PHANTOM HIGH-SPEED CAMERA

S641 Set-up and Operation **GUIDE**

When it's too fast to see, and too important not to.®

PHANTOM®

VISION RESEARCH / AMETEK®

S641 Set-up and Operation Guide

CONTENTS

The contents of this guide are subject to change without notification.

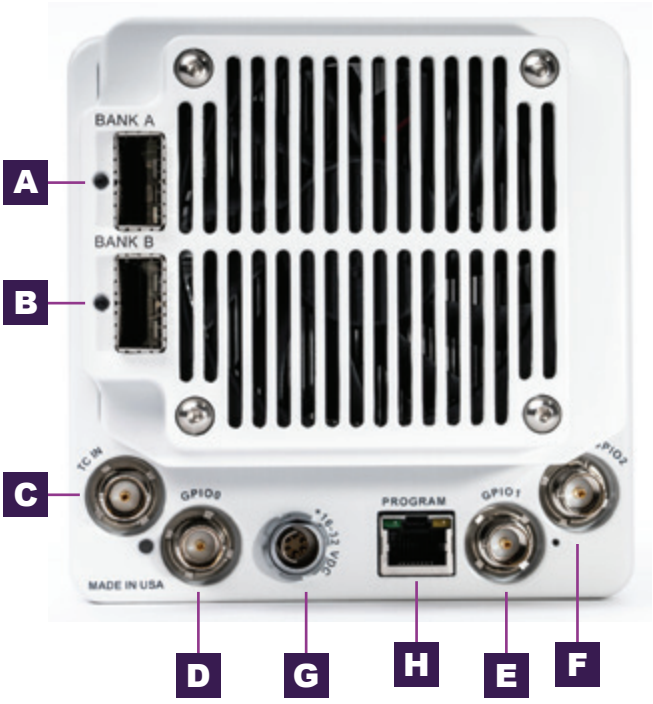
Last Updated: January 2023

PHANTOMHIGHSPEED.COM

1

INTRODUCTION

This Set-up and Operation Guide for the Phantom S641 explains how to set up and operate the S641 and access its unique features. The S641 uses GenICam and CXP 2.0 protocols and this guide should be used in conjunction with the documentation for those protocols. Please refer to that documentation for topics not covered in this Guide.



Connectors

A Bank A Slot for either multimode or single mode QSFP+ Transceiver
Use for 1-bank operation

B Bank B Slot for either multimode or single mode QSFP+ Transceiver
Use with Bank A for 2-bank operation

C Dedicated BNC for Timecode-in for IRIG-B (Modulated/Unmodulated)

D **E** **F** Selectable BNC for GPIO

G 3-pin Fischer for 16-32V DC power

H RJ45 for Firmware uploads

2

IMAGE STREAMING

The Phantom S641 can stream a full image across a single bank/frame grabber or enable multibank streaming with 2 frame grabbers to reach the maximum sensor throughput.

Single Bank Streaming

The S641 can stream a full frame over one bank at ½ the maximum sensor throughput. Typically, this will reduce the maximum frame rates at a given resolution to ½ the maximum frame rate in multibank mode.

To configure single bank streaming, set the **Banks** register to **Banks_A**.

At smaller resolutions, the maximum frame rates may be the same regardless of bank configuration or pixel format. Be sure to reference the maximum frame rate tables when determining your optimum configuration.

Sensor Readout

The S641 sensor readout is not from top to bottom. Instead, rows are read out in groups of 4, starting around the center of the frame and move to the top/bottom edges. Each read contains 4 rows of the horizontal resolution length, with 2 rows above the center of the image and 2 rows below the center. Every subsequent packet contains 4 additional rows, with 2 rows above/2 rows below the preceding packet. Depending on the Banks mode configured, these 4 row packets will be distributed, round-robin, to the selected banks.

For example: a 1024 row image would be read out as follows:

Read Order	Rows	Banks_A	Banks_AB
1	510, 511, 512, 513	A	A
2	508, 509, 514, 515	A	B
3	506, 507, 516, 517	A	A
4	504, 505, 518, 519	A	B

Table 1: Image Row Reordering

Using the Euresys egrabber driver, the framegrabber can be configured to reorganize the incoming frames in line, with no data copy necessary. To set up the stream, adjust the following registers in the GenICam **Stream** module to match the camera setup.

