# Shure Battery Rack Mount Charger (SBRC)

## Command Strings

Command strings documentation for the SBRC Rack Mount Charger
Version: 1 (2020-G)

# Table of Contents

# Shure Battery Rack Mount Charger (SBRC) Command Strings

## Command Strings Overview

Shure command strings are a set of commands and status reports used by control system programmers to interface to Shure devices. The Shure device is connected via Ethernet to a control system, such as

- AMX, Crestron or Extron
- Symetrix, Biamp, other digital signal processors (DSP)
- Specialized custom programs

The Shure device is considered to be the server and the control system is considered to be the client.

**Connection:** Ethernet (TCP/IP; select “Client” in the AMX/Crestron program)
**Port:** 2202

### Conventions

There are 4 types of strings

| | |
|---|---|
| **GET** | Finds the status of a parameter. After the AMX/Crestron sends a GET command, the system responds with a REPORT string |
| **SET** | Changes the status of a parameter. After the AMX/Crestron sends a SET command, the system responds with a REPORT string to indicate the new value of the parameter. |
| **REP** | • When the system receives a GET or SET command, it replies with a REPORT command to indicate the status of the parameter.<br>• REPORT is also automatically sent by the device when a status changes, for example:<br>As a battery charges, the charger sends the reports without any GET commands:<br>< REP 1 BATT_TIME_TO_FULL 00107 ><br>< REP 1 BATT_TIME_TO_FULL 00106 ><br>< REP 1 BATT_TIME_TO_FULL 00105 > |
| **SAMPLE** | Used for metering audio levels. (Not applicable with some Shure devices.) |

**Note:**

- All messages sent and received are ASCII. Note that the level indicators and gain indicators are also ASCII.
- It is not necessary to constantly query parameters because most parameters send a REPORT command when they change.

# Device Command Strings

## ALL

| | |
|---|---|
| **Description** | Discovery of device properties. |
| **Commands** | < GET *x* ALL ><br>< REP ... > |
| **Variables** | When **x** is zero, the device responds with REP for all device-specific properties and ALL channel, module, or bay-related properties including all metered properties.<br><br>When **x** is a channel, module, or bay number, the device responds with REP for all device-specific properties and ALL channel, module, or bay **x**-related properties, including all metered properties. |
| **Notes** | None. |

## DEVICE_ID

| | |
|---|---|
| **Description** | Controls the Device ID. |
| **Commands** | < GET DEVICE_ID ><br>< REP DEVICE_ID {Name1yyyyyyyyyyyyyyyyyyyyyyyyyy} ><br><br>< SET DEVICE_ID {Name1} ><br>< REP DEVICE_ID {Name1yyyyyyyyyyyyyyyyyyyyyyyyyy} > |
| **Variables** | Where the repeating **y** represents the spaces returned by the device to pad the Device ID to 31 characters. |
| **Notes** | The device always responds with 31-character ID.<br><br>SET accepts 1-8 Characters from the set: A-Z,a-z,0-9,!"#$%&'()*+,-./:;<=>?@[\]^_`~ and space. |

## FW_VER

| | |
|---|---|
| **Description** | Discovery of the firmware version. |
| **Commands** | Self test passed:<br><br>< GET FW_VER ><br>< REP FW_VER {2.0.15.2yyyyyyyyyyyyyyyy} ><br><br>Self test failed:<br><br>< GET FW_VER ><br>< REP FW_VER {2.0.15.2*yyyyyyyyyyyyyyy} > |

| | |
|---|---|
| **Variables** | Where the repeating **y** represents the spaces returned by the device to pad the response to 24 characters. |
| **Notes** | Package version number reported as Maj.Min.Pack.Build. |

## FLASH

| | |
|---|---|
| **Description** | Controls the flash to identify a device. |
| **Commands** | < SET FLASH ON ><br>< REP FLASH ON > |
| **Variables** | None. |
| **Notes** | Device initiates an Identify then stops flashing. |

## MODEL

| | |
|---|---|
| **Description** | Discovery of the model name of the device. |
| **Commands** | < GET MODEL ><br>< REP MODEL {SBRCyyyyyyyyyyyyyyyyyyyyyyyyyyyy} > |
| **Variables** | Where the repeating **y** represents the spaces returned by the device to pad the model name to 32 characters. |
| **Notes** | The device always responds with a 32-character model name. |

## STORAGE_MODE

| | |
|---|---|
| **Description** | Controls the storage mode setting. |
| **Commands** | < GET STORAGE_MODE ><br>< REP STORAGE_MODE OFF ><br><br>< SET STORAGE_MODE ON ><br>< REP STORAGE_MODE ON ><br><br>< SET STORAGE_MODE TOGGLE ><br>< REP STORAGE_MODE OFF > |
| **Variables** | None. |
| **Notes** | TOGGLE switches between ON and OFF. |

