FLEXIPILOT 1.35

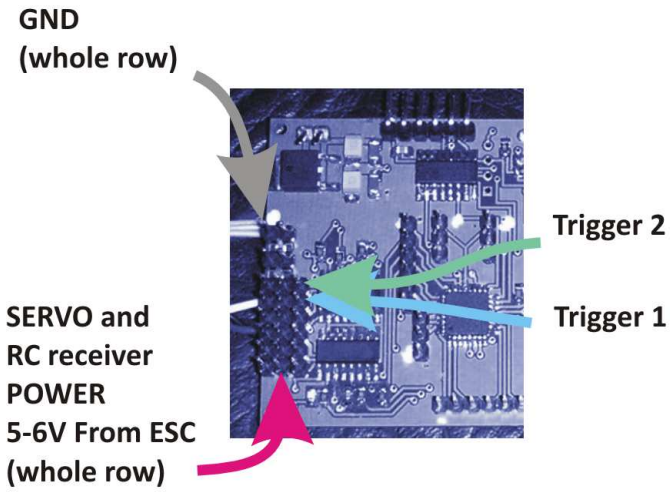
Using triggers for aerial photography

The document describes connection and setup of the triggers, which are additional servo outputs used for controlling external equipment either mechanically or by using custom electronics. Anything than can be plugged into a standard RC receiver, can be used as a receiver of trigger events. The list of devices found on the market includes electrical switches and camera-specific interfaces for triggering photos.

Connection

The FLEXIPILOT has 2 connectors at the edge of the board.
On some versions, 3-wire connectors are routed from the board and the wires are ordered as follows:

Black: ground
Red: power 5-6V depending on external servo power rail that is plugged on the same connectors
Yellow: PWM signal from the autopilot, scaled to 0..5-6V depending on external servo power rail



The signal output is a PWM output with 60Hz period and 0.6-2.4ms duty cycle.
If PWMOUT_MODE option is not 0, the update rate is 300Hz (see installation manual).
Only digital servos can handle update rates higher than 120Hz, 60Hz being the standard frequency (higher frequencies tend to overheat analog servos).
The duty cycle length of 2ms corresponds to 20000 servo position value used in the autopilot.
Other possible values are 0 = no signal, 1 is high level.

Minimal recommended values for the standard servo is 6000, maximal is 24000.
The extreme limit values could damage many cheaper servos; safe limits are 10000-20000.
The rotation direction of the servos can vary from manufacturer to manufacturer.

The triggers are powered from the same source as all servos, i.e. from the ESC Motor Controller Power Output or external BEC (power regulator). The power is completely separate from the autopilot power, because even if only one battery is finally used, both power lines are using separate voltage regulating circuits.

The trigger outputs are limited to driving 1mA at 5V therefore shouldn't be connected directly to relays: use transistor and other electronic components.

Principles of operation

Trigger states

Each trigger logic has 3 states: Disabled, Enabled&Inactive and Enabled&Active.
The user has control over enabling and disabling the trigger. Once the trigger is Enabled, it can become Active using prescribed pattern depending on setup variables described below.

You can imagine the trigger as machinery driving the cuckoo clock:

-when the trigger is **Disabled**, The cuckoo is in the house
-when the trigger is **Enabled** but still **Inactive**, the doors are open, and the cuckoo is preparing for the next opportunity to be Active
-when the Trigger is **Enabled**, the cuckoo immediately makes a sound once (goes **Active**) then waits
-you can implement camera protective door using this feature

External logic or RC transmitter only toggles state between Enabled and Disabled. The changing between Inactive and Active can be automatic (many options), always ON or always OFF.

Trigger output change conditions

How the trigger state can change:

- The trigger can be enabled or disabled using selected channel of the RC receiver (INCHANNEL/OUTCHANNEL trigger logic)
- It can be enabled or disabled on selected waypoints. Also when the mission is terminated, the autopilot can disable all waypoints (see navigation manual)
- Trigger can be enabled using console command (TRIGENABLE/TRIGDISABLE)
- You can cause distance-based activation using special command (TRIGDISTADV) from the console

When a trigger will move to the positions bypassing its logic

- Trigger can be set to copy directly the value of INCHANNEL (TRIGDIRECT command)
- Normally, a trigger is connected directly to RC receiver or unconnected input when in manual mode (but it is wired to autopilot all the time in some installations).
  A digital trigger servo will remember the last valid position if the input is unconnected.
- The command SERVODIRECT will connect by logic all servos to RC receiver also in automatic mode (SCAP0..5 to PWMOUT0..5), but this will have no effect if SCAPx corresponding to trigger output is not wired to RC receiver on some installations.

