**SnapAV Wattbox Integration Protocol Document**

Integration Protocol v1.3 rev20180912

Overview

This integration protocol details how a third-party system can be used to control a SnapAV Wattbox. With the wattbox online, the integration protocol will be listening for connections on **port 23 at the controllers IP address. NOTE: 10 simultaneous connections can be made at a time.** To get started, netcat or similar software can be used to initiate a connection and test any of the following protocol commands below.

Authentication

The protocol requires authentication before proceeding with commands. Once connected, a login prompt will be received and the third-party system must provide a valid username and password. If correct, login will be successful and other commands can be issued. If incorrect, the third-party system will be prompted for login again.

Specification

THIRD-PARTY SYSTEM <-----------------------------------------------> SnapAV Wattbox  
 i.e. Wattbox IP: 192.168.0.20 Port: 23

Integration

|  |
| --- |
| **Message Structure** |
| Command and response messages are standard ASCII text. |
| ? – Request message |
| ! – Control message |
| # - Error message |
| ~ - Unsolicited message |
| \n – End of command message, ASCII hex: 0x0A dec: 11 |

Protocol

|  |  |
| --- | --- |
| **Protocol Command** | **Description/Response** |
| ?Firmware\n | Request Firmware Version.  Response: ?Firmware=1.0.0.0\n |
| ?Hostname\n | Request Hostname.  Response: ?Hostname=Wattbox\n |
| ?Serial\n | Request Serial Number.  Response: ?Serial=12345678\n |
| ?Model\n | Request Model Number.  Response: ?Model=WB-700-IPV-12\n |
| ?OutletCount\n | Request Outlet Count.  Response: ?OutletCount=16\n |
| ?OutletStatus\n | Request Outlet States.  Response: ?OutletStatus=0,0,0,0,0,0,0,0,0,0,0,0\n  Where the array index is the outlet number and the value at the index indicates state. 0 for off, 1 for on. |
| ?OutletPowerStatus=OUTLET\n  Where OUTLET is the Outlet number. | Request Outlet Power Status for a specific outlet.  Response: ?OutletPowerStatus=OUTLET,POWER,CURRENT\_RMS,VOLTAGE\_RMS\n  Where OUTLET is the outlet index you requested, POWER is the power consumption, CURRENT\_RMS is the current RMS, and VOLTAGE\_RMS is the voltage RMS. |
| ?OutletName\n | Request Outlet Names for all outlets.  Response:  ?OutletName={Outlet 1},{Outlet 2},{Outlet 3},{Outlet 4},{Outlet 5},{Outlet 6},{Outlet 7},{Outlet 8},{Outlet 9},{Outlet 10},{Outlet 11},{Outlet 12}\n |
| ?UPSStatus\n | Request UPS Status if there is a UPS attached.  Response:  ?UPSStatus=50,0,Good,False,25,True,False\n  Where 50 is the battery charge perecentage, 0 is the battery load as a percentage, Good indicates battery health, False indicates power lost, 25 indicates battery runtime in minutes, True indicates alarm enabled, False indicates alarm muted.  Battery Charge: 0-100%  Battery Load: 0-100%  Battery Health: Good/Bad  Power Lost: True/False  Battery Runtime: Number in Minutes  Alarm Enabled: True/False  Alarm Muted: True/False |
| ?PowerStatus\n | Request Power Status for the system.  Response:  ?PowerStatus=600,6000,1100,1\n  Where 600 is the current in amps, 6000 is the power in watts, 1100 is the voltage in volts, and 1 is the safe voltage status. |
| !OutletSet=OUTLET,ACTION\n  Where OUTLET is the specific outlet and ACTION is ON/OFF/TOGGLE/RESET. | Request to set a specific outlet to a new state. RESET does not adhere to power on delay so implementing your own is required if a delay is needed.  Response:  OK\n |
| !AutoReboot=STATE\n  Where STATE is 1 for enabled or 0 for disabled | Request to set auto reboot to a new state.  Response:  OK\n |
| #Error\n | Sent whenever an invalid command was received or an internal device error has occurred. |