Zebra Battery Programmer Box V2 User Manual

Table of Contents

Overview

This manual covers the use of the Zebra Battery Programmer Box V2 which allows for the programming and reading/verifying of most Zebra batteries.

Battery Programmer Features

Simple serial interface over USB. Only requires serial terminal software such as TeraTerm.

Compatible with Windows, Mac OS, and Linux.

Support for all Zebra EEPROM sizes/addresses, including 2 cell packs

Support for reading and writing hex files to/from EEPROM, Authentication IC, internal storage and the serial interface.

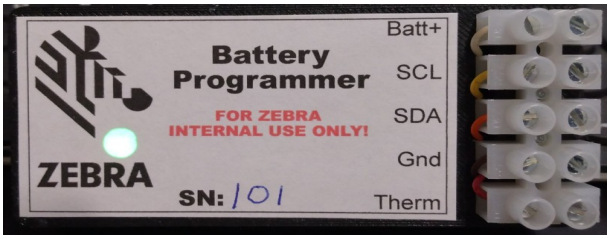
Support for “updating” battery and gas gauge data.

Batteries Currently Supported

MPA2/MPA3 Smart battery
MPA3 Gifted battery (MC18, Rogue, Frenzy)
Pollux battery
Falcon/IronMan/Thunder/Lightning Battery
Hawkeye Battery
Sentry Battery
Galactus/Badger/Frozone battery
Meteor/Gravity/Simba battery
Value Tier Battery
Value Tier Battery w/EEPROM 16K
PP+ Version 2 Battery (New Gas Gauge)

Hardware

As can be seen in the image below it has a terminal block for the battery connections and a status LED. On the left side (not shown) there is a USB connector for power and host PC hookup



When plugged into USB on your PC the programmer shows up as both a USB serial port, and a USB disk drive. Hex files can be copied to/from the drive, and long file names are supported. (Up to 42 characters long)

NOTE: The USB drive only has about 2MB of storage space.
NOTE: When saving an EEPROM/Auth chip hex file to the USB drive the programming box will disconnect/reconnect the PC USB to force the PC to reread the drive.

Status LED

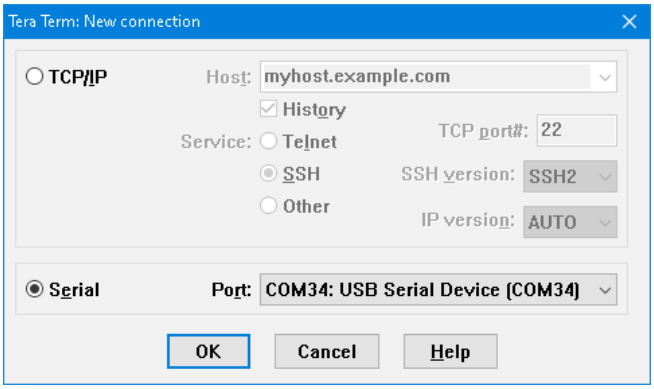
The Battery Programmer has an LED as seen above. The LED shows the following colors:
Yellow – The device is powered up but not connected to serial terminal program.
Blue – The device is connected to a serial terminal program, but no battery type is selected.
Green – The device is connected to a serial terminal program, and a battery type is selected.
Red – Battery communication is active. Do **not** disconnect the battery at this time!

Software Installation

A serial communications program is required to use the Battery Programmer. We recommend TeraTerm, but other serial terminal programs may work. TeraTerm can be found on the Battery Programmer SharePoint site: https://zebra.sharepoint.com/sites/BatteryProgrammer

NOTE: The newer V2 version of the box is NOT compatible with the Arduino IDE serial monitor

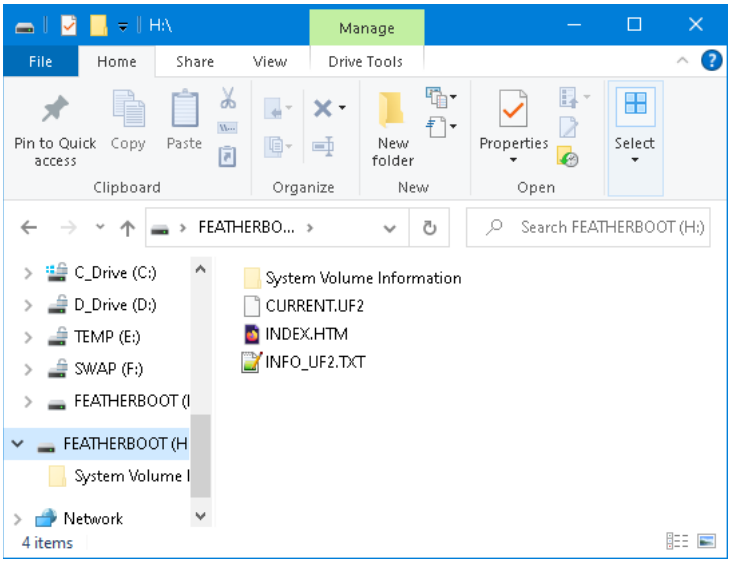
Plug the box into your PC and make a note of the serial port it was assigned. After installing TeraTerm start it and you will see the following dialog, select “Serial” and then pick the port assigned to the box. It will probably be labeled “USB Serial Device”.



