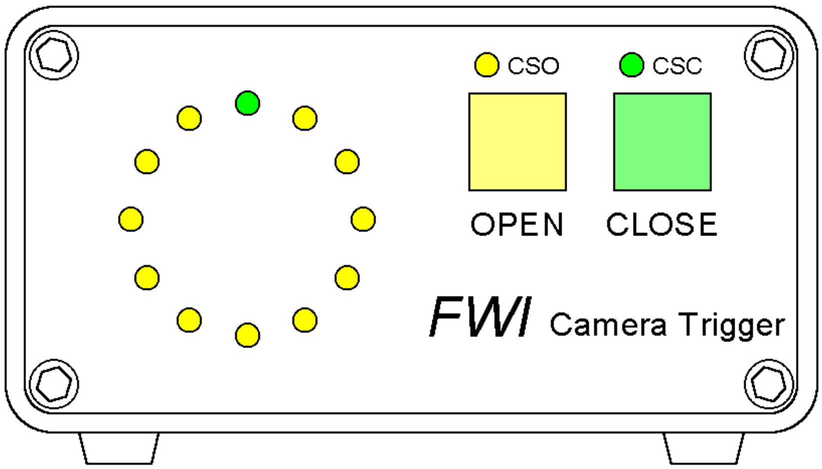
Users Manual – FWI Still Camera Interface

- Allows Kuper software to easily control digital cameras
  - Output signals track the Kuper camera output channel, just like a film camera, allowing all the "special" Kuper exposure modes, including intervalometer and synthetic shutter
  - No unusual Kuper setup is necessary, once the interface box is programmed, it operates automatically, tracking the "camera" output.
- Generates three separate, finely adjustable, output signals
  - For industrial cameras, the signals can control senor reset, shutter, and frame grab.
  - For consumer cameras, the signals can control "two-step" shutter releases
  - All signal timing is completely adjustable with 1º of resolution
- On-board high speed serial port
  - Transmits a continuous stream of status information for "smart" applications.
  - Supports easy programming and setup via a standard PC terminal program like Window's Hyperterminal
- Outputs signals are totally isolated from the Kuper inputs
- Intuitive, easy to understand controls and displays
- Equipped with a universal 100-240V power supply

Front Panel



Close switch & CSC LED

- "CLOSE" switch – Lights green to indicate the shutter is currently closed.
- A short push manually closes the shutter, but does not affect the current phase count.
- A long push manually resets the current phase count to 0^{o}, nominally center shutter closed.
- "CSC" LED - lights green to indicate a phase of 0^{o}, usually the exact Center of Shutter Closed. This is where Kuper cameras are usually "zeroed".

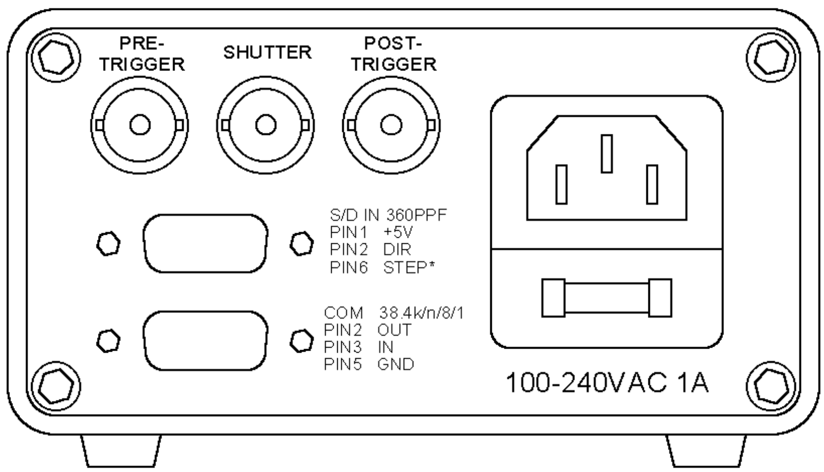
Open switch & CSO LED

- "OPEN" switch – Lights yellow to indicate the shutter currently open.
- A short push manually opens the shutter, but does not affect the current phase count.
- A long push manually resets the current phase count to 180^{o}, nominally center shutter open.
- "CSC" LED - lights green to indicate a phase of 180^{o}, usually the exact Center of Shutter Open. This is where Kuper cameras usually pause when opening.

Phase Display

- The phase display operates like a clock face, to provide a visual reference of current camera angle and movement. The display advances clockwise
- The 12 O'Clock position is 0^{o} of camera phase, nominally center shutter closed.

Rear panel



Shutter, Pre and Post Trigger outputs,

These three signals are the main outputs of the box. Nominally, they are used for a pre-exposure "setup" signal, the actual shutter open/close signal, and a post-exposure "store" signal, but the three signals are electrically identical and any signal can be configured to start and end at any phase angle.

These output signals can be any combination of 1) normally open relay contacts, 2) normally closed relay contacts, 3) logic "active high" signals, or 4) logic "active low" signals. Signal type and polarity are set via internal jumpers, see section 3.

Step/Direction in

Step & Direction in are standard current loop opto-isolator inputs. Signals are electrically isolated from the outputs and AC ground. The shell is connected to the case ground. This connector is electrically identical to a Point Blank format step-motor driver input.

Pin 1 = +5V
Pin 2 = Direction in
Pin 6 = Step in (active low, box counts on trailing edge)

The box is calibrated for 360 steps per frame, allowing easy control in increments of one degree.

The number of steps per camera frame (essentially, the camera PPU) is adjusted by using the Kuper “PullDown” command (pd), or the “camera” button on the main page.

Select “single pull down”, and 360 pulses per frame. Depending on how your system direction lines are wired, you may have to select –360 pulses per frame to get the camera to run in the correct direction.

