



# SBC220 SBC240

## Command Strings

Shure SBC220 and SBC240 command strings for third-party control systems, such as Crestron or Extron. Includes all supported programming commands.

Version: 1 (2019-F)

---

# Table of Contents

<b>SBC220 SBC240Command Strings</b>	<b>3</b>	<b>Command Strings Overview</b>	<b>3</b>
		<b>Command Strings</b>	<b>3</b>

# SBC220 SBC240

## 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

<b>GET</b>	Finds the status of a parameter. After the AMX/Crestron sends a GET command, the system responds with a REPORT string
<b>SET</b>	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.
<b>REP</b>	<ul style="list-style-type: none"> <li>• When the system receives a GET or SET command, it replies with a REPORT command to indicate the status of the parameter.</li> <li>• REPORT is also automatically sent by the device when a status changes, for example:  As a battery charges, the charger sends the reports without any GET commands:  &lt; REP 1 BATT_TIME_TO_FULL 00107 &gt; &lt; REP 1 BATT_TIME_TO_FULL 00106 &gt; &lt; REP 1 BATT_TIME_TO_FULL 00105 &gt;</li> </ul>
<b>SAMPLE</b>	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.

# Command Strings

## ALL

<b>Description</b>	Discovery of device properties.
<b>Commands</b>	< GET x ALL > < REP ... >
<b>Variables</b>	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.  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.
<b>Notes</b>	None.

## BATT\_BARS

<b>Description</b>	Discovers the number of bars for a battery.
<b>Commands</b>	< GET x BATT_BARS > < REP x BATT_BARS 003 >  When the number of bars changes:  < REP x BATT_BARS 004 >
<b>Variables</b>	Where x is the bay number.  Using 0 returns information for all bays.
<b>Notes</b>	Numeric, three characters  000 - 005 : Number of bars reported  254 : An error has occurred, the value is not applicable at this time  255 : Unknown

## BATT\_CAPACITY\_MAX

<b>Description</b>	Discovers the manufacturer's battery maximum capacity in mAh.
<b>Commands</b>	< GET x BATT_CAPACITY_MAX > < REP x BATT_CAPACITY_MAX 02393 >

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

## BATT\_CHARGE

<b>Description</b>	Discovers the charge in percent for a battery.
<b>Commands</b>	< GET x BATT_CHARGE > < REP x BATT_CHARGE 027 > < REP x BATT_CHARGE 028 > ... < REP x BATT_CHARGE 099 > < REP x BATT_CHARGE 100 >
<b>Variables</b>	Where x is the bay number. Using 0 returns information for all bays.
<b>Notes</b>	Numeric, three characters 000 - 100 : Percentage of charge 254 : An error has occurred, the value is not applicable at this time 255 : Unknown

## BATT\_CURRENT\_CAPACITY

<b>Description</b>	Discovers the current battery capacity in mAh.
<b>Commands</b>	< GET x BATT_CURRENT_CAPACITY > < REP x BATT_CURRENT_CAPACITY 02189 >
<b>Variables</b>	Where x is the bay number. Using 0 returns information for all bays.

<b>Notes</b>	<p>Numeric, five characters</p> <p>00000 - 65533 : The current battery capacity in mAh</p> <p>65534 : An error has occurred, the value is not applicable at this time</p> <p>65535 : No battery or not applicable</p>
--------------	---

## BATT\_CURRENT\_CAPACITY\_MAX

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

## BATT\_CYCLE

<b>Description</b>	Discovers the number charging cycles for a battery.
<b>Commands</b>	<p>Battery placed into charger bay x:</p> <pre>&lt; REP x BATT_CYCLE 00006 &gt; ... &lt; GET x BATT_CYCLE &gt; &lt; REP x BATT_CYCLE 00006 &gt;</pre>
<b>Variables</b>	<p>Where x is the bay number.</p> <p>Using 0 returns information for all bays.</p>
<b>Notes</b>	<p>Numeric, five characters</p> <p>00000 - 65533 : Number of charging cycles</p> <p>65534 : An error has occurred, the value is not applicable at this time</p>

