AUTOSPORT LABS

Megajolt Lite Jr. Controller API Documentation
Version 4.x.x

MEGAJOLT LITE JR.
PROGRAMMABLE IGNITION CONTROLLER

http://www.autosportlabs.net

Revision 1: 10 Jan 2007
Revision 2: 11 Jan 2007 (version 3.2.x)
Revision 3: 09 Sept 2008 (Version 4.x)

Table of Contents

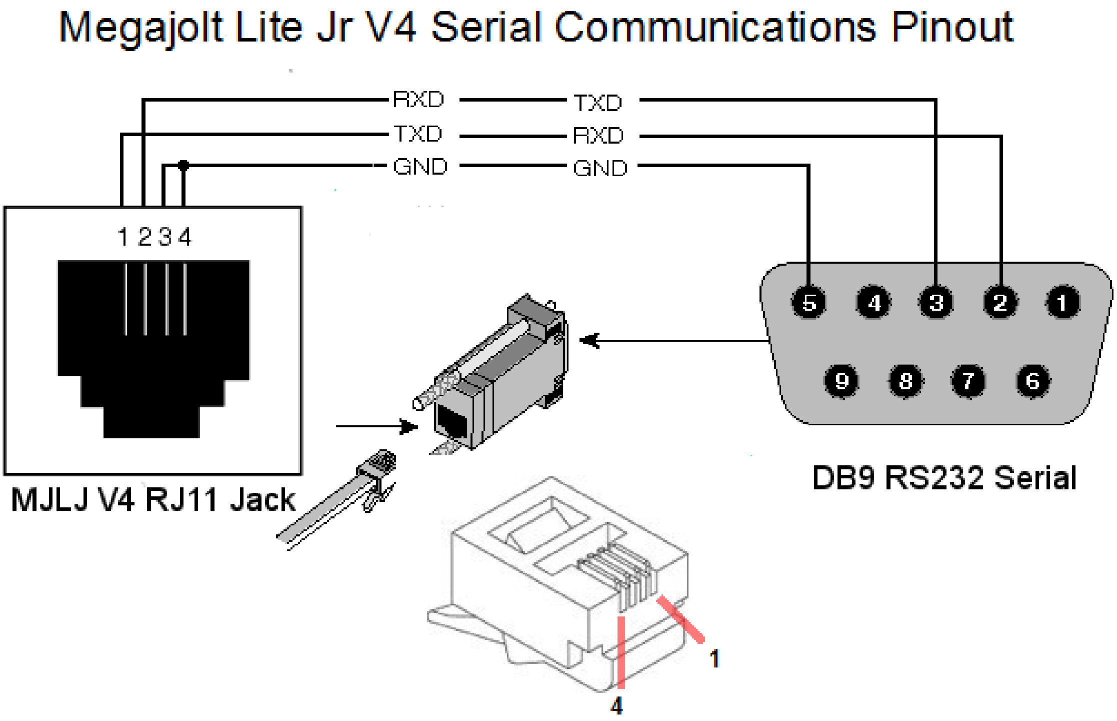
1 Overview

This document specifies the communication protocol used for the MegaJolt Lite Jr. (MJLJ) Ignition Controller.

1.1 Communications

Serial Pinout

The MJLJ Controller provides a standard RS232 Serial interface via case-mounted RJ11 modular jack.



Port Configuration

The port configuration commands are described in Table 1.

Baud Rate	38400
Data Bits	8
Parity	None
Stop bits	1
Flow Control	None

Table 1: Serial Port Specifications

1.2 Protocol Description

The MJLJ Ignition Controller employs a simple command-response sequence to perform various operations upon the controller.

Commands are coded with a single character and, if applicable, a stream of data which flows to or from the controller. Figure 1 illustrates a command which updates a single cell in the ignition table:

Byte	1	2	3
Data	0x75 (ASCII 'u')	0x11	0x14
Description	Command	RPM / Load Bin	Ignition advance value

Figure 1: Example Update Cell Command

2 API Reference

2.1 Get Version

The Get Version command returns the current firmware version of the controller. The version indicates both hardware and API versions.

Byte	Description
1	character 'V' : ASCII 0x56

Table 2: Get Version Command

Byte	Description
1	Major version
2	Minor version
3	Bugfix version

Table 3: Get Version Response

2.2 Get State

The Get State command returns the current runtime state of the controller.

