PGAD-067
07152002A

ADC1000-USB A/D Converter:

Serial Port Interface Communications and Control Information

Offices: **Ocean Optics, Inc.**
380 Main Street, Dunedin, Fla., USA
727.733.2447 • 727.733.3962 Fax
8:30 a.m.-6 p.m. EST

Ocean Optics B.V. (Europe)
Nieuwgraaf 108 G, 6921 RK DUIVEN, The Netherlands
31-(0)26-3190500 • 31-(0)26-3190505 Fax

E-mail: **Info@OceanOptics.com (general sales inquiries)**
Info@OceanOpticsBV.com (for sales questions in Europe)
Orders@OceanOptics.com (questions about orders)
TechSupport@OceanOptics.com (for technical support)

Ocean Optics Inc.
First in Photonics™

Table of Contents

Overview

The ADC1000-USB A/D Converter is a microcontroller-based A/D card that can communicate via the Universal Serial Bus or Serial Bus (RS-232). This document contains the necessary command information for controlling the unit via the RS-232 interface. For driver-level USB control information, refer to the OOIWinIP Windows Interface Package.

Hardware Description

The ADC1000-USB utilizes a Cypress AN2131 microcontroller, which has a high speed 8051, combined with an USB ASIC. Program code and data coefficients are stored in external E^2PROM and are loaded at boot-up via the I^2C bus. The microcontroller has 8K of internal SRAM and 128K of external SRAM. Due to memory mapping and paging constraints, only 96K of the external SRAM is useable.

Instruction Set

Command Syntax

The list of the commands and the microcode version number used when each was introduced are shown in the table on the following page. All commands consist of an ASCII character passed over the serial port, followed by some data. The length of the data depends on the command. The format for the data is either ASCII or binary (default). The ASCII mode is set with the "a" command and the binary mode is set with the "b" command. To insure accurate communications, all commands respond with an ACK (ASCII 6) for an acceptable command or a NAK (ASCII 21) for an unacceptable command (i.e. data value specified out of range).

In the ASCII data value mode, the ADC1000-USB "echo's" the command back out of the RS-232 port. The ADC1000-USB generates a prompt ">" that is displayed at the beginning of each line to indicate it is ready to accept a command. This prompt is generated in ASCII mode only.

In binary mode, all data, except where noted, passes as 16-bit unsigned integers (WORDs) with the MSB followed by the LSB. By issuing the "v" command (microcode version number query), the data mode can be determined by viewing the response (ASCII or binary).

In a typical data acquisition session, the user sends commands to implement the desired spectral acquisition parameters (integration time, A/D Channel, etc.). Then the user sends commands to acquire spectra ("S" command) with the previously set parameters. If necessary, the baud rate can be changed at the beginning of this sequence to speed up the data transmission process.

Command Summary

Letter	Description	Version
A	Add Scans	1.00.0
B	Set Pixel Boxcar Width	1.00.0
C		
D		
E		
F	***Nonfunctional but follows SAD500 command format***	1.00.0
G	Set Data Compression	1.00.0
H	Set A/D Channel	1.00.0
I	Set Integration Time	1.00.0
J	Set Lamp Enable Line	1.00.0
K	Change Baud Rate	1.00.0
L		
M		
N		
O		
P	Specify Partial Pixel Mode	1.00.0
Q	Initialize Operating Parameters	1.00.0
R		
S	Start Spectral Acquisition (with previously set parameters)	1.00.0
T	Set Trigger Mode	1.00.0
U		
V		
W		
X		
Y		
Z		
a	Set ASCII Data Mode	1.00.0
b	Set Binary Data Mode	1.00.0
f	Set Continuous Strobe Rate	1.00.0
k	Set Checksum Mode	1.00.0
t		
v	Query Microcode Version Number	1.00.0
x	Set Calibration Coefficients	1.00.0
?	Query Parameter Values	1.00.0
-	Identify ADC1000-USB	1.00.0

Note: "Nonfunctional but follows SAD500 command format" means that the ADC1000-USB requires a data value following the command variable and responds with just one NAK (instead of 3). This allows customer device drivers that were written for the SAD500 Serial Port Interface A/D Converter to be functional with minimal changes.

Command Descriptions

The following are detailed descriptions of all ADC1000-USB serial commands. The {} indicates a data value that is interpreted as either ASCII or binary (default). The default value indicates the value of the parameter upon power up.