65535 : Unknown or not applicable

## BATT\_DETECTED

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

## BATT\_ERROR

<b>Description</b>	Discovers the error status of a battery.
<b>Commands</b>	< GET x BATT_ERROR > < REP x BATT_ERROR 000 >
<b>Variables</b>	Where x is the bay number. Using 0 returns information for all bays.
<b>Notes</b>	Numeric, three characters 000 : No Active Error 001 : Unknown Module 002 : Unrecognized Battery 003 : Deep Discharge Recovery Failed 004 : Charge Failed 005 : Check Battery 006 : Check Charger 007 : Communication Failure 255 : No Battery Present

## BATT\_HEALTH

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

## BATT\_MODULE\_TYPE

<b>Description</b>	Discovers the type of the battery module.
<b>Commands</b>	For an SBC220  < GET x BATT_MODULE_TYPE > < REP x BATT_MODULE_TYPE 129 >  For an SBC220 ganged system of 3 units  < GET 0 BATT_MODULE_TYPE > < REP x BATT_MODULE_TYPE 129 > < REP x BATT_MODULE_TYPE 129 > < REP x BATT_MODULE_TYPE 129 > < REP x BATT_MODULE_TYPE 000 >
<b>Variables</b>	Where x is the module number.  Using 0 returns information for all modules
<b>Notes</b>	Numeric string, 3 characters  000 : No module installed  129 : Slave  133 : Master or Slave  255 : Invalid or unsupported module



## BATT\_STATE

<b>Description</b>	Discovers the state of a battery.
<b>Commands</b>	<p>&lt; GET x BATT_STATE &gt;          &lt; REP x BATT_STATE NORMAL &gt;</p> <p>After some period of time, battery becomes fully charged:</p> <p>&lt; REP x BATT_STATE FULL &gt;</p>
<b>Variables</b>	<p>Where x is the bay number.</p> <p>Using 0 returns information for all bays.</p>
<b>Notes</b>	<p>Fixed String</p> <p>FULL          CALCULATING          NORMAL          WARM          WARM_FULL          HOT          COLD          PRECHARGE          READY_TO_STORE          DISCHARGE_CALC          DISCHARGING          DISCHARGING_WARM          DISCHARGING_COLD          ERROR          NO_BATT</p>

## BATT\_TEMP\_C

<b>Description</b>	Discovers the temperature in Celsius.
<b>Commands</b>	<p>&lt; GET x BATT_TEMP_C &gt;          &lt; REP x BATT_TEMP_C 055 &gt;</p> <p>There is an offset of 40 so the actual value = 55 - 40 = 15° C.</p>
<b>Variables</b>	<p>Where x is the bay number.</p> <p>Using 0 returns information for all bays.</p>
<b>Notes</b>	<p>The actual value = the reported value - 40</p> <p>Numeric, three characters</p>

	<p>000 - 253 : Temperature in Celsius</p> <p>254 : An error has occurred, the value is not applicable at this time</p> <p>255 : Unknown</p>
--	---

## BATT\_TEMP\_F

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

## BATT\_TIME\_TO\_FULL

<b>Description</b>	Discovers the number of minutes for a battery to reach the target charging level.
<b>Commands</b>	<p>&lt; GET x BATT_TIME_TO_FULL &gt;</p> <p>&lt; REP x BATT_TIME_TO_FULL 00060 &gt;</p> <p>Battery placed into charger bay x:</p> <p>&lt; REP x BATT_TIME_TO_FULL 65533 &gt;</p> <p>...</p> <p>&lt; REP x BATT_TIME_TO_FULL 00060 &gt;</p> <p>...</p> <p>&lt; REP x BATT_TIME_TO_FULL 00001 &gt;</p> <p>&lt; REP x BATT_TIME_TO_FULL 00000 &gt;</p> <p>&lt; REP x BATT_TIME_TO_FULL 65529 &gt;</p> <p>Battery removed:</p> <p>&lt; REP x BATT_TIME_TO_FULL 65535 &gt;</p>

