



NINJATRADER 8 INDICATOR · NINJAScript, C#

User Guide

An adaptive Supertrend driven by a Trend Quality Index, scored BUY and SELL signals on closed bars, stop and three targets, a dashboard, alerts, and series a strategy can read.

PRODUCT	Algo Shadow Matrix for NinjaTrader 8
VERSION	1.0.2
DATE	18 September 2026
NINJATRADER	8.1.x Desktop, tested on 8.1.8.2
PACKAGE	AlgoShadowMatrix.zip , compiled NinjaScript Add-On
SHA - 256	bea280794bdd8491390a2cc3c2762495b21c291ab8ab9084b2513f3d05466d11
AUTHOR	Kestutis Balciunas, https://kestutisbalciunas.com/
PRODUCT PAGE	https://kestutisbalciunas.com/ninjatrade/algo-shadow-matrix/
SUPPORT	https://kestutisbalciunas.com/contact/

Trading involves substantial risk. Futures, foreign currency and options trading contains substantial risk and is not for every investor. Past, simulated and backtested performance is not necessarily indicative of future results.

Contents

01 What it is

02 Installation

03 Quick start

04 Calculate mode and bar timing

05 Outputs for strategies and the Market
Analyzer

06 Parameters

07 Sessions, time and volume

08 Compatibility

09 Troubleshooting

10 How the port was verified

11 Licensing

12 Support

13 Risk disclosure

What it is

Chart indicator for NinjaTrader 8 Desktop (NinjaScript, C#): an adaptive Supertrend whose band width is set by a Trend Quality Index (market efficiency, volatility regime, price structure, momentum persistence), scored BUY/SELL signals on every confirmed flip (six factors, 0 to 102), entry / structure stop / three take-profit levels, a three-column dashboard with a self-learning R ledger, watermark and alerts. Signals only: it never places, modifies or closes orders.

Installation

Requirements

- NinjaTrader 8.1.x Desktop for Windows (tested on 8.1.8.2). Not for NinjaTrader 8.0 or NinjaTrader Web.
- Any data connection that delivers bars for the chart (the indicator needs no Order Flow+ features, no depth, no tick history). Volume is used when the feed supplies it.

Import the Add-On package

1. Control Center > Tools > Import > NinjaScript Add-On.
2. Select `AlgoShadowMatrix.zip` and confirm. NinjaTrader compiles the assembly and reports "Import complete".
3. No restart is needed. If a chart already had an older version, remove and re-add the indicator (a compile does not reload running instances).

Verify installation

1. Open a chart (any instrument, a 1 to 60 minute interval is a good first look), press Ctrl+I.
2. In the Available list find **Algo Shadow Matrix**, click add, click OK.
3. The chart shows the Supertrend line, BUY / SELL pills, the ENTRY / SL / TP set of the last signal, the dashboard in the top-right corner and the watermark at the bottom. The dashboard's Preset row reads `Ready` once 50 bars have loaded (longer lookback windows extend the warm-up).

Uninstall / update

- Remove: Control Center > Tools > Remove NinjaScript Assembly > Algo Shadow Matrix.
- Update: import the new zip over the installed version (NinjaTrader replaces the assembly), then remove and re-add the indicator on every chart that uses it, or reload the workspace.

Quick start

First chart

1. Add **Algo Shadow Matrix** to a chart (Ctrl+I). Leave every setting at its default.
2. Wait for the dashboard's Preset row to read **Ready** (50 bars with the defaults).
3. Leave **Preset** on **Auto**: it resolves **Scalping** on charts up to 5 minutes, **Default** up to 240 minutes, **Swing** above. **Crypto 24/7** and **Custom** are picked by hand.

Reading the chart

- **Supertrend line**: green while bullish, red while bearish. With **Color Line by TQI** on, the line is bright when trend quality is high and fades as it drops.
- **BUY / SELL pill**: a flip confirmed on bar close, with its score out of 102 (six factors: momentum, efficiency, volume, RSI depth, structure, break depth). Pills never repaint.
- **ENTRY / SL / TP1 / TP2 / TP3**: drawn on every signal, extended to the right while the trade is open. The stop is the last pivot minus (plus) **SL Buffer** × **ATR** and at least that far from the entry; the targets sit at 1R, 2R and 3R by default. A reached target turns cyan with a check mark. The set stays until the next signal.
- **Trend background** (off by default): tints the chart by trend direction.