COM in/out

Standard PC-type RS232 serial port, configured as 38.4 Kbaud / no parity / 8 data bits / 1 stop bit. This box supports, but does not require, hardware handshaking.

The COM port is directly compatible with a standard 9-pin PC serial port, using a straight-through type, 9 pin male-to-female serial cable.

These 3 pins are required

Pin 2 = TX data out
Pin 3 = RX data in
Pin 5 = signal ground

These 2 pins are optional, they provide handshaking signals for devices that require them.

Pin 8 = RTS out
Pin 6 = DTR out

AC in

“Universal input” 100-240 VAC/DC power in, uses a standard IEC type power cable. Case is connected to the ground (earthing) pin. The power inlet also contains a fuse holder with an integral spare. The proper fuse is a standard 1A 5x20mm SB type.

Setting Output Signal Parameters

This box provides three separate output signals for camera control at the rear-panel BNC connectors.

Typically, they will be used for a pre-trigger function, like flushing a CCD or turning on an SLR's metering, shutter control for triggering and timing the shutter, and a post trigger function, for triggering an image capture board or triggering a download.

However, the three signals are identical and interchangeable, and any signal can be used for any purpose.

All three signals can be assigned as a relay closure, a relay opening, or a TTL logic high or low. The signal parameter is determined via internal jumpers, and described in the section titled "Selecting the type of output signal"

Any of the signals can start and end at any point in the cycle. Signals can overlap at will. Signal start and end times are set via the serial port, and may be specified to 1^{o} degree of shutter angle.

Setting Timing Parameters via the Serial Port

Signal timing is set via the serial port, in a manner similar to setting up parameters in a servo drive. Once parameters are set, they can be stored to EEPROM as new defaults, and they will be automatically re-loaded at the next power-up.

You may use any RS-232 type terminal to communicate with the camera trigger. Most users will use a Windows based PC for this task and will have HyperTerminal available by default. If you want to use HyperTerminal, and are unfamiliar with its setup, please consult Appendix C.

Whichever terminal program you use, the communication parameters are…

- 38.4 Kbaud
- No parity
- 8 data bits
- 1 stop bit
- No handshaking (This box supports hardware handshaking, but does not require it)

Appendix "A" contains a full description of all the serial commands and responses available to the box.

Realistically, most users will be better off simply attaching a serial cable to the box, typing "help", and following the directions on the screen.

Most parameters are set with a simple command dialog. For example, "List" returns a list of all the current parameter settings, "shutter open = 090" sets the shutter opening angle to 90 degrees, "write eprom" writes the parameters to eeprom so that they will be loaded automatically on the next power-up.

Note that all commands sent to the box have to terminate with the standard ASCII characters <carriage return><line feed>. These are the tokens used by the camera trigger to determine that an incoming command has arrived.

Most ANSI terminal programs will happily append these standard characters to each transmitted line, but many programs (like HyperTerminal) require you to check off a box specifying this action. If necessary, it's usually in a sub-menu titled something like "ASCII mode".

Selecting the type of output signal

The camera trigger will output 4 kinds of trigger signals on the rear-panel BNC jacks.

They are
1) Relay contacts, normally open;
2) Relay contacts, normally closed;
3) TTL level logic signals, active high; or
4) TTL level logic signals, active low

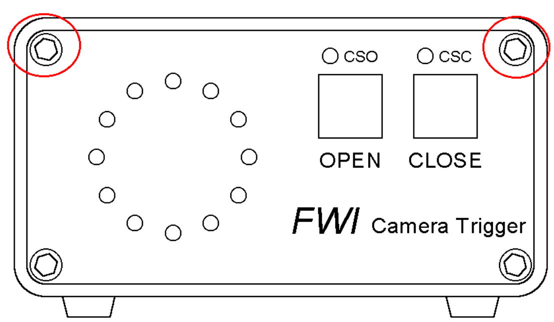
Any output can use any type of signal, in any combination

The box ships with all outputs set for normally open relay contacts, but for your application, you may want to change this to another signal type or combination of types, depending on your particular camera and capture system.

The desired type of signal is individually selected for each output by using programming jumpers on the main board.

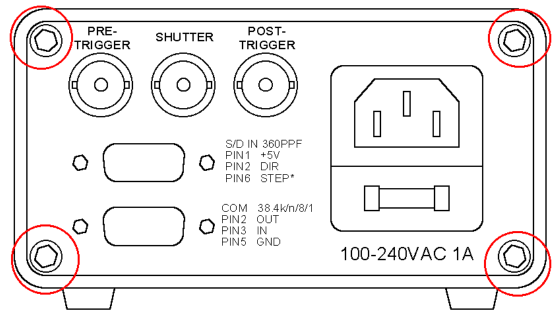
To do this, you will have to open top lid of the case.

Whenever you open the case, always first disconnecting the power.



Then remove the top 2 case screws on the front panel, and all 4 case screws on the rear panel.

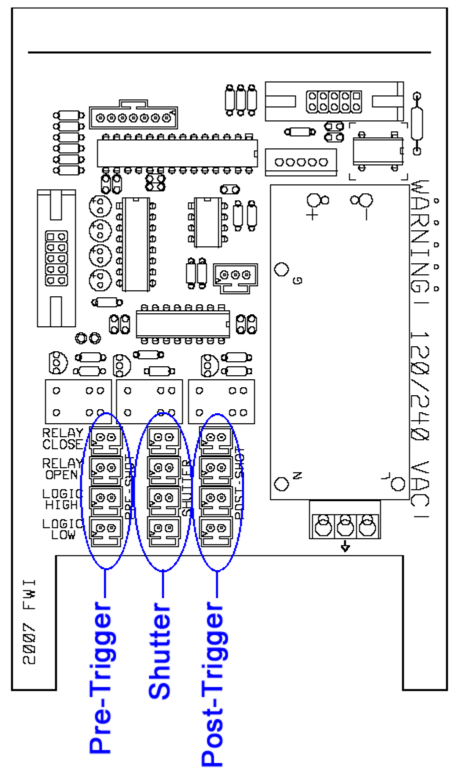
This will loosen the rear and top panels. Slide the top panel backwards out of the unit, exposing the main circuit board.