StreamControl Register	Value in Banks_AB Mode
LineHeight	<Horizontal Resolution> * <Bytes Per Pixel>
LinePitch	<Horizontal Resolution> * <Bytes Per Pixel>
StripeHeight	4
StripePitch	4
BlockHeight	4
StripeOffset	0
StripeArrangement	Geometry_1X_2YM

Table 2: Stream Configuration for Bank A Mode

Multibank Streaming

The S641 has two QSFP+ ports that can be used in parallel to get the fastest throughput the sensor offers. In this set up, GenICam will discover the S641 as 2 unique **Devices**. To determine which GenICam **Device** is connected to either bank, read the

ConnectedBankID register and reference the table below:

Bank	ConnectedBankID
A	0
B	1

Table 1: ConnectedBankID Association

Banks_AB allows for a full frame to be split in half with one section being streamed over bank A and the other over bank B. Each bank streams 4 lines, in a round-robin order, starting from the middle of the frame. For a full resolution stream, 2560 x 1600, split across 2 banks the readout would be:

Line	Bank
798, 799, 800, 801, 794, 795, 804, 805...	A
796, 797, 802, 803, 792, 793, 806, 807...	B

Table 2: Line Ordering for 'Banks_AB' Mode

It is important to remember that the **Height** register details the number of lines to be streamed across one of the banks. For example, if a 2560 x 1600 image is required, the **Width** and **Height** registers should be set to 2560 and 800 respectively.

To start a stream on 2 banks:

1. Write **Banks_AB** to the **Banks** register.
2. Start a stream on bank B. No data will be received
3. Start a stream on bank A. Now both streams will begin sending data.

It is critical when stitching the frame that each sub-frame is coming from the same exposure. Each sub-frame will contain a frame ID (called **SourceTag**) in the header and this can be used to correlate the two sub-frames during stitching.

In general, use bank A for all register read and write commands. Bank B should only be used to stream image data.

Auto-stitching in GenICam

The framegrabber can be configured so the buffer readout is the complete stitched image from both banks. Euresys has implemented these features in the Coaxlink QSFP+ board and egrabber libraries. In egrabber version 19.0+, the auto-stitching setup is automatically configured by the driver.

For reference, the following is an explanation of the auto-stitching registers. Reference the different frame components below, in an example sub-frame from bank A:

Figure 1: Frame Components of Bank A Sub-frame

To set this up for the S641, configure the following registers, found in the GenICam **Stream** module:

StreamControl Register	Value in Banks_AB Mode
LineHeight	<Horizontal Resolution> * <Bytes Per Pixel>
LinePitch	<Horizontal Resolution> * <Bytes Per Pixel>
StripeHeight	4
StripePitch	8
BlockHeight	4
StripeOffset	0 (if bank A stream)/4 (if bank B stream)
StripeArrangement	Geometry_1X_2YM

Table 3: Stream Configuration for Bank AB Mode

For more detail on the Euresys/egrabber implementation, please reference the sample code '310-high-frame-rate' in the Euresys sample documentation package.

Frame Grabber Setup for Ultra-High Speed Streaming

Some frame grabber manufacturers have some additional features to support the ultra-high frame rates (5000+ fps) that the S641 can reach. Using Euresys Coaxlink cards and egrabber libraries, there are 2 main controls to focus on if the maximum rates cannot be sustained.

1. **Buffers** – count of frame buffers to be stored in memory. Buffer is reused once it's read, but the buffer pool can fill up if the camera stream speed is faster than the buffer read speed.
2. **BufferPartCount** (in GenICam **Stream** module) – Number of frames that can be stacked in a single buffer, default 1. Increasing this value will allow the DMA controller to gang up multiple frames in a single DMA transfer. Frames will need to be split apart by the application layer, but increasing this control will allow the PC to keep up with the camera throughput.

Grabber errors like *Rejected Frame Error* or *Input buffer pool is empty* indicates that the framegrabber or CPU cannot keep up with the camera and these controls will need to be used.

Other frame grabber manufacturers may have different methods to allow for ultra-high frame acquisition, please refer to their documentation for more support.

General Guidelines for Camera Control

Though not required, it is best to change the following parameters while the camera is not streaming:

- **ShutterMode**
- Resolution – **Height, Width**
- **PixelFormat**
- **Banks**

3

IMAGE CALIBRATION

Each S641 is factory calibrated and allows the user to download a gain and offset value, per pixel, to perform a first-order calibration of the image, if desired.