The dashboard

Three columns, position and size from **08 Dashboard**. Rows and colours reproduce the original table cell for cell.

ROW	COLUMN 2	COLUMN 3
ALGO SHADOW MATRIX	by KESTUTIS BALCIUNAS	instrument and period
MARKET CONTEXT		
Trend Regime	Bullish / Bearish	Supertrend
Preset	resolved preset	Ready or the bars still needed
Regime / Vol	Trending (ER at or above 0.50), Mixed (0.25 to 0.50), Choppy	Low Vol (ATR ratio under 0.7), Norm Vol, High Vol (1.3 and above)
TQI ENGINE		
TQI Score	0 to 1	Q.Str effective quality strength
ER / RSI / VolZ	efficiency ratio and RSI	volume z-score
Eff / Vol, Struct / Mom	the four TQI factors (with Show TQI Breakdown)	
ACTIVE SIGNAL		
Direction	LONG / SHORT / WAITING	bars since the signal

ROW	COLUMN 2	COLUMN 3
Entry, Stop Loss	prices	distance in percent
TP1 / TP2 / TP3	prices, check mark after a hit	
PERFORMANCE	Sample	closed signals in memory / 100
W/L	rolling average R	win and loss streaks (current / max)
Window DD	worst run in the calibration window	all-time drawdown
Regime Edge	average R of the current efficiency x volatility cell with its count	

The ledger is rebuilt from the chart's history on every load, exactly as the original does; it is a description of the loaded bars, not a forecast.

Alerts

Evaluated once per closed bar and only for bars that close after the indicator went live (never during the history pass). Each alert fires once per bar and kind and appears in the Alerts Log with an optional sound (Sound File inside NinjaTrader's sounds folder) and an optional email through NinjaTrader's Mail share service.

TOGGLE	FIRES WHEN
Alert: BUY Signal / SELL Signal	a confirmed signal (body: price, TQI, score, SL, TP1, TP2, TP3; JSON with Webhook JSON Format)
Alert: Any Signal	either
Alert: Bullish / Bearish Trend Shift	the Supertrend flips, before the warm-up gate

Using the values in a strategy or the Market Analyzer

The indicator exposes three series: `SuperTrend` (the line), `Signal` (+1 BUY bar, -1 SELL bar, 0 otherwise) and `Trend` (+1 bullish, -1 bearish). Example: `if (AlgoShadowMatrix(...).Signal[0] > 0) .`

Calculate mode and bar timing

- Default `Calculate.OnBarClose`. Every engine state change happens on a closed bar (`Pine barstate.isconfirmed`).
- `OnEachTick` / `OnPriceChange`: the forming bar is previewed on every tick (Supertrend value, line colour, dashboard follow the live bar like the TradingView realtime bar) and committed once at `IsFirstTickOfBar` of the next bar. Signals, levels, ledger and alerts never repaint.
- Historical pass: one `OnBarUpdate` per closed bar. The last historical bar (still forming when the chart loads) is previewed, not committed, and committed by the first realtime close. The engine slot is derived from `CurrentBar`, so the historical to realtime transition can neither double-feed nor skip a bar; a skipped range is refilled from the `Bars` series.
- Tick Replay: handled as per-tick realtime.

Outputs for strategies and the Market Analyzer

PLOT / SERIES	VALUES
<code>SuperTrend</code> (plot 0, <code>SuperTrend</code> series)	the Supertrend line, NaN before warm-up or with bands off
<code>Signal</code> (plot 1, hidden)	+1 on a confirmed BUY bar, -1 on a confirmed SELL bar, 0 otherwise
<code>Trend</code> (plot 2, hidden)	+1 while bullish, -1 while bearish

Example in a strategy: `var asm = AlgoShadowMatrix(...); if (asm.Signal[0] > 0) ...` (the accessor is generated by NinjaTrader on compile).

