POINT GREY

Technical Application Note TAN2004001

Configuring and testing the RS-232 serial port
Revised December 7, 2010

1.1. Subject

Technical Application Note (TAN2004001): Configuring and testing the RS-232 serial port.

1.2. Applicable Product(s)

Dragonfly®2, Flea®2, Flea®3, Grasshopper® and Grasshopper®2.

1.3. Application Note Description

The purpose of this Technical Application Note is to provide the user with a set of basic instructions on how to configure and test the RS-232 serial port functionality for the Applicable Product(s) above. The Applicable Product(s) is/are equipped with a set of general purpose input/output (GPIO) pins that can be accessed via the Hirose connector on the back of the camera. Different products may use different Hirose connectors; consult your camera's *Technical Reference* or *Getting Started* manual for part numbers and specifications.

Specific GPIO pins are used for RS-232 serial port communications. Consult your camera's *Technical Reference* or *Getting Started* manual for GPIO connector pin layouts, specific wiring information and GPIO electrical characteristics.

In addition to providing some detail on how the buffer system is arranged in the camera system, this document provides a series of examples that illustrate how a user could use their camera system to:

- Transmit Characters to a PC
- Receive Characters from a PC
- Transmitting and Receiving Data Simultaneously

1.3.1. Understanding the camera's serial buffer system

The camera associates two distinct buffers with the serial port; one for transmitting data, and one for receiving data. These two buffers are accessed and controlled via a single set of data access registers. The table below provides a high level description of the data access registers.

Register Address Offset	Register Name	Description
Base offset + 000h	SERIAL_MODE_REG	Shared transmit and receive configuration and status registers.
Base offset + 004h	SERIAL_CONTROL_REG /SERIAL_STATUS_REG	
Base offset + 008h	RECEIVE_BUFFER_STATUS_CONTROL	Receive buffer status control.
Base offset + 00Ch	TRANSMIT_BUFFER_STATUS_CONTROL	Transmit buffer status control.
Base offset + 100h	SIO_DATA_REGISTER	Shared access to transmit and receive buffers. Data written to this range is directly reflected in the transmit buffer. Reads from this register range only provides access to data from the receive buffer after the data has been exposed via a write to register 2004h as illustrated in the examples below.
Base offset + 104h – 1FFh	SIO_DATA_REGISTER_ALIAS	

To obtain the base offsets for these SIO control register addresses, read SIO_CONTROL_CSR_INQ 488h. This register returns the offset for this register set from the 1394 base offset FFFF F0F0 0000h. Because this register returns a quadlet offset, multiply the value returned by 4 to get the actual base offset. Then, treat all of the registers in the table above as offsets from this value.

For example, in many Point Grey cameras, the SIO_CONTROL_CSR_INQ returns 3C0800h. When multiplied by 4, we arrive at 2000h as the actual base offset for this register set.

Offsets vary by camera model. Query SIO_CONTROL_CSR_INQ register 488h to obtain the correct offset for your camera.

For more information, refer to the sections “SIO Control and Inquiry Registers” and “Calculating Register Addresses using Quadlet Offsets” in the *Point Grey Digital Camera Register Reference* or your camera’s *Technical Reference Manual*.

1.4. Application Note Examples

To configure and test these examples:

1. **Connect the camera’s serial port to your PC’s serial port.** Consult your camera’s *Technical Reference* or *Getting Started* manual for:
    a. GPIO connector pin layouts, including which pins are designated for RS-232; and
    b. GPIO electrical characteristics.
2. **Access the camera’s register space.** The easiest way to try this is using the FlyCap demo software included with the *PGR FlyCapture SDK*.

For register definitions and individual bit descriptions, please refer to the Serial Port Input/Output section of the *Point Grey Digital Camera Register Reference* or your camera's *Technical Reference Manua*l.

1.4.1. Transmit Characters to a PC

This example describes how to send four (4) characters from the camera to the serial port on a PC. Microsoft's HyperTerminal program (*Start Menu > All Programs > Accessories > Communications*) is used to display the characters received from the camera. The process detailed by the table below involves the user enabling transmit, verifying that the transmit buffer is ready, writing four characters to the transmit buffer via the data access registers and then verifying that the characters are ready before finally transmitting them.