Electric installations

Several electric installations are possible:

1. In the most basic installation, the autopilot has control over trigger when it is enabled (automatic mode), otherwise the output is 0.
2. Autopilot controls trigger servos all the time (special wiring on the autopilot PCB)
3. Autopilot controls servo when enabled, RC transmitter sends servo position when autopilot is disabled (if SERVOCAP inputs corresponding to trigger outputs are wired to RC receiver)

Photolock

In order to protect the autopilot from maneuvering when shooting photos, the PHOTOLOCK feature is possible which freezes servo movements for a short time when the trigger is activated. During this period the autopilot is not steering the plane's course and the roll and pitch are more likely to remain constant if the airframe is stable.
During Photolock:
-stabilized head movement is frozen (removing jitter-induced blur on the photos)
-stabilized head angle errors increase due to short-term turbulence
-all output channels defined in MIXOUT logic as photo-lockable are not updated

The most important timestamp is the Activation Time. After Enabling (either by remote or by hitting a waypoint), the trigger becomes active for the first time. All delays and PHOTOLOCK are offset relative to this timestamp.

Diagnostics

In order to control the state of the triggers you may use s/S command to show actual servo input list, q/Q command to show trigger state list, @@@TRIGENABLE1 , @@@TRIGENABLE2, @@TRIGDISABLE1 and @@@TRIGDISABLE2 commands, also y/Y for showing the output servo list. You can test distance-triggered triggers using @@@TRIGDISTADV1, @@@TRIGDISTADV2. You can use @@@TRIGDIRECT1 or @@@TRIGDIRECT2 in order to connect trigger input channel directly to output channel, overriding the remaining logic. This allows noting servo positions when setting up mechanics.

s/S: Servo input listing displays in order:

```
ch0    ch1  ch2   ch3                      …ch12 live-timer
0 0 0 0 0 0 0 0 0 0 0 0 18771
```

q/Q: Trigger status list displays:

```
   enabled=no   active:no photolock:no  servo position
T1 E:0          A:0        P:0          S:20000
```

y/Y: Output servo displays:

```
ch0=rudd      ch1=elev   void    ch3=thr   ch4=trigger1   ch5=trigger2
13999       14309        0       0           20000         20000
```

Recorded events are logged in the event log. For example, @@@TRACEDUMP displays:

```
#ART00000:61045.6,+21.134138,+037.134558,+0000.1,-011,+003,+337,2009-08-10,16:57
:25.600,1#
```

The number in bold indicates trigger number, 1 or 2. Use ‘C’ or @@@TRACECLEAR to erase the TRACE after testing.

Waypoint logic

While the waypoint syntax is described in detail in navigation manual,
we repeat it here for easier access.

Syntax: @@@WPTWRITE*wptnumber*:*param1*,*param2*,*AGL*,*airspeed*,
position_mode,*triggers*

Triggers:

Trigger parameter defines what actions will be taken when waypoint is met

3	TRIG1ON
1	TRIG1OFF
12	TRIG2ON
4	TRIG2OFF
7	1ON, 2OFF
13	2ON, 1OFF
15	both on
5	both off
16	Deploy parachute
32	Arm parachute (will open when plane descends below safe altitude)
64	Disarm parachute (stop altitude monitoring)
128	Disable Throttle
256	Enable Throttle
512	Disable ThrottleB
1024	Enable ThrottleB
2048	Radio Silence until next waypoint (modem off)
4096	Disable manual mode until next waypoint (RXOVR disable)

Trigger parameter defines what actions will be taken when waypoint is met.
Trigger parameters are in fact binary values:

trigger bits:	***15 Reserved, Keep 0***	***14 Reserved, Keep 0***	***13 Reserved, Keep 0***	***12 RXOVR Disable 1 leg***	***11 Radio Silence 1 leg***	***10 THRB on***	***9 THRB off***	***8 THR on***
trigger bits:	***7 THR off***	***6 disarm parachute***	***5 arm parachute***	***4 deploy parachute immediately***	***3 Trigger2 Enable/Disable***	***2 Trigger2 Change State***	***1 Trigger1 Enable/Disable***	***0 Trigger1 Change State***

