may use ‘sensor-data-print: True’ to find the names of the parameters.  id of another sensor of which only the listed parameters should be logged. |

# The recovery.txt file

All previous tables are intended for the **config.txt** file. Below table is the complete **recovery.txt** file.

The recovery file should only be put on the FLASH drive if cellular communication is used with remote control (webserver\_access: True in the config file).

|  |  |
| --- | --- |
| Complete recovery.txt file | Comments |
| # ----System---- #  Omc048:  usb\_mode: repl  system\_id: RECOVERY  # ----Modem-Settings---- #  Modem:  id: onboard\_modem  port: modem  sim\_username:  sim\_password:  apn:  # ---- External connection settings ---- #  Connection\_settings:  webserver\_access: True  webserver\_url: blue2cast.com  webserver\_port: 5000  webserver\_password: <TO\_BE\_GENERATED>  transmit\_interval: "0 0,5,10,15,20,25,30,35,40,45,50,55 \* \* \*"  # ---- Power monitor ---- #  Input\_power\_monitor:  - id: power\_mon  hysteresis\_high: 12.5  hysteresis\_low: 11.5  sample\_interval: "30 0 \* \* \*" | Leave as-is  Make sure you have the correct SIM details filled-in (see config.txt)  Leave as-is  Leave as-is  Leave as-is  Leave as-is (Blue2Cast will fill-in a value)  Leave as-is, or choose other time points..  If the application is battery-powered, put this block in to prevent the batteries from being deep-discharged in case the logger remains in recovery mode for a long time.  Remove the power monitor if the application is mains-powered.  Adapt the thresholds for your specific case (e.g. battery type). |



**www.observator.com**