Calibration Data Format

The S641 calibration data contains 2560 x 1600 16-bit unsigned integers (1 per pixel), indexed in order from top left to the bottom right of the image. For the gains table, the gain multiplier is scaled by 16384, so it is necessary to bring the uint16 value back to a float prior to the calibration. The offset table is not scaled but is saved as a 12 bit value. Bit shift to match the image bit depth, then simply subtract these values from the raw image.

$$[\text{Image}_{cal}] = ([\text{Image}_{raw}] - [\text{Offset}]/2^{12\text{-bitdepth}}) \left(\frac{[\text{Gains}]}{16384} \right)$$

Please note that the offset image is simply a black reference image measured at the factory at 50 fps and a long exposure time. It is suggested to always take a new black reference image using your specific setup (lighting, rate, exposure settings, etc.) for a more accurate image calibration. For convenience, there is a mechanical shutter that can be open/closed using the **LensShutter = Open/Close** command.

To easily set the camera up to output a raw image, set **OutputRawImage = On**

Windowing

The Phantom S641 supports resolutions smaller than 2560 x 1600 and will window around the center of the frame. It is especially important to consider frame windowing when applying calibration to a raw, live image. The calibration arrays are of full frame size so be sure to window these arrays to the same resolution as the raw, live image.

Display Name	Name
Live Image	Imgsrc0
Offset Table @ Global	Imgsrc1
Gain Table @ Global	Imgsrc2

Table 4: Image Source Descriptions

How to Download

Each calibration file is saved to camera flash and can be accessed by changing the **ImageSource** register, located under **ImageFormatControl**, to one of the following options:

To download:

1. Select one of these options from the **ImageSource** register
   a. This will automatically change the resolution to maximum and the **PixelFormat** to **Mono16**
2. Start acquisition on bank A
3. Send command to **ImageSourceGrab** register
   a. This will start the readout from flash and the data transfer over CXP.
   b. Please note that this image is saved in camera flash, in which data reads are quite slow. Streaming one frame of a 16bpp image can take approximately 60 seconds.
   c. Due to this delay, the **ImageSourceGrab** command will timeout - ***clear this error***.
4. Get the next frame buffer and save to disk.

Since this calibration data is factory set and will not change, it is suggested to save this to file locally and access it from the PC instead of the camera every time it is needed.

In Camera Flat Field Correction (FFC)

The S641 can stream a flat field corrected image by setting the **FlatFieldCorrection** parameter to **FlatFieldCorrectionOn**. This will apply a column wise flat field correction in the camera.

If this image correction is not high enough quality for your application, a full FFC can be applied by using the gain and offset tables provided. These tables are based on a non-corrected image, so be sure to stream with **FlatFieldCorrectionOff** if you are intending to apply an FFC in the framegrabber or software.

4

GENERAL PURPOSE INPUT/OUTPUT (GPIO)

The S641 has 3 GPIO ports, 1 timecode input port and 11 available GPIO types, described below:

Types of GPIO

Type	I/O	Description
Trigger In	Input	On falling edge, exposure will start, selectable through the TriggerSource register.
Trigger Out	Output	Outputs a falling edge at the start of every exposure. Active only in TriggerMode=Off
Software Trigger Out	Output	Outputs a rising edge when a software trigger is received from the frame grabber.
Strobe	Output	Waveform that represents the frame rate and exposure time of each frame. Exposure start happens on the falling edge, exposure end happens on the rising edge. Falling edge to falling edge represents the frame rate.
Event	Input	If 0V, the event flag in the timestamp metadata will be TRUE.
User In	Input	If 3.3V, UserInputStatus register = TRUE. If 0V, register = FALSE.
User Out	Output	If 'UserOutputSet' register = Low, output = 0V. If register = High, output = 3.3V
Ready	Output	If 3.3V, stream is inactive. If 0V, stream is active.
Memgate	Input	If 3.3V, allow streaming. If 0V, pause streaming.
Timecode In	Input	IRIG-B (modulated or unmodulated) input
Timecode Out	Output	IRIG-B unmodulated output

Table 5: Available GPIO Types

How to Configure

To configure the GPIO lines, access the registers **GPIO0, GPIO1, GPIO2** under the **DigitalIOControl** section.