Add Scans

Description: Sets the number of discrete spectra to sum together. Since this routine can add up to 15 spectra, each with a maximum intensity of 4096, the maximum returned intensity is 65535.

Command Syntax:	A{DATA WORD}
Response:	ACK or NAK
Range:	1-15
Default value:	1

Set Pixel Boxcar Width

Description: Sets the number of pixels to average together. A value of *n* specifies the averaging of *n* pixels to the right and *n* pixels to the left. This routine uses 32-bit integers so that intermediate overflow will not occur; however, the result is truncated to a 16-bit integer prior to transmission of the data. This math is performed just prior to each pixel value being transmitted out. Values greater than ~3 will exceed the idle time between values and slow down the overall transfer process.

Command Syntax:	B{DATA WORD}
Response:	ACK or NAK
Range:	0-15
Default value:	0

Set Data Compression

Description: Specifies whether the data transmitted from the ADC1000-USB is compressed to speed data transfer rates. For more information on ADC1000-USB Data Compression, see the Technical Note 1 on page 12.

Command Syntax:	G{DATA WORD}
Response:	ACK or NAK
Range:	0 – Compression off !0 – Compression on
Default value:	0

Set A/D Channel

Description: Sets the A/D channel to digitize. The data value either activates or deactivates the rotator feature. If the rotator is active, then the pixels are interlaced in the following manner:

Pixel	0	1	N	N+1	N+2	2N
Channel	Master	Slave1	Slave N	Master	Slave 1	Slave N

Command Syntax:	H{DATA WORD}
Response:	ACK or NAK
Range:	0-7, 256 – 263 If the value is > 255 then the rotator is active, otherwise it is inactive
Default value:	0

Set Integration Time

Description: Sets the ADC1000-USB's integration time, in milliseconds, to the value specified.

Command Syntax:	I{DATA WORD}
Response:	ACK or NAK
Range:	5 - 65535
Default value:	100

If the Integration clock is using the 8-bit timer ("y" command) and a value greater than 255 is desired, the integration time is set to the LSB of the data word (i.e. value MOD 255).

Set Lamp Enable Line

Description: Sets the ADC1000-USB's lamp enable line to the value specified.

Command Syntax:	J{DATA WORD}
Value:	0 = Light source/strobe off—Lamp Enable low !0 = Light source/strobe on—Lamp Enable high
Response:	ACK or NAK
Default value:	0

Change Baud Rate

Description: Sets the ADC1000-USB's baud rate.

Command Syntax:	K{DATA WORD}
Value:	0=2400 1=4800 2=9600 3=19200 4=38400 5=57600 6=115200
Response:	See below
Default value:	2

When changing baud rates, the following sequence must be followed:

1. Controlling program sends K with desired baud rate, communicating at the old baud rate
2. A/D responds with ACK at the old baud rate, or with NAK and the process is aborted
3. Controlling program waits longer than 50 milliseconds
4. Controlling program sends K with desired baud rate, communicating at the new baud rate
5. A/D responds with ACK at the new baud rate, or with NAK and the old baud rate is used

If a deviation occurs at any step, the previous baud rate is utilized.

Specify Partial Pixel Mode

Description: Specifies which pixels are transmitted. While all pixels are acquired on every scan, this parameter determines which pixels are transmitted out the serial port.

Command Syntax:	P{DATA WORD}	
Value:	Description	Example
	0 = All 2048 pixels	P 0 (spaces for clarity only)
	1 = Every n^{th} pixel with no averaging	P 1 n
	2 = N/A	P 2 n
	3 = Pixel x through y every n pixels	P 3 x y n
	4 = Up to 10 randomly selected pixels between 0 and 2047 (denoted p1, p2, … p10)	P 4 n p1 p2 p3…p10
Response:	ACK or NAK	
Default value:	0	

Since most applications only require a subset of the spectrum, this mode can greatly reduce the amount of time required to transmit a spectrum while still providing all of the desired data. This mode is helpful when interfacing to PLCs or other processing equipment.

Initialize Operating Parameters

Description: Sets all operating parameters to their default value.

Command Syntax:	Q
Response:	ACK or NAK

Start Spectral Acquisition

Description: Acquires spectra with the current set of operating parameters. When executed, this command determines the amount of memory required. If sufficient memory does not exist, an ETX (ASCII 3) is immediately returned and no spectra are acquired, otherwise an STX (ASCII 2) is returned followed by the data.

