Solid State Variac

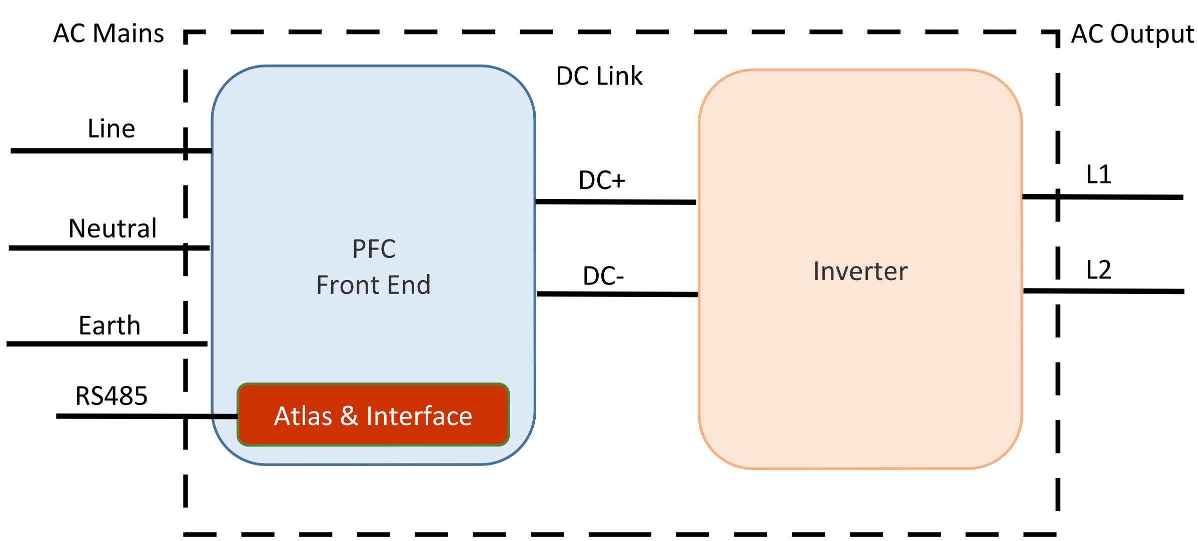
Operation Manual v3.1

Solid State Variac Theory of Operation

The Solid State Variac (SSV) is a compact, lightweight, digitally controlled variable AC power supply. It can supply AC voltage with variable voltage magnitude and variable frequency for regulated microgrids, transformer or CT excitation circuits, or test equipment. The SSV can replace standard much heavier rotary variacs in most applications. The SSV has the following specifications:

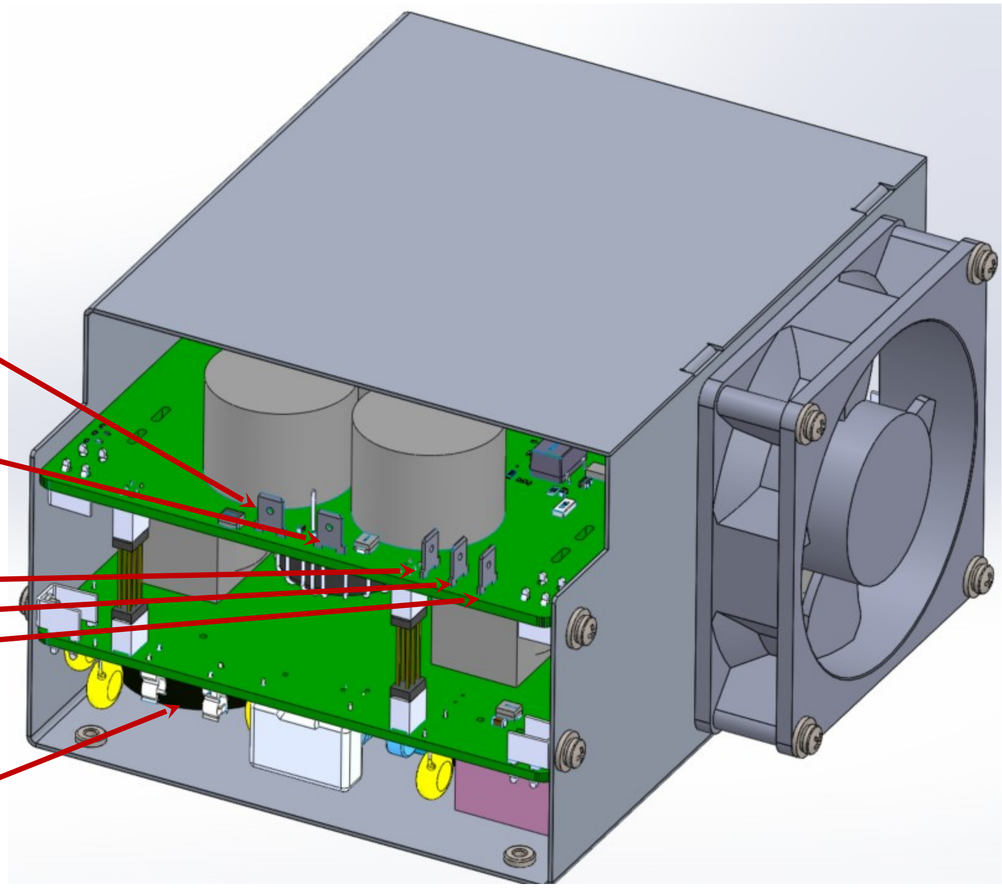
Power	3.5kVA
Current Output	$26A_{RMS}$
Peak Instantaneous Current	45A
Output Voltage	0-$135V_{AC}$ Variable
Output Frequency	0-400Hz <500W output, 50-60 Hz matching input frequency >500W output
Maximum Input Power	1500W
Input Voltage	104 – $260V_{AC}$
Undervoltage Fault	$90V_{RMS}$ In
Operational Ambient Temperature	-20 to 50°C