Trigger In can be enabled and configured using the **TriggerMode, TriggerSource** and **TriggerSelector** registers under the **AcquisitionControl** section.

GPIO Input Priority

The camera can only accept one event channel, one memgate channel, and one user input channel at a time. In the case that more than one GPIO line is configured as event, memgate, or user input, the camera will only monitor the top-most GPIO line as the input, ignoring the others. For example:

GPIO Configuration	Camera Monitors...
GPIO0 = eventin0	GPIO0 for event
GPIO1 = eventin1	
GPIO0 = strobe0	GPIO1 for event
GPIO1 = eventin1	
GPIO2 = eventin2	

Table 6: GPIO Input Priority Examples

Note: this priority does not apply to output channels as they can be set up for parallel output.

Electrical

The S641's GPIO are accessible the rear connector panel via BNC connectors. All GPIO signals are 3.3V LVTLL tolerant. Also, an internal pullup resistor allows for the shorting of the GPIO line to ground to toggle the line.

Type	Count	GenICam Label	Available Features
General Purpose IO	3	GPIO0, GPIO1, GPIO2	Trigger In, Trigger Out, SW Trigger Out, Strobe, Event, Ready, Timecode Out, Memgate, User In, User Out
Timecode In	1	N/A	Timecode In

Table 7: GPIO Electrical Configurations

Trigger Types

The S641 implements the **ExposureStart** triggering type only.

ExposureStart

On falling edge, the sensor will expose for the duration of time in the exposure register. Note: be sure that the exposure time in the register is less than the input clock period. Also, be sure that the input clock rate is supported by the camera at your specific set resolution.

5

SERIAL PORT PASSTHROUGH

The S641 6 pin power connector contains pins for RS232 Tx and Rx and are accessible for bi-directional communication through the fiber CXP connection.

To connect a serial device to the camera power port, a serial breakout cable will need to be used. Connect the cable to the camera power port and connect the DC power port to the breakout connector. Connect the external serial device to the breakout DB9 cable.

Once connected the application can write bytes to the serial port using the **UserSerialTxReg** register. Write ASCII characters to this register, one byte at a time. The application can read serial port bytes using the **UserSerialRxReg**. Read ASCII bytes, one at a time, from this register. Reading a null character, 0x00, indicates that the Rx buffer is empty.

6

REMOTE LENS AND SHUTTER CONTROL

The Phantom EOS lens mount works with the S641. This allows the user to remotely change the aperture size and lens focus. If the EOS mount is installed the **LensAperture**, **LensFocusStep**, and **LensFocus** registers will all be enabled.

To change the aperture, select the size from the list in the **LensAperture** register.

To change the focus, first set the encoder count step size in the **LensFocusStep** register, which ranges from -200 to +200. Then, send the **LensFocus** command to move the focus ring this amount of steps.

There is also a mechanical shutter to block light from the sensor for a convenient, remote black reference frame using the **LensShutter** register. The mechanical shutter comes standard with all S641s.

7

METADATA STREAMING

There is various metadata that can also be streamed via CXP Events. All CXP events contain an event ID, event timestamp and event payload. Each S641 event payload contains the frame ID, or **SourceTag**, to correlate the metadata to a specific frame.

The S641 will send events for two different reasons: the **EventRefresh** time has elapsed or there was a change in the event state. Set **EventRefresh** to 0 to disable these refresh events. On Change events cannot be disabled.

Event	ID	Payload	Send	Description
Timestamp	0x01	word0: sourcetag (low 16bit) word1: extension (bit31-16), fraction(bit15-2), event(bit1), lock(bit0) word2: csec (32bit)	Refresh	IRIG timestamp, current lock state (1 = freerun, 0 = locked to IRIG), and current GPIO event state
Exposure Time	0x02	word0: sourcetag (low 16bit) word1: exp (bit31-16), extension(bit15-0)	Refresh, On Change	Exact exposure time of the frame. Will send on change of exposure time.
GPIO Event	0x03	word0: sourcetag (high 16bit), event state (low 16bit)	On Change	Sends sourcetag on the GPIO event state transition
Temperature	0x04	word0: sourcetag (low 16bit) word1: camera temp (bit31-16), sensor temp(bit15-0)	Refresh	Current camera and sensor temperatures (°C)
Frame Rate	0x06	word0: sourcetag(low 16bit) word1: framerate (float)	On Change	Sends frame rate and sourcetag at the exact time of rate change