Example:
when flying over forest, you might want to disable parachute deployment at this specific seciont.
Use binary value of
0100 0000bin=32dec
Example:
Enable Trigger1, disable Trigger2:
0000 0111bin=7dec
Example:
Deploy parachute for landing:
0001 0000bin=16

Default values

Consider the following values

```
@@@TRIGGER1_SERVOPOS_ACTIVE=10000
@@@TRIGGER1_SERVOPOS_INACTIVE=20000
@@@TRIGGER1_SERVOPOS_DISABLED=20000
@@@TRIGGER1_ACTIVE_TIMEOUT=2
@@@TRIGGER1_INACTIVE_TIMEOUT=0
@@@TRIGGER1_PHOTOLOCK_TIMEOUT=0
@@@TRIGGER1_SERVO_ACTIVATION_DELAY=0
@@@TRIGGER1_TRACE_DELAY=0
@@@TRIGGER1_ATD_REACTIVATE_DIST=0
@@@TRIGGER1_TRIP_REACTIVATE_DIST=0
@@@TRIGGER1_TIMEOFFSET=0
@@@TRIGGER1_DISTOFFSET=0
@@@TRIGGER1_OUTCHANNEL=4
@@@TRIGGER1_RX_CHANNEL=-1
@@@TRIGGER1_RX_OFFZONE_MIN=6000
@@@TRIGGER1_RX_OFFZONE_MAX=12000
@@@TRIGGER1_RX_ONZONE_MIN=18000
@@@TRIGGER1_RX_ONZONE_MAX=24000
@@@TRIGGER1_MAXNSHOTS=-1
@@@TRIGGER1_SHOTSRELOAD=0
@@@TRIGGER1_SHOT2THR=0
@@@TRIGGER1_SHOT2THRMAX=0
@@@TRIGGER1_SHOT2THRB=0
@@@TRIGGER1_SHOT2THRBMAX=0

@@@TRIGGER2_SERVOPOS_ACTIVE=10000
@@@TRIGGER2_SERVOPOS_INACTIVE=20000
@@@TRIGGER2_SERVOPOS_DISABLED=20000
@@@TRIGGER2_ACTIVE_TIMEOUT=2
@@@TRIGGER2_INACTIVE_TIMEOUT=0
@@@TRIGGER2_PHOTOLOCK_TIMEOUT=0
@@@TRIGGER2_SERVO_ACTIVATION_DELAY=0
@@@TRIGGER2_TRACE_DELAY=0
@@@TRIGGER2_ATD_REACTIVATE_DIST=0
@@@TRIGGER2_TRIP_REACTIVATE_DIST=0
@@@TRIGGER2_TIMEOFFSET=0
@@@TRIGGER2_DISTOFFSET=0
@@@TRIGGER2_OUTCHANNEL=4
@@@TRIGGER2_RX_CHANNEL=-1
@@@TRIGGER2_RX_OFFZONE_MIN=6000
@@@TRIGGER2_RX_OFFZONE_MAX=12000
@@@TRIGGER2_RX_ONZONE_MIN=18000
@@@TRIGGER2_RX_ONZONE_MAX=24000
@@@TRIGGER2_MAXNSHOTS=-1
@@@TRIGGER2_SHOTSRELOAD=0
@@@TRIGGER2_SHOT2THR=0
@@@TRIGGER2_SHOT2THRMAX=0
@@@TRIGGER2_SHOT2THRB=0
@@@TRIGGER2_SHOT2THRBMAX=0
```

This trigger has only 2 positions, it goes active for 2s once enabled, then is inactive and MUST be disabled (either by waypoint, by remote control or @@@TRIGDISABLE1) before it can be enabled again. There are no delays, and no RC control over the trigger. The total number of shots that can be taken this way is infinite.
Default output channel numbers are PWMOUT4 and PWMOUT5.

