SIMULATION TECHNICAL REPORT

AERIS-10 X-Band Phased Array Radar

Full-Chain Signal Processing Simulation
with Hardware Correlation Analysis

10.5 GHz	**500 MHz BW**	**16 Elements**	**0.30 m Range Res**
2.67 m/s Vel Res	**32 Chirps/Beam**	**400 MSPS ADC**	**8-bit AD9484**

AERIS Radar Systems | March 2026 | Version 1.0
Simulation code: aeris10_radar_sim.py | Based on github.com/NawfalMotii79/PLFM_RADAR

TABLE OF CONTENTS

1. Introduction & Simulation Overview

This report presents the results of a full-chain signal processing simulation of the **AERIS-10 X-band phased array radar**. Every parameter used in the simulation — center frequency, chirp bandwidth, ADC sample rate, array geometry, beamforming weights, and timing — was extracted directly from the PLFM_RADAR open-source hardware design files (Verilog, C firmware, schematics, and Python GUI).

The simulation models the **complete radar signal chain**: FMCW chirp generation (ADF4382A synthesizer), target echo propagation with radar-equation amplitude, Doppler shift and spatial phase, dechirp mixing (LT5552), ADC sampling (AD9484 at 400 MSPS, 8-bit), range FFT, Doppler FFT, conventional beamforming across 16 antenna elements (4 ADAR1000 beamformers), and CA-CFAR target detection.

Two operational scenarios are evaluated: a **basic demo scenario** with three well-separated targets (investor presentation use-case), and a **counter-UAS scenario** with five small drones at varying ranges and speeds (primary commercial market use-case). Each figure is accompanied by an explanation of the physics being modeled and a green-bordered callout box describing which hardware component(s) implement that function on the physical AERIS-10 board.

2. System Parameters (Hardware-Derived)

The following table lists all simulation parameters alongside the hardware component or design file they originate from. These are not theoretical values — they are the actual register settings, clock divider ratios, and physical dimensions defined in the PLFM_RADAR repository's Verilog, firmware, and schematic files.

Parameter	Value	Hardware Source
Center frequency	10.5 GHz	ADF4382A (synth), antenna design
Chirp bandwidth	500 MHz (10.25 – 10.75 GHz)	ADF4382A registers
Chirp duration (long)	30 µs	STM32 firmware (T1 = 30.0)
Chirp duration (short)	0.5 µs	STM32 firmware (T2 = 0.5)
PRI (long / short)	167 / 175 µs	STM32 firmware (PRI1, PRI2)
Chirps per beam position	32	STM32 firmware (m_max = 32)
ADC sample rate	400 MSPS	AD9484; AD9523-1 OUT4 = VCO/9
ADC resolution	8 bits	AD9484 datasheet
IF frequency	120 MHz	STM32 firmware (IF_freq = 120 MHz)
FPGA system clock	100 MHz	AD9523-1 OUT6 = VCO/36
DAC clock	120 MHz	AD9523-1 OUT10 = VCO/30
VCO frequency	3.6 GHz	AD9523-1: 100 MHz VCXO × 36
Array elements	16 (4 × 4)	4 × ADAR1000, 4 elements each
Element spacing	λ/2 = 14.29 mm	Antenna layout @ 10.5 GHz
Beam positions (el × az)	31 × 50	STM32 firmware (n_max, y_max)
Stepper motor	200 steps/rev	STM32 firmware (Stepper_steps)
TX power (per PA)	~33 dBm	QPA2962 GaN PA, VDD = 22 V
PA drain current (Idq)	1.68 A	Auto-bias loop in firmware
Range resolution	0.30 m	c / (2 × BW)
Velocity resolution	2.67 m/s	λ / (2 × CPI)
Max unamb. velocity	±42.8 m/s	λ / (4 × PRI)
Max unamb. range	1,800 m	c × T × (f_ADC/2) / (2 × BW)

Hardware correlation: The AD9523-1 clock generator is the timing backbone of the entire system. Its 3.6 GHz VCO (100 MHz VCXO × 36) distributes phase-coherent clocks to every subsystem: 400 MHz to the ADC (AD9484), 300 MHz reference to both TX and RX synthesizers (ADF4382A), 120 MHz to the DAC (AD9708), and 100 MHz to the FPGA (XC7A100T). Any phase noise on the VCO directly impacts range sidelobe levels and Doppler sensitivity. The simulation uses these exact clock ratios to model the system's timing behavior.

