ST-7 Microcontroller Interface

November 8, 2011

This document describes the interface between the ST-7's microcontroller and the host computer. Physically the interface consists of a 4-bit bidirectional data bus with handshaking both ways. This document describes the nibble level signaling, the command packet protocol and the high level commands.

The Physical Layer

At the lowest level the interface between the microcontroller and the host computer consists of a 4 bit bidirectional data bus with handshake lines going both directions and an additional sync line from the PC to the microcontroller. Transmission from the host to the microcontroller is handled a nibble at a time, always transmitting nibbles in pairs, using the following protocol:

State	PC	Micro	Notes
Idle	-	-	Both Handshake Lines Low
Send Start Pulse	Raise Sync	-	Start is Sync and HS Out High
Send High Nibble	Write Data	-	
	Raise HS Out	-	HS Out Signals Data Ready
	-	Read Data	
	-	Raise HS In	HS In Signals Data Ack.
Negate Start Pulse	Lower Sync		Sync high for 1st nibble only
Send Low Nibble	Write Data	-	Data written in Pairs of Nibbles
	Lower HS Out	-	
	-	Read Data	
	-	Lower HS In	
Repeat of Nibble Pair Transmission for all Pairs			
Ready for Response	Raise HS Out	-	PC is Ready for Response after acknowledging Micro's reception of the Last Nibble sent
Rec. High Nibble	-	Write Data	Micro sends response when
		Raise HS In	HS Out goes High
	Read Data	-	
	Lower HS Out	-	
Rec. Low Nibble		Write Data	
		Lower HS In	
	Read Data		
	Raise HS Out*		*On last nibble PC does not raise HSO
Repeat of Nibble Pair Reception for all Pairs			

The Packet Layer

Data is sent from the host to the microcontroller and vice versa in packets. The host acts as the master, initiating communications, and the microcontroller acts as a slave, sending data to the host only in response to a query from the host. The packet consists of a start nibble pair (hex A followed by 5) followed by a command nibble (0 through F), a data length nibble (0 thru F) followed by N data nibble pairs where N is the value in the data length nibble. The Packet is shown below:

Nibble	Value	Comment
1	A	Start Sequence
2	5	" "
3	0-F	Command, 16 supported
4	0-F	N=Data Length in Nibble Pairs or Bytes Note some packets have no data and N=0
5	Data 1 High	High Nibble of First Byte
6	Data 1 Low	Low Nibble of First Byte
.	.	.
.	.	.
.	.	.
3+2N	Data N High	High Nibble of Last Byte
3+2N	Data N Low	Low Nibble of Last Byte

In response to a packet from the host, the microcontroller can have several responses, which are sent only after the complete packet has been received by the microcontroller. If the command is an unimplemented command then the microcontroller should return to the host a CAN byte (CAN = $18). If the command is a valid command but the data length is incorrect then the microcontroller should return a NAK byte (NAK = hex $1F). If the command is valid and the length is valid then the microcontroller should either return a single ACK nibytebble (ACK = $06) or a response packet, as appropriate for the command being issued. Response packets start with the same 3 nibble sequence of start nibbles and data length that the command packets have but the length is the length of the response.

When packets are sent from the host to the micro or back they are send in bursts. If the next nibble in the packet is not received or able to be sent within 25 ticks (1 tick = 0.01 second) the sender or receiver resets to the idle condition which is with HSO and HSI both 0. The start of a packet is designated by the A nibble with HSI high and HSO low.

The Commands

This section describes each of the commands supported by the microcontroller.

