**EMA Crypto Trading Bot (with RSI/MACD as “Advisors”)**

This bot watches multiple crypto markets on Coinbase in real time. It makes simple, explainable trades:

* **EMA** is the captain, the core leader. Gives the main signal: when a short-term moving average crosses a long-term one, that’s a buy/sell signal.
* **MACD** is the commodore. A strategist. Looks at momentum and trend confirmation; sits higher in the hierarchy than RSI.
* **RSI** is the skipper. Short term tactical and can veto bad entries like overbought/oversold.
* Orders are **post-only limit orders** by default (to earn maker rates), with a **daily spend cap** and a **cool-down** between trades per coin. It keeps a lightweight portfolio and P&L.

**Quick start (plain English)**

1. **Add your API keys** in an env file (default name APIkeys.env) or environment variables:
   * COINBASE\_API\_KEY, COINBASE\_API\_SECRET, and optional COINBASE\_PORTFOLIO\_ID.
2. **Run the bot**:
3. python main.py

It sets up logging, loads keys, starts the bot, and idles until you stop it. Ctrl-C triggers a clean shutdown.

**What each file does (in friendly terms)**

* **main.py** – The entry point. Loads your keys, wires up Ctrl-C handling, starts the bot, and keeps the event loop alive until you stop it.
* **bot/config.py** – All the knobs. Pick which coins to trade, EMA lengths, spend limits, cool-down, advisor (RSI/MACD) settings, and maker offsets. Includes per-coin overrides and an EMA “dead-band” to avoid flapping.
* **bot/tradebot.py** – The brain. Connects to Coinbase (REST + WebSocket), subscribes to Coinbase for chosen products, aggregates price data into candles (default 1m/5m), updates indicators per candle close, and decides when to trade, enforces daily cap/cool-down, places orders (maker by default), and updates portfolio/P&L from fills. Also logs a session P&L footer on exit.
* **bot/indicators.py** – Lightweight, streaming indicators:
  + **EMA** (short & long) for the captain signal,
  + **RSI(14)** by default,
  + **MACD(12,26,9)** with signal line & histogram.
* **bot/strategy.py** – Advisor rules. RSI must be within a safe range; MACD must not contradict the side (default: buy only if MACD≥0, sell only if MACD≤0). It can normalize MACD by price (basis points).
* **bot/orders.py** – Builds post-only limit orders near the current bid/ask with an offset, and rounds price/size to the exchange increments. Returns the price & size to place.
* **bot/persistence.py** – The little database:
  + **SpendTracker** (daily spend cap),
  + **LastTradeTracker** (per-coin cool-down),
  + **PortfolioStore** (positions, cost basis, realized P&L),
  + **ProcessedFills** (avoid double-counting),
  + Trade log writer (appends human-readable lines).
* **bot/constants.py** – Points the bot to a .state/ folder and names the JSON/text files it uses.
* **bot/logging\_setup.py** – Optional rotating file logs; sets formats and quiets noisy libraries.
* **bot/utils.py** – Thin re-export layer so tradebot.py can import the persistence & constants helpers from one place.

**How the bot decides to trade (simple view)**

1. **Real-time prices** stream in for every chosen product.
2. For each candle close, EMA/RSI/MACD update… waits for a minimum warm-up of candles first. When **short EMA crosses long EMA** (with a small “dead-band” buffer), that’s a new buy/sell event.
3. **Advisors can veto** if RSI is out of range or MACD disagrees with the side.
4. It obeys **per-coin cooldown** and a daily **BUY** cap. Once the cap is hit, it writes the session P&L footer and skips further BUYs (for the current **UTC** day) but continues to place SELLs until you stop it (Ctrl+C).
5. **Orders**: by default it places **post-only limit** buys/sells using your maker offset (bps) relative to bid/ask; market orders are optional.

**Key settings you’ll actually tweak**

All of these live in **bot/config.py** unless noted.

**What to trade & how much**

* product\_ids: the coins you want. More coins = more potential trades overall.
* usd\_per\_order: size of each order in USD.
* max\_usd\_per\_day: daily spend cap; hitting it will stop the bot for the day.
* dry\_run: simulate without sending live orders.

