SPECTRUM ONE

CCD Detection System

Programming Manual

Revised October 30, 2002

Copyright © October, 02 Jobin Yvon, Optical Spectroscopy Division. All rights Reserved. Portions of the software described in this document Copyright © Microsoft Corporation and Galactic Industries Corporation. All rights Reserved.

No part of this document may be reproduced, stored in a retrieval system, or transmitted in any form by any means, including electronic or mechanical, photocopying and recording without prior written permission of Jobin Yvon, Optical Spectroscopy Division. Requests for permission should be submitted in writing.

Information in this document is subject to change without notice and does not represent a commitment on the part of the vendor.

ABOUT THE MANUALS

You may have more than one manual, depending on your system configuration. To find the manual that has the information you need, these guidelines may help.

- Each manual generally covers a product and the features and accessories peculiar to and/or contained within that product.
- Accessories that can be applied to other products are normally covered by separate documentation.
- Software that is exclusively used with one instrument or system is covered in the manual for that product.
- Software that can be used with a number of other products is covered in its own manual.
- If you are reading about a product that interacts with other products, you will be referred to other documentation as necessary.

TABLE OF CONTENTS:

OVERVIEW:

The Spectrum One series detectors are cooled Charge Coupled Devices (CCD's) and InGaAs Arrays (IGAs) which provide photodetection for spectrometric applications. These detectors can be interfaced to the exit port of most SPEX and J-Y spectrographs, including those in fluorometers and Raman instruments.

The glossary section in Appendix A of this manual contains definitions of terms and information about essential topics relating to CCD detection of spectra. Reading the glossary is recommended.

The Spectrum One CCD system is well suited to applications such as:

- Low or very low signal levels such as Raman, fluorescence, and absorption spectroscopy.
- Recording the spectra from multiple sources or locations that are imaged along the height of the spectrograph entrance slit.
- Near IR measurements to 1100 nm.

INTRODUCTION

CCD-3x00 Family of Detectors:

This manuals covers the programming of the CCD-3x00 family of detectors. This family includes the CCD-3000, CCD-3000V, CCD-3500, CCD3500V, IGA-3000, and IGA-3000V controllers. All controllers will have the same basic Command Set Structure. However, the various types of detectors will have different methods for reading the data.

Command Set Structure:

The controller family supported by this command set has been designed with a multi-purpose interface. This interface will communicate with a simple ASCII "terminal" or an "intelligent" computer program. Commands are sent to the CCD controller by sending these ASCII strings over the GPIB or IEE488 interface. Before trying to program the CCD controller, please review the correct methods in your programming language for sending and receiving ASCII strings over the GPIB interface. For more information, please refer to the National Instruments GPIB Manuals.

GPIB SETUP

Supported IEEE 488 Computer Interface Boards

Of National Instrument's GPIB offerings, we support the following cards and drivers for use in Windows 95 or Windows 98.

- AT-GPIB/TNT 488 Interface board: The driver supplied by National should be version 1.2 or later. This board comes in Plug-and-Play and standard formats, either is suitable for use with the CCD-3000.

- PCMCIA-GPIB: Should have National Instruments driver 1.2 or newer.

- PCI-GPIB: Fits most newer Pentiums and takes advantage of free PCI slots. Should use National Instruments driver 1.2 or newer.

- GPIB-PCII/PCIIA 488.2 Interface boards: The driver supplied by National should be version 1.2 or newer, and their BASIC support disk should be version 2.0 or newer. These boards should only be used in PCII mode, as the PCIIA driver can conflict with Windows 95.

There are other boards by National and other suppliers for IBM compatible and Macintosh computers. Many of these boards can function in a *SIMILAR* fashion, however the program communicates specifically with the National Instruments driver therefore we **do not** support or guarantee reliable communications with other boards and software, we **strongly** recommend that you use those National Instruments products described above.

Installing the GPIB Card

Please refer to the Installation Instructions from National Instruments that accompanied your GPIB card.

Setting Up the GPIB Card for the CCD-3x00

GPIB Device Addresses

The CCD-3x00 detectors uses one of two GPIB Dev Addresses for all GPIB communication. The GPIB Dev Address used depends on the actual type of detector

CCD-3000, CCD-3000V, CCD-3500, CCD3500V use GPIB Device 5

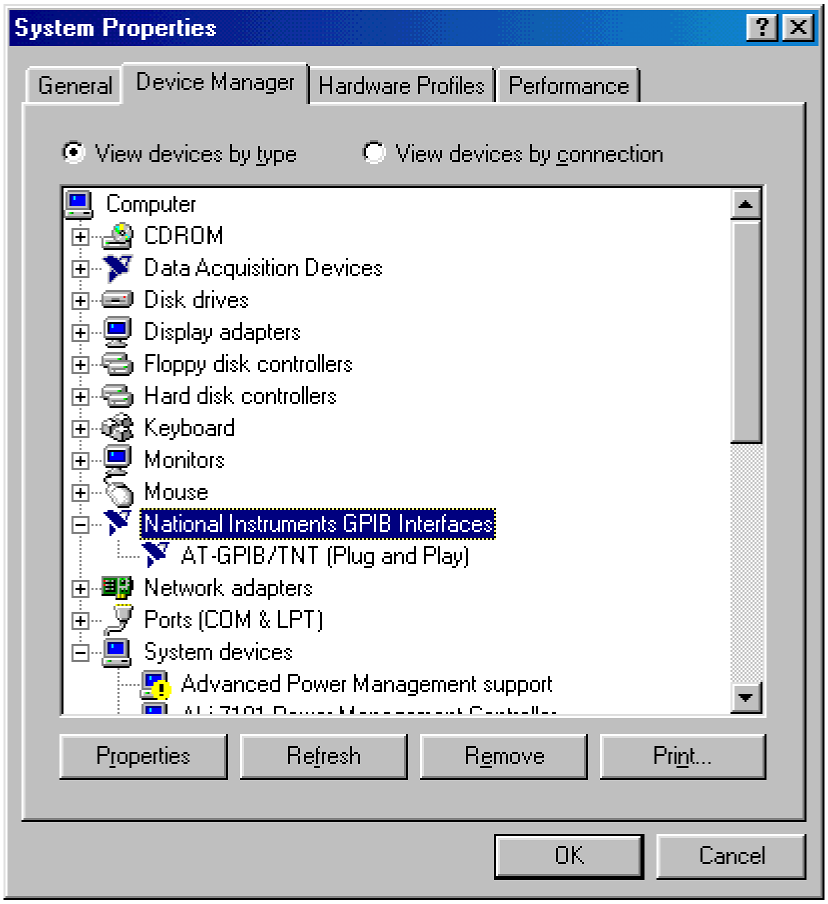
IGA-3000, IGA-3000V use GPIB Device 6

These GPIB Device Addresses are not settable by the user.

