

Chaos Alias V3.5 Instruction Manual

- Chaos Alias V3.5 — Instruction Manual
 - 1. Entering programming mode
 - 2. Trigger programming — step by step
 - 3. Button programming — step by step
 - 4. Complete settings reference
 - 4.1 — Option list (1-10, all reachable both ways)
 - 4.2 — Full value tables (trigger programming)
 - 5. Presets (option 4)
 - 6. Firing modes (option 5)
 - 7. Eye mode (option 8)
 - 8. Manual eye bypass (hold-to-toggle)
 - 9. Important: not independently verified end-to-end
 - 10. Known gaps / things to confirm on hardware

Chaos Alias V3.5 — Instruction Manual

Covers the current firmware build only. This build renumbers the option list to 1-10, consecutive, with the non-functional options removed entirely (not just hidden in documentation – actually unreachable in the firmware now). Earlier official manuals (e.g. V3.1.1) describe a different firing-mode list and ROF range – don’t mix the two.

⚠ **Dwell timing was corrected in this build – re-tune before you trust your old settings.** A real firmware bug made Dwell and Anti-bolt-stick added dwell (options 1 and 9) run at roughly **half** their labeled ms value in every prior build using this timer setup – a Dwell of “16” fired for ~7.9ms of real valve-open time, not 16ms. That’s now fixed at the firmware level, so Dwell and ASD-dwell numbers mean what they say again. The flip side: **flashing this build roughly doubles the real dwell time your marker produces at any Dwell/ASD-dwell setting number you’d already dialed in.** Re-verify your dwell from scratch after flashing – don’t assume your old numbers still make sense. See the timing note under the Dwell table in section 4.2 for the full technical detail.

There are two ways to change settings: **trigger programming** (count trigger pulls, LED blinks confirm) and **button programming** (two buttons step through options). Both write the same settings.

1. Entering programming mode

Mode	How to enter
Trigger programming	Hold the trigger down while turning the marker on . LED is solid orange (or solid green, depending on your LED) while you hold it, confirming you’re in. Release to begin.
Button programming	Press the programming button on the board. LED blinks red then green to show the current option and value.

On the orange LED: the firmware lights red and green together here – on most LEDs that reads as orange. Some LED parts have a weaker red channel, which makes this same red+green signal look mostly or solid green instead of orange. Either color means the same thing – you’re in trigger programming – this is a brightness difference between LED parts, not a setting or a firmware issue. (Confirmed: red works fine on its own, so it’s specifically the mixed red+green blend that reads differently brand to brand.)

2. Trigger programming — step by step

- Power on holding the trigger.** LED solid orange (or solid green on some LED parts – see note in section 1). Release – LED goes off.
- Pull and release the trigger the number of times** for the option you want (table in section 3). Each pull gets a quick green flash. You must let go between pulls – they’re counted as separate taps, not a hold.
- After your last tap, pull the trigger one more time and hold it down.** This is a separate, additional pull – not a continuation of your last tap. You can pause beforehand (up to a few seconds is fine). You’ll see one more quick flash as this pull begins – that’s expected, the firmware briefly counts it and then un-counts it about a third of a second in, once it recognizes you’re holding rather than tapping. Keep holding through that – about 1.3 seconds total – until the LED locks solid to confirm.
- If you let go too early, or never follow up with a hold, the LED flashes red 3 times (rejected) and the count resets – just try that step again.
- Once confirmed, **tap out the value** you want within that option the same way, then **pull one more time and hold** to confirm it.

6. LED alternates green/blue – the setting is saved.
7. You're back at option selection – pull for the next setting, or power off to exit.

A rejection at any point just means retry that same step; nothing already saved is lost.

3. Button programming — step by step

1. Press the programming button to enter. LED blinks red (option number) then green (current value).
2. **Button 1** steps to the next option.
3. **Button 2** steps the value up within the current option.
4. **Hold the trigger in for about 1.3 seconds** to save the shown option/value and move to the next option. A quick trigger pull with nothing held exits (LED blinks blue 3x).
5. If you stop pressing buttons for a few seconds, the LED re-blinks the current option and value so you can confirm where you are.