Simplified Block Diagram:



Universal 104V_{AC} to 260V_{AC} mains voltage enters the Power Factor Correction (PFC) front end where it is converted to a DC link of 210V_{DC} via an interleaved buck-boost stage. Additionally, the PFC front end provides fusing, surge suppression, and filtering per the system requirements. The generated DC link is fed to the Inverter where a three-level neutral point clamped (NPC) inverter chops and filters the DC to form the desired AC output voltage. The software and firmware control systems actuating the inverter switching devices as well as performing higher level processes are programmed into the Atlas Control Board. The Atlas Control Board resides on top of an Interface board which conditions sensor feedback, provides low voltage bias power, and provides communications pathways. Inverter commands are received over a full duplex RS485 communications link.

Solid State Variac Hardware & Connections Diagram



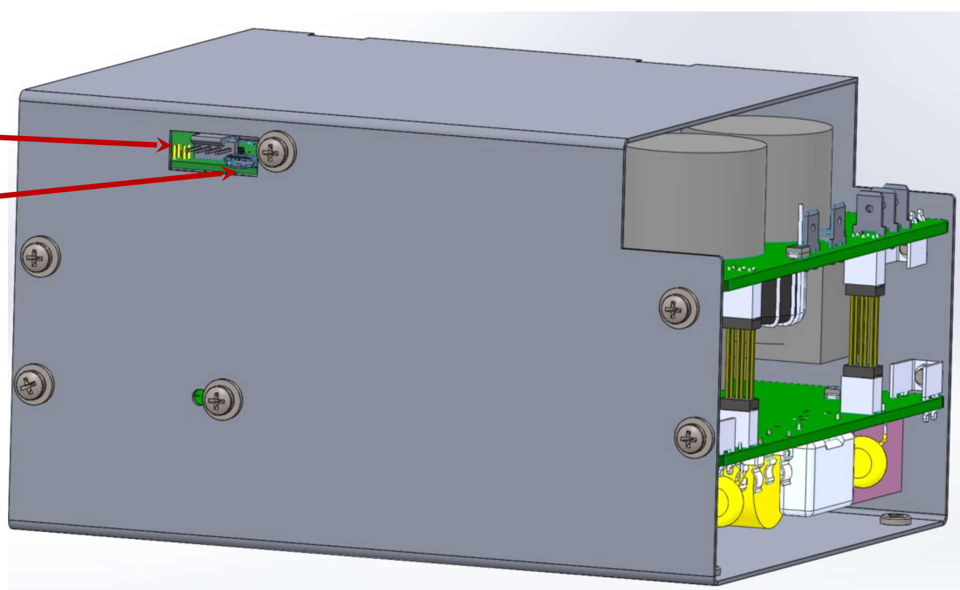
*All AC Terminals

V_{AC} Output1

V_{AC} Output2

V_{AC} Input L1, L2, Gnd

Mains Fuse: 15A, Slow Blow 5x20mm



RS485

USB Debug and Comms

Solid State Variac Communication Specification

Overview:

Communication to the Solid State Variac (SSV) is over a full duplex RS485 link. The SSV acts as a slave. It will only respond to messages from the master. It will never asynchronously transmit a message to the master. Messages are either of the 'set' or 'get' format. The master either sets a parameter on the SSV, or it gets the value of parameter.

The SSV will always respond to a command, whether it understands the intent or not. For 'set' commands, it echoes the command that it received. For 'get' commands, it echoes the command except with the data value altered to contain the desired information. For commands it cannot parse, it responds with a standard 'do not understand' message.

Serial Specifications

- RS485, full duplex
- 57600 baud
- 8 bit characters
- No parity
- 1 stop bit
- No flow control

Note: Commands may be tested using a PC with a terminal emulator such as Hyperterminal or TeraTerm and the FTDI RS422 to USB Converter Cable Part Number: USB-RS422-WE-1800-BT. Appropriate macros may be written to quickly test the various messages.

It is intended that the output voltage be controlled in “PWM counts” in an open loop fashion. The range of PWM counts is from 0 to 1000. For example, setting the PWM count to 500 sets the modulation to 50%. The resulting output voltage will be a function of the load impedance and the modulation set point.

Alternatively, the SSV can be put into closed-loop output voltage control mode. In this mode, the SSV will vary its switching pattern to actively control the output AC voltage to a preset or commanded level.

Generalized Message Format:

All messages will have the following ASCII format:

`CNNNNCSCS<CR>`

where:

C	Single letter command character which specifies the command type
NNNN	Numeric value associated with the command. Must have at least one digit.
CSCS	Checksum based on Fletcher Checksum algorithm (Appendix D).
<CR>	CR or Carriage Return character acts as the end of message identifier

Command Summary Table

		Example	
Letter	Command Description	SSV Receives	SSV Transmits
S	Get **S**tate	S0Axxxx	S000xxxx
F	Clear **F**ault	F0xxxx	F000xxxx
R	Set **R**un State	R1xxxx	R1xxxx
I	Get Current Output	I0xxxx	I555xxxx
O	Set Normal Res Voltage **O**utput	O350xxxx	O350xxxx
V	Get **V**oltage Output	V0xxxx	V0xxxx
Z	Set Enable Hi-Res Mode	Z1xxxx	Z1xxxx
H	Set **H**igh Res Voltage Output	H1000xxxx	H1000xxxx
L	Get **L**ine Frequency	L601xxxx	L601xxxx
P	Set Line Frequency	P554xxxx	P554xxxx
X	Get Software Version	X0xxxx	X124xxxx
Y	Get Firmware Version	Y0xxxx	Y111xxxx
N	Load New Software Version	N0xxxx	N/A
Response to Unknown Command: ?0E1A xxxx – Appended Checksum			

Command Get State:

Use this command to get the current state of the solid state variac.