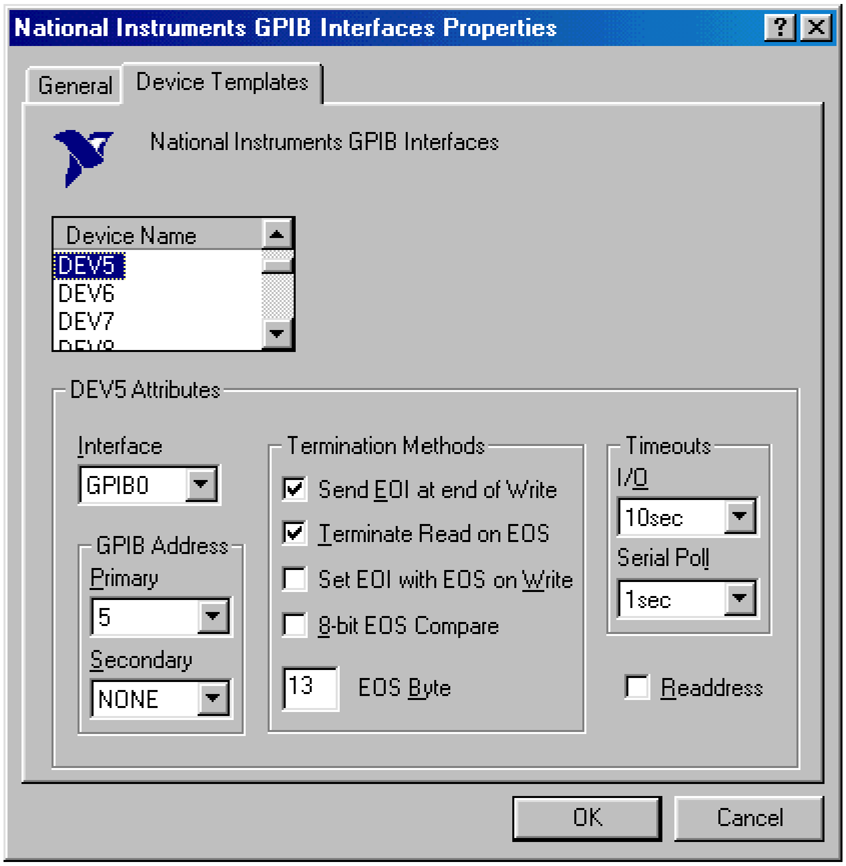
GPIB Device Setup

The CCD-3x00 family of controllers requires specific parameters to be set in the Device Setup of the GPIB Interface. To access the Device Setup in Windows, press the Start button on the Task Bar and go to Settings>>Control Panel>>System and choose the Device Manager Tab

Click on the National Instrument GPIB Interface



Select DEV5 for CCD-3x00s or DEV 6 for IGA3000s



Place a Check in the Terminate Read on EOS Check Box and enter 13 as the EOS Byte. After these values have been set, press the OK button and exit the GPIB Device Setup.

Testing GPIB Communication with the CCD

Win32 Interactive Control allows you to send commands directly to the GPIB interface, while ***NI Spy*** allows you to monitor and capture the GPIB calls which are made. To test communications you may use the following procedure to write and read a character from the CCD-3000 via the GPIB card using ***Win32 Interactive Control*** and monitoring this with ***NI Spy***.

Activate ***NI Spy*** and *start capture* then enter ***Win32 Interactive Control*** and enter the following lines.

IBFIND GPIB0	Finds the GPIB board in the PC
IBFIND DEV5	Finds the CCD3000 controller at address 5
IBWRT " "	Send one space character (must be enclosed in Quotation marks)
IBRD 1	Read 1 character.

If you have successfully communicated, you should receive a B or an F. If you have an error message displayed, refer to the National Instruments documentation to interpret it.

NI Spy should display the capture of the above similar to figure C-19.

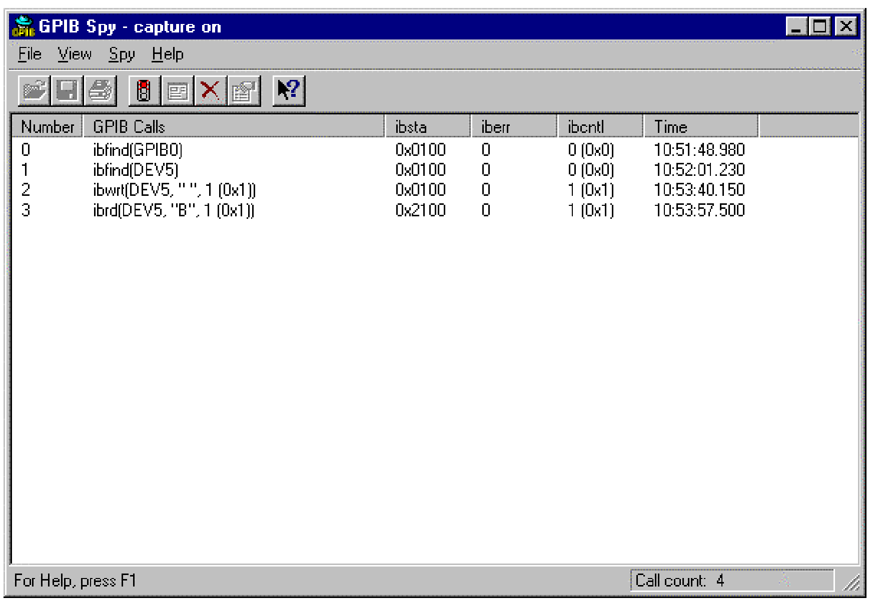


Figure C-19: GPIB Spy capture of NI-488.2 Interface Test Procedure

Preparing to program via IEEE 488:

After success in communicating using the National Instruments hardware and software you may proceed to command the controller using the Programmer's Command Set.

If you construct programs based on the Command Set, we strongly recommend that you add prudent error trapping and protection features to your program to protect your system and enhance ease-of-use.

INITIALIZING CCD3X00 CONTROLLER

GPIB Communications Startup:

The part of your program that will establish communications must follow the steps outlined in the GPIB Start Up Procedure flow chart below.

When the controller is addressed on the IEEE 488 bus, it automatically sets itself in the intelligent communications mode and you need only establish whether you are talking to the BOOT or the MAIN internal controller program.

Before you start collecting data with the CCD, you must first initialize the controller and download information about the CCD chip to the controller. To do this, perform the following sequence of commands.

1. <Space> *(ASCII code 32)*
   - Read Confirmation: The Controller will respond with a 'B' or an 'F'. If the controller responds with a 'F', please skip to step 3.
2. O2000<Null> *The <Null> character is ASCII code 0*
   - Read Confirmation: The Controller will respond with an *.
3. <Space> *(ASCII code 32)*
   - Read Confirmation: The Controller will respond with an 'F'.
4. Z300,0<CR> *(CCD INIT Command)*

– Read Confirmation: The controller should respond with 'o0<CR>'

5. z
   - Read Response: The Controller will respond with 'Vx.xx CCD-3000<CR>'
   - The version number should be read into a string and stored. Some commands and procedures will depend on the Version Number.

Skip to Step 7 if your CCD controller is Version 1.68 or lower.

6. Z352,0<CR> *(CCD Select ADC Command. Use a 0 for all CCD-3000 controllers and 16-bit mode of CCD-3000, or 1 for 14 bit mode in CCD-3500)*

   - Read Response: The controller will respond with 'nnn<CR>' when 'nnn' is a number. This number represents the number of placeholder data values you will need to skip over when reading the data from the CCD. Store this value as a variable in your program, it will be required in later steps.

   - If you send this command to a CCD-3000 controller with Version 1.68 or lower, you will receive a 'b' representing a Bad Command. Ignore this and go on to step 7.