Command Syntax:	S
Response:	If successful, STX followed by data If unsuccessful, ETX

The format of returned spectra includes a header to indicate scan number, channel number, pixel mode, etc. The format is as follows:

WORD 0xFFFF – start of spectrum
WORD A/D channel number
WORD scan number ALWAYS 0
WORD scans in memory ALWAYS 0
WORD integration time in milliseconds
WORD integration time counter ALWAYS 0
WORD pixel mode
WORDs if pixel mode not 0, indicates parameters passed to the Pixel Mode command "P"
WORDs spectral data
WORD 0xFFFD – end of spectrum

Set Trigger Mode

Description: Sets the ADC1000-USB's external trigger mode to the value specified.

Command Syntax:	T{DATA WORD}
Value:	0 = Normal – Continuously scanning 1 = Software Trigger 2 = External Synchronization Trigger 3 = External Hardware Trigger
Response:	ACK or NAK
Default value:	0

Set ASCII Data Mode

Description: Sets the mode in which data values are interpreted as ASCII. Only unsigned integer values (0 – 65535) are allowed in this mode and the data values are terminated with a carriage return (ASCII 13) or linefeed (ASCII 10). In this mode, the ADC1000-USB "echos" the command and data values back out of the RS-232 port.

Command Syntax:	aA
Response:	ACK or NAK
Default value	N/A

- This command requires that the string "aA" be sent without any CR or LF. This requirement helps to insure that this mode is not entered inadvertently.
- A legible response to the microcode version number query ("v" command) indicates that the unit is in the ASCII data mode.

Set Binary Data Mode

Description: Sets the mode in which data values are interpreted as binary. Only 16-bit unsigned integer values (0 – 65535) are allowed in this mode with the MSB followed by the LSB.

Command Syntax:	bB
Response:	ACK or NAK
Default value	Default at power up – not changed by command "Q"

This command requires that the string "bB" be sent without any CR or LF. This requirement helps to insure that this mode is not entered inadvertently.

Set Continuous Strobe Rate

Description: Sets the continuous strobe period, in milliseconds, to the value specified. A value greater than 255 results in the strobe rate being set to 255.

Command Syntax:	f{DATA WORD}
Response:	ACK or NAK
Range:	1 - 255
Default value:	10

Set Checksum Mode

Description: Specifies if the ADC1000-USB generates and transmits a 16-bit checksum of the spectral data. This checksum can be used to test the validity of the spectral data, and its use is recommended when reliable data scans are required. See Technical Note 2 on page 14 for more information on checksum calculation.

Command Syntax:	k{DATA WORD}
Value:	0 = Do not transmit checksum value !0 = Transmit checksum value at end of scan
Response:	ACK or NAK
Default value:	0

Query Microcode Version Number

Description: Returns the version number of the code running on the microcontroller. A returned value of 1000 is interpreted as 1.00.0.

Command Syntax:	v
Response:	ACK followed by {DATA WORD}
Default value	N/A

Set Calibration Coefficients

Description: Writes one of the 44 possible calibration coefficients to EEPROM. A calibration coefficient is specified by the first DATA WORD that follows the x. The coefficient is stored as an ASCII string with a max length of 15 characters. The string is not checked for viability. To query the coefficients, use the ?x{DATA WORD} format to specify the desired coefficient.

Command Syntax:	x{DATA WORD}{ASCII STRING}
Value:	DATA WORD Index description 0 – Serial Number 1 – Channel Enabled Register – bit n, channel n, … 2 – 0 Order Wavelength Calibration Coefficient: Channel 0 3 – 1st Order Wavelength Calibration Coefficient: Channel 0 4 – 2nd Order Wavelength Calibration Coefficient: Channel 0 5 – 3rd Order Wavelength Calibration Coefficient: Channel 0 6 – 0 Order Wavelength Calibration Coefficient: Channel 1 7 – 1st Order Wavelength Calibration Coefficient: Channel 1 8 – 2nd Order Wavelength Calibration Coefficient: Channel 1 9 – 3rd Order Wavelength Calibration Coefficient: Channel 1 … 30 – 0 Order Wavelength Calibration Coefficient: Channel 7 31 – 1st Order Wavelength Calibration Coefficient: Channel 7 32 – 2nd Order Wavelength Calibration Coefficient: Channel 7 33 – 3rd Order Wavelength Calibration Coefficient: Channel 7 34 – 44 – Reserved
Response:	ACK or NAK
Default value:	N/A

Query Parameter Values

Description: Returns the current value of the parameter specified. The syntax of this command requires two ASCII characters. The second ASCII character corresponds to the command character which sets the parameter of interest (acceptable values are B, A, I, K, T, J). A special case of this command is "?x" (lower case), which requires an additional data word be passed to indicate the calibration constant to query.