SSV Receives: S0xxxx
SSV Transmits: SNFFMxxxx

Where the first digit in “NFFM” is a number from 0 to 7 which corresponds to the current state as described in the table below:

State	Description
0	Idle
1	AC Contactor Closing
2	Enabling PFC
3	Waiting for DC Link Voltage to Stabilize
4	Energized (DC Link Stabilized, No Output Voltage Generation)
5	Running (Output Voltage Produced)
6	Returning to Idle State
7	Engineering Debug State (for 3L Engineers)

The middle two digits in “NFFM” are the two-digit fault code (see Appendix B).

Example:

SSV Receives: `S0xxxx`
SSV Transmits: `S0000xxxx`

FF = 00 which indicates the SSV is in the idle state.

The last digit in “NFFM” is the operational mode of the SSV

M	Mode Description
0	Normal Resolution Mode
1	High Resolution Mode
2	Transitional State Between Normal and High Resolution

Normal Resolution Mode: The DC link is at its maximal level and only “O” commands may be used to command the output from maximal current to lower current levels.

High Resolution Mode: When High Resolution mode is enabled (using Z command), when sweeping the output current down through 1A, the DC link is brought down such that the AC modulation is renormalized to about 80% thusly increasing the resolution at lower current levels. Upon this renormalization, the system will be in the High Resolution Mode. During this mode “H” commands are used to modulate the voltage output with the increased resolution. ‘O’ commands may also be issued to return to Normal Resolution mode. The system must again be swept down through 1A to return to High Resolution Mode.

Transitional State: During the transition from Normal Resolution Mode to High Resolution Mode, new “H” and “O” commands are not respected. The SSV will still respond to other pertinent commands such as S commands.

Command Set Enable Hi-Res Mode:

Use this command to enable High Resolution mode. Upon receiving a “Z” command, the SSV will echo back the “Z” state. High Resolution mode is enabled by default on a power cycle. Enabling High Resolution mode allows for the renormalization of the modulation index by lowering the DC link voltage. When enabled, the transition from Normal Resolution mode to High Resolution mode happens automatically as the current passes through 1A. It is important to note that enabling High Resolution mode does not put the system in High Resolution mode, it merely *allows* for the renormalization to happen when appropriate. The renormalization happens as the output is swept down from a high output to a low output through the 1A threshold.

Z	Description
0	High Resolution NOT Enabled
1	High Resolution Enabled

Example

SSV Receives: `Z0xxxx`

SSV Transmits: `Z0xxxx`

Command Clear Fault:

Use this command to clear the solid state variac faults.

Example:

SSV Receives: `F0xxxx`

SSV Transmits: `F000xxxx`

The numeric field NNN in the response indicates the most recent fault in the fault queue. In the example NNN = 000 and there are no faults in the fault queue.

Command Set Run State:

Use this command to set the solid state variac’s internal run state variable to either TRUE (1) or FALSE (0). This command is used to turn on/off the output voltage. Note that in order to produce an output voltage, the run state must be set to 1 and the output voltage must be set to a non-zero number.

Example:

SSV Receives: `R1xxxx`

SSV Transmits: `R1xxxx`

The SSV is commanded to the run state and echos the run state variable.

Command Get Current:

Use this command to get the RMS output current of the SSV.

SSV Receives: `I0xxxx`
SSV Transmits: `INNNNxxxx`

The 'NNNN' is the RMS current in Amps multiplied by 10. For example, if 'NNNN' is 55 then the SSV is reporting $5.5A_{RMS}$. The lowest reported increment of current is 0.1A.

Command Set Voltage Output:

Use this command to set the output voltage based on PWM counts with normal resolution. Note that the output voltage is set by the PWM count number in an open loop fashion and the actual output voltage will vary based on load impedance. The number of PWM counts can be set from 0 to 1000.

Example:

SSV Receives: `O350xxxx`
SSV Transmits: `O350xxxx`

The SSV is commanded with a PWM count of 350. The SSV echos the PWM count value of 350.

Command Set High Resolution Voltage Output:

Use this command to set the output voltage based on PWM counts when in high resolution mode. The Get State 'S' command should be used to poll the resolution state to know if one is in the high resolution state. The SSV will echo back the H command.

SSV Receives: `H800xxxx`
SSV Transmits: `H800xxxx`

Command Get Voltage Output:

Use this command to get the measured solid state variac output voltage in volts.

Example:

SSV Receives: `V0xxxx`
SSV Transmits: `V0xxxx`

The SSV is commanded with a get voltage command. The SSV responds with a numeric field of "0" indicating there is 0V present at the output.

Command Get Line Frequency:

Use this command to get the mains line frequency.

SSV Receives: `L0xxxx`
SSV Transmits: `LNNNxxxx`

The digits ‘NNN’ are the line frequency in Hz multiplied by 10. For example, if ‘NNN’ is 605 then the SSV is reporting a line frequency of 60.5Hz.

Command Set Line Frequency:

Use this command to set the AC output frequency of the solid state variac. Note this command is only available when in High Resolution Mode (i.e. the current is below 1A). The output frequency may ± 6Hz from the nominal 60Hz with 0.1Hz resolution. The SSV will echo back the commanded frequency

SSV Receives: `PNNNxxxx`
SSV Transmits: `PNNNxxxx`

The digits ‘NNN’ are the line frequency in Hz multiplied by 10. For example, if ‘NNN’ is 555 then the SSV is commanded to output AC voltage at 55.5Hz.