StartExposure - Command 0

(4 Byte Command, No Response)

The StartExposure command starts an integration in one of the CCDs. At the start of the integration the shutter may need to be opened, the integration is timed, and at the end of the exposure the

shutter may then be closed. The shutter activity during the exposure is controlled by data within the StartExposure command. The data for this command is shown below:

Nibble	Meaning
1	High Nibble of the Exposure Time (b_{23}-b_{20})
2	Mid Nibble " " (b_{19}-b_{16})
3	Mid Nibble " " (b_{15}-b_{12})
4	Mid Nibble " " (b_{11}-b_8)
5	Mid Nibble " " (b_7-b_4)
6	Low Nibble of the Exposure Time (b_3-b_0)
7	Upper Nibble of the Flags
8	Lower Nibble of the Flags
	b_1b_0 = ABG State -0=Low, 1=Clk Low, 2=Clk Med, 3=Clk Hi
	b_3b_2 = Shutter Control - 0 = Leave Shutter Alone,
	1 = Open Shutter at Start, Close at End
	2 = Close Shutter at Start, Leave Closed
	b_4 = CCD - 0 = Imaging CCD, 1 = Tracking CCD
	b_5 = External Trigger Out when 1
	b_6 = External Tracking CCD when 1
	b_7 = 1 for exposure times in milliseconds, 0 for exposure
	times in hundredths of a second

The exposure time is in hundredths of a second. If an exposure is already in progress for the specified CCD then abort the old exposure and start the new one. *The host response to a StartExposure command is an ACK.*

EndExposure - Command 1

(1 byte Command, No Response)

The EndExposure command is used by the host computer to terminate an exposure prematurely. The data for this command is shown below:

Nibble	Meaning
1	Upper Nibble of the Flags
2	Lower Nibble of the Flags
	b_0 = CCD - 0 = Imaging CCD, 1 = Tracking CCD
	b_1-b_7 = Spare

If the requested CCD does not have an exposure in progress then ignore the command, otherwise close the shutter as was requested in the StartExposure command and terminate the exposure. *In either case return an ACK.*

RegulateTemp - Command 2

(3 Byte Command, No Response)
The RegulateTemp command is used by the PC to enable or disable temperature regulation in the camera. The data for this command is shown below:

Nibble	Meaning
1	Flag Nibble
	b_0 = Enable Regulation - 0 = Disabled, 1 = Enabled
	b_1 = Setpoint / Override - 0 = Setpoint, 1 = Override
2	Upper Nibble of the Setpoint / Override
3	Middle Nibble " "
4	Low Nibble of the Setpoint / Override
5	High Nibble of the TE Cooler Preload
6	Low Nibble of the TE Cooler Preload

If the Setpoint / Override flag is set then use the Setpoint value for the temperature controller output rather than as a setpoint. *The host response to a RegulateTemp command is an ACK.*

TempStatus - Command 3
(No Command, 5 Byte Response)
The TempStatus command is used by the PC to interrogate the camera's temperature regulation. There is no data for this command. *The response data for this command is shown below:*

Nibble	*Meaning*
1	*High Nibble of the Flags*
2	*Low Nibble of the Flags*
	b_0 = Regulation Enabled - 0 = Disabled, 1 = Enabled
3	*High Nibble of the Setpoint*
34	*Low Nibble of the Setpoint*
45	*High Nibble of the CCD Thermistor*
6	*Low Nibble of the CCD Thermistor*
7	*High Nibble of the Ambient Thermistor*
8	*Low Nibble of the Ambient Thermistor*
9	*High Nibble of the Power Output*
10	*Low Nibble of the Power Output*

Relay - Command 4
(5 Byte Command, No Response)
The Relay command is used by the PC to activate one or more relays in the camera. The data for this command is shown below:

Nibble	Meaning
1	High Nibble of the +X Time
2	Low Nibble of the +X Time

3	High Nibble of the -X Time
4	Low Nibble of the -X Time
5	High Nibble of the +Y Time
6	Low Nibble of the +Y Time
7	High Nibble of the -Y Time
8	Low Nibble of the -Y Time
9	High Nibble of the Flags
10	Low Nibble of the Flags b0 = X16 time

The relay times are in multiples of a hundredth of a second. Zero signifies forcing the relay off. *The host response to a Relay command is an ACK.*

Pulse - Command 5

(5 Byte Command, No Response)