# Module Command Strings

## BATT_MODULE_TYPE

| | |
|---|---|
| **Description** | Discovers the type of the battery module. |
| **Commands** | < GET *x* BATT_MODULE_TYPE ><br>< REP *x* BATT_MODULE_TYPE 002 ><br><br>< GET 0 BATT_MODULE_TYPE ><br>< REP 1 BATT_MODULE_TYPE 002 ><br>< REP 2 BATT_MODULE_TYPE 004 ><br>< REP 3 BATT_MODULE_TYPE 005 ><br>< REP 4 BATT_MODULE_TYPE 000 > |
| **Variables** | Where *x* is the module number.<br><br>Using 0 returns information for all modules |
| **Notes** | Numeric string, 3 characters<br><br>Identifies the model of the module:<br><br>000 : No module installed<br><br>001 : AXT902<br><br>002 : AXT901<br><br>003 : SBC-AX (For SB900x)<br><br>004 : SBM920<br><br>005 : SBM910<br><br>006 : SBM910M<br><br>255 : Invalid / Unsupported module |

# Bay Command Strings

## BATT_BARS

| | |
|---|---|
| **Description** | Discovers the number of bars for a battery. |
| **Commands** | < GET *x* BATT_BARS ><br>< REP *x* BATT_BARS 003 ><br><br>When the number of bars changes: |

| | |
|---|---|
| | < REP *x* BATT_BARS 004 > |
| **Variables** | Where *x* is the bay number.<br>Using 0 returns information for all bays. |
| **Notes** | Numeric, three characters<br>000 - 005 : Number of bars reported<br>254 : An error has occurred, the value is not applicable at this time<br>255 : Unknown |

## BATT_CAPACITY_MAX

| | |
|---|---|
| **Description** | Discovers the manufacturer's battery maximum capacity in mAh. |
| **Commands** | < GET *x* BATT_CAPACITY_MAX ><br>< REP *x* BATT_CAPACITY_MAX 02393 > |
| **Variables** | Where *x* is the bay number.<br>Using 0 returns information for all bays. |
| **Notes** | Numeric, five characters<br>00000 - 65533 : The manufacturer's battery maximum capacity in mAh<br>65534 : An error has occurred, the value is not applicable at this time<br>65535 : No battery or not applicable |

## BATT_CHARGE

| | |
|---|---|
| **Description** | Discovers the charge in percent for a battery. |
| **Commands** | < GET *x* BATT_CHARGE ><br>< REP *x* BATT_CHARGE 027 ><br>< REP *x* BATT_CHARGE 028 ><br>...<br>< REP *x* BATT_CHARGE 099 ><br>< REP *x* BATT_CHARGE 100 > |
| **Variables** | Where *x* is the bay number. |

| | |
|---|---|
| | Using 0 returns information for all bays. |
| **Notes** | Numeric, three characters<br>000 - 100 : Percentage of charge<br>254 : An error has occurred, the value is not applicable at this time<br>255 : Unknown |

## BATT_CURRENT_CAPACITY

| | |
|---|---|
| **Description** | Discovers the current battery capacity in mAh. |
| **Commands** | < GET *x* BATT_CURRENT_CAPACITY ><br>< REP *x* BATT_CURRENT_CAPACITY 02189 > |
| **Variables** | Where *x* is the bay number.<br>Using 0 returns information for all bays. |
| **Notes** | Numeric, five characters<br>00000 - 65533 : The current battery capacity in mAh<br>65534 : An error has occurred, the value is not applicable at this time<br>65535 : No battery or not applicable |

## BATT_CURRENT_CAPACITY_MAX

| | |
|---|---|
| **Description** | Discovers the current maximum capacity in mAh. |
| **Commands** | < GET *x* BATT_CURRENT_CAPACITY_MAX ><br>< REP *x* BATT_CURRENT_CAPACITY_MAX 02393 > |
| **Variables** | Where *x* is the bay number.<br>Using 0 returns information for all bays. |
| **Notes** | Numeric, five characters<br>00000 - 65533 : The current battery maximum capacity in mAh<br>65534 : An error has occurred, the value is not applicable at this time<br>65535 : No battery or not applicable |

## BATT_CYCLE

| | |
|---|---|
| **Description** | Discovers the number charging cycles for a battery. |
| **Commands** | Battery placed into charger bay *x*:<br><br>< REP *x* BATT_CYCLE 00006 ><br>...<br>< GET *x* BATT_CYCLE ><br>< REP *x* BATT_CYCLE 00006 > |
| **Variables** | Where *x* is the bay number.<br><br>Using 0 returns information for all bays. |
| **Notes** | Numeric, five characters<br><br>00000 - 65533 : Number of charging cycles<br><br>65534 : An error has occurred, the value is not applicable at this time<br><br>65535 : Unknown or not applicable |