Command Get Software Version:

Use this command to get the software version that is currently loaded in the Atlas DSP. The SSV numeric field response will be three digits e.g. ABC indicating version A.BC.

Example:
SSV Receives: `X0xxxx`
SSV Transmits: `X124xxxx`

The SSV receives the get software version command and responds with the number field “124” indicating software version 1.24.

Command Get Firmware Version:

Use this command to get the firmware version that is currently loaded in the Atlas FPGA. The SSV numeric field response will be three digits e.g. ABC indicating version A.BC.

Example:
SSV Receives: `Y0xxxx`
SSV Transmits: `Y111xxxx`

The SSV receives the get firmware version command and responds with the number field “111” indicating software version 1.11.

Command Load New Software Version:

Use this command to begin the process of loading new software onto the SSV. Begin by issuing the command:

SSV Receives: `N0xxxx`
SSV Transmits: `N/A`

The SSV will the reboot itself and enter into the bootloader mode over the period of a few seconds. The file may then be transferred via the XMODEM protocol at the 56700 Baud. Once loaded, the SSV will burn the image to flash and then reboot itself. The SSV may be polled using the get state command to determine when it is ready to operate again after the new code has been loaded.

Appendix A: System States

State	Description
0	Idle
1	AC Contactor Closing (No Longer Used)
2	Enabling PFC
3	Waiting for DC Link Voltage to Stabilize
4	Energized (DC Link Stabilized, No Output Voltage Generation)
5	Running (Output Voltage Produced)
6	Returning to Idle State
7	Engineering Debug State (for 3L Engineers)

Appendix B: Fault Codes

#	Name	Description
0	FAULT_DC_BUS_TIMEOUT	DC Bus Not Settled in Time
1	FAULT_WATCHDOG	Processor Watchdog Timer Trip
2	FAULT_OC_IOUT	Overcurrent on I_{AC} Out
3	FAULT_OC_IPWM	Overcurrent on PWM Current
4	FAULT_VDC_TOO_HIGH	Internal DC Link Voltage Too High
5	FAULT_VDC_TOO_LOW	Internal DC Link Voltage Too Low
6	FAULT_HI_TEMP_INVERTER	Inverter Board Over Temperature
7	FAULT_HI_TEMP_PFC	PFC Board Over Temperature
8	FAULT_FREQ_TOO_HIGH	Mains Frequency Too High
9	FAULT_FREQ_TOO_LOW	Mains Frequency Too Low
10	FAULT_OV_VACIN	Overvoltage on V_{AC} Input
11	FAULT_OV_VACOUT	Overvoltage on V_{AC} Output
12	FAULT_OV_VDCP	Overvoltage on V_{DC} Positive to Neutral
13	FAULT_OV_VDCN	Overvoltage on V_{DC} Negative to Neutral
14	FAULT_DC_BUS_UNBALANCED	Internal Three Level Bus Unbalanced
15	FAULT_PLL_LOST	Mains PLL Not Locked
16	FAULT_FPGA_BELLYUP	FPGA not Responding to DSP
17	FAULT_NO_HEARTBEAT	Heartbeat not Echoing
18	FAULT_ADC_TIMEOUT	Analog to Digital Converter Response Error
19	FAULT_BAD_SOFTWARE	Loaded Software is not Expected Software
20	FAULT_0020	Unused Fault
21	FAULT_BAD_FLASH_MEMORY	Error in The Flash Memory
22	FAULT_VREF_OOR	Reference Voltage Out of Expected Range
23	FAULT_V5DIV2_OOR	5V Bus Out of Expected Range
24	FAULT_FAULT_HST_CORRUPT	Fault History Error
25	FAULT_PARAM_CORRUPT	Settable Parameters Corrupt

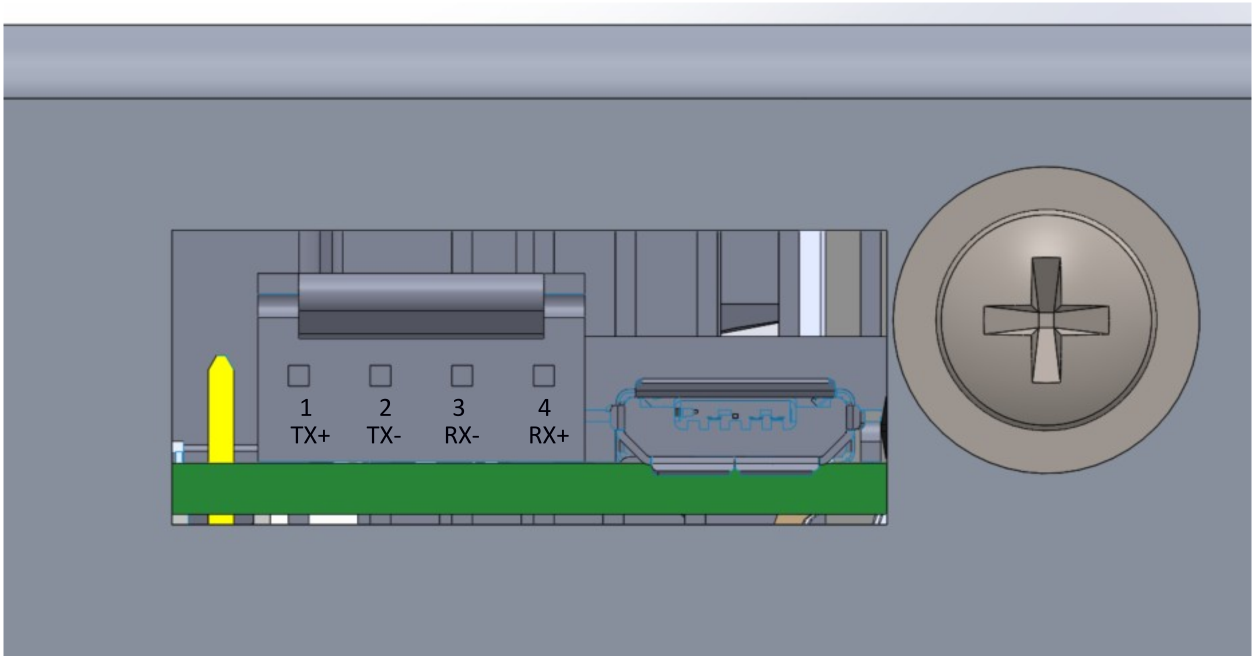
26	FAULT_CALIB_CORRUPT	Calibration Error
27	FAULT_BAD_PWM_FREQUENCY	Inverter Devices Switching Frequency Error
28	FAULT_VDCP_TOO_HIGH	VDC+ to Neutral Point Too High
29	FAULT_VDCN_TOO_HIGH	VDC- to Neutral Point Too High
30	FAULT_VDCP_TOO_LOW	VDC+ to Neutral Point Too Low
31	FAULT_VDCN_TOO_LOW	VDC- to Neutral Point Too Low