Parameters

Of the 71 inputs of the original script, 70 keep their titles, groups, order and defaults (the group names carry a number prefix because NinjaTrader sorts groups alphabetically) and the `Source` input is NinjaTrader's own **Input series** setting of the indicator (Close by default). The 13 NinjaTrader-only additions (the original has one alert switch and the JSON format) are the five per-event alert toggles and the sound and email settings in group 10, and the label cap, score memory mode and the two diagnostic switches in group 11: 83 parameters in total.

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
01 Main Settings	Preset	AsmPreset	Auto		Preset	Quick presets that override key parameters. Auto picks by timeframe: Scalping up to 5 min, Default up to 4 h, Swing above.
01 Main Settings	ATR Length	int	13	5 to 100	AtrLen	
01 Main Settings	Base Band Width (xATR)	double	2.0	0.5 to 5.0	BaseMult	
02 Adaptive Engine	Enable Vol-Adaptive Bands (legacy ER)	bool	true		UseAdaptive	Legacy linear ER modulation. Kept for backwards compatibility; the TQI Engine is recommended.
02 Adaptive Engine	Efficiency Window	int	20	5 to 100	ErLength	
02 Adaptive Engine	Adaptation Strength	double	0.5	0.0 to 1.0	AdaptStrength	
02 Adaptive Engine	ATR Baseline Length	int	100	20 to 500	AtrBaselineLen	
03 Trend Quality Engine	Enable Trend Quality Engine	bool	true		UseTqi	Master switch. Computes a continuous Trend Quality Index (0..1) each bar from 4 factors and modulates the band width non-linearly.
03 Trend Quality Engine	Quality Influence	double	0.4	0.0 to 1.0	QualityStrength	How aggressively TQI compresses/expands bands. 0 = no effect, 1 = maximum.
03 Trend Quality Engine	Quality Curve Power	double	1.5	1.0 to 3.0	QualityCurve	Non-linearity. 1.0 = linear, above 1 bands expand slowly for mild quality drops, rapidly for severe ones.
03 Trend Quality Engine	Smooth Adaptive Multipliers	bool	true		MultSmooth	EMA-smooths band multipliers before applying ratchet logic. Recommended on.

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
03 Trend Quality Engine	Asymmetric Bands	bool	true		UseAsymBands	Active side (in direction of trend) tightens, passive side widens.
03 Trend Quality Engine	Asymmetry Strength	double	0.5	0.0 to 1.0	AsymStrength	
03 Trend Quality Engine	Efficiency-Weighted ATR	bool	true		UseEffAtr	Scale ATR by (0.5 + 0.5 x ER). Clean trending volatility counts full; noisy volatility is halved.
03 Trend Quality Engine	Character-Flip Detection	bool	true		UseCharFlip	Allow the Supertrend to flip on quality collapse, not just price break.
03 Trend Quality Engine	Char-Flip: Min Trend Age	int	5	1 to 50	CharFlipMinAge	
03 Trend Quality Engine	Char-Flip: High TQI	double	0.55	0.3 to 0.9	CharFlipHigh	
03 Trend Quality Engine	Char-Flip: Low TQI	double	0.25	0.0 to 0.5	CharFlipLow	
03 Trend Quality Engine	Weight: Efficiency	double	0.35	0.0 to 1.0	TqiWeightEr	
03 Trend Quality Engine	Weight: Volatility Regime	double	0.20	0.0 to 1.0	TqiWeightVol	
03 Trend Quality Engine	Weight: Structure	double	0.25	0.0 to 1.0	TqiWeightStruct	
03 Trend Quality Engine	Weight: Momentum Persist	double	0.20	0.0 to 1.0	TqiWeightMom	
03 Trend Quality Engine	Structure Window	int	20	5 to 100	TqiStructLen	
03 Trend Quality Engine	Momentum Persist Window	int	10	3 to 50	TqiMomLen	
04 Signal Filters (Display)	Use Structure in Score (display)	bool	true		UseStructure	
04 Signal Filters (Display)	Pivot Strength	int	3	2 to 10	PivotLen	
04 Signal Filters (Display)	Use RSI in Score (display)	bool	true		UseRsi	
04 Signal Filters (Display)	RSI Length	int	14	5 to 50	RsiLen	
04 Signal Filters (Display)	RSI Overbought	int	70	55 to 90	RsiOB	
04 Signal Filters (Display)	RSI Oversold	int	30	10 to 45	RsiOS	
04 Signal Filters (Display)	RSI Memory (bars)	int	20	3 to 100	RsiLookback	
04 Signal Filters (Display)	Use Volume in Score (display)	bool	true		UseVol	
04 Signal Filters (Display)	Volume Z Window	int	20	5 to 100	VolLen	
04 Signal Filters (Display)	Min Signal Score (display only)	int	60	0 to 100	MinScore	Declared by the original script and never read; kept for dialog parity.