Byte	Description	Type
1	character 'S' : ASCII 0x53	unsigned byte

Table 6: Get State Command

Byte	Description	Type
1	Current ignition advance (degrees BTDC)	unsigned byte
2	Raw RPM Count, high byte	unsigned byte
3	Raw RPM Count, low byte	unsigned byte
4	Current RPM / Load Bins High 4 bits: RPM Bin Low 4 bits: Load Bin	unsigned byte
5	Current Load Value (KPa or TPS%)	unsigned byte

6	Controller State Bit 0: User Output 1 state (set=active; clear=inactive) Bit 1: User Output 2 state (set=active; clear=inactive) Bit 2: User Output 3 state (set=active; clear=inactive) Bit 3: User Output 4 state (set=active; clear=inactive) Bit 4: Rev Limit state (set=active; clear=inactive) Bit 5: Shift Light state (set=active; clear=inactive) Bit 6: Reserved Bit 7: Currently active ignition configuration (set=Configuration 1; clear=Configuration 2)	unsigned byte
7	Auxiliary input value (scaled)	unsigned byte
8	Current Advance Correction Bin	unsigned byte
9	Current Advance Correction (degrees)	signed byte

Table 7: Get State Response

The raw RPM count represents the number of microseconds elapsed between PIP signals emitted by the EDIS module. The actual RPM can be calculated using the following formula:

```
RPM = 60 * (1/((rpmTicks/1000000)*cylinder))
where:
cylinder = 2 for 4 cylinder engines
cylinder = 3 for 6 cylinder engines
cylinder = 4 for 8 cylinder engines
rpmTicks = 16 bit value-  High Byte:Low Byte
```

2.3 Get Ignition Configuration

The Get Ignition Configuration returns the currently active Ignition Configuration in the controller's RAM memory.

Byte	Description	Type
1	character 'C' : ASCII 0x43	unsigned byte

Table 8: Get Ignition Configuration Command

Byte	Description	Type
1 - 10	RPM Bin values, from low to high Encoding: RPM value / 100	unsigned byte
11 - 20	Load Bin values, from low to high Encoding: KPa (0 – 255) or TPS% (0 - 100)	unsigned byte
21 - 120	Ignition Map Encoding: Ignition advance values (0 – 59 degrees BTDC) organized in a 100 element, 10 x 10 array in Row–major order. (Row == Load axis)	unsigned byte
121	User Output Types User output types are encoded in a two bit number, with 4 configurations packed into a single byte. Value: 1 = RPM; 2 = Load; 3 = Aux Input Bit 0-1: Output 1 Type Bit 2-3: Output 2 Type Bit 4-5: Output 2 Type Bit 6-7: Output 3 Type	unsigned byte
122	User Output Mode Configurations Defines the operational mode for each output. Normal = Output is activated when configured threshold is reached Inverted = Output is de-activated when configured threshold is reached Bit 0: Output 1 Mode (set = Normal; clear = Inverted) Bit 1: Output 2 Mode (set = Normal; clear = Inverted) Bit 2: Output 3 Mode (set = Normal; clear = Inverted) Bit 3: Output 4 Mode (set = Normal; clear = Inverted) Bit 4: Reserved Bit 5: Reserved	unsigned byte

	Bit 6: Reserved Bit 7: Reserved	
123	User Output 1 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
124	User Output 2 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
125	User Output 3 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
126	User Output 4 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
127	Rev Limit threshold value Encoding: RPM (x100): 0 – 100	unsigned byte
128	Shift Light threshold value Encoding: RPM (x100): 0 – 100	unsigned byte
129-138	Advance Correction Bins, from load to high Encoding: Aux Input scaled input value	unsigned byte
139-148	Advance Correction Values Encoding: ignition advance (degrees)	signed byte
149-150	Auxiliary Input Peak Hold Decay Encoding: Number of ignition events to hold peak Aux	unsigned short

	Input value (0-65535)	

Table 9: Get Ignition Configuration Response

2.4 Update Ignition Configuration

The Update Ignition Configuration command loads new ignition configuration values to the controller's RAM. There is no response from the controller upon command completion. .