Firmware Updating

Firmware updates are supplied in a “uf2” file. The naming convention for these files is: BattprogV2_VerX_YJ.uf2 where X_Y is the version number of the firmware. Connect to the box using TeraTerm and issue the “UF” (update firmware) command.

The box will reboot into bootloader mode and you should see a drive appear as shown below:



Drag and drop the uf2 file onto this drive. The box will load the new firmware and then reboot.

Command List

Cmd	Function	Notes
B	Select battery type	Pick the type of battery, must be done before other operations.
R(E|A)[1]	Read battery data	Reads entire battery data storage into the buffer.
W(E|A)[1]	Write battery data	Overwrites entire battery data storage with the data in the buffer.
U(E|A)[1]	Update battery data	Updates the battery using the data in the buffer.
V(E|A)[1]	Verify battery data	Checks the battery data against the data in the buffer.
UHD	Auto update hex data	Updates battery hex data to the latest rev.
E	Erase buffer	Flush buffer to all 0xFF's.
LS	Load hex file into buffer	Loads a hex file from serial port. Does NOT clear buffer first.
DS	Dump hex file from buffer	Dumps a hex file out the serial port from the buffer.
LF	Load buffer from file	Loads hex file from the internal drive. Does NOT clear buffer first.
SF	Save buffer to file	Writes buffer to a hex file on the internal drive.
DIR	Directory of drive	Display contents of the internal drive.
VBD	Validate Battery Data	Perform battery data validation. (Check checksums, etc..)
A	Authenticate Battery	Performs a full authentication on the battery auth chip.
RTC	Read Temp Chip	Reads battery temperature from temp chip
UGG	Update Gas Gauge	Updates the gas gauge firmware/settings and the battery data
VER	Version	Shows the software/hardware versions.
STAT	Show status	Shows the current box status.
UF	Update Firmware	Reboots device into bootloader mode to load new firmware.
?	Help	Displays context sensitive list of available commands.

1. E or A specifies which device EEPROM or Auth chip to read/write.

Detailed Command Descriptions

Note: Many of the commands use a "buffer" located in RAM to store the data from the battery or to get the data to write into the battery. Data can be loaded into the buffer from the serial port (In Intel hex file format), or an Intel hex file on the internal drive. The data in the buffer can also be written out the serial port in Intel hex file format, or saved onto the internal drive as an Intel hex file.

Where mention is made to "erasing" or "flushing" the buffer, that means that the buffer will be filled with hex 0xFF bytes.

B – Select battery type

This command allows the user to choose which battery type is being used. The current choices are:

0: No Battery – This will float the I2C lines so they don't load or drive the battery I2C pins.
1: MPA2 PP – MPA2 style power precision battery. Also know as a "Smart Battery".
2: MC95 PP

3: MC18 PP+
4: Rouge/TC8000
5: Frenzy/WT6000
6: IronMan/Lightning – TC51/56 PP+ battery.
7: Pollux – TC70/75 PP battery.
8: Falcon/Thunder – TC71/76 PP+ battery.
9: Hawkeye/TC20
10: Sentry/Elektra
11: Galactus/Badger/Firebird/Frozone Auth
12: Frozone EEPROM – Frozone EEPROM, fixed data.
13: Value tier battery
14: PP+ V2 Battery – PP+ battery using new gas gauge

R(E|A) – Read battery data

Read the contents of the battery EEPROM (RE) or the auth chip data area (RA) into the buffer.

W(E|A) – Write battery data

Overwrite the entire contents of the battery EEPROM (WE) or the auth chip data area (WA) using the data currently in the buffer. After writing the data it will verify the write by reading the data back and comparing it to the buffer.