Setup variables

TRIGGER1_OUTCHANNEL
TRIGGER2_OUTCHANNEL
Possible values: -1…5
Logic **output** channel.
-1 Disable trigger output (but not its logic, can be used for other purposes)
0-5 Selecting PWMOUT0-5 as output
Note: default positions are
TRIGGER1_OUTCHANNEL=4
TRIGGER2_OUTCHANNEL=5 but PWMOUT 2 is often unused and free.
If both triggers use the same output channel,
the one with higher priority is affecting servo position.
The priority counted in ascending order is: unconfigured (invalid), disabled, inactive, active.
If both triggers are in the same state, TRIGGER1 is selected.
Examples:
- {TRIGGER1_OUTCHANNEL=-1, TRIGGER2_OUTCHANNEL=3}
Regardless on the state of TRIGGER2, output channel 3 will be written and TRIGGER1 is ignored.
- {TRIGGER1_OUTCHANNEL=5, TRIGGER2_OUTCHANNEL=5, TRIGGER1 disabled, TRIGGER2 inactive}
Output channel 5 will be written by TRIGGER2_SERVOPOS_INACTIVE
- {TRIGGER1_OUTCHANNEL=4, TRIGGER2_OUTCHANNEL=4, TRIGGER1 disabled, TRIGGER2 disabled}
Output channel 4 will be written by TRIGGER1_SERVOPOS_DISABLED
The prioritization allows mixing the use of a single camera, two trigger logic and custom stabilized head movement using STABn*_TRIG1COEFF, STABn_TRIG2COEFF, STABn_TRIGINHIBIT_ROLL, STABn_TRIGINHIBIT_PITCH, STABn_TRIGINHIBIT_YAW*.

TRIGGER1_RX_CHANNEL
TRIGGER2_RX_CHANNEL
Possible values: -1..11
Logic **input** channel.
-1 disables remote triggering, 0..11 is input channel number. Some of the channels are of course already used by regular airplane controls like rudder (by default PWMOUT0) elevator (by default PWMOUT1) throttle (by default SCAP3), RPM1 counter (optional, SCAP10) and autopilot override (always SCAP11).
One can label this ‘trigger input channel’.

TRIGGER1_RX_OFFZONE_MIN
TRIGGER2_RX_OFFZONE_MIN
TRIGGER1_RX_OFFZONE_MAX
TRIGGER2_RX_OFFZONE_MAX
TRIGGER1_RX_ONZONE_MIN
TRIGGER2_RX_ONZONE_MIN
TRIGGER1_RX_ONZONE_MAX
TRIGGER2_RX_ONFZONE_MAX
Possible values: 0…24000
The RC receiver can either enable or disable the trigger. Use non-overlapping zones (like 8000…13000 OFF, 17000…22000 ON) when using repetitive triggers to avoid jumping

between states due to noise. The value 1-5999 is disabling as servo input never reaches those values and condition is never met.

TRIGGER1_SERVOPOS_ACTIVE
TRIGGER1_SERVOPOS_INACTIVE
TRIGGER1_SERVOPOS_ DISABLED
TRIGGER2_SERVOPOS_ACTIVE
TRIGGER2_SERVOPOS_INACTIVE
TRIGGER2_SERVOPOS_DISABLED
Possible values: 0, 1, 6000…24000
Default:
Active=16000
Inactive=15000
Disabled=12000
0 is low level, 1 is always high level, safe values are 10000-20000 (corresponding to 1ms…2ms active pulse).

TRIGGER1_ACTIVE_TIMEOUT
TRIGGER2_ACTIVE_TIMEOUT
Possible values: 0…1e6s
This timer defines how long after being activated; the trigger is going to remain active.
0 is infinite, the trigger will remain active until is disabled.

TRIGGER1_INACTIVE_TIMEOUT
TRIGGER2_INACTIVE_TIMEOUT
Possible values: 0…1e6s
Once the trigger has been enabled, it can become inactive after certain time.
This timer defines how low after being inactivated; the trigger is going to reactivate itself.
Combining this value with ACTIVE_TIMEOUT gives repetitive action.
0 is infinite, the trigger will remain inactive after activity period elapses, until is disabled.

TRIGGER1_PHOTOLOCK_TIMEOUT
TRIGGER2_PHOTOLOCK_TIMEOUT
Possible values: 0…8s
Defines for how long the control surfaces will be blocked after enabling the trigger. Use with extreme caution with times above few s, at low altitude and with repeating triggers.

TRIGGER1_SERVO_ACTIVATION_DELAY
TRIGGER2_SERVO_ACTIVATION_DELAY
Possible values: 0…1e6s
This is the delay between activating the timer and the servo going to active state.
Those delays are useful for using both triggers on the same object, i.e. opening the protective cover, another making the photo. This is the exception to the rule when the trigger is already activated, but until the activation delay elapses, the servo is in inactive position.