Byte	Description	Type
1	Character 'U': ASCII 0x55	unsigned byte
2 - 11	RPM Bin values, from low to high Encoding: RPM value / 100	unsigned byte
12 - 21	Load Bin values, from low to high Encoding: KPa (0 – 255) or TPS% (0 - 100)	unsigned byte
22 - 121	Ignition Map Encoding: Ignition advance values (0 – 59 degrees BTDC) organized in a 100 element, 10 x 10 array in Row–major order. (Row == Load axis)	unsigned byte
122	User Output Types Encoding: User output types are encoded in a two bit number, with 4 configurations packed into a single byte. Value: 1 = RPM; 2 = Load; 3 = Aux Input Bit 0-1: Output 1 Type Bit 2-3: Output 2 Type Bit 4-5: Output 2 Type Bit 6-7: Output 3 Type	unsigned byte
123	User Output Mode Configurations Defines the operational mode for each output. Encoding: Normal = Output is activated when configured threshold is reached Inverted = Output is de-activated when configured threshold is reached	unsigned byte

	Bit 0: Output 1 Mode (set = Normal; clear = Inverted) Bit 1: Output 2 Mode (set = Normal; clear = Inverted) Bit 2: Output 3 Mode (set = Normal; clear = Inverted) Bit 3: Output 4 Mode (set = Normal; clear = Inverted) Bit 4: Reserved Bit 5: Reserved Bit 6: Reserved Bit 7: Reserved	
124	User Output 1 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
125	User Output 2 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
126	User Output 3 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
127	User Output 4 threshold value Encoding: Load, KPa : 0 – 255 Load, TPS%: 0 – 100 RPM (x100): 0 – 100 Aux Input (scaled): 0 – 255	unsigned byte
128	Rev Limit threshold value Encoding: RPM (x100): 0 – 100	unsigned byte

129	Shift Light threshold value Encoding: RPM (x100): 0 – 100	unsigned byte
130-139	Advance Correction Bins, from load to high Encoding: Aux Input scaled input value	unsigned byte
140-149	Advance Correction Values Encoding: Ignition advance (degrees)	signed byte
150-151	Auxiliary Input Peak Hold Decay Encoding: Number of ignition events to hold peak Aux Input value (0-65535)	unsigned short

Table 11: Update Ignition Configuration Command

2.5 Update Ignition Cell

The Update Ignition Cell command updates a single cell in the ignition table. There is no response from the controller upon command completion.

Byte	Description	Type
1	Character 'u' : ASCII 0x75	unsigned byte
2	RPM / Load Bin Encoding: high 4 bits: RPM Bin low 4 bits: Load Bin	unsigned byte
3	New ignition advance value (0 – 59 degrees)	unsigned byte

Table 12: Update Ignition Cell Command

2.6 Write Ignition Configuration to Flash

The Write Ignition Configuration to Flash command copies the ignition configuration values in controller RAM to the non-volatile storage in the controller's flash memory. There is no response from the controller upon command completion.

If the switchable ignition configuration option is enabled, the controller will write Ignition Configuration 1 when the Option port on the controller is logical low, or Configuration 2 when the Option port is logical high. If the switchable ignition configuration option is disabled, the controller will default to Configuration 1.

Byte	Description	Type
1	Character 'W' : ASCII 0x57	unsigned byte

Table 13: Write Ignition Configuration to Flash

2.7 Get Load Calibration

The Get Load Calibration command retrieves the calibration data used for the internal MAP sensor or TPS input, depending on controller configuration.

Byte	Description	Type
1	Character 'l' : ASCII 0x6C	unsigned byte

Table 14: Get Load Calibration Command

Byte	Description	Type
1-256	Load calibration data. Encoding:	unsigned byte

	The raw analog-digital converter from the MAP or TPS sensor value is mapped to the value specified in this table. This mapped, or scaled, value is used to select the appropriate Load bin during ignition advance calculations.	

Table 15: Get Load Calibration Response

2.8 Update Load Calibration

The Update Load Calibration Command writes a new load axis calibration table directly to the controller's flash memory.

During command processing, ignition advance calculation and communication with the EDIS module is not active.