Command Syntax:	?{ASCII character}
Response:	ACK followed by {DATA WORD}
Default value:	N/A

Identify ADC1000-USB

Description: Device drivers can differentiate between an ADC1000-USB A/D Converter, a USB2000 Spectrometer and a SAD500 Serial Port Interface A/D Converter with the use of this command. The ADC1000-USB generates an ACK in response to this command while the other instruments generate a NAK.

Command Syntax:	-
Response:	ACK
Default value	N/A

Examples

Below are examples on how to use some of the commands. Commands are in **bold** and descriptions are in parenthesis. For clarity, the commands are shown in the ASCII mode ("a" command) instead of the default binary mode.

The desired operating conditions are: acquire spectra from spectrometer channel 0 (master) with a 200 ms integration time, set number of scan to add to 5 and operate at 57,600 baud.

aA	(Set ASCII data mode)
K5<CR>	(Start baud rate change to 57,600)
	Wait for ACK, change to 57600, wait for 50 ms
K5<CR>	(Verify command, communicate at 57600)
A5<CR>	(Add 5 spectra)
I200<CR>	(Set integration time to 200 ms)
S	(Acquire spectra)
…	Repeat as necessary

Application Tips

- During the software development phase of a project, the operating parameters of the ADC1000-USB may become out-of-synch with the controlling program. It is good practice to cycle power on the ADC1000-USB when errors occur.

- If you question the state of the ADC1000-USB, you can transmit a space (or another non-command) using a terminal emulator. If you receive a NAK, the ADC1000-USB is awaiting a command; otherwise, it is still completing the previous command.

- For Windows 95/98 users, use HyperTerminal as a terminal emulator after selecting the following:
  1. Select **File | Properties**.
  2. Under **Connect using**, select **Direct to Com x**.
  3. Click **Configure** and match the following **Port Settings**:
     Bits per second (baud rate): Set to desired rate
     Data bits: 8
     Parity: None
     Stop bits: 1
     Flow control: None
  4. Click the **Advanced** button and slide the **Receive Buffer** and **Transmit Buffer** arrows to the Low 1 value. Click **OK**.
  5. Click **OK** in **Port Settings** and in **Properties** dialog boxes.

Technical Note 1: ADC1000-USB Data Compression

Transmission of spectral data over the serial port is a relatively slow process. Even at 57,600 baud, the transmission of a complete 2048 point spectrum takes about 600 msec. The USB2000 Spectrometer implements a data compression routine that minimizes the amount of data that needs to be transferred over the RS-232 connection. Using the "G" command (Compressed Mode) and passing it a parameter of 1 enables the data compression. Every scan transmitted by the USB2000 is then compressed. The compression algorithm is as follows:

1. The first pixel (a 16-bit unsigned integer) is always transmitted uncompressed.
2. The next byte is compared to 0x80.
   - If the byte is equal to 0x80, the next two bytes are taken as the pixel value (16-bit **unsigned** integer).
   - If the byte is not equal to 0x80, the value of this byte is taken as the difference in intensity from the previous pixel. This difference is interpreted as an 8-bit **signed** integer.
3. Repeat step 2 until all pixels have been read.

Using this data compression algorithm greatly increases the data transfer speed of the ADC1000-USB. The table below shows the data transfer speed, in milliseconds, for various light sources and baud rates. Keep in mind that these rates are for demonstration purposes only, and the speed of your computer may affect the data transfer rates.

	Comp	57.6 kb	% faster	38.4 kb	% faster	19.2 kb	% faster	9.6 kb	% faster
dark	on	426	45.2%	624	46.7%	1148	47.5%	2247	48.8%
	off	778		1170		2188		4391	
LS-1	on	429	44.9%	624	46.6%	1141	49.6%	2192	50.1%
	off	779		1169		2266		4390	
HG-1	on	465	40.2%	679	41.9%	1238	43.5%	2424	44.8%
	off	777		1169		2193		4391	

The following shows a section of a spectral line source spectrum and the results of the data compression algorithm.