TRIGGER1_TRACE_DELAY
TRIGGER2_TRACE_DELAY
Possible values: 0…1e6s
The time elapsing between servo move and camera shutter being triggered can take several seconds. You can measure this time with a stopwatch and use for more precise information.
0 is immediate logging.

TRIGGER1_ATD_REACTIVATE_DIST
TRIGGER2_ATD_REACTIVATE_DIST
Possible values: 0…1e6km
One method to shot continuous photos is to use timed trigger, TRIGGERX_INACTIVE_TIMEOUT. In order to avoid wind influence and to shot photos at regular intervals over the ground, use Along-Track Distance retrigger. Note the distances are in km. The distance counter goes 0 once the waypoint is met.
Note that with WAYPOINT_ROUNDCORNER_RADIUS > 0 the plane will not be passing precisely over waypoint when switching to the next and the ATD triggering precision will be diminished.
0 is disabled.

TRIGGER1_TRIP_REACTIVATE_DIST
TRIGGER2_TRIP_REACTIVATE_DIST
Possible values: 0…1e6km
Similar as ATD retrigger, but uses the distance flown over ground, which is slightly different than ATD particularly in tight patterns when there is a lot of overshoot around waypoints.
In general ATD gives better results inside the zone of interest. TRIP retrigger allows easier photo stitching along the whole trip.
0 is disabled.

TRIGGER1_TIMEOFFSET
TRIGGER2_ TIMEOFFSET
Possible values: 0..1e6s
All trigger actions that are based on timers will be delayed by this time offset. Therefore, it is possible to activate both timers simultaneously, but the servos will move in sequence, allowing implementation of camera protection doors.
0 means no offset.

TRIGGER1_DISTOFFSET
TRIGGER2_ DISTOFFSET
Possible values: 0..1e6km
Similar to timeoffset, but this time distance will be added to TRIP and ATD triggers, allowing implementation 2 cameras for increased photo coverage, using 2 separate triggers.
0 means no offset.

TRIGGER1_MAXNSHOTS
TRIGGER2_MAXNSHOTS
Possible values: -1…32000
If the photo storage is limited, or you want just N photos per flight, use this counter.
The counter is diminished during activation.
-1 is infinite number of activations.

TRIGGER1_SHOTSRELOAD
TRIGGER2_SHOTSRELOAD
Possible values: 0, 1
When reloading the number of shots at waypoint is enabled, the shots counter will effectively allow shooting N photos each time a trigger is being enabled.
Once again, this can occur at waypoint or using RC receiver.

TRIGGER_HOME_DISABLE
Possible values: 0, 1, 2
Sometimes you don't want to make photos when the mission has been aborted for any reason (like going out of range). This could mean opening a parachute, exposing costly equipment for damage.
0 Triggers can be enabled anytime
1 Triggers will disable automatically after finishing or aborting a mission (but not when using RC override, loiter or magnet-home using RC)
2 Triggers will be held disabled also before takeoff
The function always affects both triggers. This action will be logged in TRACE with ID=7. After disabling, using WPTRESET will restart navigation engine. In such case a trigger could be enabled again. Until then, no trigger enabling will be possible.
Since parachute deployment is also causing return home, it will disable selected trigger, make enabling it impossible, however if trigger enabling RC channel will remain in ON zone all the time – the trigger will remain enabled.

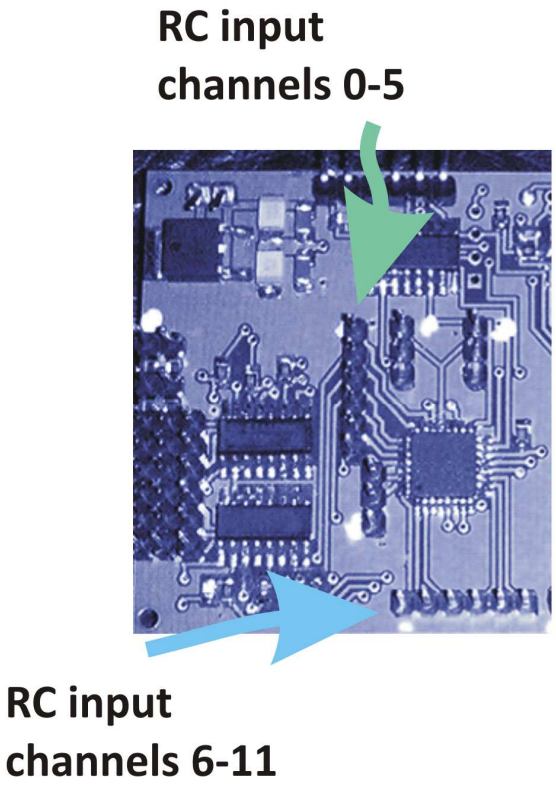
TRIGGER_PHOTOLOCK_MINAGL
Possible values: -1000, 30000m
The PHOTOLOCK feature blocking the autopilot can be unsafe at low altitude. It will not be used UNLESS the altitude above takeoff point is at leas N meters. Use *-1000* in order to always enable PHOTOLOCK, *30000* for always disabling PHOTOLOCK regardless on other settings. The photolocking function freezes update of specific servos:
- all stabilized head servos will not move
- all channels (0-5) marked at MIXOUT module with a special flag 128 will not be updated: this can include any control surface or throttle