Byte	Description	Type
1	Character 'L' : ASCII 0x4C	unsigned byte
2-257	Load calibration data. Encoding: The raw analog-digital converter from the MAP or TPS sensor value is mapped to the value specified in this table. This mapped, or scaled, value is used to select the appropriate Load bin. **Note, for this segment, a minimum 150ms transmission delay is required every 32 bytes.**	unsigned byte

Table 16: Update Load Calibration command

2.9 Read Load ADC

The Read Load ADC command retrieves a raw analog-digital reading

Byte	Description	Type
1	Character 'a' : ASCII 0x61	unsigned byte

Table 17: Read Load ADC command

Byte	Description	Type
1	The raw ADC load axis value. Encoding:	unsigned byte

	0 – 255 value linearly scaled from 0 – 5v	

Table 18: Read Load ADC response

2.10 Get Auxiliary Calibration

The Get Auxiliary Calibration command retrieves the calibration data used to convert, or scale, the raw value read from the Auxiliary input to a more meaningful value, such as Degrees C or Ratio of air to fuel.

Byte	Description	Type
1	Character 'x' : ASCII 0x78	unsigned byte

Table 14: Get Auxiliary Calibration Command

Byte	Description	Type
1-256	Auxiliary calibration data. Encoding: The raw analog-digital converter from the Auxiliary Input is mapped to the value specified in this table. This mapped, or scaled, value is used to select the appropriate Advance Correction Bin during ignition advance calculations.	unsigned byte

Table 15: Get Auxiliary Calibration Response

2.11 Update Auxiliary Calibration

The Update Auxiliary Calibration Command writes a new auxiliary calibration table directly to the controller's flash memory.

During command processing, ignition advance calculation and communication with the EDIS module is not active.

Byte	Description	Type
1	Character 'X' : ASCII 0x58	unsigned byte
2-257	Auxiliary calibration data. Encoding: The raw analog-digital converter from the Auxiliary Input is mapped to the value specified in this table. This mapped, or scaled, value is used to select the appropriate Advance Correction Bin during ignition advance calculations.	unsigned byte

	Note, for this segment, a minimum 150ms transmission delay is required every 32 bytes.	

Table 16: Update Auxiliary Calibration command

2.12 Get Global Configuration

The Get Global Configuration Command retrieves global controller configuration values. These values contain options and configuration data which are not part of the standard ignition configuration. These values are read from non-volatile flash memory.

Byte	Description	Type
1	Character 'g' : ASCII 0x67	unsigned byte

Table 19: Get Global Configuration command

Byte	Description	Type
1	Number of cylinders Encoding: 4 = 4 cylinder (EDIS-4) 6 = 6 cylinder (EDIS-6) 8 = 8 cylinder (EDIS-8)	unsigned byte
2	PIP Noise filter level Encoding: 0 = no filtering 255 = maximum filtering	unsigned byte
3	Cranking Advance Encoding: Ignition advance, degrees BTDC Valid: 0-59	unsigned byte
4	Trigger wheel offset Encoding: Valid: -5 to +5 (degrees)	signed byte
5-64	Reserved for future use	unsigned byte

Table 20: Get Global Configuration Response

2.13 Update Global Configuration

The Update Global Configuration command updates the global controller configuration values. These values contain options and configuration data which are not part of the standard ignition configuration. This command writes the values directly to non-volatile flash memory.

During command processing, ignition advance calculation and communication with the EDIS module is not active.

Note: a 150ms delay is required every 32 bytes, starting with the beginning of the Global Configuration data section (byte 2)

Byte	Description	Type
1	Character 'G': ASCII 0x47	unsigned byte
2	Number of cylinders Encoding: 4 = 4 cylinder (EDIS-4) 6 = 6 cylinder (EDIS-6) 8 = 8 cylinder (EDIS-8)	unsigned byte
3	PIP Noise filter level Encoding: 0 = no filtering 255 = maximum filtering	unsigned byte
4	Cranking Advance Encoding: Valid: 0-59 (degrees BTDC)	unsigned byte
5	Trigger wheel offset Encoding: Valid: -5 to +5 (degrees)	signed byte
6-65	Reserved for future use	unsigned byte

Table 21: Update Global Configuration Command

3 Revision History

Revision 1: Initial document version
Revision 2: Corrected various grammatical mistakes; added missing description for Get/Update Global Configuration Data commands; clarified various sections.
Revision 3: Updates for V4 controller