4. Complete settings reference

4.1 — Option list (1-10, all reachable both ways)

Pulls	Option
1	Dwell
2	Debounce (trigger)
3	Mechanical debounce
4	Load a preset
5	Firing mode
6	Rate of fire
7	After-eye delay
8	Eye mode
9	Anti-bolt-stick added dwell
10	No-bolt eye wait

Tourney lock hides options 1, 2, 3, 4, 5, and 9 while engaged – you can't touch dwell, debounce, mechanical debounce, presets, firing mode, or ASD dwell while locked. Options 6, 7, 8, and 10 (ROF, after-eye delay, eye mode, no-bolt eye wait) stay available.

4.2 — Full value tables (trigger programming)

Every table is **pulls -> resulting value**, at the value level within each option.

1 - Dwell (ms), pulls 1-16

Pulls	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Value	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

Timing note: this board's dwell timer (hardware Timer0) ticks every ~0.496ms, not 1ms. Every other "ms" setting in this manual that uses the same timer (ROF cap aside – that's a separate timer) already compensates for that in firmware by storing double the requested count. Dwell and Anti-bolt-stick added dwell (option 9, same timer) were the one place that compensation was missing – a labeled "Nms" dwell actually ran at ~N/2ms in every prior build using this timer setup.

That's now fixed at the firmware level in this build: `fire()` doubles the stored Dwell+ASD-dwell count before loading the hardware timer, matching the convention already used everywhere else in this codebase, so the Value column above is real ms again (accurate to roughly 1%, i.e. "16" really is ~15.9ms). I confirmed this by compiling the corrected source and checking the actual doubling instruction in the disassembled binary, not just reading the source – I was not able to get a clean live simulated measurement as an independent third check (the simulator I tried doesn't reliably replay this timer's reload behavior), so if you have a scope or logic analyzer on the valve driver line, that's the strongest remaining confirmation. **This is a real behavior change**, not just a documentation fix – see the warning at the top of this manual. Whatever Dwell/ASD-dwell numbers you were running before this build will now produce roughly double the real valve-open time; re-tune from scratch rather than trusting your old numbers.

2 - Debounce (trigger, ms), pulls 1-10 only

Pulls	1	2	3	4	5	6	7	8	9	10
Value	3	6	9	12	15	18	21	24	27	30

3 - Mechanical debounce, pulls 1-16

Pulls	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Value	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

4 - Preset, pulls 1-5 - see section 5 for what each preset sets.

Pulls	1	2	3	4	5
Preset	Extreme	Fast	Normal	Conservative	Default

5 - Firing mode, pulls 1-7 - see section 6 for behavior.

Pulls	1	2	3	4	5	6	7
Mode	Semi	Turbo	Auto response	Full auto	PSP ramp	NXL	Snapshot

6 - Rate of fire, pulls 1-32

Pulls	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
bps	Unlimited	20.0	19.5	19.0	18.5	18.0	17.5	17.0	16.5	16.0	15.5	15.0	14.5	14.0	13.5	13.0

Pulls	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
bps	12.5	12.0	11.5	11.0	10.5	10.0	9.5	9.0	8.5	8.0	7.5	7.0	6.5	6.0	5.5	5.0

Real-world rate can run a little under these at the fast end if the eye, after-eye delay, or no-bolt eye wait settings are the limiting factor rather than the ROF setting itself.

7 - After-eye delay, pulls 1-16

Pulls	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Value	manual eye bypass (always on)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

8 - Eye mode, pulls 1-6 - see section 7 for what each number does.

Pulls	1	2	3	4	5	6
Mode	0	1	2	3	4	5

9 - Anti-bolt-stick added dwell (ms), pulls 1-11 - same timer fix as Dwell (option 1); see the timing note under that table. This is additional dwell time added on top of the Dwell setting, now also corrected to real ms.