TRIGGER1_SHOT2THR
TRIGGER1_SHOT2THRMAX
TRIGGER1_SHOT2THRB
TRIGGER1_SHOT2THRBMAX
TRIGGER2_SHOT2THR
TRIGGER2_SHOT2THRMAX
TRIGGER2_SHOT2THRB
TRIGGER2_SHOT2THRBMAX

Possible values: -24000…24000
Those mixers add to the central and max throttle value of throttle control PID every time a trigger is activated. This allows closing throttle and connecting secondary parachute to a trigger.

Remote triggering – connecting to the RC receiver

You can connect any number of input channels to the RC receiver, you can also use secondary RC receiver. Note however, that channels 0,1 and 3 are used by RUDD, ELEV and THROTTLE. Channels 4-5 are multiplexed with trigger outputs and when the autopilot is inactive, the control over the triggers is direct from the receiver. Channel 2 is typically a parachute.
It is easiest to use channels 6-11 as inputs unless you plan to use RC receiver in order to control the shutter directly.



Examples

Continuous shooting using slow Olympus C50Z and PRISM Shutter:

```
@@@TRIGGER2_SERVOPOS_ACTIVE=10000
@@@TRIGGER2_SERVOPOS_INACTIVE=20000
@@@TRIGGER2_SERVOPOS_DISABLED=20000
@@@TRIGGER2_ACTIVE_TIMEOUT=2
@@@TRIGGER2_INACTIVE_TIMEOUT=7  //when repetitive,
@@@TRIGGER2_PHOTOLOCK_TIMEOUT=0 // ← disable photolock!
@@@TRIGGER2_SERVO_ACTIVATION_DELAY=0
@@@TRIGGER2_TRACE_DELAY=3.5
@@@TRIGGER2_ATD_REACTIVATE_DIST=0
@@@TRIGGER2_TRIP_REACTIVATE_DIST=0
@@@TRIGGER2_TIMEOFFSET=0
@@@TRIGGER2_DISTOFFSET=0
@@@TRIGGER2_OUTCHANNEL=5
@@@TRIGGER2_RX_CHANNEL=-1
@@@TRIGGER2_RX_OFFZONE_MIN=8000
@@@TRIGGER2_RX_OFFZONE_MAX=16000
@@@TRIGGER2_RX_ONZONE_MIN=18000
@@@TRIGGER2_RX_ONZONE_MAX=22000
@@@TRIGGER2_MAXNSHOTS=-1
@@@TRIGGER2_SHOTSRELOAD=0
@@@TRIGGER2_SHOT2THR=0
@@@TRIGGER2_SHOT2THRB=0
@@@TRIGGER2_SHOT2THRMAX=0
@@@TRIGGER2_SHOT2THRBMAX=0
```

The trigger activates for 2s then stays inactive for another 7s. The position of the photo is logged 3.5s after activation (photo camera response to IR trigger is very slow in this case). Remote triggering is deselected – on this case you cannot toggle trigger using RC.