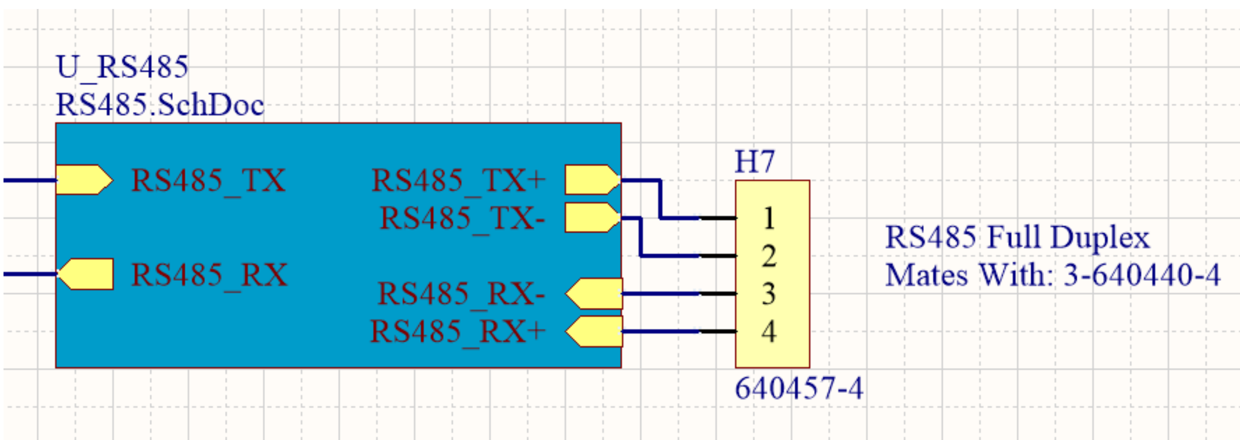
Appendix C: Fletcher Checksum Code

Wikipedia Link: https://en.wikipedia.org/wiki/Fletcher%27s_checksum

```
/*  Note:
unsigned char variables are 8 bits
unsigned short variables are 16 bits
*/

unsigned short Fletcher16( unsigned char *data, int count )
{
  unsigned char sum1 = 0;
  unsigned char sum2 = 0;
  int index;
  unsigned short finalSum;
  unsigned short finalCheckSum;

  for( index = 0; index < count; ++index ) {
     sum1 = (sum1 + data[index]) % 255;
     sum2 = (sum2 + sum1) % 255;
  }

  finalSum = (sum2 << 8) | sum1;

  {
     unsigned char f0 = finalSum & 0xff;
     unsigned char f1 = (finalSum >> 8) & 0xff;
     unsigned char c0 = 0xff - ((f0 + f1) % 0xff);
     unsigned char c1 = 0xff - ((f0 + c0) % 0xff);

     finalCheckSum = (c0 << 8) | c1;
  }

return finalCheckSum;
}
```

Appendix D: RS485 / 422 Communication Hardware Information





- Pin 1 and 2, TX+ and TX- are driven outputs of the Interface Board
- Pin 3 and 4, RX- and RX+ are inputs as seen by the Interface Board

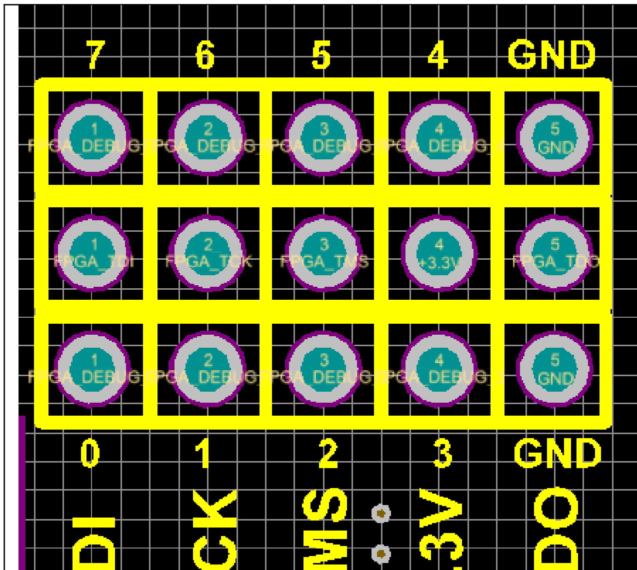
If using the FTDI RS422 to USB Converter Cable Part Number: USB-RS422-WE-1800-BT for testing, use the following connection table:

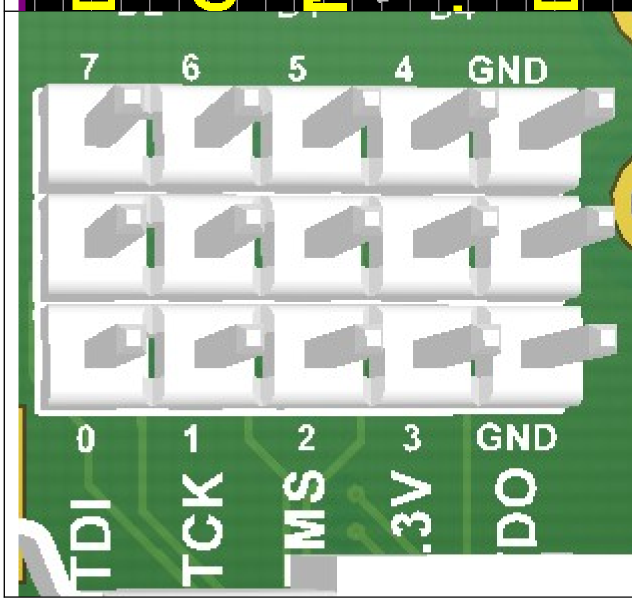
USB-RS422-WE				SSV INTERFACE BOARD		
ORANGE	TXD+	Output	→	PIN 4	RX+	Input
RED	TXD-	Output	→	PIN 3	RX-	Input
YELLOW	RXD +	Input	→	PIN 1	TX+	Output

WHITE	RXD-	Input	→	PIN 2	TX-	Output

Note: The extra wires on the USB-RS422-WE-1800-BT are not used in this application.
Note: There is a permanent 120Ω termination resistor across the RX lines on the interface board. The master should also have a 120Ω termination resistor across its RX lines as close to the receiver as possible.