Near the rear of the main board, there is a small field of 2 pin sockets, arranged in three sets of four. Each set of 4 connectors carries one of the possible 4 outputs for each signal.

The leftmost set are the outputs for the pre-trigger signal, the center set are the outputs for the shutter signal, and the right set are the outputs for the post-trigger signal.

You should be able to clearly identify the short jumper which runs from the rear-panel BNC jack to the board connector. It can be easily removed from the board connector by depressing the small latch on the side of the mating part.

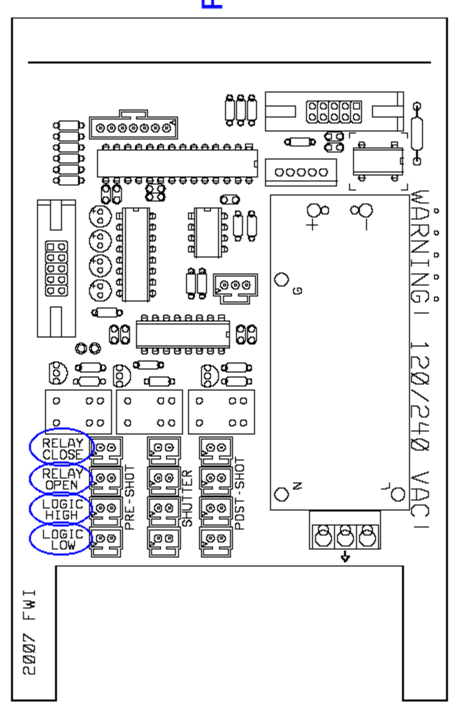


The four possible types of output signal are arranged in a vertical row.

From front to back, they are…

- Relay close (a normally open relay which closes when active)
- Relay open (a normally closed relay which opens when active)
- logic high (a TTY level, active high signal)
- logic high (a TTY level, active high signal)

Select the type of signal you want, and plug that output connector into the appropriate jack on the board



Maximum shooting speed

The counting logic, and the digital outputs, will track correctly at over 100fps.

The practical upper limit on how fast you can shoot will be determined by how long it takes for the relays to open and close (if you're using them), and how fast your camera can store captured frames.

The relay outputs are mechanical nature and have a definite lag time. Even though this box uses low-inertia relays, they still take about 4 milliseconds to change state. That's quick, but still a significant part of the 20 millisecond frame time at 24fps.

If you plan to use this box above about 8 fps, you should use the digital output signals rather than the relay outputs.

More significantly for most applications is the speed with which you can take, and especially, store, frames with a digital camera.

Most consumer SLR type cameras, and many industrial CCD cameras, take a relatively long time to store a captured image to ROM or disk after the actual exposure is done. Some cameras have a buffer, which can store a small number of frames, but once this fills up, shooting more frames gets delayed till the existing frames are stored.

You will have to experimentally determine how long it takes your particular camera to shoot and store a frame, and be careful to stay below this timing threshold. Otherwise, your camera may introduce unknown, unreproducible delays in shutter response as it struggles to store frames.

You should perform all such testing using the real subject or subject of similar complexity, as there may be variations in storage time caused by the variable runtime of image compression algorithms inside the camera.

One last area of timing concern should be noted by anyone who intends to use the automatic outputs to control an intelligent downstream device (such as a PC based frame grabber)

All the serial status signals that come out of the box during normal step/direction operation are all 4 characters long (op<cr><lf>, cl<cr><lf>, etc). At 38400 Kbaud it takes just over 1 millisecond to transmit this information.

If you're shooting fast enough, and your transitions are close enough, you can have transitions spaced so closely that there isn't enough time for the serial port to finish reporting one transition before the next has to start. In this case, the port will "flush" the current message it's trying to finish, and cue up the new one.

The limiting speeds are pretty high, and the interval spacings are pretty small, but you should still be aware of them.

If (and only if) the serial port data is important to you, then you should use the following limits on your edge spacing to allow one status message to completely finish before the next one begins. If you're not using the serial port data, you can safely ignore these limits.

At 24 fps, transition edges should be no closer to each other than 9 degrees.

At 12 fps, transition edges should be no closer to each other than 5 degrees.

At 8 fps, transition edges should be no closer to each other than 3 degrees

At 4 fps, transition edges should be no closer to each other than 2 degrees

Below 2.5 fps there is no limit on edge spacing.

Appendix A; Connecting to Cameras

This box can control virtually any type of camera you can attach, since most have the need of only a few signals that can be readily converted into shutter angle equivalents.

Most users, however, will use this box with professional or semi-professional commercial still cameras, and attaching to these cameras is very straightforward.

Almost all cameras will have a provision for a remote release switch, usually a two-step switch that turns of the metering and fires the shutter.

Whenever possible, a 2-step release should be provided for the camera. This is because most digital cameras do not pay very close attention to their shutter release buttons while "idling", even if producing a live-view picture. Rather, they drop to a lower activity, lower power mode, and only check the shutter switch about 15 times a second.

If you directly apply a shutter closure, the camera has to first notice the switch is closed, then go to an alert mode, then start the exposure sequence.