## BATT_DETECTED

| | |
|---|---|
| **Description** | Discovers if a battery is detected. |
| **Commands** | < GET *x* BATT_DETECTED ><br>< REP *x* BATT_DETECTED YES > |
| **Variables** | Where *x* is the bay number.<br><br>Using 0 returns information for all bays. |
| **Notes** | Fixed String<br><br>YES<br>NO |

## BATT_ERROR

| | |
|---|---|
| **Description** | Discovers the error status of a battery. |
| **Commands** | < GET *x* BATT_ERROR ><br>< REP *x* BATT_ERROR 000 > |

| | |
|---|---|
| **Variables** | Where *x* is the bay number.<br>Using 0 returns information for all bays. |
| **Notes** | Numeric, three characters<br>000 : No Active Error<br>001 : Unknown Module<br>002 : Unrecognized Battery<br>003 : Deep Discharge Recovery Failed<br>004 : Charge Failed<br>005 : Check Battery<br>006 : Check Charger<br>007 : Communication Failure<br>255 : No Battery Present |

## BATT_HEALTH

| | |
|---|---|
| **Description** | Discovers the health in percent for a battery. |
| **Commands** | < GET *x* BATT_HEALTH ><br>< REP *x* BATT_HEALTH 099 > |
| **Variables** | Where *x* is the bay number.<br>Using 0 returns information for all bays. |
| **Notes** | Numeric, three characters<br>000 - 100 : Percentage of health<br>254 : An error has occurred, the value is not applicable at this time<br>255 : Unknown |

## BATT_STATE

| | |
|---|---|
| **Description** | Discovers the state of a battery. |
| **Commands** | < GET *x* BATT_STATE ><br>< REP *x* BATT_STATE NORMAL ><br>After some period of time, battery becomes fully charged: |

| | |
|---|---|
| | < REP *x* BATT_STATE FULL > |
| **Variables** | Where *x* is the bay number.<br><br>Using 0 returns information for all bays. |
| **Notes** | Fixed String<br><br>FULL<br>CALCULATING<br>NORMAL<br>WARM<br>WARM_FULL<br>HOT<br>COLD<br>PRECHARGE<br>READY_TO_STORE<br>DISCHARGE_CALC<br>DISCHARGING<br>DISCHARGING_WARM<br>DISCHARGING_COLD<br>ERROR<br>NO_BATT |

## BATT_TEMP_C

| | |
|---|---|
| **Description** | Discovers the temperature in Celsius. |
| **Commands** | < GET *x* BATT_TEMP_C ><br>< REP *x* BATT_TEMP_C 055 ><br><br>There is an offset of 40 so the actual value = 55 - 40 = 15° C. |
| **Variables** | Where *x* is the bay number.<br><br>Using 0 returns information for all bays. |
| **Notes** | The actual value = the reported value - 40<br><br>Numeric, three characters<br><br>000 - 253 : Temperature in Celsius<br><br>254 : An error has occurred, the value is not applicable at this time<br><br>255 : Unknown |

## BATT_TEMP_F

| | |
|---|---|
| **Description** | Discovers the temperature in Fahrenheit. |
| **Commands** | < GET *x* BATT_TEMP_F ><br>< REP *x* BATT_TEMP_F 095 ><br><br>There is an offset of 40 so the actual value = 95 - 40 = 50° F. |
| **Variables** | Where *x* is the bay number.<br><br>Using 0 returns information for all bays. |
| **Notes** | The actual value = the reported value - 40<br><br>Numeric, three characters<br><br>000 - 253 : Temperature in Fahrenheit<br><br>254 : An error has occurred, the value is not applicable at this time<br><br>255 : Unknown |

## BATT_TIME_TO_FULL

| | |
|---|---|
| **Description** | Discovers the number of minutes for a battery to reach the target charging level. |
| **Commands** | < GET *x* BATT_TIME_TO_FULL ><br>< REP *x* BATT_TIME_TO_FULL 00060 ><br><br>Battery placed into charger bay *x*:<br><br>< REP *x* BATT_TIME_TO_FULL 65533 ><br>...<br>< REP *x* BATT_TIME_TO_FULL 00060 ><br>...<br>< REP *x* BATT_TIME_TO_FULL 00001 ><br>< REP *x* BATT_TIME_TO_FULL 00000 ><br>< REP *x* BATT_TIME_TO_FULL 65529 ><br><br>Battery removed:<br><br>< REP *x* BATT_TIME_TO_FULL 65535 > |
| **Variables** | Where *x* is the bay number.<br><br>Using 0 returns information for all bays. |
| **Notes** | Numeric, five characters |

| | |
|---|---|
| | Considered as time to target where:<br><br>Charging Mode: Value is the estimated time to full charge.<br>Storage Mode: Value is the estimated time to optimal storage voltage.<br><br>00000 - 65528: Number of minutes estimated to reach the target<br><br>65529 : Battery is fully charged<br><br>65533 : Calculation in progress<br><br>65534 : Error has occurred<br><br>65535 : Unknown or not applicable |