**Signal sensitivity (frequency dial)**

* short\_ema, long\_ema: shorter numbers react faster (more trades); longer numbers react slower (fewer). Per-coin overrides available in ema\_params\_per\_product.
* ema\_deadband\_bps: extra buffer around crossovers to avoid choppy flip-flops. Bigger dead-band = fewer signals.
* cooldown\_sec: minimum seconds between trades **per coin**. Higher = fewer trades.
* min\_candles (and per-coin overrides): warm-up period (must have enough candles before signals are valid).

**Advisor (RSI/MACD) guardrails**

* Toggle with enable\_advisors / use\_advisors. Turning them off increases trades.
* **RSI band**: Buys require RSI ≤ rsi\_buy\_max (default 70). Sells require RSI ≥ rsi\_sell\_min (default 30). Tighten by **lowering** rsi\_buy\_max and/or **raising** rsi\_sell\_min
* **MACD sign-check**: Buys require MACD ≥ macd\_buy\_min and sells require MACD ≤ macd\_sell\_max (set in config.py). Use 0.0/0.0 for pure sign-check; increase magnitude (e.g., +3.0/−3.0) to tighten confirmation. Values set in config.py override default values.

**Order style (fill behavior)**

* prefer\_maker: True = post-only limit orders.
* maker\_offset\_bps (and per-coin map): bigger offset = less aggressive pricing, **fewer fills**; smaller offset = more fills.

**Sell guard (risk)**

* max\_loss\_bps: don’t sell below cost basis beyond this tolerance (bps). Lower value = fewer loss-taking sells; higher value = more sells allowed when under water.

**Tuning cheatsheet**

**If you’re seeing *too many* trades per session:**

* **Raise cooldown\_sec** (e.g., 720 → 1200).
* **Increase ema\_deadband\_bps** (e.g., 8 → 12–20 bps).
* **Lengthen EMAs** (e.g., short\_ema +10–30%, long\_ema +10–30%). Use per-coin overrides if only some assets are noisy.
* **Narrow the RSI window** (raise rsi\_buy\_floor, lower rsi\_sell\_ceiling, e.g., 55–65).
* **Increase maker\_offset\_bps** so limits are less likely to fill in chop.
* **Lower max\_usd\_per\_day** to force the bot to stop sooner once busy.

**If you’re getting *hardly any* trades:**

* **Lower cooldown\_sec** (e.g., 720 → 300).
* **Shrink ema\_deadband\_bps** (e.g., 8 → 4–0 bps).
* **Shorten EMAs** (e.g., short\_ema −10–30%, long\_ema −10–30%).
* **Widen the RSI window** (lower rsi\_buy\_floor, raise rsi\_sell\_ceiling, e.g., 45–75), or **disable advisors** temporarily.
* **Decrease maker\_offset\_bps** for more fills, or set prefer\_maker=False (market orders).
* **Raise max\_usd\_per\_day** if you’re hitting the cap early.

**Advanced MACD loosen/tighten:** In code, the default requires **MACD ≥ 0 for buys** and **≤ 0 for sells**. To make it less strict (more trades), allow small negatives/positives by changing macd\_buy\_min and macd\_sell\_max when AdvisorSettings is created in tradebot.py. (Example: set them to −5/+5 bps. macd\_buy\_min=-5, macd\_sell\_max=+5)

**Where things get saved**

* **State folder**: .state/ (configurable via BOT\_STATE\_DIR).
* **Files**:  
  daily\_spend.json, last\_trades.json, trade\_log.txt, portfolio.json, processed\_fills.json.
* **Trade log lines** look like: timestamp, side, product, USD, price, qty (and “DRY RUN” if simulated).
* **Session P&L footer** is appended when the session ends.

**Order placement (maker vs market)**

* **Maker (default):** prices are placed slightly *away* from current bid/ask using your offset (in basis points). This can reduce fees but may not fill in fast markets.
* **Market:** switch by setting prefer\_maker=False in config.

**Notes & defaults (handy)**

* **RSI default period**: 14. **MACD**: 12/26 with 9-period signal.
* **Dead-band meaning**: a small % buffer (in **bps**, where 100 bps = 1%) around the crossover to cut down on noise.
* **Sell guard**: won’t sell at a loss beyond max\_loss\_bps below cost basis unless you relax that number.