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
05 Risk Management	Show TP/SL Levels	bool	true		ShowRisk	
05 Risk Management	SL Buffer (xATR)	double	1.5	0.3 to 5.0	SLAtMult	
05 Risk Management	TP1 (R-multiple)	double	1.0	0.5 to 10.0	Tp1R	
05 Risk Management	TP2 (R-multiple)	double	2.0	0.5 to 10.0	Tp2R	
05 Risk Management	TP3 (R-multiple)	double	3.0	0.5 to 10.0	Tp3R	
05 Risk Management	Label Offset (bars)	int	10	10 to 100	LabelOffset	
05 Risk Management	Mark TP Hits (visual)	bool	true		ShowHits	
05 Risk Management	Trade Timeout (bars)	int	100	10 to 500	TradeMaxAge	
06 Self-Learning	Enable Auto-Calibration (experimental)	bool	false		UseAutoCalib	EXPERIMENTAL: adjusts Quality Influence via mean-reversion drift on poor edge. No guarantee of improvement; recommended off until validated on your instrument.
06 Self-Learning	Calibration Window	int	20	5 to 100	CalibWindow	
06 Self-Learning	Bad-Edge Threshold (R)	double	0.0	-2.0 to 1.0	CalibBadR	
06 Self-Learning	Good-Edge Threshold (R)	double	0.7	0.0 to 5.0	CalibGoodR	
06 Self-Learning	Quality Step	double	0.05	0.01 to 0.3	CalibStepQ	
06 Self-Learning	Calibration Cooldown (signals)	int	5	1 to 50	CalibCooldown	
06 Self-Learning	Quality Floor	double	0.1	0.0 to 1.0	CalibMinQ	
06 Self-Learning	Quality Ceiling	double	0.9	0.0 to 1.0	CalibMaxQ	
06 Self-Learning	Reset Learning Memory	bool	false		ResetLearning	No-op, exactly as in the original script (the ledger restarts with every chart load anyway).
07 Visual	Theme	AsmTheme	Auto		Theme	Auto follows the chart background (dark or light).
07 Visual	Show SuperTrend Bands	bool	true		ShowBands	
07 Visual	Color Line by TQI	bool	true		ShowTqiColor	Line transparency reflects the current TQI. Bright = high quality, faded = low quality.
07 Visual	Show Signal Labels	bool	true		ShowSignals	
07 Visual	Show Trend Background	bool	false		ShowBg	
07 Visual	Show Watermark	bool	true		ShowWatermark	

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
08 Dashboard	Show Info Dashboard	bool	true		ShowDash	
08 Dashboard	Show TQI Breakdown	bool	true		ShowTqiBreakdown	
08 Dashboard	Show Score Breakdown (legacy)	bool	false		ShowBreakdown	
08 Dashboard	Show Performance Stats	bool	true		ShowPerf	
08 Dashboard	Dashboard Position	AsmDashPos	TopRight		DashPos	
08 Dashboard	Dashboard Size	AsmDashSize	Tiny		DashSize	
08 Dashboard	Dashboard Header Color	colour	#1976D2		DashAccentBrush	
09 Colors	Bull	colour	#00E676		BullBrush	
09 Colors	Bear	colour	#FF5252		BearBrush	
09 Colors	SL	colour	#FF1744		SlBrush	
09 Colors	TP	colour	#00E676		TpBrush	
10 Alerts	Enable Alert Signals	bool	true		EnableAlerts	
10 Alerts	Webhook JSON Format	bool	false		WebhookJson	Signal alert body as JSON instead of text (the message only; no network call is made).
10 Alerts	Alert: BUY Signal	bool	true		AlertBuy	
10 Alerts	Alert: SELL Signal	bool	true		AlertSell	
10 Alerts	Alert: Any Signal	bool	false		AlertAny	
10 Alerts	Alert: Bullish Trend Shift	bool	false		AlertBullShift	
10 Alerts	Alert: Bearish Trend Shift	bool	false		AlertBearShift	
10 Alerts	Play Sound	bool	true		PlaySound	
10 Alerts	Sound File	string	"Alert2.wav"		AlertSoundFile	File name inside the NinjaTrader sounds folder, for example Alert2.wav.
10 Alerts	Send Email	bool	false		SendEmail	Uses the Mail share service configured in NinjaTrader (Tools > Options > Share).
10 Alerts	Email To	string	""		EmailTo	
11 NinjaTrader	Max Signal Labels	int	300	50 to 500	MaxSignalLabels	Oldest signal labels are removed beyond this count (the original keeps 500).