Notes on IEEE-488 Start Up Procedure

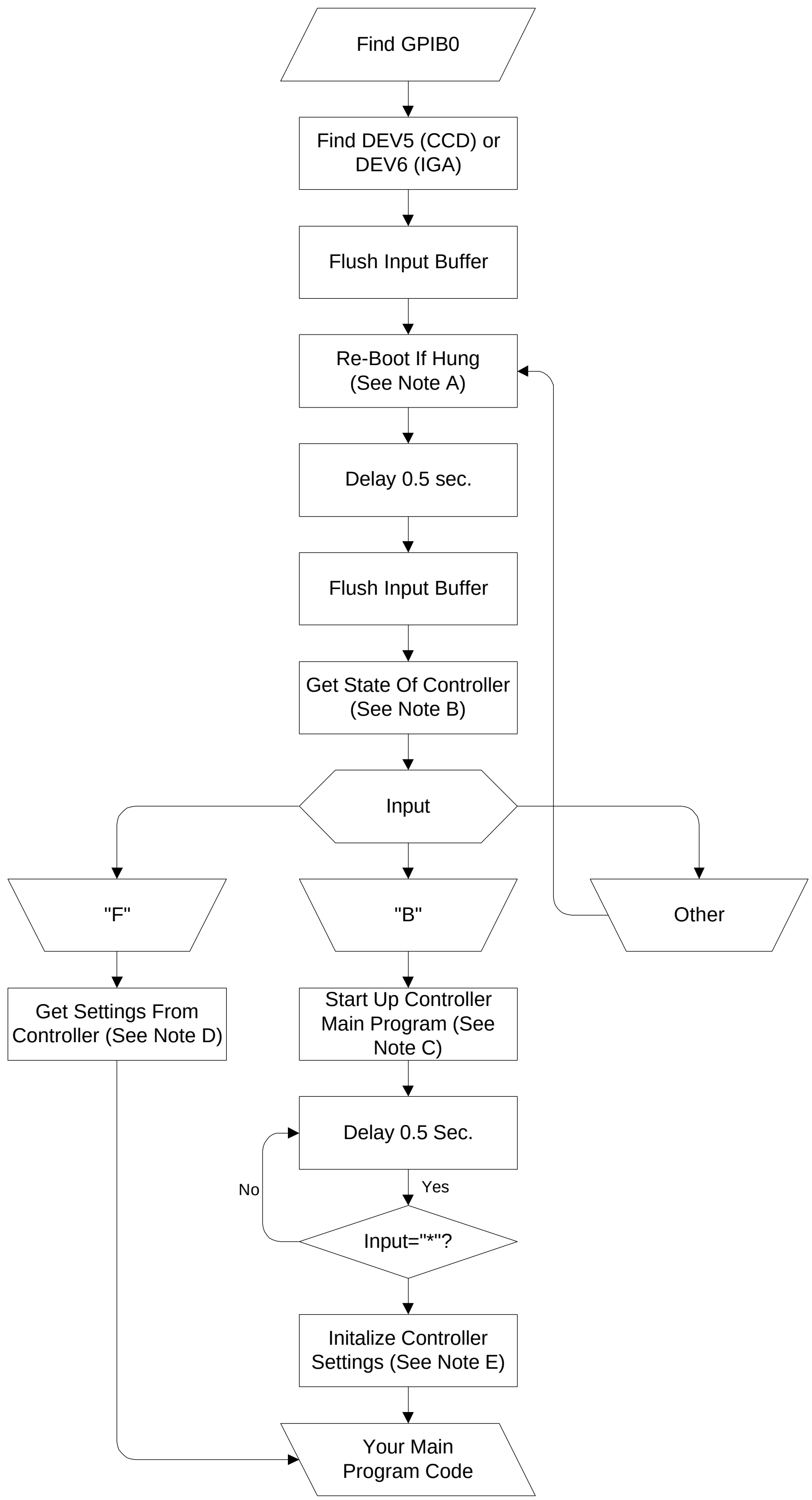
A: Send decimal value "<222>". This will force a re-boot if hung from an incomplete command.

B: Send WHERE AM I command "<Space>"; response will be "B" (for BOOT) or "F" (for MAIN) depending on the previous state of the CCD3000 controller.

C: Send "O2000<Null>". To transfer control from the BOOT to the MAIN program. You must send the "<Null>". Wait 0.5 second.

D: You can read your last position etc. from the spectrometer controller. You do not have to re-initialize the CCD3000 controller.

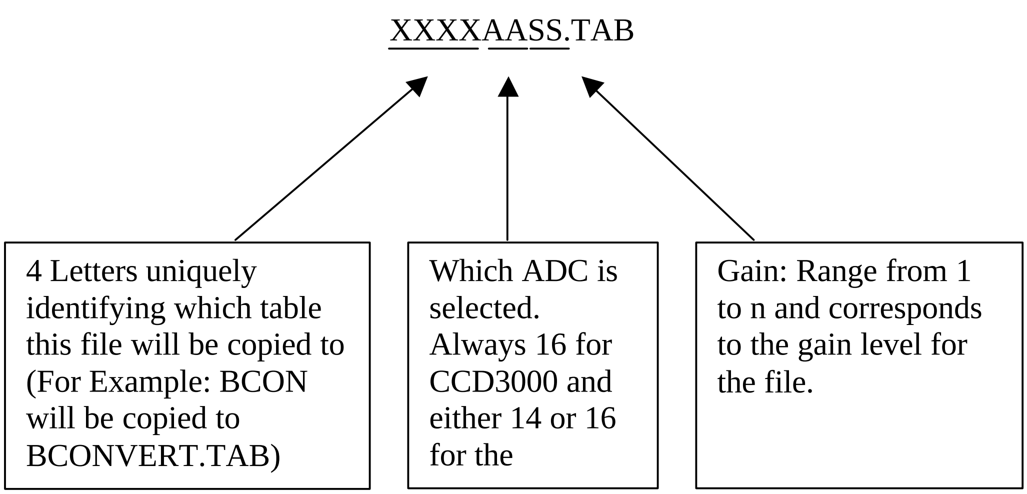
E: Initialize controller. Data tables must be loaded into the controller memory. See **TABLE LOADING PROCEDURE** on next page.



Loading Table Files

7. TAB files are files that hold binary information required by the CCD-3000 controller to correctly process the information collected by the CCD chip and convert it to values that can be read out to a computer. The information in these files must be transferred into the controller before any data acquisition can be performed. There are 8 files that need to be downloaded into the CCD-3000 before it is ready to collect data. These 8 default files are BCONVERT.TAB, ECONVERT.TAB, NIDLE.TAB, PARTRANS.TAB, SERBIN.TAB, SERCLEAR.TAB, SERWCONV.TAB, and STIDLE.TAB. The contents of these default files will change depending on the actual CCD chip, the ADC mode, and the gain level being used. On the CCD Initialization Disk, there is a complete set of TAB files for use with the available ADC modes and gain levels for that specific CCD chip and system.

   For each combination of ADC mode and gain level, there is a set of TAB files that needs to be copied into the default files listed above. These files are named using the following method.



   When loading the TAB files, the software should always load the information from the 8 default files listed above. If a change in the gain level or ADC is required, the TAB files for that particular ADC and gain level combination should be copied into the corresponding default files.

   For example, if you want to set the CCD-3000 to use gain level 1, copy all of the files ending in 1601 to the appropriate default file names.

New File (16 Bit/Gain	copy into	Default File Name
BCON1601.TAB	→	BCONVERT.TAB
ECON1601.TAB	→	ECONVERT.TAB
NIDL1601.TAB	→	NIDLE.TAB
PART1601.TAB	→	PARTRANS.TAB
SERB1601.TAB	→	SERBIN.TAB
SERC1601.TAB	→	SERCLEAR.TAB
SERW1601.TAB	→	SERWCONV.TAB
STID1601.TAB	→	STIDLE.TAB

Once the correct TAB files for the ADC and gain level have been copied into the default TAB files, the information needs to be downloaded into the CCD controller.

The TAB file information must be loaded into the CCD controller in a specific order and to specific memory addresses. The manual lists the order and address offset for each TAB file. To get the correct memory address, the Address offset is added to the Base Address. The Base Memory Address is D000 hex or 53248 in decimal. All commands being sent to the controller must be sent in decimal values.

The order to load the TAB files is as follows:

STIDLE.TAB	Offset = 0 (0 hex)	Memory Address = 53248 (D000h)
SERWCONV.TAB	Offset = 1024 (400 hex)	Memory Address = 54272 (D400h)
SERCLEAR.TAB	Offset = 2048 (800 hex)	Memory Address = 55296 (D800h)
SERBIN.TAB	Offset = 3072 (C00 hex)	Memory Address = 56320 (DC00h)
PARTRANS.TAB	Offset = 4096 (1000 hex)	Memory Address = 57344 (E000h)
BCONVERT.TAB	Offset = 5120 (1400 hex)	Memory Address = 58368 (E400h)
ECONVERT.TAB	Offset = 6144 (1800 hex)	Memory Address = 59392 (E800h)
NIDLE.TAB	Offset = 7168 (1C00 hex)	Memory Address = 60416 (EC00h)

The command to load the TAB file is Z340. The syntax is

Z340, 0, [Chip Select], [Address], [Number of Bytes]<CR>

Where [Chip Select] is defined in the manual and has values from 0 to 3, [Address] is the memory address you are writing to, and [Number of Bytes] is the number of data byte you will be sending. After each Z340 command, you will receive a confirmation 'o', after the confirmation start sending the data.

