T&C Power Conversion

AJA 13.56 MHz RF Power Supply Digital Interface Hardware Information and Protocol

Author: Paul Walker

Creation Date: 12/20/18

Version 1.00 - 12/20/18

1.0 TERMS

The following terms will be used consistently throughout this document and should be understood.

HOST	A personal computer or PLC.
DEVICE	The AJA RF Power Supply.
INTERFACE	The physical connection (or link) between the host and device.
PROTOCOL	The format and details of the information transferred over the interface.
xxh	This represents an 8-bit hex value where x can be any digit from 0 to F.
xxxxh	This represents a 16-bit hex value where x can be any digit from 0 to F.
'xx'	This represents a string of two ASCII characters that represent a single 16-bit word. (for instance 'A1' would equal 4131h)

2.0 HARDWARE

The digital interface allows for connection between the device and some type of host controller (usually a PC).

The digital interface on the device is configured for standard RS-232 serial communications.

The communications parameters are fixed at **38400** bps, **8** data bits, **NO** parity, and **1** stop bit.
No hardware handshaking signals are used.

The device is configured as **DCE** on a single DB9F connector. The pin usage on the DB9F is shown below.

PIN 2	RF Power Supply TX
PIN 3	RF Power Supply RX
PIN 5	Signal Ground

It is expected for the host to be configured as DTE on a DB9M connector. In this case a standard DB9 M/F straight-through cable can be used between the host and the device.

3.0 PROTOCOL OVERVIEW

The communications protocol between the host and device is a simple command/response interaction that uses binary data. Communications are initiated by the host sending a 'command' message to the device, and in turn the device sends back a 'response' message if required. A single command/response cycle is called a transaction.

All transactions are initiated by the host sending a command message. Once the device receives and processes the command message, it will always respond with an acknowledgement code (ACK or NACK) to inform the host of the validity of the command.

If for a particular command requires a response message from the device, the response message will be sent by the device immediately after the ACK code is sent. Typical transactions are shown below.

A transaction where the device receives a valid command, but there is no response data:

HOST: COMMAND
DEVICE: ACK

A transaction where the device receives a valid command and there is response data:

HOST: COMMAND
DEVICE: ACK
DEVICE: RESPONSE

A transaction where the device did not receive the command correctly:

HOST: COMMAND
DEVICE: NACK

In this case it doesn't matter if additional response data was expected for the particular command. Since the command was not considered valid by the device, nothing else will be transmitted by the device for this transaction.

The acknowledgement code is a single data byte. ACK (**2Ah**) means the command was received correctly and is considered valid. NACK (**3Fh**) means the command was not recognized by the device, one or more data parameters was out of range, or the particular command was not allowed at this time.

The host will always know whether or not to expect additional response data based upon the command it sends. Each command specifically defines whether or not there will be response data, and exactly what the response data will be. Therefore there is no confusion for the host as to what to expect after sending a specific command.

The host is responsible for pacing and synchronizing the communications between itself and the device. This is done by the host ensuring to never start a new transaction until the previous transaction completes. In other words, the host should not send a new command until the device ACKs/NACKs the last command and sends all of the response data when required.

The exception to the above rule is if a "timeout" occurs. If the host sends a command to the device, and the device doesn't respond with ACK or NACK in the specified time period; then the host may consider that transaction ended and try again if desired.

4.0 PROTOCOL DETAILS

4.1 MESSAGE STRUCTURE

As described in the previous section, there are two types of messages (or packets) used in this protocol. The first type is the COMMAND message (which is always sent from the host), and the second type is the RESPONSE message (which is always sent by the device). The ACK/NACK code that comes from the device is not part of either of these two messages. It is a special code that separates the two messages.

4.1.1 THE COMMAND MESSAGE

The COMMAND message is a consistent fixed-length message that provides a unique action code (ID) and up two 16-bit data parameters for that action. The COMMAND message also contains a device address for when multiple devices are connected on an RS422/485 bus. The COMMAND message completes with a 16-bit checksum to help verify message integrity. Below is the format of the COMMAND message.

