Introduction

VoltBot is an innovative DC power supply which has communication interfaces accessed by both wire and wireless ways.

We defined this protocol for three purposes:

1. Guarantee data correction and integrity.
2. Clarify the timing, throughput, and ability of VoltBot.
3. Minimize the requirement for user's knowledge so that they are able to do simplest programming.

Terminology

Operators, data type, and data format mentioned in this document apply to C programming language.

Charset in string is UTF-8, which totally compatible to ASCII when only English characters involved.

Connection

Wire

Wire means using an external data cable and connecting it into UART port on side of VoltBot.

UART has the same timing with RS232 protocol, but different voltage level. UART in VoltBot adopts 3.3V as logic '1' and 0V as logic '0'. It means:

1. If connect VoltBot with Arduino UNO board, you need to use a 5V-to-3.3V convertor.
2. If connect VoltBot with RS232 port on PLC or PC, you need to use a MAX3232 chip/module to convert the RS232 signal to 3.3V signal.

By this way, commands will be loaded on controller's TX pin and be received on VoltBot's RX pin. Meanwhile, responses will appear on VoltBot's TX pin, and be received by controller's RX pin.

Wireless

Wireless means using VoltBot IP address and UDP sockets as communication carrier.

In this case, VoltBot always listens on its 3358 port, and replies to sender's 3359 port on sender's IP address. In order to receive the response, the sender should open and listen on its 3359 port prior than sending any command. Otherwise the response will be dropped.

If either of controller & VoltBot does not possess a public IP address, Port forwarding for VoltBot's 3358 port or for controller's 3359 port might be essential.

Timing

Timing information for the VoltBot may be considered at three levels:

- Timing related to byte/command transfers
- General system timing
- Communications at Power-Up

For UART communication, the byte expects a frame size of 8 bits, no parity, one stop bit, and a baud rate of 115200.

When controller initiate one command, in normal situation, VoltBot will receive and reply it in 100ms. If it didn't reply in 500ms, you can initiate the command again.

For UDP communication, the response time will be 100ms+Downstream Delay+ Upstream

Delay. These Delay usually be tens of microsecond, but they would go significantly large when the wireless signal is bad.

We recommend that the controller shall wait for at least 3 second for the response, before it initiates another retry.

For both ways of communication, time intervals between two command should be larger than 500ms. Commands too close might get mixed up and dropped by VoltBot.

One command will always get one response as confirmation that the command is received and the link is alive. But receiving responses is not essential for operation.

During power-up, commands are not available until the disappearing of logo screen.

Protocol

Command

Format:

<start byte> <command> <length low 8bits> <length high 8bit> <payload> <parity> <end byte>

Analyze:

Start byte	**Command**	**Length**	**Length**
0xAA	CMD	N&0xff	N>>8
Byte 1	**Byte 2**	**Byte …**	**Byte N**
0x01	0x02	0x03	0x04
Parity	**End byte**		
PRT	0x0E		

Start: constant 0xAA,

CMD: lookup next section,

N: 4~65535 determined by next section,

Byte 1 ~ Byte N: the actual payload,

PRT: XOR results of all bytes in payload,

End: constant 0x0E.

Constrains:

Length N must be at least 4.

Response

Format: same with command.

Analyze: CMD is the same value in command.

Constrains: no constrains, Length N can be any value, 0~65535.

Examples

Disable Sound
Send text: 0xaa45040000000000000e CMD=0x45, N=0x0004, Payload={0x00,0x00,0x00,0x00}, PRT=0x00
Reply text: 0xaa450000000e CMD=0x45, N=0x0000, Payload={NONE}, PRT=0x00

Enable Sound
Send text: 0xaa45040001000000010e CMD=0x45, N=0x0004, Payload={0x01,0x00,0x00,0x00}, PRT=0x01
Reply text: 0xaa450000000e CMD=0x45, N=0x0000, Payload={NONE}, PRT=0x00

Read Voltage on Channel 3
Send text: 0xaab0040002000000020e CMD=0xb0, N=0x0004, Payload={0x02,0x00,0x00,0x00}, PRT=0x02 0x02 represents channel 3 0x00 represents voltage 0x00, 0x00 placeholder
Reply text: 0xaab002004402460e CMD=0xb0, N=0x0002, Payload={0x44,0x02}, PRT=0x46 The Voltage on Channel 3: (0x44+(0x02<<8))/100=5.80V

Command

Command	Read Protocol Version		
Added	2018/12/29	CMD	0x00
Returns a string that describe protocol version supported by VoltBot.			