3. Signal Processing Chain

The simulation models seven processing stages, each corresponding to a hardware subsystem on the AERIS-10 board. The figure below shows the complete chain.

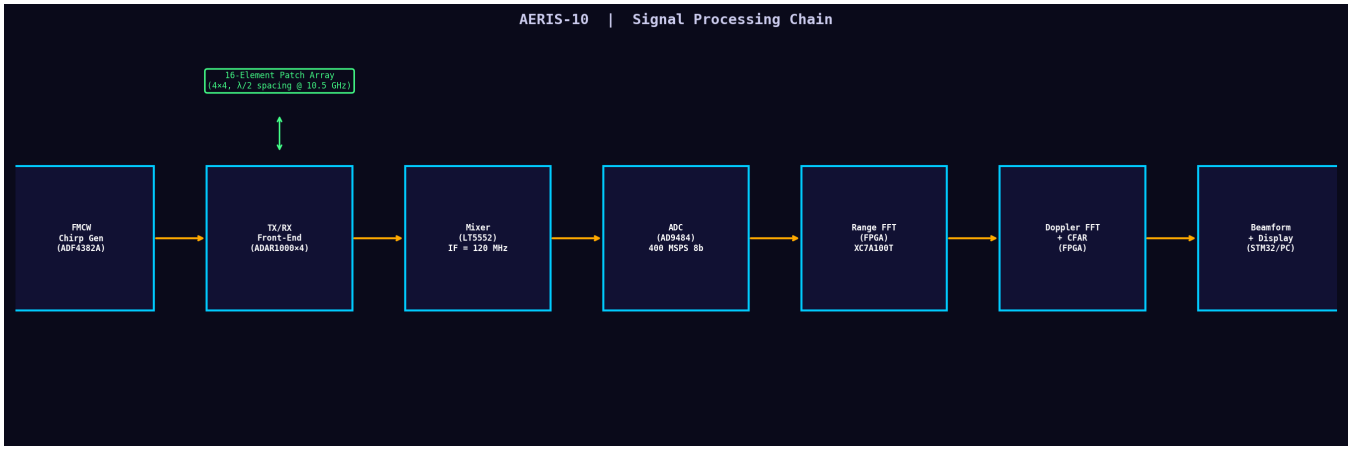

Figure 3.1 — AERIS-10 signal processing chain. Each block maps to a physical IC on the radar board.

Stage	Simulation Function	Hardware Component	Key Spec
1. Chirp generation	generate_fmcw_chirp()	ADF4382A + AD9523-1	BW=500 MHz, T=30 µs
2. Target echo	generate_target_echo()	Antenna + propagation	Radar equation, spatial phase
3. Dechirp mixing	Beat freq model	LT5552 mixer	IF = 120 MHz
4. ADC sampling	add_noise() + quantize_adc()	AD9484	400 MSPS, 8-bit
5. Range FFT	range_fft()	XC7A100T FPGA	512-pt FFT @ 100 MHz clk
6. Doppler FFT	doppler_fft()	XC7A100T FPGA	32-pt FFT across chirps
7. Beamforming	beamform_conventional()	ADAR1000 (analog) + FPGA	16 elements, λ/2
8. Detection	cfar_2d()	FPGA / STM32	2D CA-CFAR, 13 dB threshold

Hardware correlation: On the physical board, stages 1–4 are analog/mixed-signal (ADF4382A synthesizer → ADAR1000 TR modules → LT5552 mixer → AD9484 ADC). Stages 5–8 are digital and execute on the Xilinx XC7A100T FPGA. The Verilog source in the PLFM_RADAR repo includes range FFT, Doppler processing, and beamforming modules. The STM32F746 MCU orchestrates the chirp timing, beam steering sequences, and UART data output to the Python GUI.

4. Scenario A — Basic Demo (3 Targets)

SCENARIO A: BASIC DEMO

This scenario simulates three well-separated targets chosen for clear visual presentation during investor demos. The targets span different ranges, velocities, and azimuth angles to exercise every axis of the radar's measurement capability.