The Pulse command is used by the PC to activate the CFW-6. The data for this command is shown below:

Nibble	Meaning
1	High Nibble of the Number of Pulses
2	Low Nibble of the Number of Pulses
3	High Nibble of the Pulse High Width
4	Upper-Mid Nibble " "
5	Lower-Mid Nibble " "
6	Low Nibble of the Pulse High Width
7	High Nibble of the Pulse Low Width
8	Upper-Mid Nibble " "
9	Lower-Mid Nibble " "
10	Low Nibble of the Pulse Low Width

The pulse width and pulse periods are in arbitrary units. Zero pulses signifies forcing the CFW-6 output off. *The host response to a Pulse command is an ACK.* The Microcontroller ACKS the command then shuts off interrupts while generating the pulses.

GetVersion - Command 6

(No Command Bytes, 2 Byte Response)

The GetVersion command is used by the PC to interrogate the cameras features. There is no data for this command. *The data for the response to this command is shown below:*

Nibble	*Meaning*
1	*High Nibble of the Firmware Version*
2	*Upper-Mid Nibble " "*
3	*Lower-Mid Nibble " "*
4	*Low Nibble of the Firmware Version*

The firmware version is interpreted as a 4 digit BCD number in the format XX.XX.

EEPROM - Command 7

(2 Byte Command, 2 Byte Response)

The EEPROM command is used by the PC to read or write the EEPROM. The data for this command is shown below:

Nibble	Meaning
1	High Nibble of the Address
2	Low Nibble of the Address
3	High Nibble of the Data for Write
4	Low Nibble of the Data for Write

The most significant bit of the address is 0 for a Write and 1 for a Read. For Write commands the Data is contained in the packet. For read commands ignore the Data in the packet. *The host response to a EEPROM write command is an ACK and the response for an EEPROM read command is the data read from the EEPROM.*

MiscControl - Command 8
(1 Byte Command, No Response)
The MiscControl command is used by the PC to activate the fan, LED, etc. The data for this command is shown below:

Nibble	Meaning
1	High Nibble of the Flags
2	Low Nibble of the Flags
	b_0b_1 = Shutter State - 0 = Leave Shutter Alone,
	1 = Open Shutter, 2 = Close Shutter,
	3 = Reinitialize Shutter
	b_3b_2 = LED State - 0 = LED Off, 1 = LED On
	2 = LED Blink Low, 3=LED Blink High
	b_4 = Fan - 0 = Disabled, 1 = Enabled
	b_5-b_6 = Same as b0b1 for External Tracker shutter on ST-L
	b_7 = spare

The host response to a MiscControl command is an ACK. When reinitializing the shutter the Microcontroller ignores communications with the Host.

Status - Command 9
(No Command Bytes, 2 or 3 Byte Response)
The Status command is used by the PC to interrogate the status of the camera. There is no data for this command. *The response data for this command is shown below:*

ST-7 Type Status (3 Byte Response)

Nibble	*Meaning*
1	*High Nibble of the Flags1*
2	*Low Nibble of the Flags1*
	b_0b_1 = Imaging CCD state - 0 = Idle, 1=Pre Shutter,
	2 = Integrating, 3 = Post Shutter
	b_2b_3 = Tracking CCD state - 0 = Idle, 1=Pre Shutter,
	2 = Integrating, 3 = Post Shutter
	b_4b_5 = Shutter State - 0 = Open, 1=Closed
	2 = Opening, 3 = Closing
	b_6b_7 = LED State - 0 = LED Off, 1 = LED On
	2 = LED Blink Low, 3=LED Blink High
3	*High Nibble of the Flags2*
4	*Low Nibble of the Flags2*
	b_0 = Fan - 0 = Disabled, 1 = Enabled
	b_1 = CFW-6 - 0 = Inactive, 1 = Active
	b_2 = CFW Input

b_3 = ST-L: External ST-L shutter, 1=Closed, 0=Open
ST-402: 1= Shutter Rewound, 0= Shutter not Rewound
b_4= Relay +X - 0 = Inactive, 1 = Activated
b_5 = Relay -X - 0 = Inactive, 1 = Activated
b_6 = Relay +Y - 0 = Inactive, 1 = Activated
b_7 = Relay -Y - 0 = Inactive, 1 = Activated