Shot 3 photos every time you enable it by remote RC receiver, using SCAP channel 9, or at waypoints.

```
@@@TRIGGER1_SERVOPOS_ACTIVE=10000
@@@TRIGGER1_SERVOPOS_INACTIVE=20000
@@@TRIGGER1_SERVOPOS_DISABLED=20000
@@@TRIGGER1_ACTIVE_TIMEOUT=1
@@@TRIGGER1_INACTIVE_TIMEOUT=2
@@@TRIGGER1_PHOTOLOCK_TIMEOUT=0
@@@TRIGGER1_SERVO_ACTIVATION_DELAY=0
@@@TRIGGER1_TRACE_DELAY=1
@@@TRIGGER1_ATD_REACTIVATE_DIST=0
@@@TRIGGER1_TRIP_REACTIVATE_DIST=0
@@@TRIGGER1_TIMEOFFSET=0
@@@TRIGGER1_DISTOFFSET=0
@@@TRIGGER1_OUTCHANNEL=4 ← output channel is PWMOUT4, 5th pin
@@@TRIGGER1_RX_CHANNEL=9 ← input channel from RC is 9,
                            this is 10th servo capture pin, what means 4th pin in 2nd group
                            of pins (see installation manual)
@@@TRIGGER1_RX_OFFZONE_MIN=8000
@@@TRIGGER1_RX_OFFZONE_MAX=16000
@@@TRIGGER1_RX_ONZONE_MIN=18000
@@@TRIGGER1_RX_ONZONE_MAX=22000
@@@TRIGGER1_MAXNSHOTS=3
@@@TRIGGER1_SHOTSRELOAD=0
@@@TRIGGER1_SHOT2THR=0
@@@TRIGGER1_SHOT2THRB=0
@@@TRIGGER2_SHOT2THRMAX=0
@@@TRIGGER2_SHOT2THRBMAX=0
```

Manual activator relay. Some camera interfaces allow triggering photos upon high signal on specific line (+5V). You can use Trigger Logic from the FLEXIPILOT in order to translate RC receiver signals (alternating PWM signal) into steady TTL 5V signals.
Note that some systems like to see 'active is low level',
as opposed to this example (this one is 'active high').

```
@@@TRIGGER2_SERVOPOS_ACTIVE=1    // Level high, +5V
@@@TRIGGER2_SERVOPOS_INACTIVE=0  // Level low, 0V
@@@TRIGGER2_SERVOPOS_DISABLED=0  // Level low, 0V
@@@TRIGGER2_ACTIVE_TIMEOUT=0     // When it becomes active,
                                    it will stay active until you disable it
@@@TRIGGER2_INACTIVE_TIMEOUT=0   // Disable photolock
@@@TRIGGER2_PHOTOLOCK_TIMEOUT=0  // Will never disable control surfaces when making photos
@@@TRIGGER2_SERVO_ACTIVATION_DELAY=0 // Signal change from 0 to 1 will take place
                                            immediately after activation
@@@TRIGGER2_TRACE_DELAY=0        // The event is logged immediately after activation/enable
@@@TRIGGER2_ATD_REACTIVATE_DIST=0
@@@TRIGGER2_TRIP_REACTIVATE_DIST=0
@@@TRIGGER2_TIMEOFFSET=0
@@@TRIGGER2_DISTOFFSET=0
@@@TRIGGER2_OUTCHANNEL=5
@@@TRIGGER2_RX_CHANNEL=8         // Using RX channel as input
@@@TRIGGER2_RX_OFFZONE_MIN=8000  // Low channel position disables
@@@TRIGGER2_RX_OFFZONE_MAX=16000 // Low channel position disables
@@@TRIGGER2_RX_ONZONE_MIN=18000  // High channel position enables
@@@TRIGGER2_RX_ONZONE_MAX=22000  // High channel position enables
@@@TRIGGER2_MAXNSHOTS=-1         // Infinite
@@@TRIGGER2_SHOTSRELOAD=0        // Nothing to reload at waypoint
@@@TRIGGER2_SHOT2THR=0
@@@TRIGGER2_SHOT2THRB=0
@@@TRIGGER2_SHOT2THRMAX=0
@@@TRIGGER2_SHOT2THRBMAX=0
```

As you can see the trigger works as 'fully manual' here:
Input channel SCAP8 is immediately changing the state of PWMOUT5.