Though the actual delay is small from a human perspective, the wake-up sequence can add leading edge uncertainty, which is an issue during go-motion passes.

Conversely, most cameras have a "metering" mode that usually precedes the actual shooting mode, usually by means of a two-step switch. Once most cameras are in metering mode, they internally switch to a much more alert state and track the activity of the shutter button much more closely, responding to the leading edge much more consistently.

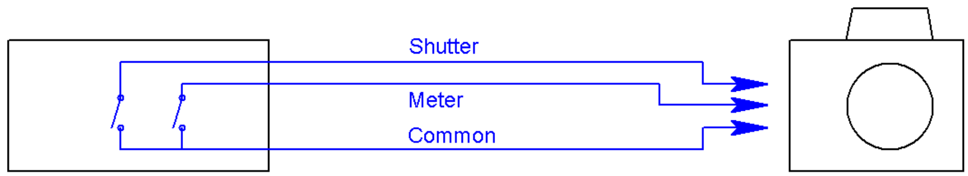
Most of the FWI boxes are preset at the factory to provide a conventional 180º shutter signal from 90º to 270º, and a pretrigger signal to turn the camera meter on from 5º to 275º, so the camera has "woken up" before the shutter signal arrives.

Depending on the camera, actual exposure timing can be controlled through the shutter signal with the camera on "bulb", or timed by the camera itself. Some cameras are better at tracking "bulb" than others, though most are good above ½ second. Canon EOS cameras seem to be very good at it, and we've even successfully done exposure ramps with these cameras, controlled from Kuper exposure tables.

When shooting go motion (or conventional motion passes) you will probably find that the minimum shooting speed is not be set by the exposure time, but rather by the time you have to give the camera *between* frames to store the image to its memory card. Most cameras can do a fairly quick burst for a few frames, but once their internal memory buffers fill up they slow down to a maximum rate of a few seconds per frame as they now have to write to the memory card in real time.

Regardless of camera chosen, the first step is usually to procure a two-step remote release for your camera, which will be sacrificed to make an interface connector for the control box.

In the simplest of cases, since these cables usually offer a switch closure, it is a simple matter to set the interface box up for relay output, then run the camera control signals directly back to the box and use the internal relays to mimic the action of the original switch.



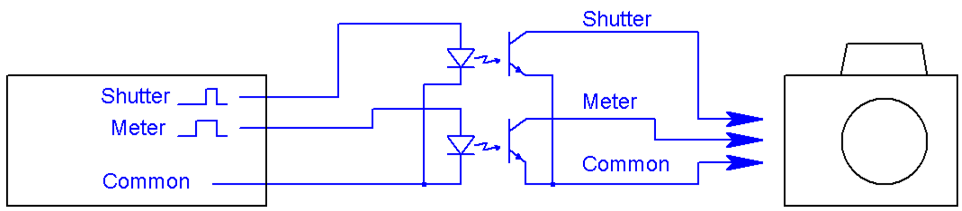
In this case, the “pretrigger” output of the box is set to go on first (at, say, 5º) and turn the meter mode on so the camera has “woken up” by the time the shutter signal arrives.

Under most circumstances, providing the distance between your controller and your camera is not very great, and the electrical environment of your stage nor expecially polluted, this setup should work fine.

However, if you have the camera mounted on a motion control rig, it will become obvious that the camera’s control signals will run down the rig for some length alongside high voltage motor cables, making them vulnerable to induced electrical noise.

Some cameras have no problem with this, while others are markedly more delicate about the situation. The Nikon D200, and cameras based on that chassis, like the Fujinon S5, for example, are know to be particularly sensitive to noise corrupting their control signals.

In cases like this, it may be necessary to isolate the camera from rig noise via an optoisolator.

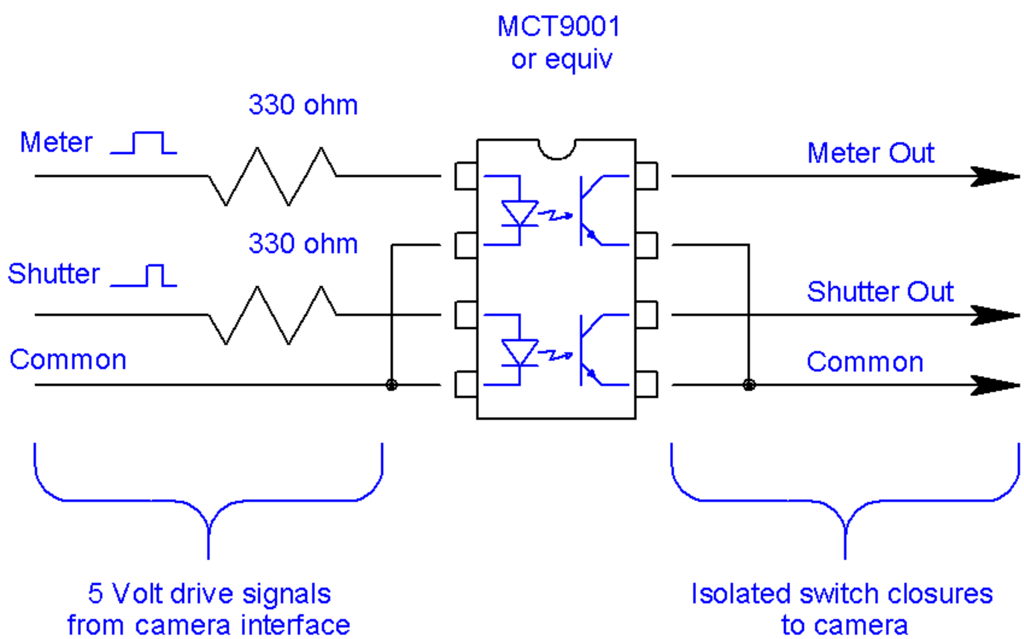


Here, the key thought is that the optoisolator should be placed near the camera, on a short pigtail. That way very little cable length is exposed to pick up rig noise.