**Parameter adjustments for preferred Trading Strategies (adjust at your own risk)**

**A) Quality-first** (fewer whipsaws, fewer trades)

* **Mode: Candles (preferred).**
  + **candle\_interval:** "1m" (or "5m" if you want even smoother signals)
* **EMA (trend gate):**
  + **1m: short\_ema** = 20, **long\_ema** = 60
  + **5m: short\_ema** = 40, **long\_ema** = 120
* **MACD veto (normalized, bps):**
  + **macd\_buy\_min** = +3.0
  + **macd\_sell\_max** = −3.0
* **RSI veto (one-sided):**
  + **rsi\_buy\_max** = 60 (don’t buy if RSI > 60)
  + **rsi\_sell\_min** =40 (don’t sell if RSI < 40)
* **When to use:** Choppy days, tight spreads, or if you’re seeing many small losers.

**B) Velocity-first** (more trades, more whipsaws)

* **Mode: Candles**
  + **candle\_interval:** "1m" (increase for quality trades)
* **EMA (trend gate):**
  + **1m: short\_ema** = 12–20, **long\_ema** = 36–60
* **MACD veto (normalized, bps):**
  + **Candles: macd\_buy\_min** = +1.0 to +2.0 / **macd\_sell\_max** = −1.0 to −2.0
* **RSI veto (one-sided):**
  + **rsi\_buy\_max** = 65–70
  + **rsi\_sell\_min** = 30–35
* **When to use:** Strong trends, wide ranges, or if the bot feels too “idle.”

**Parameter Reference (how each knob behaves)**

* **Candles vs Ticks (ticks no longer used in this version)**
  + Candles greatly reduce noise vs ticks. 1m is a good balance; 5m lowers noise further but reacts slower.
  + Ensure you have enough warm-up bars (≥ long\_ema) before trusting EMA/MACD.
* **EMA (trend gate)**
  + Buys are favored when short **EMA > long EMA**; sells when **short < long** (exact gate depends on your strategy wiring).
  + **Longer EMAs** = smoother, slower; **shorter EMAs** = faster, choppier.
* **MACD veto (normalized to price in basis points)**
  + **macd\_buy\_min:** BUY allowed only if MACD ≥ this value.
  + **macd\_sell\_max:** SELL allowed only if MACD ≤ this value.
  + **0 / 0 =** pure sign-check (fastest, choppier).
  + Increasing magnitude (e.g., +3/−3) demands more momentum → fewer but cleaner entries.
  + **Tip:** keep thresholds above your typical spread+slippage so you’re not churning.
* **RSI veto (one-sided)**
  + **rsi\_buy\_max:** cap for buys (don’t buy if overbought). Lower it (70→60) to be stricter.
  + **rsi\_sell\_min:** floor for sells (don’t sell if oversold). Raise it (30→40) to be stricter.
  + There is no “inside-band only” rule by default; it’s a ceiling for buys and a floor for sells.
* **Cooldown / Daily Cap / Runtime**
  + Cooldown: enforced per-coin to avoid rapid re-entries.
  + Daily BUY cap: once hit, the bot skips further BUYs for the rest of the UTC day, logs the P&L footer once, and continues to place SELLs until you stop it (Ctrl+C).
  + This lets you harvest exits without increasing exposure.

**Suggested Defaults:**

# config.py (starting point)

mode = "candles" # "candles". If you prefer to experiment with ticks, please use previous version.

candle\_interval = "1m" # "1m" | "5m" | etc.

# EMA

short\_ema = 20 # 1m: 20/60; 5m: 40/120

long\_ema = 60

# MACD (normalized, bps)

macd\_buy\_min = +2.0 # tighten to +3.0 if choppy

macd\_sell\_max = -2.0

# RSI (one-sided veto)

rsi\_period = 14

rsi\_buy\_max = 65 # tighten to 60 if many small losers

rsi\_sell\_min = 35 # tighten to 40 if premature exits

# Risk / Ops

daily\_spend\_cap\_usd = 100.0 # buys stop after cap; sells keep running

per\_product\_cooldown\_s = 300 # adjust per your pace