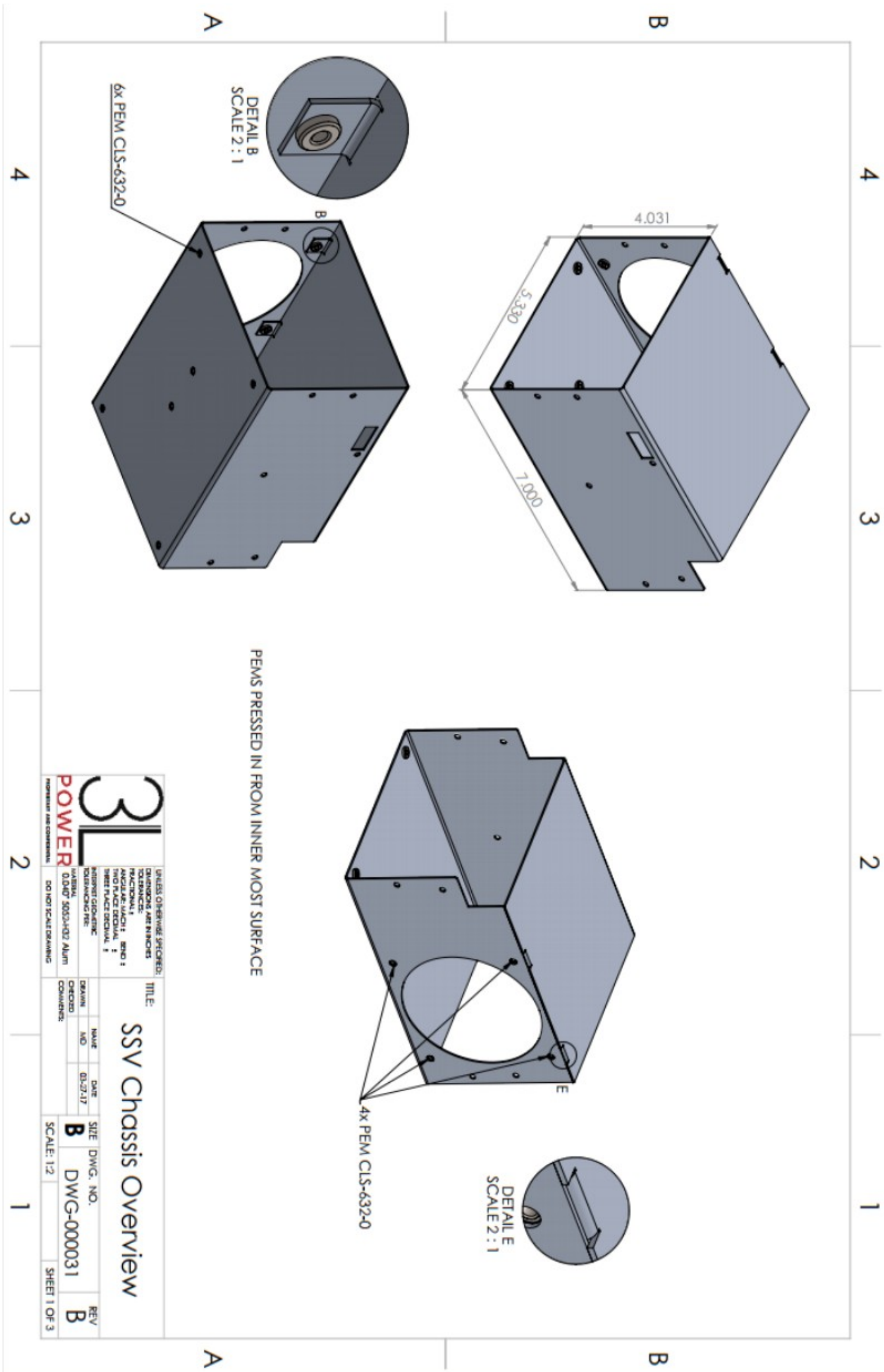
Appendix E: JTAG for FPGA

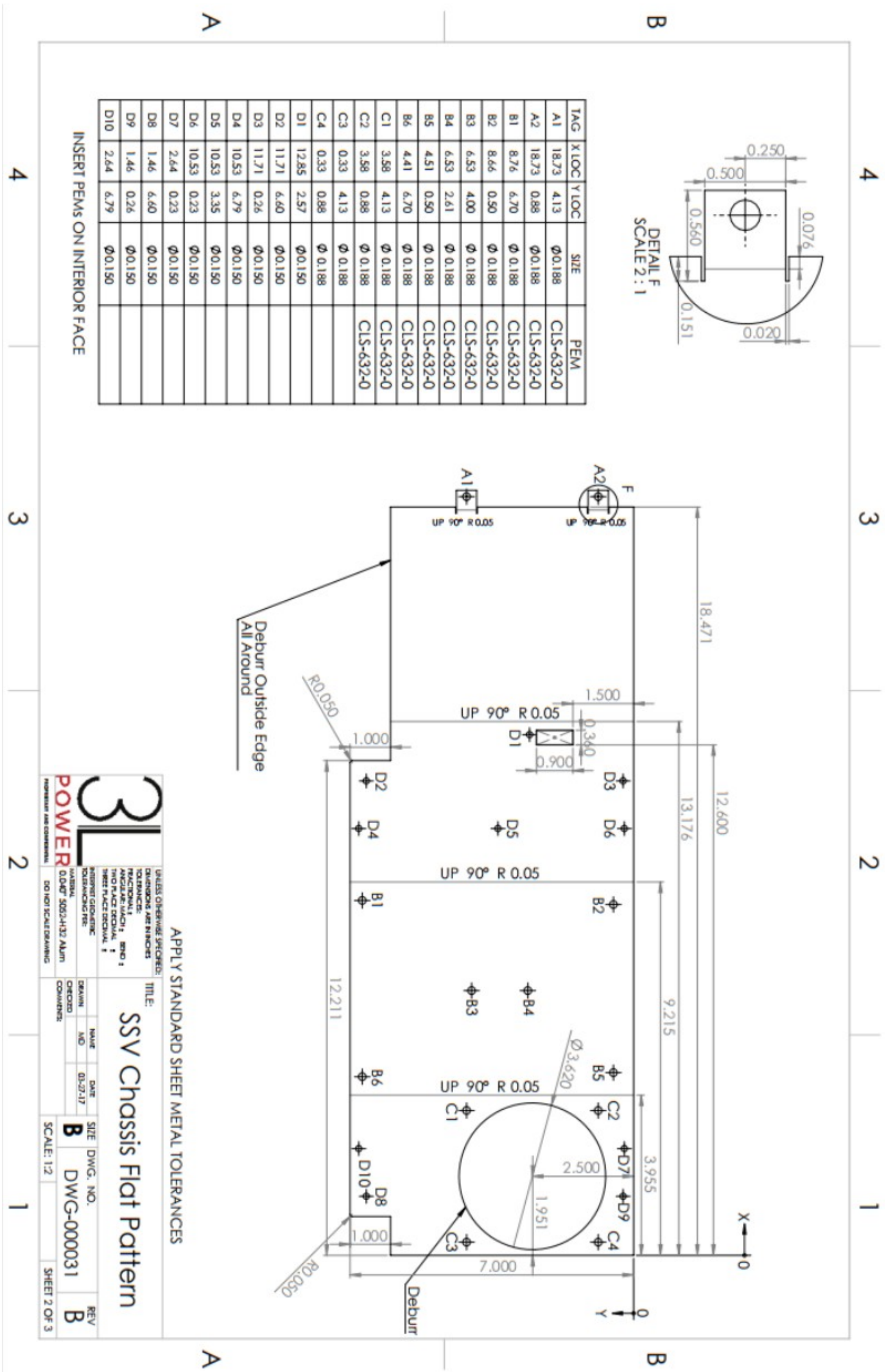




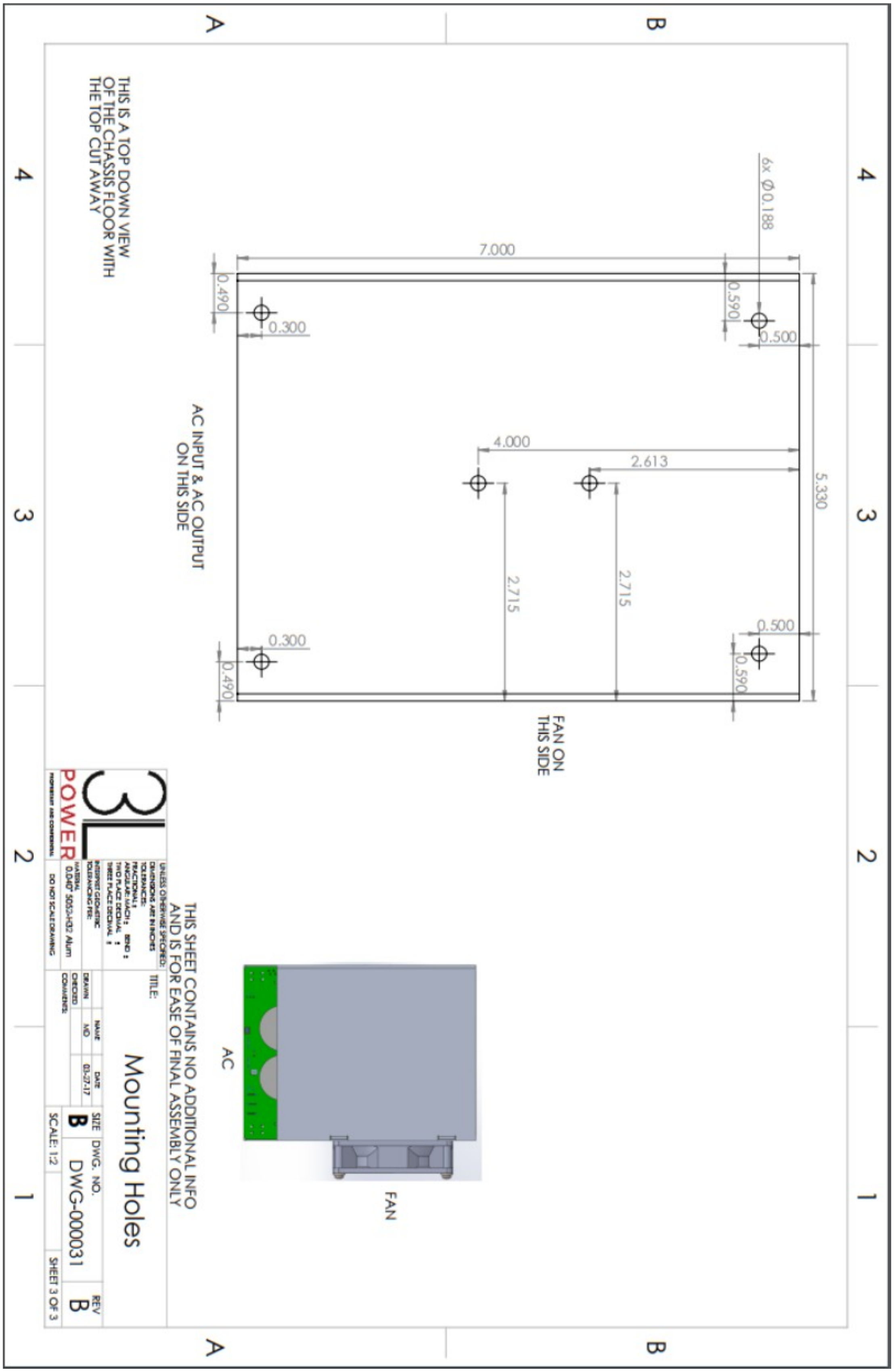
DEBUG 7	DEBUG6	DEBUG5	DEBUG4	GND
TDI	TCK	TMS	3.3V	TDO
DEBUG 0	DEBUG 1	DEBUG 2	DEBUG 3	GND

Appendix F: Dimensioned Drawing





TAG	X LOC	Y LOC	SIZE	PEM
A1	18.73	4.13	Ø0.188	CLS-632-0
A2	18.73	0.88	Ø0.188	CLS-632-0
B1	8.76	6.70	Ø0.188	CLS-632-0
B2	8.66	0.50	Ø0.188	CLS-632-0
B3	6.53	4.00	Ø0.188	CLS-632-0
B4	6.53	2.61	Ø0.188	CLS-632-0
B5	4.51	0.50	Ø0.188	CLS-632-0
B6	4.41	6.70	Ø0.188	CLS-632-0
C1	3.58	4.13	Ø0.188	CLS-632-0
C2	3.58	0.88	Ø0.188	CLS-632-0
C3	0.33	4.13	Ø0.188	
C4	0.33	0.88	Ø0.188	
D1	12.85	2.57	Ø0.150	
D2	11.71	6.60	Ø0.150	
D3	11.71	0.26	Ø0.150	
D4	10.53	6.79	Ø0.150	
D5	10.53	3.35	Ø0.150	
D6	10.53	0.23	Ø0.150	
D7	2.64	0.23	Ø0.150	
D8	1.46	6.60	Ø0.150	
D9	1.46	0.26	Ø0.150	
D10	2.64	6.79	Ø0.150	


THIS IS A TOP DOWN VIEW
OF THE CHASSIS FLOOR WITH
THE TOP CUT AWAY
6x Ø0.188
7.000
0.490
0.300
0.590
0.500
4.000
2.613
5.330
2.715
2.715
0.300
0.490
0.500
0.590
AC INPUT & AC OUTPUT
ON THIS SIDE
FAN ON
THIS SIDE
AC
FAN
THIS SHEET CONTAINS NO ADDITIONAL INFO
AND IS FOR EASE OF FINAL ASSEMBLY ONLY
3L
POWER
TITLE:
Mounting Holes
MATERIAL
0.040" 5052-H32 Alum
DO NOT SCALE DRAWING
DRAWN
CHECKED
COMMENTS:
NAME
NO
DATE
03-27-17
SIZE
B
DWG. NO.
DWG-000031
REV
B
SCALE: 1:2
SHEET 3 OF 3

Revision History	
v1.0	07/26/2016: Document originated
v1.1	07/26/2016: Fixed typos and copy paste errors. Added detailed section on RS485 hardware.
v1.2	08/01/2016: Added description of "open loop PWM commands". Updated commands for variable length numeric field. Added Teraterm Macro Appendix F.
v1.3	Added available communication bauds
v1.4	Major overhaul of command set. Updated fault list. Removed appendices with obsolete information.
v2.0	Updated Input Voltage for Universal 104V_{AC} to 260V_{AC} Updated Hardware Connection Diagram for SSV 2.0 Updated RS485 Connector in Appendix D Updated Appendix E JTAG for FPGA Added Appendix F Chassis Dimensioned Drawing
v3.0	10/09/2017 Added new commands (Z,H,I,L,P,N) Added description of the high resolution mode Changed the checksums everywhere to xxxx for futureproofing Removed the Baud command Updated dimensioned drawings to Revision B models
V3.1	1/4/2020 Updates for public release Description of uses New address Update to reflect new specifications