Each TAB file requires four Z340 commands (one for each Chip Select number from 0 to 3) to load that TAB file into memory.

Table Loading Procedure

Example using ECONVERT.TAB

Note: All tables must be loaded for controller to initialize properly.

Content/Meaning

- Table Size, 159 Long (4 bytes) follows.
- Pixel Table, each line holds 4 bytes worth of data.
- Numbers vary between table and version of table.

Line #	Table			
1	9F	ØØ	ØØ	ØØ
2	Ø1	ØØ	33	Ø1
3	Ø2	ØØ	33	Ø1
4	Ø3	ØØ	33	Ø1
M	M	M	M	M
16Ø	ØØ	ØØ	71	Ø1
	Chip Select + Ø ($R_Ø$)	Chip Select + 1 (R_1)	Chip Select + 2 (R_2)	Chip Select + 3 (R_3)

To load the following NIDLE.TAB File:

00000006
01740001
01740002
01740003
01740004
01740405
01740000

The first line shows the number of data that need to be sent, [Number of Bytes]. The remaining six line are the data values. The following four lines show the Table Load procedure for loading this file.

7a) Z340,0,0,60416,6<CR>
Read Confirmation : Controller responds with ‘o’
7b) <01><02><03><04><05><00> *The <01> indicates the sending of ASCII code 01.*

7c) Z340,0,1,60416,6<CR>
Read Confirmation : Controller responds with ‘o’
7d) <00><00><00><00><04><00>

7e) Z340,0,2,60416,6<CR>
Read Confirmation : Controller responds with ‘o’
7f) <74><74><74><74><74><74>
7g) Z340,0,3,60416,6<CR>
Read Confirmation : Controller responds with ‘o’
7h) <01><01><01><01><01><01>

8. Z328 – *Load Chip Parameters* - After loading the TAB files into the controller memory, you must use the Z328 command to load the CCD chip parameters into the controller. Most of these parameters can be found in the file CCDLOAD.INI. Two of the parameters must calculated. The Total Parallel Pixels can be calculated from the Total Active Parallel Pixels and the Number of Parallel Pixels before and after the Active area. Total Serial Pixels can be calculated in a similar manner. Below is a sample CCDLOAD.INI for a 1024x256 CCD.

   For the CCDLOAD file listed below, the Z328 command parameters will be

   Z328,0,768,1024,256,8,8,11,0,5,0,300,1,400000000,0,4,270,270,267,1040<CR>

   - The CCD number should be always be given as 0 even if the CCDLOAD.INI file lists it as 1.
   - The final two parameters are calculated by adding together the total number of pixels, the number of pixels before the active area, and the number of pixels after the active area.

- The controller will respond with a ‘o’

Example CCDLOAD.INI

```
        ; CCD hardware (chip and controller) parameters
1       ; CCD Number

768     ; Base address of RISC board (300 hex = 768 dec)

        ; Number of pixels
1024    ; total_active_x_pixels (in the horizontal direction)
256     ; total_active_y_pixels (in the vertical direction)
8       ; number_serial_pxls_before_active_area
8       ; number_serial_pxls_after_active_area
11      ; number_parallel_rows_before_active
0       ; number_parallel_rows_after_active

5       ; readout register location and direction

0       ; Minimum temperature in degrees K
300     ; Maximum temperature in degrees K

4               ; Minimum shutter time in milliseconds
400000000       ; Maximum shutter time in milliseconds

0       ; Minimum Gain
4       ; Maximum Gain

        ; Returned in driver call. Currently not used by the driver.
        ; pixel spacing/size in tenths of um
270     ;       horizontally
270     ;       vertically
```

After loading this information, the initialization sequence is complete.

SETTING ACQUISITION PARAMETERS

The next section describes how to set up the CCD Acquisition parameters.

9. Z301,0,100<CR> *(CCD Set Integration Time)*

   – The final parameter is the integration time of the acquisition in milliseconds

   – Controller will respond with an 'o'

10. Z325,0,0,1<CR> *(Define Acquisition Format)*

   – Sets the controller into Scan or Spectral mode if the second parameter is 1 or image mode if the parameter is 0.

   – Last parameter defines how many separate active areas are to be defined. *Image mode can only have 1 active area.*

   – Controller will respond with an 'o'

11. Z326,0,0,1,1,1024,256,1,1<CR> *(CCD Define Area)*

   – The command sets the parameters for the active area.

   – Area numbers start with area number 0

   – One command must be sent for each active area. For example, if you want to use three areas, you must send this command three time with Area numbers 0, 1, and 2.

   – Controller will respond with an 'o'

12. Z327,0<CR> *(CCD Get Data Size)*

   – Controller responds with 'o1024,1024<CR>

   – The first number is the area size.

   – The second number is the total number of data values that will be sent from the controller to the computer on a CCD Read Image. This value will be require when the data is read out from the controller.

STARTING AN ACQUISITION AND READING DATA

13. Z311,0,1<CR> *(CCD Start)*

- Final parameter is the shutter state during the acquisition; 0 is closed, 1 is open.
- Controller will respond with an 'o'

14. Z312,0<CR> *(CCD Status)*

- Controller responds with 'on', where n is a number indicating the status of the CCD acquisition. The status will be 0 if the acquisition is complete and non-zero if the CCD is still acquiring data.
- The software should continue to poll the CCD status until a 0 result is obtained.
- This command to read the status must be issued at least once before trying to read out the data.

15. Z315,0<CR> *(CCD Read Image)*

- After issuing this command, the controller will send a confirmation character 'o' and immediately start sending the data. After sending this command and reading the confirmation, issue a GPIB Read for a number of bytes equal to

(2* {Number of Data Points}) + 1

The final value will be a status byte and should be an 'a2' to indicate no error.

For CCD-3000's, Version 1.68 and earlier:

- The data will be returned in binary format, not ASCII codes. For each data point, there are two bytes that need to be combined into the data point value. The first byte is the Least Significant Byte (LSB) and the second byte is the Most Significant Byte (MSB). The format of the data is

[LSB, Data Point 1] [MSB, Data Point 1] [LSB, Data Point 2] [MSB, Data Point 2]…

Once the two bytes have been combined to create the data point value, add an 8000hex (32768 decimal) offset to the value to get the true data point value.

For CCD-3000's, Version 1.80 and later:

– The data will be returned in binary format, not ASCII codes. For each data point, there are two bytes that need to be combined into the data point value. The first byte is the Least Significant Byte (LSB) and the second byte is the Most Significant Byte (MSB). The format of the data is

[LSB, Data Point 1] [MSB, Data Point 1] [LSB, Data Point 2] [MSB, Data Point 2]…

– The first few data points are placeholder values and need to be ignored. The number of data points to be skipped before getting the first real data value is the number returned by the Select_ADC Command in step 6.

If data from an Image mode is being read out, the data has placeholder values on each row and these data points must be skipped on each row of data values.

– Once the real data has been obtained, an offset needs to be added to the data points to get the correct value. The offset value depends on the ADC mode that was used to collect the data.

- 16 Bit ADC Mode: Offset = 8000hex
- 14 Bit ADC Mode: Offset = 0 hex

If more that one active area is being used, one CCD Read Image Command is issued for all of the data and the data must be separated into the individual spectra. Each spectrum will have the same number of placeholder data points at the start of the spectrum that must be skipped over.

For IGA-3000's with 128, 256 or 512 x 25 mm pixels, Version 1.80 and later:

– The data will be returned in binary format, not ASCII codes. For each data point, there are two bytes that need to be combined into the data point value. The first byte is the Least Significant Byte (LSB) and the second byte is the Most Significant Byte (MSB). The format of the data is

[LSB, Data Point 1] [MSB, Data Point 1] [LSB, Data Point 2] [MSB, Data Point 2]… [LSB, Data Point n] [MSB, Data Point n]

– The first few data points are placeholder values and need to be ignored. The number of data points to be skipped before getting the first real data value is the number returned by the Select_ADC Command in step 6.