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
11 NinjaTrader	Score RSI Memory	AsmRsiMemory	TradingView		RsiMemory	TradingView reproduces the published script's score evaluation; Strict20Bars is the plain lowest/highest the script intends.
11 NinjaTrader	Debug Mode	bool	false		DebugMode	Writes diagnostics to asm_nt_debug.log in the NinjaTrader 8 folder.
11 NinjaTrader	Dump Bars + Outputs CSV	bool	false		DebugDumpCsv	Writes every committed bar with the engine outputs to asm_dump__.csv (parity checks).

83 parameters.

Sessions, time and volume

Sessions

The indicator has no session logic of its own: every bar the chart delivers is fed to the engine in order, so the chart's Trading Hours template decides which bars exist. Two consequences:

- With an ETH template (for example `CME US Index Futures ETH`) the overnight bars are part of the ATR, efficiency and structure windows; with an RTH template they are not, and the warm-up counts RTH bars only. Both are valid; the values differ, as they do for any indicator.
- The `Auto` preset reads the chart period, not the session: a 5-minute chart resolves `Scalping` whatever the template.

Time stamps

NinjaTrader stamps a bar with its CLOSE time in the platform time zone. The engine's bar open time (used for draw-object tags and alert de-duplication only) is reconstructed as close time minus the bar period; for non-time bars the observed bar spacing is used. Level lines and labels are anchored at future times (`LabelOffset` bars past the current bar), which NinjaTrader positions from the bar period; on tick, volume or range bars the x position is approximate.

Volume

`Volume[]` on a futures chart is real contract volume. The original script used `volume`, which on forex feeds is tick volume; on NinjaTrader futures the volume z-score in the Trend Quality Index and the volume factor of the signal score therefore see exchange volume, a better input for the same formula. Symbols without volume fall back to the volatility ratio exactly as the original does (`hasVolume = false`).

Contract rollover

A continuous contract (merge policy `MergeBackAdjusted` or `MergeNonBackAdjusted` in the Data Series dialog) shows a price jump at the roll unless back-adjusted. The Supertrend bands, stop and targets are price differences, so a non-back-adjusted roll produces one unusual bar; with back adjustment there is none. State the merge policy when comparing values across machines.

Compatibility

ITEM	STATUS
NinjaTrader version	8.1.x Desktop (tested 8.1.8.2). Not 8.0, not NinjaTrader Web or mobile.
Product type	Indicator, price panel overlay, single data series.