Instead of being configured with switch closures as outputs, the Interface box is now configured to provide +5volt logic outputs. These voltage signals drive currents through the input sides of the optoisolator.

Since optoisolators use light-emitting diodes as inputs devices, and light emitting diodes are current mode, not voltage mode, devices, they are immensely immune from picking up radiated noise, and the drive signals can survive trips through incredibly noisy rigs and still be very reliable.

FWI can provide optoisolators for most cameras, but you can easily build your own, using this no-frills circuit.



The optiosolator in this circuit is a common Fairchild MCT9001 (available from Mouser or Digikey with that part number). Any 2 channel optoisolator will do, so long as it is a "passive" open collector type, and not an active "logic output type" (open collector types mimic switch closures as outputs, while logic types provide an active logic signal as outputs)

Most 2-channel optoisolators in an 8 pin DIP package share this same pinout. The reisitors are typical 330 ohm, 1/8 watt resistors, any value from 300 to 510 ohms should do nicely.

Regardless of which type of signal you use, a passive switch closure or an active optoisolator, you should test to make sure that the box is configured t provide the type of output that you're expecting.

You can use a common voltmeter to test the outputs, by metering the outputs while you toggle them on and off by using the "open" and "close" buttons on the front panel.

You do not have to open the box to test the outputs.

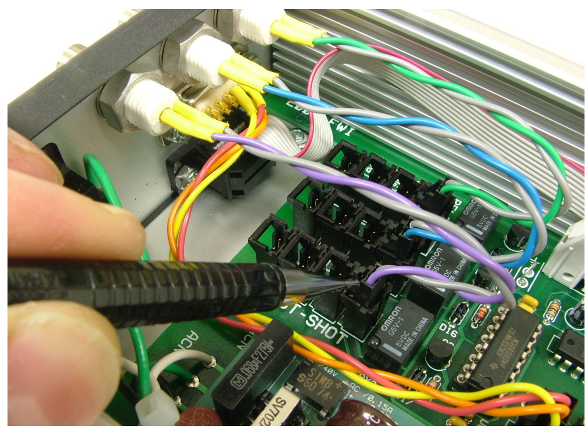
If you're using relay closure outputs, use the "ohms" setting on your meter (sometimes marked with the electrical symbol "Ω"). When the outputs are not active, they should read as a high resistance, or, depending on your meter, "open" or "over".

When the outputs are active, you should read a low resistance.

If you're using the 5 volt output to drive an optoisolator, use the "DC volts" setting on your meter. When the outputs are not active, they should read a voltage near zero.

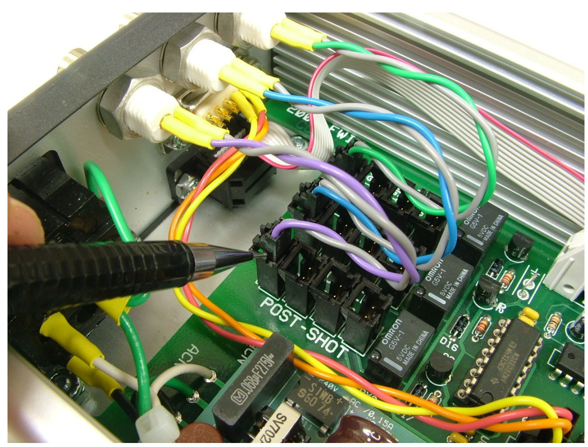
When the outputs are active, the outputs will read between 4 and 5 volts.

If you need to change the type of output you must open the box and move the output jumpers to the appropriate setting. See the instructions on page 7 for details, but the quick recap is...



The "normally open" relay outputs are the connector row that is closest to the front of the unit.

Nearly all users who are connecting to a camera with a modified cable release will be using the normally open relay output



The logic level, active hi, 5 volt outputs are the connector row that is closest to the rear of the unit.

Nearly all users who are connecting to a camera with an optoisolator will be using the logic hi 5 volt output

Specific cameras:

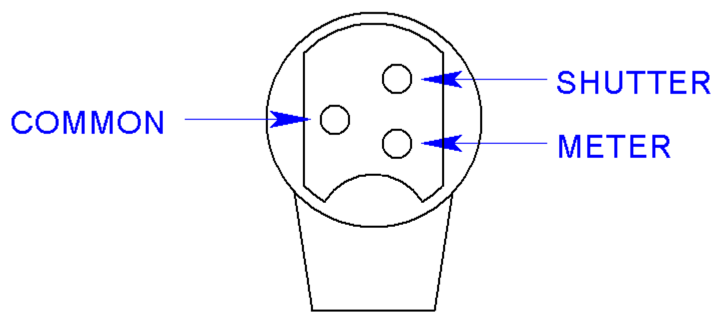
The following section discusses specific camera interfaces, bases on our current experience. While we believe these diagrams to be accurate, *you should only use them as a starting point*, since manufacturers can and do change cable colors, pinouts on camera bodies, and such similar details.

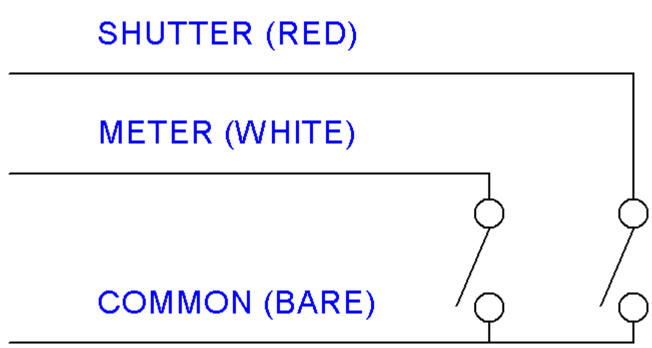
TEST! TEST! TEST! Before applying power!