For IGA-3000's with 512 x 50 mm pixels, Version 1.80 and later:

– The data will be returned in binary format, not ASCII codes. For each data point, there are two bytes that need to be combined into the data point value. The first byte is the Least Significant Byte (LSB) and the second byte is the Most Significant Byte (MSB). The format of the data is

[LSB, Data Point 257] [MSB, Data Point 257] [LSB, Data Point 1] [MSB, Data Point 1]……

[LSB, Data Point 512] [MSB, Data Point 512] [LSB, Data Point 256] [MSB, Data Point 256]

– The first few data points are placeholder values and need to be ignored. The number of data points to be skipped before getting the first real data value is the number returned by the Select_ADC Command in step 6.

16. Z314,0 *(CCD Stop)*

– Controller responds with an 'o'.

COMMAND SUMMARY

Communications Conventions:

1. This is a definition of the communications that should occur when a computer interfaces with the spectrometer controllers.

2. Whenever you see <CR>, it means to use ASCII Carriage Return. (13 in DECIMAL).

3. Any string that is placed inside of a pair of double quotes, i.e. "abcd" means that you should send all characters exactly as shown inside the double quotes. You should include spaces, But do not include the double quote characters. Also if you see a character symbolized using <> in a string, you send the character only, not the symbols. For example: if you see"1,Ø ,<Null>", you should send only 5 ASCII characters: the 1, the comma, the Ø, another comma, and the Null character (decimal Ø).

4. Anything placed inside of a pair of square brackets, i.e. [Ø..1] indicates the valid range of the parameter associated with it.

Command Syntax and Confirmation :

This section outlines the rules of syntax required to successfully send commands and recognize confirmation responses that will inform you that the spectrometer controller has or has not received a valid command string.

Standard Commands:

As a prerequisite to sending operational commands to your spectrometer controller, you must establish communications first (refer to previous sections). Only after communications have been successfully established can other commands can be sent to it.

All commands are ASCII text, unless specifically noted otherwise. The "Standard Commands" are only 1 character and are detailed in the command descriptions below. For commands requiring additional input, the command is sent, immediately followed by the relevant parameter(s). The command parameters form a string of characters representing the data that is sent or received.

Extended Commands:

The command set is not limited by the number of characters in the English alphabet.

The character "Z" has been assigned for access to extended commands. The first parameter following this command is recognized as the number representing the extended command you wish to execute. Depending on the command number you send, you may be required to send more parameters following the command.

Otherwise, extended commands are handled in the same way as the standard commands above.

Pseudo-Commands:

These commands perform some special utility functions. The syntax is the same as standard single byte commands, except that the bytes sent as commands are not text. They are ASCII values such as <222>. In this document, they are expressed as decimal ASCII values, between the < and > symbols. The pseudo-command to re-boot the spectrometer controller ASCII Code 222 is expressed as <222>. Another difference is that pseudo commands do not generate a response character from the CCD controller.

CCD COMMAND SET

The table on the following pages offers a summary of the CCD Commands. Following Command Table is a detailed description of each command.

Command	Description	# of Params	Parameters Sent	Parameters Received	Notes
Z300	CCD Init	1	CCD Number (Always 0)	Confirm, 0	No Response if CCD is not ready
Z301	CCD Set Exposure Time	2	CCD Number (Always 0), Integration Time (in milliseconds)	Confirm	
Z302	CCD Set Gain	2	CCD Number (Always 0), CCD Gain	Confirm	
Z303	CCD Read Gain	1	CCD Number (Always 0)	Confirm, CCD Gain	
Z305	CCD Set # of Flushes	2	CCD Number (Always 0), Number of Flushes	Confirm	
Z311	CCD Start	2	CCD Number (Always 0), Shutter State (0-Closed, 1-Open)	Confirm	
Z312	CCD Status	1	CCD Number (Always 0)	Confirm, Status Byte	Status Byte is 0 if acquisition is complete, Non-Zero if acquisition is still in progress
Z314	CCD Stop	1	CCD Number (Always 0)	Confirm	
Z315	CCD Read Image	1	CCD Number (Always 0)	Confirm, Data	
Z316	CCD Read Scan	1	CCD Number (Always 0)	Confirm	
Z317	CCD Reset Image	1	CCD Number (Always 0)	Confirm	
Z318	CCD Reset Scan	1	CCD Number (Always 0)	Confirm	
Z319	CCD Read Next Scan	1	CCD Number (Always 0)	Confirm	
Z320	CCD Set Shutter	2	CCD Number (Always 0), Shutter State (0-Closed, 1-Open)	Confirm	
Z325	CCD Define ACQ Format	3	CCD Number (Always 0), Acquisition Format (0 - image, 1 - scan,), Number	Confirm	

Command	Description	# of Params	Parameters Sent	Parameters Received	Notes
			of Areas		
Z326	CCD Define Area	8	CCD Number (Always 0), Area Number (Start at 0), X Origin, Y Origin, X Size, Y Size, X Binning, Y Binning	Confirm	One Command per Area
Z327	CCD Get Data Size	1	CCD Number (Always 0)	Confirm, Scan Size, Image Size	
Z328	CCD Set Chip Stuff	19	CCD Number (Always 0), port, total number of active x-pixels, total number of active y-pixels, number of serials pixels before active, number of serials pixels after active, number of parallel rows before active, number of parallel rows after active, readout register location and direction, minimum temperature (K*100), maximum temperature(K*100), minimum shutter time (msec), maximum shutter time (msec), minimum gain, maximum gain, horizontal pixel spacing, vertical pixel spacing, total parallel pixels, total serial pixel	Confirm	

Command	Description	# of Params	Parameters Sent	Parameters Received	Notes
Z329	CCD Go Blast	3	CCD Number (Always 0), Number of Acquisitions, Shutter Mode (0-Open shutter at start of experiment, 1-Open shutter for each accumulation)	Confirm	
Z330	CCD Trigger Enable	2	CCD Number (Always 0), Trigger Mode (0-Off, 1-on, 2-blast sync	Confirm	
Z331	CCD Set Blast Interval	2	CCD Number (Always 0), Blast Interval (milliseconds)	Confirm	
Z340	CCD Write X Data	4	CCD Number (Always 0), Chip Select, Address, Number of Data Bytes	Confirm	After receiving Confirm, send the data in binary. See Manual and Programming Notes
Z341	CCD Read X Data	4	CCD Number (Always 0), Chip Select, Address, Number of Data Bytes	Confirm	
Z343	CCD Set State Table Start Address	3	CCD Number (Always 0), Address Index, Address	Confirm	
Z344	CCD Get State Table Status	CCD Number (Always 0)	Confirm		
Z345	CCD Set MUX and READ ADC		2	CCD Number (Always 0), MUX Channel Code (192-STAGE 1, 193-A_EXT, 194-REF-RO, 195-VOG, 196-VABD, 197-V5V25_RO, 198-V15_R0, 199-SET_CS, 200-VTHERM, 201-CCD_TEMP, 202-	Confirm, Data

Command	Description	# of Params	Parameters Sent	Parameters Received	Notes
				SINK_TEMP, 203-ADC reference, 204 - STAGE2, 205-STAGE3, 206-STAGE4, 207-Analog GND.)	
Z348	CCD Set Flush Mode	2	CCD Number (Always 0), Flush Mode	Confirm	
Z349	CCD Get Last Readout Time	1	CCD Number (Always 0)	Confirm	
Z350	CCD Get Last Reference Sum	1	CCD Number (Always 0)	Confirm	
Z351	CCD Go ICCD	5	CCD Number (Always 0), Pulse Delay (milliseconds), Pulse Width (milliseconds), Number of Pulses to Count, Fast Burst (??)	Confirm	Version 1.68 or Higher
Z352	CCD Select ADC	2	CCD Number (Always 0), ADC (0 for 16-bit, 1 for 14-bit)	Confirm, Number of Data Points to skip	Version 1.8 EPROM or Higher

CCD Command Detailed Information

CCD_INIT **Z300**

Initializes CCD hardware. If hardware is not detected the driver will perform hardware emulation as much as possible which is very useful for software testing. This command must be called first.