<b>Variables</b>	<p>Where x is the bay number.</p> <p>Using 0 returns information for all bays.</p>
<b>Notes</b>	<p>Numeric, five characters</p> <p>Considered as time to target where:</p> <p>Charging Mode: Value is the estimated time to full charge.</p> <p>Storage Mode: Value is the estimated time to optimal storage voltage.</p> <p>00000 - 65528: Number of minutes estimated to reach the target</p> <p>65529 : Battery is fully charged</p> <p>65533 : Calculation in progress</p> <p>65534 : Error has occurred</p> <p>65535 : Unknown or not applicable</p>

## DEVICE\_ID

<b>Description</b>	<p>Controls the Device ID.</p>
<b>Commands</b>	<pre>&lt; GET DEVICE_ID &gt; &lt; REP DEVICE_ID {Name1yyyyyyyyyyyyyyyyyyyyyyyyyyyyyy} &gt;  &lt; SET DEVICE_ID {Name1} &gt; &lt; REP DEVICE_ID {Name1yyyyyyyyyyyyyyyyyyyyyyyyyyyyyy} &gt;</pre>
<b>Variables</b>	<p>Where the repeating <b>y</b> represents the spaces returned by the device to pad the Device ID to 31 characters.</p>
<b>Notes</b>	<p>The device always responds with 31-character ID.</p> <p>SET accepts 1-8 Characters from the set: A-Z,a-z,0-9,!"#\$\$%&amp;'()*+,-./:;&lt;=&gt;@[\\]^_`~ and space.</p>

## FW\_VER

<b>Description</b>	<p>Discovery of the firmware version.</p>
<b>Commands</b>	<p>Self test passed:</p> <pre>&lt; GET FW_VER &gt; &lt; REP FW_VER {2.0.15.2yyyyyyyyyyyyyyyyyy} &gt;</pre> <p>Self test failed:</p>

	<pre>&lt; GET FW_VER &gt; &lt; REP FW_VER {2.0.15.2*yyyyyyyyyyyyyyyy} &gt;</pre>
<b>Variables</b>	Where the repeating <b>y</b> represents the spaces returned by the device to pad the response to 24 characters.
<b>Notes</b>	Package version number reported as Maj.Min.Pack.Build.

## FLASH

<b>Description</b>	Controls the flash to identify a device.
<b>Commands</b>	<pre>&lt; SET FLASH ON &gt; &lt; REP FLASH ON &gt;</pre>
<b>Variables</b>	None.
<b>Notes</b>	Device initiates an Identify then stops flashing.

## MODEL

<b>Description</b>	Discovery of the model name of the device.
<b>Commands</b>	<pre>&lt; GET MODEL &gt; &lt; REP MODEL {SBC220yyyyyyyyyyyyyyyyyyyyyyyy} &gt;</pre>
<b>Variables</b>	Where the repeating <b>y</b> represents the spaces returned by the device to pad the model name to 32 characters.
<b>Notes</b>	The device always responds with a 32-character model name.

## STORAGE\_MODE

<b>Description</b>	Controls the storage mode setting.
<b>Commands</b>	<pre>&lt; GET STORAGE_MODE &gt; &lt; REP STORAGE_MODE OFF &gt;  &lt; SET STORAGE_MODE ON &gt; &lt; REP STORAGE_MODE ON &gt;  &lt; SET STORAGE_MODE TOGGLE &gt; &lt; REP STORAGE_MODE OFF &gt;</pre>
<b>Variables</b>	None.
<b>Notes</b>	TOGGLE switches between ON and OFF.