HEAD	ADDR	CMDID	PARM1	PARM2	CKSUM
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes	2 Bytes

HEAD This is a fixed 1-byte message header (43h) that starts every COMMAND message

ADDR When multiple devices are connected on an interface structure that supports a multi-drop bus (such as RS-485), this field is used to choose which device the host wants to speak to. Units on a multi-drop bus are numbered from 01h to 3Fh. 00h is a special address used to broadcast a message to all devices on the bus simultaneously. ***Presently the address field is ignored, and the device will listen to all messages.***

CMDID This is a 2-byte command ID that tells the device what specific action is being requested by the host.

PARM1 This is the first of two 16-bit parameters (data) for the specified command.

PARM2 This is the second of two 16-bit parameters (data) for the specified command.

CKSUM This is a simple 16-bit additive checksum used to help validate the message quality. The checksum is calculated by adding together each **8-bit BYTE** of the message; starting with the HEAD byte and ending with the last byte of the PARM2 field. The 16-bit result of this addition is the CKSUM value to be transmitted.

All 16-bit fields (2-byte fields) are to be transmitted with high-byte first and low-byte last.

Not every command requires PARM1 or PARM2, but these fields must always be included as part of the message. It is suggested that unused PARM fields be filled with 0000h.

4.1.2 THE RESPONSE MESSAGE

The RESPONSE message is a variable length message that provides information from the device in response to a command issued by the host. The length and meaning of the RESPONSE message depends upon the command that was issued. Although the meaning of each RESPONSE message is dynamic, the format of the message is consistent. Below is the general format of the RESPONSE message.

HEAD	ADDR	LENGTH	DATA	CKSUM
1 Byte	1 Byte	2 Bytes	VARIABLE	2 Bytes

HEAD This is a fixed 1-byte message header (52h) that starts every RESPONSE message

ADDR This is the address of the device sending back the RESPONSE message. It should always be the same value as what was transmitted in the original COMMAND message since only the addressed unit should be responding. Units on a multi-drop bus are numbered from 01h to 3Fh.
Presently the value of this field will always be 0.

LENGTH This is a 16-bit (2-byte) value that defines how many bytes are in the DATA field.

DATA This is the variable length data response for the specific command.
(see the details of each command for a description of this data)

CKSUM This is a simple 16-bit additive checksum used to help validate the message quality. The checksum is calculated by adding together each **8-bit BYTE** of the message; starting with the HEAD byte and ending with the last byte of the DATA field. The 16-bit result of this addition is the CKSUM value to be transmitted.

All 16-bit fields (2-byte fields) are to be transmitted with high-byte first and low-byte last.

4.2 TIMING

The following timing rules are required for the protocol.

The maximum message transmission time for any message (COMMAND messages from the host or RESPONSE messages from the device) is 500 ms. Once a message header (HEAD) is received, it is expected that all of the remaining bytes of the message will be received within 500 ms of the HEAD byte. If the message is not completely received within this time frame, the receiver may assume the initial incoming data was not a real message.

The device is guaranteed to ACK/NACK a COMMAND message within 200 ms after the end of message reception. If the device does not ACK/NAK within this time frame, the host may assume that there is a problem with the device, or the device is not connected.

If a particular COMMAND message requires a RESPONSE message from the device, it is guaranteed that the device will start to send the RESPONSE message within 200 ms after sending the ACK/NACK byte. If the device does not start to send the RESPONSE message within this time frame, the host may assume there is a problem with the device, or the device is not connected.

If any of these timeout conditions occur when the host tries to send a COMMAND message to the device, the host should wait at least 500 ms before trying again to send a new message to the device.

If the proper transaction protocol is followed (ie. send a message, wait for ACK/NACK, and then wait for a response if one is required); the device will be capable of immediate “back-to-back” transactions with no delays between messages for a small burst of transactions (up to 10 transactions maximum).