ITEM	STATUS
Instruments	Any instrument with bars: futures, micro futures, forex, stocks, crypto CFDs. Verified on MNQ (CME) 1-minute; the engine was verified on EURUSD M5 (stage 1 parity).
Bar types	Time-based (second, minute, day, week, month) fully; tick, volume, range, Renko and other non-time bars run, with the <code>Auto</code> preset and the drawn level positions derived from the observed bar spacing.
Calculate modes	<code>OnBarClose</code> (default), <code>OnEachTick</code> , <code>OnPriceChange</code> (forming-bar preview, no repaint of committed bars). Tick Replay handled as per-tick.
Multi-instance	Yes: every draw object and alert id carries the instrument, period and instance hash.
Market Analyzer	Loads (no chart drawing); the <code>SuperTrend</code> , <code>Signal</code> and <code>Trend</code> series are available as columns. Not advertised as a scanner.
Strategy Analyzer / Strategy Builder	The three series can be referenced from a strategy; all inputs are <code>[NinjaScriptProperty]</code> .
Data requirements	Bars only. Volume is used when present (volume z-score in the TQI and the score); without volume the volatility ratio takes its place, exactly as in the original.
Rendering	Plots and draw objects through the standard NinjaTrader API; the dashboard and watermark through SharpDX in <code>OnRender</code> with DPI scaling and full resource cleanup on render-target change and termination.
Dark and light charts	<code>Theme = Auto</code> follows the chart background.
Templates and workspaces	All settings serialize (colours through the standard brush serializer).
Minimized chart	From 1.0.1: keeps calculating and alerting while its chart window is minimized (<code>IsSuspendedWhileInactive</code> off), and never alerts for a bar that closed more than two bar periods ago.
Order handling	None. The indicator never touches orders, positions or accounts.

09

Troubleshooting

SYMPTOM	CAUSE AND FIX
Nothing is drawn, the dashboard says <code>n bars</code> in the Preset row	Warm-up: 50 bars with the defaults, more with a longer ATR baseline, efficiency, RSI, volume or structure window. Load more days in the Data Series dialog.
No pills on a long chart	Signals need a Supertrend flip after warm-up; a strongly trending chart can go a long time without one. Check the <code>Trend</code> value in the Data Box.
Dashboard missing	08 Dashboard > Show Info Dashboard on? The dashboard is drawn in the price panel; with the indicator in its own panel it is not visible.
Dashboard text tiny or huge	Pick another Dashboard Size . Sizes are scaled by the Windows display scaling.
Level labels cut off at the right edge	They sit <code>Label Offset + 2</code> bars to the right of the current bar; add right margin (chart Properties > Right side margin) or lower Label Offset .

SYMPTOM	CAUSE AND FIX
Alerts do not fire	Alerts fire only for bars that close after the chart loaded, only in realtime, and only with Enable Alert Signals plus the per-kind toggle on. Email needs the Mail share service configured under Tools > Options > Share.
Sound not played	Sound File must name a file inside <NinjaTrader 8 install>\sounds\.
Values differ from another platform on the same symbol	Bars differ by feed and session template; compare on identical bars (see the chapter on how the port was verified).
An error line names Algo Shadow Matrix in the Log tab	Set 11 NinjaTrader > Debug Mode on, reproduce, and send Documents\NinjaTrader 8\asm_nt_debug.log with the chart interval, instrument and NinjaTrader version.
Two dashboards on one chart	Two instances are configured (Ctrl+I, Configured list). Remove one.

10

How the port was verified

The verification has two stages, which measure different things and are reported separately. It is never stated as "verified against TradingView on futures bars": TradingView's CME feed and NinjaTrader's feed do not carry the same bars (sessions, rollover, continuous-contract construction), so there is no such comparison to make.

Stage 1: engine vs the TradingView original (cTrader edition, 6 September 2026)

The platform-free engine that ships here is the same code as the cTrader edition's engine apart from its namespace line. That engine was replayed on 20,141 identical ICMARKETS:EURUSD M5 bars read from the TradingView chart model and compared with a probe copy of the published Pine script:

- 55 internal series equal on every bar (tolerance 1e-9 relative), maximum float difference 3.6e-14;
- 2,707 of 2,707 signals on the same bars, all trade levels, the whole ledger and the regime grid equal;
- residue: 2 sell-score labels out of 2,707 differ by up to 5 points because the Pine calls `ta.lowest/ta.highest` inside a function that only runs on flip bars (an undefined-behaviour construct TradingView's own compiler warns about); the `Score RSI Memory = TradingView` default reproduces TradingView's behaviour, `Strict20Bars` the intent.

Stage 2: NinjaTrader host vs the engine on identical NinjaTrader bars (16 September 2026)

With **Dump Bars + Outputs CSV** on, the indicator writes every committed bar (time, open, high, low, close, volume, source) together with the engine outputs as the host saw them. The same bars are replayed through a build of the identical engine outside NinjaTrader, with no host in between, and the two files are diffed value by value.