With that disclaimer, here are some common camera connections we have used…

CANON EOS Family w/ N3 port

Use: Canon Remote switch P/N RS-80N3

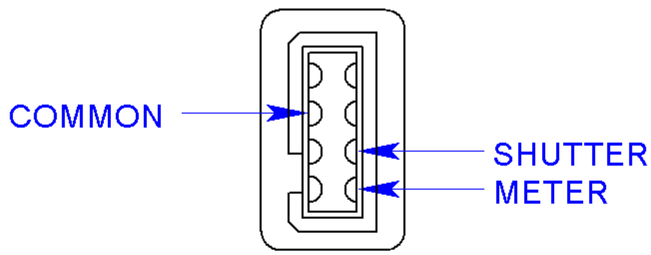


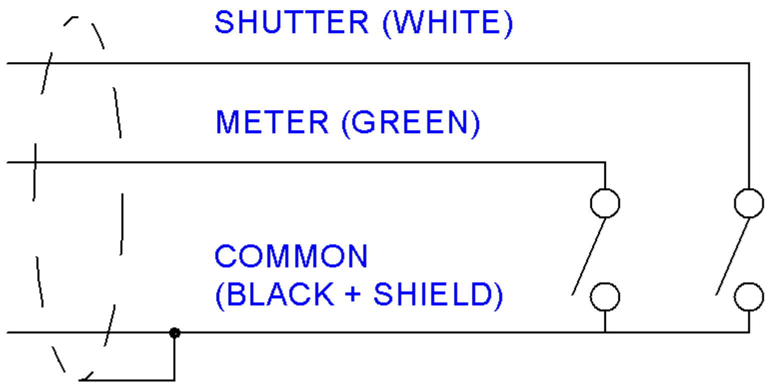


Notes: Cannibalize genuine RS-80N3 Canon remote switch. The real Canon part uses an delicate but reliable expanding ring-sleeve for very positive locking, whereas aftermarket switches seem to depend on a friction fit and tend to work loose.

NIKON D90

Use: Nikon Remote switch P/N MC-DC2

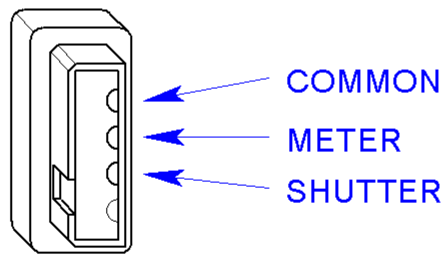


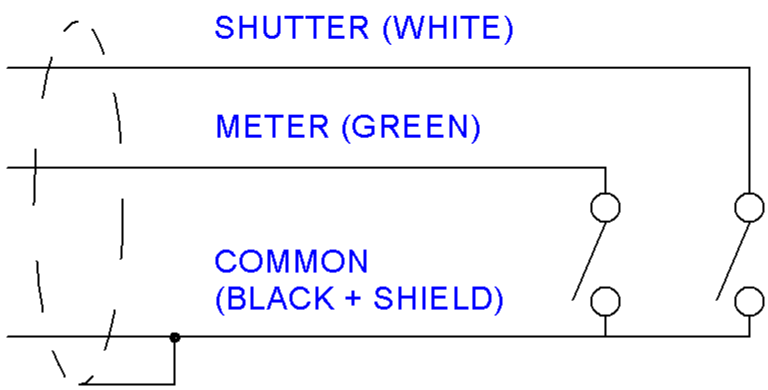


Notes: Cannibalize genuine Nikon MC-DC2 remote switch. This connector has very tiny, precise, parts, and the aftermarket switches we've seen are nowhere as precisely made as the original Nikon switches, and tend to have a loose fit with unreliable retention.

NIKON D80

Use: Nikon Remote switch P/N MC-DC1





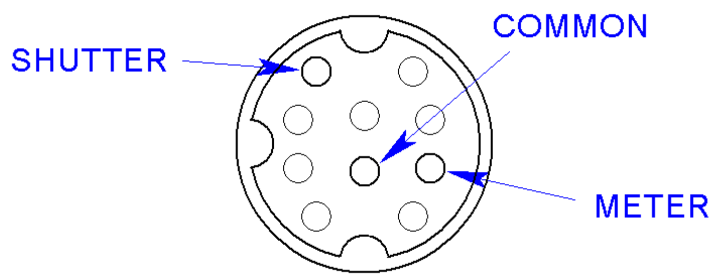
Notes: Cannibalize genuine Nikon MC-DC1 remote switch. This connector has very tiny, precise, parts, and the aftermarket switches we've seen are nowhere as precisely made as the original Nikon switches, and tend to have a loose fit with unreliable retention.

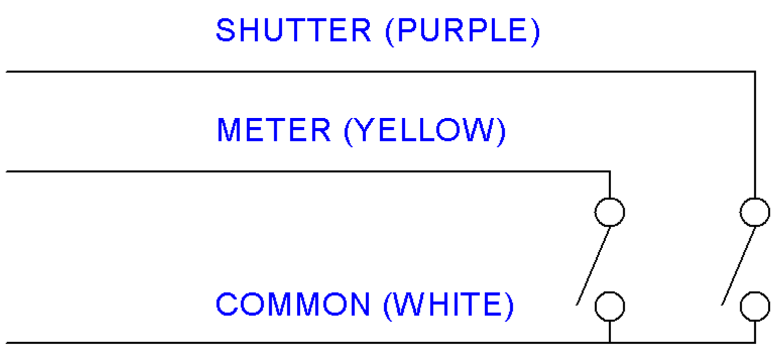
NIKON D200 / FujiFilm S5

(and other Nikon cameras using the 10 pin connector)

Use: Nikon Remote switch P/N MC-30 or equivalent

(connector is viewed looking into cable end)





Note: because this is a threaded locking connector, aftermarket switches seem to be equally reliable in this application, but note – if you're using an aftermarket switch, wiring colors may vary, so double-check your wiring.