If constant polling is required by the host, it is not recommended to do continuous transactions without some time delay between groups of transactions. If constant polling is required by the host, it is recommended to allow at least 100ms between each burst (or group) of transactions so as to not overload the device.

5.0 STANDARD COMMANDS

This section provides information on the standard COMMAND messages that the host may send to the device. The tables provide all of the command details, acceptable parameters, and expected RESPONSE message if one exists.

5.1 BASIC ACTION COMMANDS

These commands preform a specific action or function on the device.

Command	REQ CONTROL	CMDID 4243h 'BC'
Description	Request or release control of the device.	
PARM1	5555h to request control, any other value to release control.	
PARM2	<none>	
Response	[STATUS] STATUS is a 16-bit value. If the command was to request control, STATUS = 0 means the request was denied and the host does not have control of the device. STATUS = 1 means access was granted and the host now has control of the device. If the command was to release control, then STATUS will always be 0.	

Notes about requesting and maintaining control of the generator.

The host may use GET commands at any time to retrieve information from the device whether it has control of the device or not. However, for the host to use any of the SET commands, it must first request (and be granted) control of the device using the 'BC' command as described above.

The device may or may not allow the host to have control-access depending upon various factors. If the host is denied control-access it may try again later. However, once the host is granted control-access, it may assume it has control until the host releases control.

While the host has control of the unit, the generator front panel will be inoperable. The front panel will only display operating information while the host is in control. Front panel operation will be restored once the host releases control.

While the host has control of the generator, it will expect messages to be sent periodically from the host. If the device doesn't receive any messages for more than a 2 second period, it will assume the host has disconnected and revert back to front panel operation. At this point host control will be lost and can only be regained with a new request. Polling the generator status once per second with the 'GS' command is a good way to maintain a constant communication with the generator to satisfy this requirement.

Command	PING	CMDID 4250h 'BP'
Description	Simple check for communications with the device.	
PARM1	<none>	
PARM2	<none>	
Response	<none>	

Command	RF ON/OFF	CMDID 4252h 'BR'
Description	Enables or disables the RF power for any operating mode.	
PARM1	5555h for ON, any other value means OFF.	
PARM2	<none>	
Response	<none>	

5.2 SET COMMANDS

These commands set operating parameters on the device. These commands are only valid when the host has control on the device. Control-access is requested by the host via the 'BC' command. If the host sends one of these SET commands while not in control, the device will respond with NACK and the function will not be executed.

SET commands that specify power allow for parameter values that may exceed the limits of the specific model under control. This is because the protocol was designed for a wide range of device models. If the host sends a parameter value in one of these commands that is beyond a limit for a specific model (or beyond a user-set limit), the device will not respond with a NACK. The device will correctly accept the command and simply limit the setting to the MAX or MIN value allowed for that model.

Command	SET POWER LEVEL	CMDID 5341h 'SA'
Description	Sets the set-point power for all operating modes.	
PARM1	Value from 0 to 4000 that specifies a power level in watts. Even though you can send any value up to 4000 watts; if this value exceeds the internal device limits, the set-point power will be set to the device limit.	
PARM2	<none>	
Response	<none>	

Command	SET AIO SCALE	CMDID 5349h 'SI'
Description	Sets the input/output scaling for the analog interface.	
PARM1	Value from 1000 to 10000 that specifies the full-scale input and output voltage in mV. Example: a value of 5000 would provide a full-scale voltage of 5.00V.	
PARM2	<none>	
Response	<none>	

Command	SET OPERATING MODE	CMDID 534Fh 'SO'
Description	Sets the operating modes (NORMAL OR RAMP)	
PARM1	1 = NORMAL 2 = <INVALID> 3 = <INVALID> 4 = RAMP	
PARM2	<none>	
Response	<none>	

NOTE: The 'SO' command will not only change the operating mode of the device, but it will also reset the device to a default state with RF Power OFF.