Table 8: S641 CXP Event Descriptions

Event	Example Parsing Code (Python)

Timestamp	frame_count = int.from_bytes(data[2:][:2], byteorder='big')
	ext = int.from_bytes(data[4:][:2], byteorder='big')
	frac = int.from_bytes(data[6:][:2], byteorder='big')
	csec = int.from_bytes(data[8:][:4], byteorder='big')
	event = (frac & 0x2) >> 1 # get event bit
	lock = frac & 0x1 # get irig lock
	frac = (frac & 0xFFFC) >> 2 # get frac time # convert to seconds
	time = csec * 0.01 + frac * 0.000001 + ext/65536 * 0.000001

Exposure Time	frame_count = int.from_bytes(data[2:][:2], byteorder='big')
	us = int.from_bytes(data[6:][:2], byteorder='big')
	ext = int.from_bytes(data[4:][:2], byteorder='big')
	# convert to us
	exp = us + ext/65536

GPIO Event	frame_count = int.from_bytes(data[0:][:2], byteorder='big') event = bool.from_bytes(data[2:][:2], byteorder='big')
Temperature	frame_count = int.from_bytes(data[2:][:2], byteorder='big') camtemp = int.from_bytes(data[6:][:2], byteorder='big') snstemp = int.from_bytes(data[4:][:2], byteorder='big')
Frame Rate	frame_count = int.from_bytes(data[2:][:2], byteorder='big') _frameratebytes = bytearray(data[4:][:4]) _frameratebytes = frameratebytes[::-1] frame_rate = struct.unpack('f', _frameratebytes)[0]

Table 9: S641 CXP Event Sample Code

How to Access

Reference CXP standard documentation and framegrabber help documentation for more info on CXP Events and implementation details, sample code.

In general, the application will need to enable device events in the frame grabber and register callback functions for a new device event and/or event ID. In the callback functions, reference the above tables for data payload content and parsing.

IRIG Timecode Synchronization

The S641 can be synchronized to an external IRIG-B source, however IRIG-B timestamping is not implemented in the GenICam standard. All timestamping done by the framegrabber is in the PC clock time domain. It is the application layers responsibility to transform the incoming frame timestamps from the PC time domain to the camera IRIG time domain.

To do this, in the event callback for the timestamp event, determine the IRIG to PC time offset:

t_event_sec = data.timestamp * .000001

time = csec * 0.01 + frac * 0.000001 + ext/65536 * 0.000001

irig_offset = t_event_sec - time

Then, as new frame buffers are received, perform the transformation of the frame time in the PC domain to frame time in IRIG domain:

t_frame_irig_domain = t_frame_pc_domain - irig_offset

where t_frame_pc_domain is the time associated with each new frame buffer header.

Clock Drift Considerations

The PC clock will drift from the IRIG clock so the calculated irig_offset is something that should be updated frequently, depending on your timing requirements. Two ways to decrease the effect of clock drift:

1. **Increase the 'EventRefresh' time** – if you get new timestamp events more often, the effects of clock drift will be less. The minimum refresh rate is 1ms.
2. **Calculate expected drift and apply to incoming times** – clock drift is a linear function and can be calculated by the difference of the two domains between two frames. Assuming temperature of both camera and PC are stable, you can apply this first order calibration for incoming frame times for increased accuracy.

Other Timecode Features

The **TimeStampSet** register is the Unix time offset value. By default it is set to 0 at startup. Enter in the Unix time offset if necessary. This is not available if the camera is connected to an IRIG timecode source.

8

PC SETUP

To take full advantage of the S641 throughput, ensure that the PC you are running matches or exceeds the framegrabber requirements that you choose. Common things to check are:

- PCIe generation (typically gen 3)
- PCIe lanes (use maximum or greater that required by the framegrabber)
- CPU clock speeds
- RAM size
- Storage write speeds (HDD vs SSD)
- GPU

Most commonly, the camera throughput will be throttled when the framegrabber is installed into an older generation of PCIe slot, or a PCIe slot with less lanes than required. Most framegrabbers will be able to function properly in a lower throughput or older generation PCIe slot, but the camera throughput will be affected. Check the PCIe slot map of the PC's motherboard to verify the framegrabber is installed in an appropriately sized slot.