Pulls	1	2	3	4	5	6	7	8	9	10	11
Value	0	1	2	3	4	5	6	7	8	9	10

10 - No-bolt eye wait, pulls 1-16

Pulls	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Value	0	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30

5. Presets (option 4)

Pulls	Preset	Dwell (ms)	Mode	ROF	Debounce	After-eye delay
1	Extreme	12	Semi	Unlimited	5	1
2	Fast	16	Semi	Unlimited	7	2
3	Normal	16	Semi	20.0 bps	7	4
4	Conservative	20	Semi	15.0 bps	10	6
5	Default	16	Semi	Unlimited	7	4

6. Firing modes (option 5)

Pulls	Mode	Behavior
1	Semi-automatic	One shot per trigger pull.
2	Turbo	Alternates: odd pulls fire one shot, even pulls fire a second shot on release too.
3	Auto response	Every pull fires a second shot on release.
4	Full-automatic	Fires continuously while the trigger is held.
5	PSP ramping	After 3 consecutive pulls faster than ~210 ms apart, ramp engages and adds shots up to the ROF setting. Continues only as long as you keep pulling that fast – a pure hold with no further pulls drops ramp within ~210 ms.
6	NXL ramping	Ramp engages only on the 4th consecutive pull at the required speed (same ~210 ms window). Fires at a fixed ~10.4 bps while ramping – ignores the ROF setting. Same “keep pulling fast” requirement as PSP; releasing or slowing down drops ramp immediately, no grace period or carry-over.
7	Snapshot trainer	Fires up to 4 shots, then locks out the trigger for 2 seconds.

With tourney lock engaged, modes 2, 3, and 4 (Turbo, Auto response, Full-auto) aren’t selectable – the firmware resets to Semi if one of those is already saved.

7. Eye mode (option 8)

Pulls	Bolt visible to eye?	Scope mode?	Fire-after-timer?
1	No	No	No
2	No	No	Yes
3	No	Yes	No
4	Yes	No	No
5	Yes	No	Yes
6	Yes	Yes	No

“Bolt visible” determines whether the firmware expects to see the bolt cycle open-then-closed between shots, or relies on a timed wait instead (no-bolt eye wait, option 10). “Scope mode” and “fire-after-timer” are internal flags whose exact behavioral effect hasn’t been traced in detail – ask if you want that written up specifically.

8. Manual eye bypass (hold-to-toggle)

Separate from trigger/button programming – works during normal firing, marker powered on and ready to shoot.

Hold the trigger in continuously for about 3 seconds (no release) to toggle the eyes off. Hold ~3 seconds again to toggle back on.

- Doesn’t work in full-automatic mode (mode 4) – disabled there.
- Suppressed while NXL is actively ramping – a 3-second hold mid-burst won’t trigger it.

9. Important: not independently verified end-to-end

Everything above reflects a real firmware change (not just a documentation reorganization) – the option numbers 1-10 are now hard-coded in the firmware itself, consecutive, with the old non-functional options removed so they’re no longer reachable at all.

I verified this three ways: the full source diff only touches the option-numbering logic (nothing else); the compiled binary’s internal lookup table matches the intended 1-10 mapping exactly, checked byte-for-byte; and option-level pull counting in simulation produces identical, correct results whether testing the old or new numbering for an equivalent option. What I could **not** get a clean automated end-to-end confirmation of is the complete two-pull-sequence flow (option, then value, then save) in simulation – a persistent issue in my test setup affected both the old and new firmware identically, which tells me it’s very unlikely to be a bug from this change, but I wasn’t able to fully resolve it to get a clean pass/fail from the simulator on the complete flow.

Given that, **please test this build carefully before relying on it** – especially confirm that option 5 (Firing mode) actually lands on the mode you expect, and that a save genuinely sticks after a power cycle. If anything’s off, tell me exactly what you pulled and what you got, and I’ll dig further.

10. Known gaps / things to confirm on hardware

- The NXL “required trigger speed” (~210 ms between pulls) is a firmware constant, shared with PSP ramping, not separately programmable.
- The ROF table’s accuracy was simulated, not measured on a marker – worth a shot-count check at a couple of settings if you haven’t already.
- Whether to remove a particular timer read-modify-write (affects full-auto/NXL timing jitter) is still an open, un-applied decision from earlier in this project.