Input format	Output format
"Z300,PARAM<CR>" PARAM: CCD number - always 0	"o" + "0<CR>"
Example	
Send: "Z300,0<CR>"	Receive: "o" + "0<CR>"

CCD_SET_EXPOSURE_TIME **Z301**

Sets the CCD integration time in milliseconds.

Input format	Output format
"Z301,PARAM1,PARAM2<CR>" PARAM: 1. CCD number - always 0 2. integration time (msec)	"o"
Example	
Send: "Z301,0,1000<CR>"	Receive: "o"

CCD_SET_GAIN **Z302**

Sets the CCD gain.

Input format	Output format
"Z302,PARAM1,PARAM2<CR>" PARAM: 1. CCD number - always 0 2. CCD gain	"o"
Example	
Send: "Z302,0,2<CR>"	Receive: "o"

CCD_READ_GAIN Z303

Returns the current gain setting.

Input format	Output format
"Z303,PARAM<CR>" PARAM: CCD number - always 0	"o" + "RESULT<CR>" RESULT: CCD gain
Example	
Send: "Z303,0<CR>"	Receive: "o" + "2<CR>"

CCD_SET_NUMBER_OF_FLUSHES Z305

Sets the number of flushes of the CCD which will occur prior to starting an acquisition.

Input format	Output format
"Z305,PARAM1,PARAM2<CR>" PARAM: 1. CCD number - always 0 2. number of flushes	"o"
Example	
Send: "Z305,0,1<CR>"	Receive: "o"

CCD_SET_TEMPERATURE Z307

Sets the CCD to the specified temperature (degrees K multiplied by 100). Use CCD_READ_TEMPERATURE to determine stability, this typically takes 20 minutes. NOTE: you should reinitialize hardware after a temperature change; details unknown.

Input format	Output format
"Z307,PARAM1,PARAM2<CR>" PARAM: 1. CCD number - always 0 2. temperature (°K × 100)	"o"
Example	
Send: "Z307,0,29000<CR>"	Receive: "o"

CCD_READ_TEMPERATURE Z308

Reads CCD temperature in degrees K multiplied by 100.

Input format	Output format
"Z308,PARAM<CR>" PARAM: CCD number - always 0	"o" + "RESULT<CR>" RESULT: CCD temperature (°K × 100)
Example	
Send: "Z308,0<CR>"	Receive: "o" + "29000<CR>"

CCD_READ_CHIP_STUFF Z310

Returns CCD hardware specific information.

Input format	Output format
"Z310,PARAM<CR>" PARAM: CCD number - always 0	"o" + "RESULT1,...,RESULT18<CR>" RESULT: 1. port 2. total number of active x-pixels 3. total number of active y-pixels 4. number of serial pixels before active 5. number of serial pixels after active 6. number of parallel rows before active 7. number of parallel rows after active 8. readout register location and direction 9. minimum temperature (°K × 100) 10. maximum temperature (°K × 100) 11. minimum shutter time (msec) 12. maximum shutter time (msec) 13. minimum gain 14. maximum gain 15. horizontal pixel spacing ($\mu m \times 10^{-1}$) 16. vertical pixel spacing ($\mu m \times 10^{-1}$) 17. total parallel pixels 18. total serial pixels
Example	
Send: "Z310,0<CR>"	Receive: "o" + "848,1024,256,8,8,0,0,5,0,29000,0,400000000,0, 4,270,270,256,1040<CR>"

CCD_START Z311

Sets up the data acquisition sequence of:

1. close shutter
2. perform specified number of flushes (default - 1)
3. start integration (with open or closed shutter)
4. transfer data from chip to RISC memory

Input format	Output format
"Z311,PARAM1,PARAM2<CR>" PARAM: 1. CCD number - always 0 2. shutter flag - 1 if open during integration, 0 if closed	"o"
Example	
Send: "Z311,0,1<CR>"	Receive: "o"

CCD_STATUS **Z312**

Checks if an acquisition is finished or not.

Input format	Output format
"Z312,PARAM<CR>" PARAM: CCD number - always 0	"o" + "RESULT<CR>" RESULT: CCD acquisition status - non-zero integer if an acquisition is in progress, 0 if finished
Example	
Send: "Z312,0<CR>"	Receive: "o" + "2<CR>"

CCD_STOP **Z314**

Stops previously started acquisition.

Input format	Output format
"Z314,PARAM<CR>" PARAM: CCD number - always 0	"o"
Example	
Send: "Z314,0<CR>"	Receive: "o"

CCD_READ_IMAGE **Z315**

Transfer all the scan data from the controller using binary transfer conventions. The number of data points transferred is equal to the number returned from the CCD_GET_DATA_SIZE call plus 1 (Status byte, 1 data point = 2 bytes).

Input format	Output format
"Z315,PARAM<CR>" PARAM: CCD number - always 0	Binary data transfer
Example	
Send: "Z315,0<CR>"	Receive: Binary data

CCD_RESET_IMAGE **Z317**

Resets the controller so that the next CCD_READ_IMAGE call starts at the beginning of the image.

Input format	Output format
"Z317,PARAM<CR>" PARAM: CCD number - always 0	"o"
Example	
Send: "Z317,0<CR>"	Receive: "o"

CCD_SET_SHUTTER **Z320**

Opens or closes the CCD shutter.

Input format	Output format
"Z320,PARAM1,PARAM2<CR>" PARAM: 1. CCD number - always 0 2. shutter flag - 1 for open, 0 for close	"o"
Example	
Send: "Z320,0,1<CR>"	Receive: "o"

CCD_DEFINE_ACQ_FORMAT **Z325**

Sets up the type of acquisition (image or scan) and the number of areas for the acquisition. Only one area can be set in an image mode.

Input format	Output format
"Z325,PARAM1,PARAM2,PARAM3<CR>" PARAM: 1. CCD number - always 0 2. acquisition format (0 - image, 1 - scan) 3. number of areas	"o"
Example	
Send: "Z325,0,0,1<CR>"	Receive: "o"

CCD_DEFINE_AREA **Z326**

Defines physical location of a particular area on the CCD detector and the area's binning. Size and binning are specified in pixels.

Input format	Output format
"Z326,PARAM1,...PARAM8<CR>" PARAM: 1. CCD number - always 0 2. area number (starting from 0) 3. area's x-origin 4. area's y-origin 5. area's x-size 6. area's y-size 7. x-binning 8. y-binning	"o"
Example	
Send: "Z326,0,0,0,0,1024,256,1,1<CR>"	Receive: "o"

CCD_GET_DATA_SIZE Z327

Returns the number of data points that will be returned by the acquisition in Scan mode and Image Mode.

Input format	Output format
"Z327,PARAM<CR>" PARAM: CCD number - always 0	"o" + "Scan Data Size,Image Data Size<CR>"
Example	
Send: "Z327,0<CR>"	Receive: "o" + "1024,262144<CR>"

CCD_WRITE_CHIP_STUFF Z328

Returns the number of data points in the largest scan area or in one image row and total number of data points for this acquisition.