Appendix B; Serial Communications

This box uses the following communication parameters …

- 38.4 Kbaud
- No parity
- 8 data bits
- 1 stop bit
- No handshaking (This box supports hardware handshaking, but does not require it)

Pinouts conform to the PC style 9-pin D-shell connector, and are intended to directly connect to a PC port with a 1-to-1 cable. The pinouts are…

Pin 2 = TX data out
Pin 3 = RX data in
Pin 5 = signal ground
Pin 8 = RTS out (optional if required by terminal)
Pin 6 = DTR out (optional if required by terminal)

The box does not require hardware handshaking lines (CTS, DSR) and ignores them if provided.

Types of output data

There are two kinds of signals that come out of the serial port, automatic outputs, essentially status messages generated internally, and dialog outputs, when the box communicates with an outside terminal for programming.

Automatic outputs

Automatic signals are generated by the trigger box on important transitions, like shutter open, close, etc, and broadcast as status information. They are intended to be read by external applications that need information about camera operation.

All automatic outputs except shutter closed reset are 4 characters long, including a terminating <cr> and <lf>. Automatic outputs do not start with the ">" character.

These status outputs are issued during normal operation from following step commands

op(<cr><lf>) Shutter open automatically (from step pulses)

cl(<cr><lf>) Shutter closed automatically (from step pulses)

p1(<cr><lf>) Pre-trigger signal started

p2(<cr><lf>) Post-trigger signal started

These status outputs are issued during manual operation of the front panel buttons

mo(<cr><lf>) Shutter opened manually (from the front-panel button)

mc(<cr><lf>) Shutter closed automatically (from front-panel button)

phase = 000(<cr><lf>) Phase has been reset to center shutter closed using front panel button

Dialog outputs

Dialog outputs are information transmitted in response to user commands. Dialog outputs always start with a ">" character so they can easily be separated from automatic outputs. Dialog outputs often refer to a phase angle, ###, where "###" = 000 to 359 degrees

Dialog outputs include

>phase = ###(<cr><lf>)	Current camera phase angle
>shutter open = ###(<cr><lf>)	Opening edge angle of the Shutter pulse
>shutter close = ###(<cr><lf>)	Closing edge angle of the Shutter pulse
>pre start = ###(<cr><lf>)	Leading edge angle of the Pre-trigger (P1) pulse
>pre end = ###(<cr><lf>)	Trailing edge angle of the Pre-trigger (P1) pulse
>post start = ###(<cr><lf>)	Leading edge angle of the Post-trigger (P2) pulse
>post end = ###(<cr><lf>)	Trailing edge angle of the Post-trigger (P2) pulse
>bad command(<cr><lf>)	Last received line was not understood.

Input Commands

h(<cr><lf>) or **help**(<cr><lf>)	Returns a list of valid commands
phase(<cr><lf>)	Query for the current camera step angle Returns **>phase = ###**(<cr><lf>)
list(<cr><lf>)	Returns a list of current angle settings for all signals
shut(<cr><lf>)	Query for information about the shutter signal Returns **>shutter open = ###**(<cr><lf>) **>shutter close = ###**(<cr><lf>)

shutter open = ###(<cr><lf>)
Sets the opening angle for the shutter pulse
Returns **>shutter open = ###**(<cr><lf>)
>shutter close = ###(<cr><lf>)

shutter close = ###(<cr><lf>)
Sets the closing angle for the shutter pulse
Returns **>shutter open = ###**(<cr><lf>)
>shutter close = ###(<cr><lf>)

pre(<cr><lf>)
Query for info about the pre-trigger (P1) pulse
Returns **>pre start = ###**(<cr><lf>)
>pre end = ###(<cr><lf>)

pre start = ###(<cr><lf>)
Sets the starting angle for the pre-trigger (P1) pulse
Returns **>pre start = ###**(<cr><lf>)
>pre end = ###(<cr><lf>)

pre end = ###(<cr><lf>)
Sets the ending angle for the pre-trigger (P1) pulse
Returns **>pre start = ###**(<cr><lf>)
>pre end = ###(<cr><lf>)

post(<cr><lf>)
Query for information about the post-trigger (P2) pulse
Returns **>post start = ###**(<cr><lf>)
>post end = ###(<cr><lf>)

post start = ###(<cr><lf>)
Sets the starting angle for the post-trigger (P2) pulse
Returns **>post start = ###**(<cr><lf>)
>post end = ###(<cr><lf>)

post end = ###(<cr><lf>)
Sets the ending angle for the post-trigger (P2) pulse
Returns **>post start = ###**(<cr><lf>)
>post end = ###(<cr><lf>)

set default(<cr><lf>)
Reloads the phase angle information with the default values.
(Pre trigger start = 30°, end = 60°, shutter open = 90°, shutter close = 270°, post trigger start = 300°, end= 330°)
Returns a list of updated angles

set nikon(<cr><lf>)
Loads the phase angle information with values that work well for the two-stage shutter release used by some Nikon and other consumer-type SLR cameras, The pre-trigger signal becomes release stage 1 (turn camera and metering on). The shutter signal becomes stage two (shutter release)
(Pre trigger start = 5°, end = 271°, shutter open = 90°, shutter close = 270°, post trigger start = 300°, end= 330°)
Returns a list of updated angles

write eprom(<cr><lf>)	Saves the current shutter, pre-trigger and post-trigger angles to non-volatile memory. These values will be reloaded on the next power-up. Function returns a confirmatory message

Appendix C; using HyperTerminal

HyperTerminal comes installed as a standard desktop accessory on most Windows PC's and is a convenient and readily available tool. On most PC's HyperTerminal is accessible directly from the Start menu, it's usually installed as…

Start > Programs > Accessories > Communications > HyperTerminal > Hyperterm.exe

When invoking HyperTerminal, a screen appears asking you to choose a name (and possibly an icon) for the connection. Type in a throwaway name (try "cam") and hit <OK>.