Command	Enable/Disable Channel		
Added	2018/1/1	CMD	0x40
Byte 1 defines Channel: Value 0~3 represents Channel 1~4. Byte 2 defines On/Off, value 1=On, value 0=Off			

Command	Channel Mode		
Added	2018/1/1	CMD	0x41
Byte 1 defines Channel, value 0~3 Byte 2 defines Mode: Value 0=charge mode, value 1=DC source mode Byte 3,4 defines voltage setting for DC source mode, no used in charge mode. The value should be a 16bit integer in little endian, ranging from 250~1250, defining voltage from 2.5V~12.5V. Byte 5,6 defines voltage setting for DC source mode, no used in charge mode. The value should be a 16bit integer in little endian, ranging from 5~400, defining voltage from 0.05A~4V. Example: 0xaa400400010108020a0e will set Channel 2 into DC source mode, and voltage setting will be 5.2V.			

Command	Charger Mode		
Added	2018/1/1	CMD	0x43
Byte 1 defines Channel, value 0~3 Byte 2 defines Mode: Value 0=quickcharge off, value 1=quickcharge on. Example: 0xaa41040001000000010e will set Channel 2 into charger mode, and 0xaa43040001000000010e will set quickcharge on.			

Command	Set Backlight		
Added	2018/1/1	CMD	0x42
Byte 1 defines Backlight Mode: Value 0=auto mode, value 1=manual mode Byte 2 defines intensity in manual mode: value 0=weakest, value 10=brightest.			

Command	Set Unique ID		
Added	2018/1/1	CMD	0x44
Byte 1 defines unique ID: value 0=unique ID canceled, value 1~99=activate ID.			

Command	Set Sound		
Added	2018/1/1	CMD	0x45
Byte 1 defines on/off: value 0=sound off, value 1=sound on.			

Command	Read Voltage/Current		
Added	2018/1/1	CMD	0xB0

Byte 1 defines Channel, value 0~3 Byte 2 defines Type: value 0=voltage, value 1=current. Returns the average value of last second, data format will be a 16bit little endian unsigned integer. Example: refer to last paragraph of the section above.

Command	Read Voltage/Current Curve		
Added	2018/1/1	CMD	0xB1
Byte 1 defines Channel, value 0~3 Byte 2 defines Type: value 0=voltage, value 1=current. Byte 3 defines Period: Value 0=curve in a second, value 1=curve in a minute, Value 2=curve in a hour, value 3=curve in a day Returns a data array group by 16bit little endian unsigned integers, the first integer represent the latest data multiplied by 100.			

Command	Read Channel On/Off		
Added	2018/11/27	CMD	0xB5
Returns four bytes describing on/off status of every channel			

Command	Read Channel Mode/Type		
Added	2018/11/27	CMD	0xB6
Returns 24 bytes: 4 bytes describe channel mode of every channel, value 0=charger mode, value 1=DC source mode, value 2=Current source mode. 4 bytes describe charge mode of every channel, value 0=non-quickcharge, value 1=quickcharge. 8 bytes describe voltage settings of every channel. 8 bytes describe current settings of every channel.			

Command	Read Unique ID		
Added	2018/11/27	CMD	0xB7
Returns 1 bytes: Value 0xff=ID canceled, value 1~99=the active ID			

Command	Read IP in External WiFi		
Added	2018/12/29	CMD	0xB8
Returns a string describing the IP address. Value 0.0.0.0 means no IP got yet.			

Command	Read Start-up Time		
Added	2018/12/31	CMD	0xB9
Returns 8 byte describing the milliseconds since last power up. All 8 byte comprises a 64bit little endian unsigned integer.			

Technique Support

Email: xiongzechong@sillycomm.com