Input format	Output format
"Z328,PARAM1,….PARAM19<CR>" PARAM: 1. CCD number - always 0 2. port (value is not important) 3. total number of active x-pixels 4. total number of active y-pixels 5. number of serial pixels before active 6. number of serial pixels after active 7. number of parallel rows before active 8. number of parallel rows after active 9. readout register location and direction 10. minimum temperature (°K × 100) 11. maximum temperature (°K × 100) 12. minimum shutter time (msec) 13. maximum shutter time (msec) 14. minimum gain 15. maximum gain 16. horizontal pixel spacing ($\mu m \times 10^{-1}$) 17. vertical pixel spacing ($\mu m \times 10^{-1}$) 18. total parallel pixels 19. total serial pixels	"0"
Example	
Send: "Z328,0,848,1024,256,8,8,0,0,5,0,29000,0,400000 000,0,4,270,270,256,1040<CR>"	Receive: "o"

CCD_GO_BLAST **Z329**

Set the controller to take the specified number of acquisitions as fast as possible.

Input format	Output format
"Z329,PARAM1,PARAM2,<CR>" PARAM: 1. CCD number - always 0 2. Number of acquisitions	"o"
Example	
Send: "Z329,0,1<CR>"	Receive: "o"

CCD_WRITE_XDATA **Z340**

Setting the data memory of the controller.

Input format	Output format
"Z340, 0,' + [Chip Select] + ',' + [Address] + ',' + [Number of Bytes] + <CR>" PARAM: 1. CCD number - always 0 2. Chip select 3. Address to write in decimal 4. Size to write in decimal	"o"
Example	
Send: 1. "Z340,512,0,159<CR>" 2. Send data in binary transfer. (159 bytes in this example)	Receive: "o"

CCD_SET_MUX_AND_READ_ADC   Z345

Use to read CCD temperature and other sensors.

Input format	Output format
‘Z345,’ + [CCD #] + ‘,’ + [MUX] + <CR> PARAM: 1. CCD number - always 0 2. MUX channel code – in Decimal 207 - CF hex - Analog Ground 203 - CB hex - ADC Reference 193 - C1 hex - A_EXT 194 - C2 hex - REF_RO 195 - C3 hex - VOG 196 - C4 hex - VABD 197 - C5 hex - V5V25_RO 198 - C6 hex - V15_RO 199 - C7 hex - SET_CS 200 - C8 hex - VTHERN 201 - C9 hex - CCD_TEMP 202 - CA hex - SINK_TEMP 192 - CØ hex - hex - STAGE1 204 - CC hex - STAGE2 205 - CD hex - STAGE3 206 - CE hex - STAGE4	"o" + "RESULT1<CR>"
Example	
Send: "Z345,0,201<CR>"	Receive: "o" + Value (2 Bytes)

Reading Detector Temperature

Use the Z345 to read the C9 hex (201 dec), CB hex(203 dec), and CF hex (207 dec) MUX Channels. The following formula can be used to convert the values to Temperature in Kelvin.

$$\mathrm{K} = \frac{(C9 - CF)\times 3000}{(CB - CF)}$$

Binary transfer protocol.

Binary transfer takes place between the host computer and the controller. We can assign the names "Sender" and "Receiver" to them, depending on which unit sends or receives binary data. The protocol is presented below:

1. Issue the command (size of the transfer usually a part of the parameter list)
2. Get conformation
3. Send the data which has is equal to the size specified in Step 1.

Error Codes

In the case that the input from the host computer causes problem for the controller the latter will return symbolic warning that error has occurred. The warnings can be of the following formats:

1. "b" - an incomplete or erroneous command has been sent.
2. "e" + "ERR_CODE<CR>" - controller attempted operation, but error occurred.

ERR_CODE	Meaning
1	Hardware problem
2	Not available
3	Parameter problem
4	Not initialized
20	CCD: NULL user pointer
21	CCD: Not enough memory
22	CCD: Alt param
23	CCD: Load
24	CCD: Read program
25	CCD: Timeout
26	CCD: Zero loop
30	PDA: Multiscan error
31	Remote: Not enough memory
32	Remote: No data available
33	Remote: Binary transfer error
34	Remote: Illegal call sequence

Appendix A: Glossary

The discussion of detection with charge coupled devices requires some familiarity with the terminology used. This section includes definitions specific to this context for some familiar terms, as well as several unique terms, abbreviations and acronyms.

ADC:

An Analog to Digital Converter (ADC) converts a sample of an analog voltage or current signal to a digital value. The value may then be communicated, stored, and manipulated mathematically. The value of each conversion is generally referred to as a datapoint.

Backthinning:

The depletion layer of a CCD (where the photoelectric effect occurs) is normally partially obscured under the electrode gates, which are formed in layers above the depletion layer. This is due to the constraints of the chip fabrication process. The substrate (or back) of a CCD chip can be etched down to be very thin. Then the chip is mounted so that the signal light is incident on the back rather than the front. The chance a photon has of reaching the depletion region is greater. Thus, the Quantum Efficiency is higher. See the graphs on page showing the greater QE of the backthinned CCD.

Binning:

The charges of adjacent pixels can be combined in the readout register cell for that column or row. This combining is called binning. Binning can be used to select only the illuminated pixels. Binning enables adjustment of the effective detector height from one pixel up to the full height of the CCD. More than one binned area can exist in a given readout.

The signal level increase is directly proportional to the number of pixels binned. However, shot noise only increases as the square root of the number of pixels (Felgett's S/N Advantage). Thus, signal to noise ratio is improved. Readout-associated noise is also reduced because the total signal from the binned number of pixels is combined into a single analog to digital conversion.

Binning also enables selective readout of multiple spectra. The signals from several samples can be optically collected simultaneously and imaged to vertically separated parts of the spectrograph entrance slit. This will result in vertically separated spectra imaged across the CCD. By binning the heights of these spectra, each binned area can be captured as a separate spectrum from the same readout cycle. The dark signal from unused pixels between the spectra can be discarded. Signal/Noise is improved by discarding dark signal from non-illuminated pixels.

Charge Coupled Device:

The CCD is a solid state photodetector array made of silicon. It is essentially one continuous photosensitive material. Individual pixels or picture elements are defined by a grid of three electrode gates in the X and Y directions. The charge is collected under the gate with the greatest potential. During the readout cycle, the voltages applied to the gate electrodes are manipulated to move the charge across the pixels to the output register at the edge of the array. In contrast, PMT's and other single channel detectors can measure only one intensity at a time. PDA detectors measure both intensity and wavelength. The CCD simultaneously measures intensity, wavelength and position differences projected along the height of the spectrograph image plane.

Charge Transfer Efficiency (CTE):

The percentage of charge moved from one pixel to the next is the charge transfer efficiency. The CCD has a high CTE if the pixels are read out slowly. As the speed at which the charge is transferred is increased, increasing amounts of the charge is left behind. The residual charge combines with the charge of the next pixel as it is moved into the cell. Therefore, using too high a transfer rate deforms the image shape; it smears the charge over the pixels that follow in the readout cycle. Temperature also affects CTE. Below -140°C the movement of the charges becomes sluggish, and, again, the image becomes smeared.

Correlated Double Sampling:

This is a technique used to increase the signal to noise ratio of each datapoint detected. Minute charges are unavoidably retained in the readout register between one sample and the next. Even though the readout register is reset after each point is read, some charge will persist. At the extremely low signal levels that are typical for cooled CCD detection, these charges are significant. By sampling this retained charge, amplifying and inverting it, it can be canceled by combining it with the actual signal which is amplified, but not inverted. The combined signal is then passed to the ADC to be processed as a datapoint. This process ensures that only the charge due to the signal in each pixel is measured.

Cosmic Ray Events:

Cosmic Rays are high energy particles from the sun. Although they penetrate all detectors, their effect usually is masked by dark current. The dark signal of an LN_2 cooled CCD is so low that cosmic rays may be detected. In the active area of a typical array, about 10 events per minute may occur. Compared to very weak signals, detected cosmic ray events can be quite distracting. To minimize the effects of cosmic rays, one can use the smallest section of the chip that the experiment allows, and use the least integration time possible. **Variable Gain** can help to reveal weak signals. Mathematical treatment of the data can also be used to remove the spurious spikes in spectra. Refer to the software manual for more about cosmic ray spike removal.