The first of two session setup screens appears and asks you to choose a connection. Choose an available hardware serial port on your machine.

Most desktop PC support two serial ports on the motherboard, COM1 and COM2.

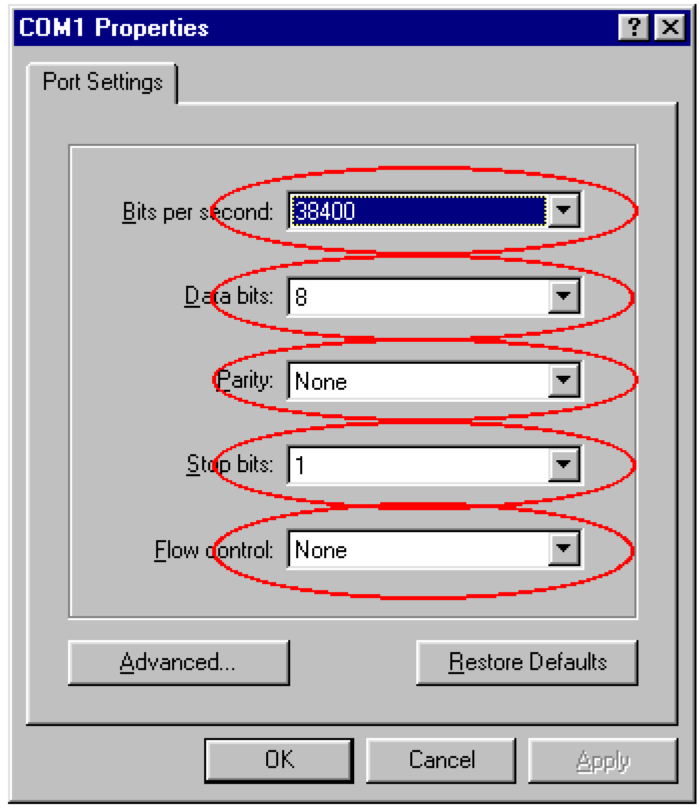
Many more recent laptops no longer have built-in serial ports. If this is the case with your laptop, you may have to use a USB-to-RS232 adapter dongle.

Identify the serial port you intend to use, then press <OK>.

The second session setup screen appears and asks you to specify the serial parameters for communication. The Camera trigger box uses the following serial parameters:

- 38400 K baud
- 8 data bits
- no parity
- 1 stop bit
- flow control none

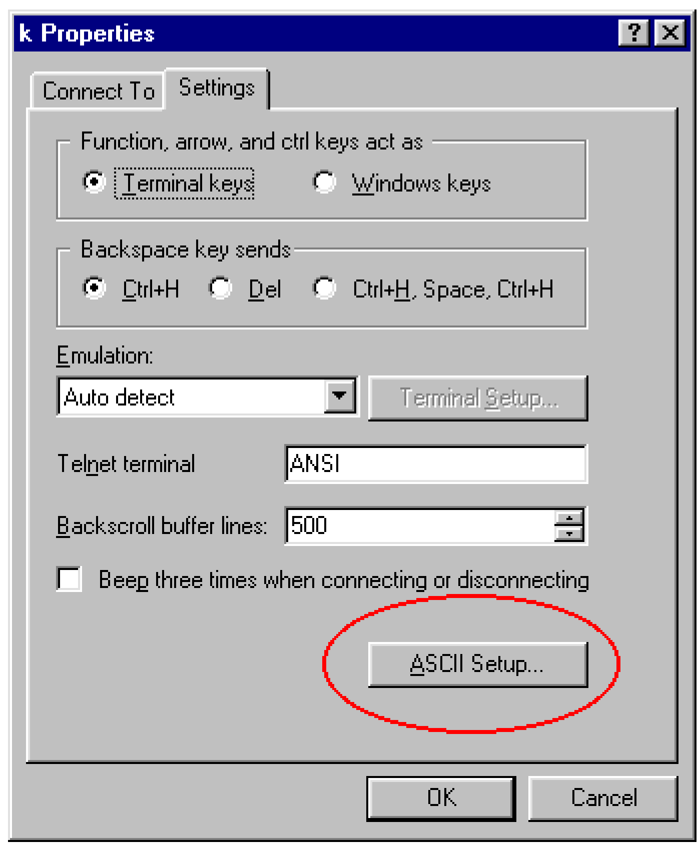
(hardware flow control is *supported* by the box, but is not *required*)



In addition to the serial parameters, you will have to specify that you want to use standard ASCII end-of-line signaling (This option places a standard carriage return and line feed character at the end of each transmitted line)

Open the "Properties" window from the "File" tab.

Choose "ASCII Setup"



In the "ASCII setup" menu, choose

"Send line ends with line feeds"
(this adds carriage returns and line feeds to the transmitted string, so the box can tell when you've hit return)

and "Echo typed characters locally"
(this displays the sent characters on the PC screen, so you can see both sides of the conversation)