Command	SET RF SOURCE	CMDID 5353h 'SS'
Description	Sets the RF source (INTERNAL or EXTERNAL) for NORMAL operating mode.	
PARM1	1 = INTERNAL SOURCE 2 = EXTERNAL SOURCE	
PARM2	<none>	
Response	<none>	

Command	SET USER LIMIT	CMDID 5355h 'SU'
Description	Sets one of the user power-limits for ALL operating modes.	
PARM1	1 = FORWARD POWER LIMIT 2 = REVERSE POWER LIMIT	
PARM2	Value from 0 to 4000 that specifies a power level in watts. Even though you can send any value up to 4000 watts; if this value exceeds the internal device limits, the user-limit will be set to the device limit.	
Response	<none>	

Command	SET RAMP START POWER	CMDID 5250h 'RP'
Description	Sets the start-power level for RAMP mode.	
PARM1	Value from 1 to 4000 that sets the ramp start-power in watts. The start-power is the initial power level that will be set at each RF OFF/ON transition while in RAMP mode.	
PARM2	<none>	
Response	<none>	

Command	SET RAMP RATE	CMDID 5252h 'RR'
Description	Sets the ramp rate for when in RAMP mode.	
PARM1	Value from 1 to 99 that sets the ramp rate in watts per second. The ramping rate is used at all times when changing power levels in RAMP mode and RF power is enabled.	
PARM2	<none>	
Response	<none>	

Command	SET TUNER CAP POSITIONS	CMDID 5443h 'TC'
Description	Sets the position for one of the Tuner capacitors when the Tuner is in manual-tune mode.	
PARM1	Selects which capacitor to adjust. 1 = LC POSITION 2 = TC POSITION	
PARM2	16-bit integer value from 0 to 100 that represent the position as a percent of range.	
Response	<none>	

NOTE: The 'TC' command is only allowed when the Tuner is set to MANUAL tune mode.
If this command is given while in AUTO tune mode, the Generator will respond with NACK.

Command	SET TUNER MODE	CMDID 544Dh 'TM'
Description	Sets the operating mode for the Tuner (AUTO MANUAL).	
PARM1	1 = AUTO 2 = MANUAL	
PARM2	<none>	
Response	<none>	

5.3 GET COMMANDS

These commands retrieve various types of information from the device, and they may be used at any time whether or not the host has control-access of then device.

Command	GET FREQUENCY	CMDID 4746h 'GF'
Description	Retrieves the present operating frequency.	
PARM1	<none>	
PARM2	<none>	
Response	[FRQH] [FRQL] **FRQH** is a 16-bit value that represents the HI WORD of a 32-bit frequency value. **FRQL** is a 16-bit value that represents the LO WORD of a 32-bit frequency value.	

Command	GET POWER SET POINT LEVEL	CMDID 474Ch 'GL'
Description	Retrieves the present set point level from the unit.	
PARM1	<none>	
PARM2	<none>	
Response	[SETP] **SETP** is a 16-bit value that represents a power level in tenths of watts. For example 3295 would equal 329.5 watts.	

Command	GET POWER READINGS	CMDID 4750h 'GP'
Description	Retrieves the present true-power readings from the unit.	
PARM1	<none>	
PARM2	<none>	
Response	[FORWARD POWER] [REVERSE POWER] [LOAD POWER] Each value is a 16-bit number that represents a power level in tenths of watts. For example 3295 would equal 329.5 watts.	

Command	GET RAMP PARAMETERS	CMDID 4752h 'GR'
Description	Retrieves all of the RAMP mode settings.	
PARM1	<none>	
PARM2	<none>	
Response	[START] [RATE] **START** is a 16-bit value that represents the ramp starting power level in watts. **RATE** is a 16-bit value that represents the ramp rate in watts/second.	

