

# Technical specifications for the use of the Aquatronica ACQ225 module (Ethernet module) with third party software applications via HTTP

## IMPORTANT!! - ATTENTION!!

*The specifications regarding the operation of the Aquatronica module ACQ225 (Ethernet module) that are described in the following document are exclusively informational.*

***Aquatronica, a division of AEB srl, will not offer any technical support or guarantee of correct operation for any applications that are created based on the above-stated technical information.***

*The following document is offered "as is" without guarantees or technical support. Users who decide to develop systems based on this document will do so at their own risk. **Aquatronica will not be held responsible for damages caused by systems developed by users based on the information provided in this document.***

*Aquatronica may modify the contents of this manual at any moment and without notification; furthermore, Aquatronica reserves the right to modify its product's hardware/software with no obligation to update this document. Finally, Aquatronica may modify its product's hardware/software rendering them incompatible with the specifications described in this document.*

## Contents

1.	Introduction .....	3
2.	Data structure of aquarium parameters (rtMon.xml) .....	3
2.1.	DTD rtMon.xml .....	3
2.2.	General description .....	5
2.3.	Request result <result> .....	5
2.4.	Sensor <sensor> .....	5
2.5.	Power Unit <powerUnit> .....	5
2.5.1.	Plug <plug> .....	6
3.	Authorization .....	6
3.1.	Login .....	7
3.2.	Logout .....	7
4.	Remote plug state settings .....	7
5.	Legal .....	8

## 1. Introduction

The Aquatronica module ACQ225 (Ethernet module) is designed to transfer specific parameters and information of the connected Aquarium controller system via internet/intranet (refer to the ACQ225 user manual for more details on use and connection).

The ACQ225 module contains memorized web pages that display aquarium parameters and, if logged-in, allow the user to control the state of the power plugs (ON/OFF) of the Aquatronica Power Units.

This document contains the necessary technical specifications for the use of the above-stated services without using the default web page.

In this way, the ACQ225 module will become controllable service provider by using "HTTP GET."

Software that is developed by a user cannot be programmed into the ACQ225 module's memory and module access requires its IP address (the HTTP address will be omitted in this document).

Please note that names and parameters that are used by the module are "case sensitive" and the set of characters is UTF-8

## 2. Data structure of aquarium parameters (rtMon.xml)

The aquarium data that the ACQ225 module is capable of exporting via Ethernet is listed in XML document and can be recalled by performing a GET of the file **rtMon.xml**.

### 2.1. DTD rtMon.xml

The following is the data type definition (DTD) of the rtMon.xml document:

```
<!ELEMENT xmlResponse (login ,result ,Monitor ) >
  <!ATTLIST xmlResponse xml:lan  >
<!ELEMENT login (status ) >
<!ELEMENT status (#PCDATA) >
<!ELEMENT result (#PCDATA) >
<!ELEMENT Monitor (sensorList ,powerList ) >
<!ELEMENT sensorList (sensor*) >
<!ELEMENT sensor (name ,value ,status ) >
<!ELEMENT name (#PCDATA) >
<!ELEMENT value (#PCDATA) >
<!ELEMENT powerList (powerUnit*) >
<!ELEMENT powerUnit (name ,plugList ) >
  <!ATTLIST powerUnit id CDATA #REQUIRED >
<!ELEMENT plugList (plug*) >
<!ELEMENT plug (name ,status ) >
  <!ATTLIST plug id CDATA #REQUIRED >
```

**List 1 - DTD rtMon.xml**

An example of the XML documents is shown in Listing 2:

```
<?xml version="1.0" encoding="UTF-8"?>
<xmlResponse xml:lan="eng">
  <login>
    <status>0</status>
  </login>
  <result>0</result>
  <Monitor>
```

```

<sensorList>
  <sensor>
    <name>Temperatura</name>
    <value>26.7°C</value>
    <status>2</status>
  </sensor>
  <sensor>
    <name>Livello</name>
    <value>LIV2</value>
    <status>2</status>
  </sensor>
</sensorList>
<powerList>
  <powerUnit id="1">
    <name>Unità di Potenza</name>
    <plugList>
      <plug id="1">
        <name>Alim UP1</name>
        <status>138</status>
      </plug>
      <plug id="2">
        <name>Neon blu</name>
        <status>138</status>
      </plug>
      <plug id="3">
        <name>HQI</name>
        <status>138</status>
      </plug>
      <plug id="4">
        <name>Neon blu 2</name>
        <status>138</status>
      </plug>
      <plug id="5">
        <name>Ventola mob.</name>
        <status>148</status>
      </plug>
      <plug id="6">
        <name>Frigo</name>
        <status>218</status>
      </plug>
      <plug id="7">
        <name>Pompa mov. sin.</name>
        <status>138</status>
      </plug>
      <plug id="8">
        <name>Pompa mov. destr</name>
        <status>138</status>
      </plug>
    </plugList>
  </powerUnit>
  <powerUnit id="2">
    ...
  </powerUnit>
</powerList>
</Monitor>
</xmlResponse>

```

**List 2 – Example of rtMon.xml**

## 2.2. General description

The XML response of the ACQ225 module is enclosed inside the tag `xmlResponse`. The tag `login` contains information on login state (see paragraph 3). The tag `result` reports the result of the request while information on the system's state is contained inside the `Monitor` tag. The `Monitor` tag contains the `sensorList` block that encloses sensor information, and the `powerList` block that contains information on the power units.

Every sensor is defined by the `sensor` block and every power unit is defined by the `powerUnit` block.

Inside every `powerUnit`, the single plugs are described through the `plug` tag list inside the `plugList`.