Shot photos in regular intervals between waypoints, every 100m.

```
@@@TRIGGER1_SERVOPOS_ACTIVE=10000
@@@TRIGGER1_SERVOPOS_INACTIVE=20000
@@@TRIGGER1_SERVOPOS_DISABLED=20000
@@@TRIGGER1_ACTIVE_TIMEOUT=1
@@@TRIGGER1_INACTIVE_TIMEOUT=0
@@@TRIGGER1_PHOTOLOCK_TIMEOUT=0
@@@TRIGGER1_SERVO_ACTIVATION_DELAY=0
@@@TRIGGER1_TRACE_DELAY=1
@@@TRIGGER1_ATD_REACTIVATE_DIST=0.1
@@@TRIGGER1_TRIP_REACTIVATE_DIST=0
@@@TRIGGER2_TIMEOFFSET=0
@@@TRIGGER2_DISTOFFSET=0
@@@TRIGGER1_OUTCHANNEL=4
@@@TRIGGER1_RX_CHANNEL=-1
@@@TRIGGER1_RX_OFFZONE_MIN=8000
@@@TRIGGER1_RX_OFFZONE_MAX=16000
@@@TRIGGER1_RX_ONZONE_MIN=18000
@@@TRIGGER1_RX_ONZONE_MAX=22000
@@@TRIGGER1_MAXNSHOTS=-1
@@@TRIGGER1_SHOTSRELOAD=0
@@@TRIGGER1_SHOT2THR=0
@@@TRIGGER1_SHOT2THRB=0
@@@TRIGGER2_SHOT2THRMAX=0
@@@TRIGGER2_SHOT2THRBMAX=0
```

How to interface with a camera

There are several existing solutions for triggering a photo.
It is normally up to the user to research a solution on the Internet.

In order to get an idea of the possibilities:
*** Using mechanical lever** pressing the button and standard RC model servo. This offers typically maximum shot rate and easily interfaces with the autopilot (a servo plug)
It could be made like this one:
http://www.uscoles.com/kapservobracket.html

*** Using IR pilot shutter** like Hexpert Systems PRISM remote IR
http://www.hexpertsystems.com/prism/

*** Using Servo Camera Shutter**
URBI Shutter
http://www.blip.com.au/Products/UrbiRC.aspx
-PicoSwitch
http://www.dimensionengineering.com/PicoSwitch.htm

*** Using modified CHDK Canon firmware**, a modified USB connector directly connected to the autopilot.
http://chdk.wikia.com/wiki/CHDK
Example: Canon IXUS:
Trigger output from the autopilot can be connected directly to the cameras USB. I have ripped an old USB cable, and fitted the red and black wires directly to the red and black wires on a standard servo-cable. (The rest of the wires can be cut off). Next step would be to set up the correct parameters in the autopilot, in order to have it to output linear 5V current (instead of standard servo signals). Then run a mission in simulation mode in the autopilot, in order to measure +5V current on the wires when trigger is fired.
In the end connect camera, start CHDK script, and check that it fires when it is supposed to.
The CHDK scripts I have been running are remote.bas, for the procedure as described, and ult_interv.bas - for fixed time-interval firing.
In order to use CHDK you need SD card with max 4GB capacity.

remote.txt

```
@title Remote button
:loop
wait_click 1
is_key k "remote"
if k=1 then shoot
goto "loop"
end
```

Corresponding autopilot settings:

```
@@@TRIGGER1_SERVOPOS_ACTIVE=1 // 5.0V
@@@TRIGGER1_SERVOPOS_INACTIVE=0 // 0V
@@@TRIGGER1_SERVOPOS_DISABLED=0 // 0V
```

*** Using modified CHDK Canon firmware** interval shooting script

ult_interv.bas

```
rem Author - Keoeeit
rem Written for S-Series
rem Should be okay on others
rem Use Endless mode with caution
rem See documentation for important info
@title Ultra Intervalometer
@param a Delay 1st Shot (Mins)
@default a 0
@param b Delay 1st Shot (Secs)
@default b 3
@param c Number of Shots
@default c 5
@param d Interval (Minutes)
@default d 0
@param e Interval (Seconds)
@default e 3
@param f Interval (10th Seconds)
@default f 5
@param g Endless? No=0 Yes=1
@default g 0

p=a*60000+b*1000
t=d*60000+e*1000+f*100
if c<1 then let c=5
if t<100 then let t=100
if g<0 then let g=0
if g>1 then let g=1
if p<0 then let p=0
z=t*c
y=p+z

print "1 Cycle Time:", y/60000; "min", y%60000/1000; "sec"

goto "interval"

:interval
  if p>0 then gosub "pause"
  print "Shot 1 of", c
  shoot
  if c=1 then end
  for n=2 to c
  sleep t
  print "Shot", n, "of", c
  shoot
  next n
  if g=1 then goto "interval" else end

:pause
  n=(a*60)+b
  for m=1 to n
  q=n-m
  print "Intvl Begins:", q/60; "min", q%60; "sec"
  sleep 930
  next m
  return
```