9

UPGRADING CAMERA FIRMWARE

To update S641 firmware in the field, use the Phantom Camera Firmware Loader and a new firmware file, *.phfw.

To upgrade firmware, power on the camera and connect the 'Program' ethernet port of the S641 to the PC. The PC NIC will need to be configured to be:

IP Address	100.100.X.X
Subnet Mask	255.255.0.0

Table 10: IP/Mask Configuration

Open up the Phantom Camera Firmware Loader and select the discovered camera and click 'Next'

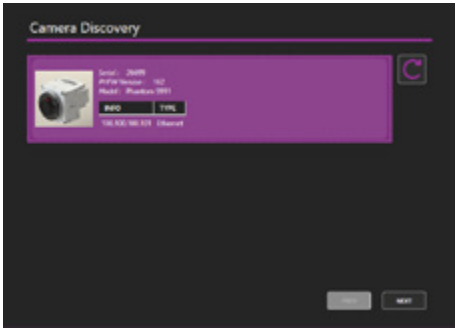


Figure 2: Camera Loader Wizard Discovery

Select the PHFW file using the file dialog and click 'Load PHFW'.

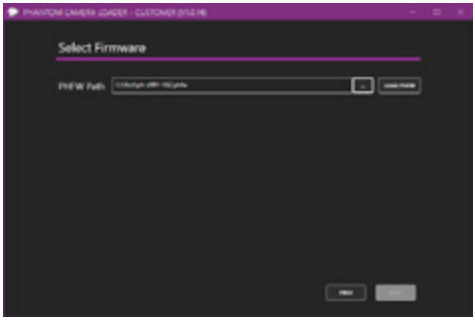
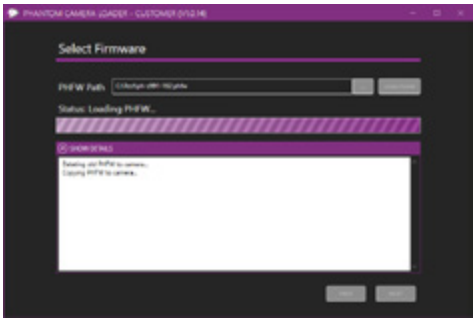


Figure 3: Camera Loader Wizard PHFW Select

During the firmware load, progress and status will be displayed.

Figure 4: Phantom Camera Firmware Loader During PHFW Load



At completion of the load, you will be prompted to reboot the camera. For reference and debug purposes, two log files will also be created: output.txt and supportOutput.txt. If any issues arise during the load process, please contact Vision Research Support and send these log files.

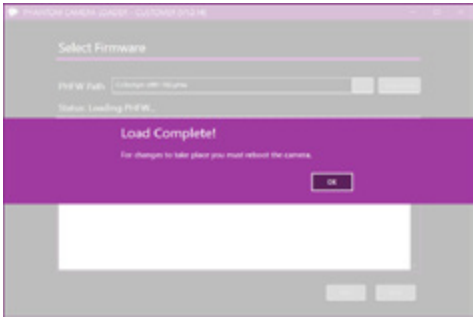


Figure 5: Phantom Camera Firmware Loader PHFW Load Complete

CONTACT SUPPORT

GLOBAL HEADQUARTERS

Vision Research, Inc.—Wayne, New Jersey

100 Dey Road
Wayne, New Jersey 07470 USA
T: +1.973.696.4500

For answers to most questions, please visit us at: ***www.phantomhighspeed.com*** and search the camera product pages, tutorials, support knowledgebase and FAQs.

SUBMITTING A SUPPORT TICKET

For technical product support, operation and application information or to request an RMA, please submit a ticket by filling out a form at ***www.phantomhighspeed-service.force.com*** or by emailing us at **phantom-support@ametek.com.**

LIVE CUSTOMER AND TECHNICAL SUPPORT

Serving the Americas:
M-F 8:00 AM to 5:00 PM EST (GMT -4:00)
T: +1.973.696.4500
Customer Support, ext. 4002
Technical Support, ext. 4003

Serving Europe, the Middle East and Africa:
M-F 9:00 AM to 6:00 PM GMT +3:00

Vision Research, Inc.—Bucharest, Romania
T: +40 21 210 8587

Serving Asia Pacific:
M-F 8:00 AM to 5:00 PM GMT +8:00

Vision Research, Inc.—Shanghai, China
T: 86-21-58685111, ext. 141

CONTACT SUPPORT