5	*High Nibble of Shutter Edge/Filter Position*
6	*Low Nibble of Shutter Edge/Filter Position* *ST-7, etc – Shutter Edge - Nominally 9 ± 1. 255 = No edge detected* *ST-402 – Filter position - 0 = Moving, 1-5=In position, 6=position unknown*

ST-5C/237 Status (2 Byte Response)

Nibble	*Meaning*
1	*High Nibble of the Flags1*
2	*Low Nibble of the Flags1* *b_0b_1 = CCD state - 0 = Idle, 1=Clearing 2 = Integrating, 3 = Transferring* *b_2-b_4= Filter Position - 0 = Moving, 1-5=In position, 6=position unknown* *b_5= Vane/Filter detected - 0 = Vane, 1 = Filter wheel* *b_6= Filter Inited- 0 = Filter wheel type not determined, 1 = Filter wheel determined, set in b_5* *b_7= Spare*
3	*High Nibble of the Flags2*
4	*Low Nibble of the Flags2* *b_0= Relay +X - 0 = Inactive, 1 = Activated* *b_1 = Relay -X - 0 = Inactive, 1 = Activated* *b_2 = Relay +Y - 0 = Inactive, 1 = Activated* *b_3 = Relay -Y - 0 = Inactive, 1 = Activated*

System Test - Command A

(1 Command Byte, ACK Response)

The System Test command is used by the PC to test hardware that only the Microcontroller accesses. Not all cameras support this command. The command data is shown below

Nibble	*Meaning*
1	*High Nibble of the Flags*
2	*Low Nibble of the Flags*
	b_0 = Test Clocks
	b1 = Test Motor
	b2 = Test 5800
	b3 = Activate motor phases for ST-L cameras to align shutter

Tx Bytes - Command B

The Tx Bytes command is used by the PC to send data to the AO or other high-speed Serial Port devices. The length nibble in the outgoing packet is 0 to indicate the packet has extended length and the first byte of the data packet specifies the number of bytes that follow (and are to be sent to the AO). The command returns with the command / length byte set to 0xB1 and one byte of data is returned, indicating the actual number of bytes accepted (ACKed) by the AO. The following AO commands are supported:

A5 + Instrument ID + Length + Command + Data

Command 1 = Tip Tilt (Total of 9 bytes to pass through)

- Data 1 = MS X
- Data 2 = LS X
- Data 3 = MS Y
- Data 4 = LS Y

Command 2 = Set Focus

- Data 1 = Command
  - 1=step out, 2=step in, 3=hard center, 4=soft center
- Data 2 = Checksum? (what does that mean?)

Control CCD - Command C
(2 Command Bytes, ACK Response)
This command was used with the ST-237 Camera to control the CCD. The only command that was implemented was to Clear the CCD

Nibble	Meaning
1	High Nibble of the Flags
2	Low Nibble of the Flags b_0 = Clear the CCD when 1 b_1-b_7 = spare
3	Not used, set to 0
4	Not used, set to 0

The host response to a ControlCCD command is an ACK.

Command 13 – Command D
This command was never implemented.

System - Command E
(Up to 12 Command Bytes, Up to 8 Byte Response)
The System command is a general purpose command whose current use is to read or write memory in the camera. The command data is shown below:

Byte	*Meaning*
1	*Sub-Command (defined below)*
2	*Data length*
3	*LSB of Destination Address*
4	*MSB of Destination Address*
5-12	*Data Bytes*

Sub-Commands: SYS_READ_INT=0, SYS_WRITE_INT=1, SYS_READ_EXT=2, SYS_WRITE_EXT=3, SYS_GET_ROM_SUM=4, SYS_WRITE_SFR=5, SYS_INIT_GA=6, SYS_SET_MOTOR_PHASE=7

The response is ACK for the Writes and Init GA, 2 Bytes for the Get ROM Sum and 8 bytes for the Reads with 1-8 of the bytes filled in.