	Action	Register	Input / Expected Output
1. Plug the camera in and start FlyCap.			
2. Open the Camera Control Dialog and select the Register tab.			
3. Get the current baud rate, character length setting, parity setting and stop bit setting.	Get Register	0x2000	0x060800FF ▪ *0x06 = 19200bps* ▪ *0x08 = 8bit, no parity, 1 stop* ▪ *0xFF = 255 byte buffer*
4. Open a HyperTerminal window and create a new connection, setting the COM Port Settings to match the current camera settings obtained in step 3.			
5. Enable the serial output (transmit).	Set Register	0x2004	0x40000000
6. Verify transmit buffer ready.	Get Register	0x2004	0x40**80**0000
7. Send four (4) characters to the output buffer on the camera.	Set Register	0x2100	0x31323334 ▪ *ASCII = 1234*
8. Verify that the transmit buffer is currently storing 4 bytes worth of characters.	Get Register	0x200C	0xFF**04**0000 ▪ *0xFF = 255 bytes of buffer space remaining* ▪ *0x04 = 4 bytes currently stored and waiting to be transmitted*
9. Send the characters from the output buffer to the PC's serial port.	Set Register	0x200C	0xFF040000 ▪ *HyperTerminal should echo the characters "1234"*

Other Comments

- To send more than four characters, either:
  a. Repeat steps 7 through 9 above, and send characters in sets of four using register 0x2100; or
  b. Do a block write of all the characters using registers 0x2104 – 0x21FF (see the PGR FlyCapture API documentation for information on doing block transfers).
- Although both types of writes to the transmit buffer may have to be quadlet aligned, the number of characters transmitted does not. Subsequent writes to the buffer will simply overwrite characters that were not transmitted during a previous transmit.
- The actual transmit buffer size may be larger than that reported in step 3 above. Consult the Serial Port Input/Output section of the *Point Grey Digital Camera Register Reference* or your camera's

Technical Reference Manual. When this is the case, the "buffer space remaining" that is reported in step 8 will not decrease until the actual buffer space remaining is less than 255 bytes.

1.4.2. Receive Characters from a PC

This example describes how to send four (4) characters from the PC to the camera's serial port. Microsoft's HyperTerminal program (*Start Menu* > *All Programs* > *Accessories* > *Communications*) is used to send the characters received from the camera. The process detailed by the table below involves the user enabling receive, having characters sent to the camera, checking to insure that the receive buffer is ready to be read, verifying that the characters have arrived and then having them transferred to the data access registers before they are read out.

	Action	Register	Input / Expected Output
1. Repeat steps 1 to 4 described above in section 1.4.1.			
2. Enable the serial input (receive).	Set Register	0x2004	0x80000000
3. Verify no receive data framing errors.	Get Register	0x2004	0x80**00**0000 • *0x80040000 indicates a receive data framing error, possibly due to a noisy RS-232 line or incorrect baud rate/port settings.* • *0x80020000 indicates a receive data parity error*
4. Send four (4) characters to the input buffer on the camera. For test purposes, type the characters "ABCD" in the HyperTerminal window.			By default, characters will not be displayed in the HyperTerminal window. To echo typed characters to the screen, select *File* > *Properties* > *Settings* tab > *ASCII Setup…*
5. Verify that the receive data buffer is ready to be read.	Get Register	0x2004	0x80**20**0000
6. Verify that the receive buffer is currently storing 4 bytes worth of characters, which are waiting to be read.	Get Register	0x2008	0x**04**000000
7. Send four (4) characters from the input buffer to the data access register.	Set Register	0x2008	0x00040000
8. Verify that four (4) characters are ready to be read from the data access register.	Get Register	0x2008	0x00**04**0000
9. Read the four (4) characters from the data access register.	Get Register	0x2100	0x41424344 • *Assumes input was "ABCD"*

Other Comments

- To receive more than four characters, either:
  a. Repeat steps above, and receive characters in sets of four using register 2100h; or
  b. Do a block read of all of the characters using registers 0x2104 – 0x21FF. For example, if 12 characters were received (0x2008 = 0x0C000000), Set Register 0x2008 to 0x000C0000 and begin reading the bytes starting at 0x2104 (see the PGR FlyCapture documentation for information on doing block transfers).
- Although both types of reads from the receive buffer may have to be quadlet aligned, the number of characters received does not. Extra characters read will simply be filled with 0's.

- The actual receive buffer size may be larger than that reported in step 3 above. Consult the Serial Port Input/Output section of the *Point Grey Digital Camera Register Reference* or your camera's *Technical Reference Manual*.

1.4.3. Transmitting and Receiving Data Simultaneously

Simultaneous transmitting and receiving of data can be achieved in a manner very similar to that illustrated by the previous two examples. The primary difference is that register 2004h must be set to 0xC0000000 to enable both transmit and receive. Once this has been done transmit and receive transactions can be interleaved as may be required by the application.