Command	GET GEN STATUS	CMDID 4753h 'GS'
Description	Retrieves the present operating status of the Generator.	
PARM1	<none>	
PARM2	<none>	
Response	[STATUS] [TEMP] [OPMODE] [TUNER] **STATUS** is a 16-bit field that provides various status information about the unit. B15: 0 B07: UNUSED B14: ANALOG INTERFACE ENABLED B06: UNUSED B13: UNUSED B05: UNUSED B12: UNUSED B04: EXTERNAL RF SOURCE ACTIVE B11: INTERLOCK OPEN B03: UNUSED B10: OVER TEMPERATURE B02: UNUSED B09: REVERSE POWER LIMIT B01: UNUSED B08: FORWARD POWER LIMIT B00: RF POWER ON **TEMP** is a 16-bit value that represents the heat sink temperature in tenths of degrees Celsius. For example 482 would equal 48.2° C. **OPMODE** is a 16-bit value that represents the current operating mode. 1 = NORMAL, 2 =<INVALID>, 3 = <INVALID>, 4 = RAMP **TUNER** is a 16-bit value that represents the type of Tuning available. 1 = NONE, 2 = AFT GENERATOR, 3 = ANALOG TUNER, 4 = DIGITAL TUNER	

Command	GET TUNER STATUS	CMDID 4754h 'GT'
Description	Retrieves the present operating status of the Tuner unit.	
PARM1	<none>	
PARM2	<none>	
Response	[STATUS] [LC POS] [TC POS] [VDC] [PRESET] **STATUS** is a 16-bit field that provides various status information about the unit. B15: 0 — B07: TUNE CAP AT UPPER LIMIT B14: DIGITAL TUNER — B06: TUNE CAP AT LOWER LIMIT B13: UNUSED — B05: LOAD CAP AT UPPER LIMIT B12: UNUSED — B04: LOAD CAP AT LOWER LIMIT B11: UNUSED — B03: UNUSED B10: UNUSED — B02: UNUSED B09: UNUSED — B01: MANUAL MOVE IN PROCESS B08: UNUSED — B00: MANUAL MODE **LC POS** is a 16-bit value that represents the present LOAD cap position as a percentage of full-scale. The value is an integer from 0 to 1000 that represents percentage in tenths of percent (eg. 455 = 45.5%) **TC POS** is a 16-bit value that represents the present TUNE cap position as a percentage of full-scale. The value is an integer from 0 to 1000 that represents percentage in tenths of percent (eg. 455 = 45.5%) **VDC** is a 16-bit value that represents the chamber DC voltage (in volts). **PRESET** is a 16-bit value that represents the currently selected (active) preset location. ***This value has no meaning and can be ignored.***	

Command	GET FIRMWARE VERSIONS	CMDID 4766h 'Gf'
Description	Retrieves the system's firmware versions.	
PARM1	<none>	
PARM2	<none>	
Response	[UIP MAJOR] [UIP MINOR] [RFP MAJOR] [RFP MINOR] **UIP MAJOR** is an 8-bit value that represents the major portion of the UI processor firmware. **UIP MINOR** is an 8-bit value that represents the minor portion of the UI processor firmware. **RFP MAJOR** is an 8-bit value that represents the major portion of the RF processor firmware. **RFP MINOR** is an 8-bit value that represents the minor portion of the RF processor firmware. Firmware version numbers should be interpreted and displayed as MAJOR.MINOR	

Command	GET ID STRINGS	CMDID 4769h 'Gi'
Description	Retrieves one of the system ID strings.	
PARM1	Value from 1 to 2 that defines which ID string to retrieve. 1 = UNIT NAME STRING 2 = UNIT SERIAL NUMBER STRING	
PARM2	<none>	
Response	[TAG] [STRING] **TAG** is a 16-bit value that represents which ID string is sent (this will match PARM1 from the command). **STRING** is the data string (14 bytes long). The first 13 bytes are printable characters, and the last byte will always be 00h to terminate the string.	