Pixel Value	Value Difference	Transmitted Bytes
185	0	0x80 0x00 0xB9
2151	1966	0x80 0x08 0x67
836	-1315	0x80 0x03 0x44
453	-383	0x80 0x01 0xC5
210	-243	0x80 0x00 0xD2
118	-92	0xA4
90	-28	0xE4
89	-1	0xFF
87	-2	0xFE
89	2	0x02
86	-3	0xFD
88	2	0x02
98	10	0x0A
Pixel Value	**Value**	**Transmitted Bytes**

	Difference	
121	23	0x17
383	262	0x80 0x01 0x7F
1162	779	0x80 0x04 0x8A
634	-528	0x80 0x02 0x7A
356	-278	0x80 0x01 0x64
211	-145	0x80 0x00 0xD3
132	-79	0xB1
88	-44	0xD4
83	-5	0xFB
86	3	0x03
82	-4	0xFC
91	9	0x09
92	1	0x01
81	-11	0xF5
80	-1	0xFF
84	4	0x04
84	0	0x00
85	1	0x01
83	-2	0xFE
80	-3	0xFD
80	0	0x00
88	8	0x08
94	6	0x06
90	-4	0xFC
103	13	0x0D
111	8	0x08
138	27	0x1B

In this example, spectral data for 40 pixels is transmitted using only 60 bytes. If the same data set were transmitted using uncompressed data, it would require 80 bytes.

Technical Note 2: ADC1000-USB Checksum Calculation

For all uncompressed pixel modes, the checksum is simply the unsigned 16-bit sum (ignoring overflows) of all transmitted spectral points. For example, if the following 10 pixels are transferred, the calculation of the checksum would be as follows:

Pixel Number	Data (decimal)	Data (hex)
0	15	0x000F
1	23	0x0017
2	46	0x002E
3	98	0x0062
4	231	0x00E7
5	509	0x01FD
6	1023	0x03FF
7	2432	0x0980
8	3245	0x0CAD
9	1984	0x07C0

Checksum value: 0x2586

When using a data compression mode, the checksum becomes a bit more complicated. A compressed pixel is treated as a 16-bit **unsigned** integer, with the most significant byte set to 0. Using the same data set used in Technical Note 1, the following table shows a section of a spectral line source spectrum and the results of the data compression algorithm.

Data Value	Value Difference	Transmitted Bytes	Value added to Checksum
185	0	0x80 0x00 0xB9	0x0139
2151	1966	0x80 0x08 0x67	0x08E7
836	-1315	0x80 0x03 0x44	0x03C4
453	-383	0x80 0x01 0xC5	0x0245
210	-243	0x80 0x00 0xD2	0x0152
118	-92	0xA4	0x00A4
90	-28	0xE4	0x00E4
89	-1	0xFF	0x00FF
87	-2	0xFE	0x00FE
89	2	0x02	0x0002
86	-3	0xFD	0x00FD
88	2	0x02	0x0002
98	10	0x0A	0x000A
121	23	0x17	0x0017
383	262	0x80 0x01 0x7F	0x01FF
1162	779	0x80 0x04 0x8A	0x050A
634	-528	0x80 0x02 0x7A	0x02FA
356	-278	0x80 0x01 0x64	0x01E4
211	-145	0x80 0x00 0xD3	0x0153
132	-79	0xB1	0x00B1
Data Value	**Value**	**Transmitted Bytes**	**Value added to**

	Difference		Checksum
88	-44	0xD4	0x00D4
83	-5	0xFB	0x00FB
86	3	0x03	0x0003
82	-4	0xFC	0x00FC
91	9	0x09	0x0009
92	1	0x01	0x0001
81	-11	0xF5	0x00F5
80	-1	0xFF	0x00FF
84	4	0x04	0x0004
84	0	0x00	0x0000
85	1	0x01	0x0001
83	-2	0xFE	0x00FE
80	-3	0xFD	0x00FD
80	0	0x00	0x0000
88	8	0x08	0x0008
94	6	0x06	0x0006
90	-4	0xFC	0x00FC
103	13	0x0D	0x000D
111	8	0x08	0x0008
138	27	0x1B	0x001B

The checksum value is simply the sum of all entries in the last column, and evaluates to 0x2C13.