Readout - Command F
(Up to 17 Command Bytes, Varying Response)
The Readout command is used by the PC to readout pixel data on the USB versions of the cameras. The command data varies for each sub-command as defined below:

Sub-Commands:

RS_DIG_ROW=0 – Camera Digitizes one row of pixels but doesn't download them.
RS_DLP_ROW=1 –
RS_DL_ROW=2 – Camera Downloads one row of pixels.
RS_DLP_ROWS=3 -
RS_DUMP_FIFO=4 – Camera dumps the contents of the Pixel FIFO
RS_DL_SETUP=5 – Camera
RS_DUMP_ROWS=6 – Camera
RS_CLEAR_CCD=7 – Camera clears one of the CCDs
RS_SET_VDD=8 – Turn on or Turn off the Vdd and return it's previous state
RS_WRITE_AD=9 – Write data to one of the A / D Registers
RS_DLPP_ROWS,=10 – Download a number of pipelined Rows
RS_END_READOUT=11 – End the readout for one of the CCDs
RS_MAN_CLOCKS=12 – Enable or disable Manual clocking of the CCD
RS_TRANSFER_KAI=13 – Do a transfer fr a KAI based CCD
RS_SETUP_TDI=14 – Setup for the Biorad TDI mode
RS_GET_PIXCNT=15 – Get the number of pixels in who knows what
RS_OFFSET_PIXELS=16 – Get a row of pixel data for use in the Offset routine
RS_READ_AD=17 – Read the contents of an A / D register
RS_DL_SETUP2=18 – Tell the camera the readout area for cameras with frame buffers
RS_DL_SETUP3=19 – Same as above but with additional data (bottom and right)
RS_DL_IMAGE=20 – Start downloading an entire image
RS_RX_PIXELS=21 – Ask for up to 65535 pixels worth of data

The readout sub-commands have a variable length data packet as required for each individual command. The packet structure is such that the first byte of data to the camera is the Sub-Command as shown below:

Byte	*Meaning*
1	*Sub-Command*
2	*Sub-command Dependent*
3…	" "

Sub-Commands RS_DIG_ROW, RS_DLP_ROW, RS_DL_ROW, RS_DLP_ROWS, RS_DLPP_ROWS, RS_CLEAR_CCD have the following packet structure for data sent to the camera:

Byte	*Meaning*
1	*Sub-Command*
2	*CCD (0=Imager, 1=Tracker, 2=Ext Tracker)*
3	*MSB of Columns*
4	*LSB of Columns*
5	*MSB of Rows*
6	*LSB of Rows*

The response to the RS_DIG_ROW and RS_CLEAR_CCD sub-commands is an Ack. The response to the RS_DLP_ROW and RS_DL_ROW sub-commands is Column Pixels worth of data on the Pixel Pipe. The response to the RS_DLP_ROWS and RS_DLPP_ROWS sub-commands is Column * Rows Pixels worth of data on the Pixel Pipe.

Sub-Command RS_DL_SETUP, Response = Ack

Byte	*Meaning*
1	*Sub-Command*
2	*CCD (0=Imager, 1=Tracker, 2=Ext Tracker)*
3	*Horizontal binning*
4	*Vertical Binning*
5	*MSB of Left*
6	*LSB of Left*
7	*MSB of Right*
8	*LSB of Right*

Sub-Command RS_DUMP_ROWS, Response = Ack

Byte	*Meaning*
1	*Sub-Command*
2	*CCD (0=Imager, 1=Tracker, 2=Ext Tracker)*
3	*Vertical Binning*
4	*MSB of rowWidth*
5	*LSB of rowWidth*
6	*MSB of Rows*
7	*MSB of Rows*
8	*VToHMask*