Result on MNQ SEP26, 1-minute bars, NinjaTrader 8.1.8.2, Simulation connection (delayed CME data):

bars compared	4,780
bar identity (time, open, high, low, close)	0 mismatching rows
series compared	19 (warm-up flag, Supertrend value and trend, TQI, ER, RSI, volume z, buy and sell scores, confirmed buy and sell, trade direction, entry, stop, TP1, TP2, TP3, ledger count, cumulative R)
cells compared	90,820
mismatches	0
max abs difference	0

Second run on the shipped build with `Calculate = OnEachTick`, the chart left live for twenty minutes so the dump spans the historical to realtime transition and bars committed through the per-tick path: 4,805 bars, 91,295 cells, 0 mismatches, 0 bar-identity mismatches.

What stage 2 proves: the host feeds the engine exactly one closed bar per chart bar with the right OHLCV and source, never double-feeds or skips a bar across the historical to realtime transition, and draws the values the engine produced. What it does not prove: anything about future performance.

11 Licensing

License model

The product is distributed as a compiled NinjaScript Add-On and licensed through NinjaTrader's user-based Vendor Licensing (NinjaTrader 8.1.5.0 or later). A licence is tied to the customer's NinjaTrader account login, not to a machine: the same login works on every installation of NinjaTrader Desktop the customer uses, and no Machine ID has to be sent to the vendor. Licences are created in the NinjaTrader Ecosystem Vendor Dashboard against the customer's NinjaTrader account email address, with a licence type and an expiry date.

The indicator carries the `VendorLicense(productId)` call in its constructor, with the Product ID of its add-on in the Vendor Dashboard. NinjaTrader verifies the licence whenever the indicator is used.

No free trial

The product is sold as a paid licence only; no free trial is offered. A licence is issued to the buyer's NinjaTrader account email after payment, normally within a few hours.

Activation and transfer

Nothing to activate: log in to NinjaTrader with the licensed account. A new PC needs no transfer. NinjaTrader allows two simultaneous market-data sessions per account, which is what prevents a licence from being shared.

Behavior without a valid licence

NinjaTrader shows its own licence notice (invalid or expired) and stops calling the indicator's calculation and plot methods: the indicator draws nothing and raises no alerts, while its settings still load. It never places orders (it has no order code) and exposes no proprietary data. A licensing-server outage is handled by NinjaTrader's verification (it retries against a secondary server) and must never crash the chart.

Licensing is a practical access-control layer, not a guarantee against copying.

12

Support

Questions go through the contact page at <https://kestutisbalciunas.com/contact/> with the NinjaTrader version, the instrument and the chart interval. For an error, switch **11 NinjaTrader > Debug Mode** on, reproduce it once and attach `asm_nt_debug.log` from the `Documents/NinjaTrader 8` folder; switch Debug Mode off afterwards.

The product page, with the documentation and the change log, is <https://kestutisbalciunas.com/ninjatrader/algo-shadow-matrix/>. Updates and refunds follow the terms of the NinjaTrader listing once it is live.

13

Risk disclosure

Trading involves substantial risk. Past, simulated, and backtested performance is not necessarily indicative of future results.

Futures, foreign currency and options trading contains substantial risk and is not for every investor. An investor could potentially lose all or more than the initial investment. Risk capital is money that can be lost without jeopardizing one's financial security or lifestyle. Only risk capital should be used for trading and only those with sufficient risk capital should consider trading.

Algo Shadow Matrix is a technical analysis tool. It draws levels and signals computed from historical bars; it does not place, modify or close orders and it gives no trade advice. The dashboard's win rate, average R and drawdown describe the signals inside the bars currently loaded on the chart and are not a forecast. Nothing in the software or its documentation is a promise of returns, win rates or funded-account success.

NinjaTrader is a registered trademark of NinjaTrader Group, LLC. No NinjaTrader company has any affiliation with the owner, developer, or provider of the products or services described herein, or any interest, ownership or otherwise, in any such product or service, or endorses, recommends or approves any such product or service.