U(E|A) – Update battery data

Update the data in the EEPROM/Auth chip using the data in the RAM buffer. You must load valid data (for the selected battery) into the buffer before using this command or you will corrupt the battery. The command will update the battery data differently depending on what battery type is selected, as follows:

a. For PP+ data the following is done:
The data blocks 0-2 is copied over from the new data.
The data in blocks 3-7 is left untouched. (Dynamic blocks)
The data in blocks 8-14 is copied from the new data.
The data in blocks 15-16 is left untouched. (Cell identifying data)
The data in blocks 17-31 (25 for auth chip data) is copied from the new data.

b. For Pollux data the following is done:
The format revision byte will be updated, only values 0-3 are supported.
Bytes 2-12 will be left untouched. (Part number, rev, serial number, date made)
Bytes 13-239 will be updated.
The checksum at byte 0 will be updated.
Bytes 240-252 will be left untouched. (aggregate charge)
Bytes 253-295 will be updated.
Bytes 296-323 will be left untouched. (Health, part number)
Bytes 324-511 will be updated.

c. Devices that have both types of data, Falcon, IronMan, etc. will have both sets of changes applied.

d. For Hawkeye data the following is done:
The format revision byte will be updated, only values 0-3 are supported.
Bytes 4-39 will be left untouched. (Part number, rev, serial number, date made)
Bytes 40-199 will be updated.
The checksum at byte 0 will be updated.
Bytes 201-215 will be left untouched. (aggregate charge)
Bytes 216-251 will be updated.
Bytes 252-259 will be left untouched. (Health)
Bytes 260-415 will be updated.

e. For PP+ V2 data the entire Auth chip data is overwritten with the new hex file.

V(E|A) – Verify battery data

Checks that the entire contents of the battery EEPROM (VE) or the auth chip data area (VA) matches the data currently in the buffer. It will stop at the first mismatch.

UHD – Auto update hex data

Updates the hex battery data to the latest rev. Uses files loaded into the internal disk drive in the GG directory.

E – Erase buffer

Flushes the buffer to all 0xFF's.

LS – Load Hex file from serial port

Load data in Intel hex file format into the buffer thru the serial port connection to the PC. It does not flush the buffer first so it is possible to load multiple hex files into different sections of the buffer, or to over write just part of the buffer with a small hex file.

DS – Dump Hex file to the serial port

Dump the contents of the buffer in Intel hex file format out the serial port connection to the PC.

LF – Load Hex file from internal drive

Load the buffer from an Intel hex file located on the internal drive. It does not flush the buffer first so it is possible to load multiple hex files into different sections of the buffer, or to over write just part of the buffer with a small hex file.

SF – Save Hex file to internal drive

Saves the contents of the buffer into an Intel hex file on the internal drive. You are limited to about 2MB of space on the internal drive. Long file names are supported.

NOTE: When you save the buffer to a file on the internal drive the box will disconnect/reconnect the USB to force the PC to reread the drive.

VBD – Validate battery data

Performs a full validation of the data in the battery. Does basically the same steps a terminal does when checking a battery.

1. Checks all check sums and data types for valid values. (Both EEPROM/Auth chip and gas gauge data)
2. For PP+ batts checks that the data in the EEPROM matches the data in the gas gauge.
3. For PP+ batts checks that unused blocks have the correct default pattern.

Note: This command will overwrite the buffer contents with the current EEPROM/Auth chip data.

DIR – Directory of SD card

Displays a list of the files on the SD card plugged into the box.

A – Authenticate battery

Checks the validity of the battery auth chip to insure it’s a valid Zebra battery.

RTC – Read Temp Chip

Displays the battery temperature read from the temp chip.

UGG – Update Gas Gauge

Updates the firmware and settings in the gas gauge as well as updating the data in the battery EEPROM/Auth IC. The process is driven by files loaded into the internal disk drive.

VER – Show version information

Displays the version of the software running on the box as well as the hardware revision.

STAT – Show device status

Shows the current status of the box. This includes the type of battery selected as well as the size of the battery memory, the current verbose mode setting, the clock drive type, the pull up resistors and voltage selected, the current battery voltage, and the resistance of the battery thermistor if one was detected.

UF – Update Firmware

Tells the box to reboot into bootloader mode so new firmware can be programmed. If you change your mind, just unplug the box from USB for a few seconds.

SB – Scan I^2C Bus

Scans the I2C bus for devices and prints the 8 bit addresses of any it finds.

? – Display help

Displays a list of the available commands.

NOTE: This list is context sensitive and will not show commands the are not available, e.g. there will be no SD card commands shown if there is no SD card plugged in.