Dark Signal:

Dark signal is generated by thermal agitation. This signal is directly related to exposure time and increases with temperature. The dark signal doubles with every 7^{o}C increase in chip temperature above -25^{o}C. The more dark signal, the less dynamic range for experimental signal. This signal accumulates for the entire time between readouts or flushes, regardless of whether the shutter is open or closed. Dark signal is also generated during the charge transfer cycles of the CCD.

Dynamic Range:

The **Dynamic Range** is the ratio of the maximum and minimum signal measurable. The dynamic range of the chip can be greater than that of the system which is limited by the **ADC**. A 16 bit ADC limit is 65,535 (2^{16}-1) counts. A 14 bit ADC is limited to 16,383 counts. **Variable Gain** can be used to shift the ADC range to match the potential well capacity or signal levels of a given spectral measurement. In this way, stronger or weaker signals can be accommodated with optimal Dynamic Range.

On a pixel by pixel basis, the most intense detectable signal, the saturation level, is the lesser of the **Potential Well Capacity** of the pixel or the ADC maximum limit. When pixels are binned, individual pixels within a binned area may saturate if the intensity is concentrated. Also, the well capacity of the readout register will limit the total signal that can be binned from a given row or column of a binned area.

The weakest detectable signal is limited by the **Dark Level** plus the **Readout Noise**.

Electrons/Count:

Electrons per count is a value indicating how many electrons are needed to be identified by the **ADC** as the smallest measurable unit, or Count. The total “Counts” of a given datapoint are comprised of electrons from a variety of sources, including: Photoelectrons (signal), Dark Level, Read Out Noise, Bias Level, and Amplifier Noise.

Felgett's Advantage:

Multichannel detection provides an improvement in signal to noise ratio, as compared to single channel (scanned) spectral detection. Because the multichannel detection acquires a number of spectral elements simultaneously, the S/N is improved by a factor proportional to the square root of the number of channels acquired.

Flush:

To reduce noise and maximize dynamic range at the CCD, the dark charge that has accumulated on the chip can be rapidly removed by flushing. The effect of flushing the array is similar to a readout cycle in that the charges are cleared from the pixels. But a flush dumps the charges without conversion. A flush is much faster than a readout. Flushing is only necessary when there is an appreciable time between readouts.

Full Well Capacity:

Full well capacity is the measure of how much charge can be stored in an individual pixel. This specification varies for each chip type. It depends on the doping of the silicon, architecture and pixel size. The quantum well capacity is usually around 300,000 electrons. The greater the well, the greater the **Dynamic Range**. A chip with a larger full well capacity can record a higher signal level before saturating. See also **Variable Gain**.

Linearity:

When photo response is linear, if the light intensity doubles, the detected signal will double in magnitude as well. Nonlinear response at medium to high intensities is usually due to amplifier problems, and at very low light levels poor charge transfer efficiency. A CCD's response is linear, once the bias is subtracted. Another definition of linearity is applied to the spectral positioning accuracy or tracking error of a spectrometer drive mechanism.

Noise:

Noise is common to all detectors. The total amount of signal that exists is less important than the ratio of signal magnitude to noise magnitude (S/N). With a high signal to noise ratio a signal peak can be discerned even though signal counts per second may be low. The noise components for CCD arrays are as follows:

Amplifier Noise: Some noise is introduced in the process of electronically amplifying and conditioning the signal read from the detector before conversion to a digital value. Part of **Readout Noise**.

Conversion Noise: During the conversion of an analog signal to a digital datapoint some electronic noise is introduced, statistical variations occur in the least significant bit of the converted data. Part of **Readout Noise**.

Dark Noise: The detector will integrate a thermally generated **Dark Current** at all times, whether light is reaching the detector or not. Most of the dark current signal is a steady state level that can be subtracted, and so will not ultimately contribute to the noise. However, a component of **Dark Current** is **Dark Noise** due to statistical variations in the Dark Current. The Dark Noise component increases as the square root of the Dark Current. Dark Current, and therefore Dark Noise, can be reduced by cooling. The LN_2 cooled CCD is one of the least noisy detectors available, with less than one electron/pixel/hour of dark signal. If the signal level is below saturation, increasing the signal integration time per readout will minimize the effect that dark noise has on the acquired signal. If the signal level is too high, summing multiple reads can give similar improvement. (See **Readout Noise** below.)

Readout Noise: The electronic noise impressed on the signal during the readout and digitizing of the signal. For convenience, usually all of the noise associated with resetting, amplifying, and converting the signal are considered as readout noise. When averaging signal by acquiring over a long interval of time, increasing the signal integration time per readout rather than summing multiple readouts is preferred. This will proportionately reduce the readout noise component in the acquired signal. However, the integration time must be short enough to prevent saturation of any individual pixels and to keep the digital signal for any datapoint below the ADC limit.

Reset Noise: Following pixel or bin readout, the readout register is reset to a level approaching zero charge. **Reset Noise** is the non-uniformity in the resetting. This is canceled by **Correlated Double Sampling**. Part of **Readout Noise**.

Shot Noise: This is due to the random statistical variations of light. It includes both experimental and dark signal components. Shot noise is equal to the square root of the number of electrons generated. Its effect can be minimized by increasing signal intensity, signal integration time, or summing a number of readouts.

Photoelectric Effect:

Some materials respond to illumination from photons by releasing electrons. When light of sufficient energy hits a photosensitive material, an electron is freed from being bound to a specific atom. Such materials include the P-N junctions of the silicon photodiodes used in CCD arrays. The energy of the light must be greater than or equal to the binding energy of the electron to free an electron. The shorter the wavelength, the higher the energy the light has.

Photoelectron

A photoelectron is an electron that is released through the interaction of a photon with the active element of a detector. The photoelectron could be released either from a junction to the conduction band of a solid state detector, or from the photocathode to the vacuum in a PMT. A photoelectron is indistinguishable from other electrons in any electrical circuit.

Photo Response NonUniformity (PRNU)

PRNU is the peak to peak difference in response between the most and least sensitive elements of an array detector, under a uniform exposure giving an output level of $V_{Sat}/2$. These differences are primarily caused by variations in doping and silicon thickness.

Quantum Efficiency (QE):

The efficiency of the photoelectric effect of a detector can be quantified. The quantum efficiency of a detector is the ratio of number of photoelectrons produced to the number of photons impinging on a photoactive surface. A QE of 20% would indicate that one photon in five would produce a distinguishable photoelectron. CCD's are made of silicon which has a high QE, about 45-50% at its peak at 750 nm. The quantum efficiency of a detector is determined by several factors. These include the material's intrinsic electron binding energy or band gap, the reflectivity of the surface, the thickness of the surface, and energy of the impinging photon (h). The QE varies with the wavelength of incident light. Standard CCD's typically have a peak QE of about 50%. Back thinned CCD's may peak at about 85%. The QE at short wavelengths can be improved by coating with fluorescent dye that converts UV light to longer wavelengths where the quantum efficiency of the chip is higher. The graphs on page **Error! Bookmark not defined.** show the QE of several available CCD's.

Readout Time:

The **Readout Time** of a CCD is the interval required to move the charges from their locations in the array to the readout registers, sample the charges, amplify them and convert them to datapoints. A consideration with a CCD is that the time between sample exposures can be longer than linear array detectors. This is because the readout requires that the charges be moved across the array (charge coupled). Also, the correlated double sampling readout technique requires more time per pixel.

Responsivity

Responsivity is the ratio of output voltage to corresponding exposure ($\mu J/cm^2$). Technically it is measured at $V_{Sat\ min}/2$ under specified conditions of illumination, readout rate, and temperature.

Saturation Level

The maximum signal level that can be accommodated by a device is its saturation level. At this point, further increase in input signal do not result in a corresponding increase in output. This term is often used to describe the upper limit of a detector element, an amplifier, or an ADC.

Spectral Response:

Most detectors will respond with higher sensitivity to some wavelengths than to others. The spectral response of a detector is often expressed graphically in a plot of responsivity versus wavelength.