Details on the various blocks can be found in the following paragraphs.

## 2.3. Request result <result>

The result of each request that is sent to the ACQ225 module is contained in the `result` tag and contains the system's state according to the codes found in Table 1:

Code <result>	Meaning
0	Request followed successfully (OK)
1	Request syntax error (SYNTAX ERROR)
2	Request failed (FAILED)
3	Access denied (DENIED)
4	Multiple access error (MULTIPLE)
5	User disconnected (DISCONNECTED)
255	ACQ225 module internal error (INTERNAL ERROR)

**Table 1 – State codes of <result>**

Refer to paragraph 3 for a detailed explanation of the state codes.

## 2.4. Sensor <sensor>

<name>

name assigned to the sensor

<value>

sensor value along with measuring unit

<status>

sensor state (one byte): if the less significant bit (LSB) is equal to 1, the sensor is in alarm state.

## 2.5. Power Unit <powerUnit>

"id"

univocal numerical identification of the power unit connected to the aquarium

<name>

name assigned to the power unit

<plugList>

plug list (outlets) of the power unit

### 2.5.1. Plug <plug>

"id"

univocal numerical identification of the power unit's plug

<name>

name assigned to the plug

<status>

the state of the plug (ON/OFF - 1 byte). If the most significant bit (0X80) is equal to 1, then the plug is ON; otherwise, the plug is OFF. The other bits indicate the programs in use on the plug as shown in table 2:

State	Description
0	No program
1	Program not defined
3	Output with a program (like timer program)
10	Timer program
20	ON/OFF program
30	Tide effect
40	Wave effect
50	Power cut
119	
60	Summer function
70	Redox sensor program
80	pH sensor program
90	Temperature sensor program
100	Conductivity sensor program
103	Density sensor program
110	Level sensor program
114÷117	Function key program (1÷4)
120	Manual function
123	Water leakage sensor program
126	Power unit not powered
127	Blocked plug (in emergency state)

**Table 2 – Plug state**

In addition to the states listed in table 2, are the states of the "complex programs" (XY). These states are associated with sensor programs +1. The following states are subject to complex programs: 71, 81, 91, 101, 104, 111, 124.

## 3. Authentication

The ACQ225 module is capable of activating and deactivating the single plugs of the power units; this function is only available if the user undergoes authentication by means of a password.

The login process determines the beginning of a session defined by a key (SID – session identifier) that is casually generated by the ACQ225 module. Once inside this session, the user is authorized to use the protected commands; if the authorized user does not make any requests for more than two consecutive minutes, the session will expire. In order to keep the session active, it is necessary to periodically send the ACQ225 module the SID key as the first field of the GET query:

rtMon.xml?K=sid

where *sid* is the SID of the session.

The ACQ225 module allows only one session at a time to avoid resource acquisition conflicts: only one user at a time can be authorized to command plug state.

### 3.1. Login

The login is done by performing a GET HTTP of the XML document `login.xml` with the query string:

```
login.xml?L=pwdB64
```

in which `pwdB64` is the previously set password of the ACQ225 module, only it is encoded in "base64".

The response (contents of `login.html`) is shown in list 3:

```
<?xml version="1.0" encoding="UTF-8"?>
<xmlResponse xml:lan="eng">
  <login>
    <status>1</status>
    <sid>123131231</sid>
  </login>
  <result>0</result>
</xmlResponse>
```

**List 3 – login result (login.xml)**

`<status>`

the values of this tag are shown in table 3:

Value	Description
0	User not authorized
1	User authorized
255	Internal error

**Table 3 – Value of `<status>` in login**

`<sid>`

If the user is logged-in correctly, the value of the `sid` tag is the session key (session identifier). The session key needs to be periodically sent to the ACQ225 module in order to keep the session open ( $K=sid$ ).

### 3.2. Logout

Logout takes place by performing a GET HTTP of the XML document `logout.xml` utilizing the query string:

```
logout.xml?K=sid&X
```

where *sid* is the SID of the opened session; the *x* parameter is the logout command. The type of response given by the ACQ225 module is shown in list 3.

## 4. Remote plug state settings

Once authorized, the user can set the state of the plugs on the power unit by using a remote command. The following is the syntax of the GET HTTP:

```
rtMon.xml?K=sid&U=idUnit&P=idPlug&S=status
```

where:

sid = SID;

idUnit = identifier of the power unit where the plug that must be modified is found;

idPlug = identifier of the plug that must be modified;

status = new plug status. The values are shown in table 4:

Status	Meaning
0	Plug is functioning in automatic
2	Plug is forced off
3	Plug is forced on

**Table 4 – Plug command codes**

The response to the command is a rtMon.xml document equivalent to those shown in list 2; to see the result of the request, it is necessary to evaluate the `result` tag (see 2.3). You will see that the request of the activation/deactivation of the plug is not recognized instantly on the selected plug but requires a certain period of time. This is due to the fact that the ACQ225 module must re-route the request to the selected ACQ002T/3T device.

## 5. Legal

The firmware of the ACQ225 module device (Ethernet module) is property of A.E.B. srl, based in Cavriago (RE), Via dell'Industria, n.20.

The publication of specifications for the use of the data made public by the device (ACQ225) does not attribute any right on the above-stated firmware.

A.E.B. srl is the sole holder of all copyright and all other rights of the firmware, the moral rights and the rights of economic use.

It is thus forbidden to reproduce, translate, adapt, or transform the firmware, as well as distribute, under any form and with no restrictions whatsoever, including the cession, or in any way make the firmware available to third parties whether for sale or for free. The licensee shall not hire or lease the firmware or any part of it.