Sub-Command RS_SET_VDD

Byte	*Meaning*
1	*Sub-Command*
2	*Set Vdd (0=Lower, 1=Raise)*

Response:

Byte	*Meaning*
1	*Sub-Command*
2	*Vdd Was Low (0=Was High, 1=Was Low)*

Sub-Command RS_WRITE_AD, Response = Ack

Byte	*Meaning*
1	*Sub-Command*
2	*Register to Write to*
3	*Data*

Sub-Command RS_END_READOUT, Response = Ack

Byte	*Meaning*
1	*Sub-Command*
2	*CCD (0=Imager, 1=Tracker, 2=Ext Tracker)*

Sub-Command RS_MAN_CLOCKS, Response = Ack

Byte	*Meaning*
1	*Sub-Command*
2	*Manual Clocking (0=Auto Clocking, 1=Manual Clocking)*

Sub-Command RS_TRANSFER_KAI, Response = Ack

Byte	*Meaning*
1	*Sub-Command*

Sub-Command RS_SETUP_TDI

Byte	*Meaning*
3	*Sub-Command*
4	*Row Period*

Response:

Byte	*Meaning*
1	*Sub-Command*
2	*b_0 = Scheduling Error*
	b_1 = Overrun Error

Sub-Command RS_GET_PIXCNT

Byte	*Meaning*
1	*Sub-Command*

Response:

Byte	*Meaning*
1	*Pixel Count, High Byte*
2	*Pixel Count, Low Byte*

Sub-Command RS_OFFSET_PIXELS (STX, STT and STF-8300 Only)

Byte	*Meaning*
1	*Sub-Command*
2	*Channel*
3	*Left, High Byte*
4	*Left, Low Byte*
5	*Length, High Byte*
6	*Length, Low Byte*

Response:

Length pixels from the Pixel Pipe

Sub-Command RS_READ_AD

Byte	*Meaning*
1	*Sub-Command*
2	*Register*

Response:

Byte	*Meaning*
1	*Sub-Command*
2	*Unused*
3	*Register Value, High Byte*
4	*Register Value, Low Byte*

Sub-Command RS_DL_SETUP2, Response = Ack

Byte	*Meaning*
1	*Sub-Command*
2	*CCD (0=Imaging, 1=Tracking, 2=Ext Tracking)*
3	*Horizontal Binning (1 – 255)*
4	*Vertical Binning (1 – 255)*
5	*Top, High Byte*
6	*Top, Low Byte*
7	*Left, High Byte*
9	*Left, Low Byte*
10	*Height, High Byte*
11	*Height, Low Byte*
12	*Width, High Byte*
13	*Width, Low Byte*

Sub-Command RS_DL_SETUP3, Response = Ack

Byte	*Meaning*
1	*Sub-Command*
2	*CCD (0=Imaging, 1=Tracking, 2=Ext Tracking)*
3	*Horizontal Binning (1 – 255)*
4	*Vertical Binning (1 – 255)*
5	*Top, High Byte*
6	*Top, Low Byte*
7	*Left, High Byte*
9	*Left, Low Byte*
10	*Height, High Byte*
11	*Height, Low Byte*
12	*Width, High Byte*
13	*Width, Low Byte*
14	*Bottom, High Byte*
15	*Bottom, Low Byte*
16	*Right, High Byte*
17	*Right, Low Byte*

Sub-Command RS_DL_IMAGE

Byte	*Meaning*
1	*Sub-Command*

Response: An image worth of pixel data whose size was defined by the previous RS_DL_SETUP3 call.

Sub-Command RS_RX_PIXELS

1	*Sub-Command*
2	*Unused*
3	*Pixel Count High Word, High Byte*
4	*Pixel Count High Word, Low Byte*
5	*Pixel Count Low Word, High Byte*
6	*Pixel Count Low Word, High Low*

Response: Pixel Count pixels on the Pixel Pipe