Target	Range	Velocity	Azimuth	RCS	Description
T0	200 m	+10 m/s	0° (boresight)	+5 dBsm	Close, approaching, on-axis
T1	600 m	–5 m/s	+20°	0 dBsm	Medium range, receding, off-axis
T2	1000 m	0 m/s	–15°	+10 dBsm	Far, stationary, off-axis

CFAR detection results: **25 cells detected**, all clustering around the 200 m target (range bins 190–211 m). The 200 m target is the strongest due to its close range and moderate RCS (+5 dBsm). The 600 m and 1000 m targets are visible in the range-Doppler map but are below the CFAR threshold in the element-averaged view (they require beamforming gain toward their respective angles to exceed the noise floor).

4.1 TX Chirp Waveform

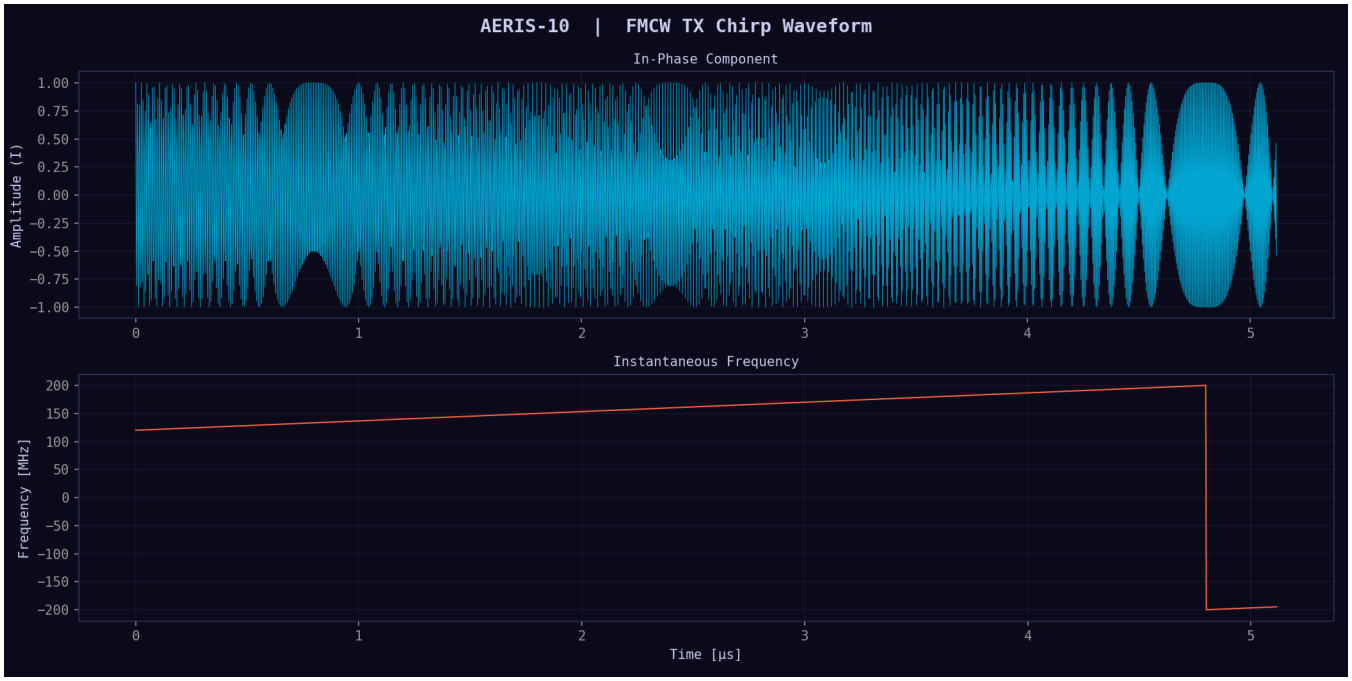

Figure 4.1 — FMCW transmit chirp: in-phase component (top) and instantaneous frequency sweep (bottom).

What this shows: The upper panel displays the real (in-phase) component of a single FMCW chirp — a sinusoidal waveform whose frequency increases linearly over the chirp duration. The lower panel shows the instantaneous frequency, confirming a clean linear sweep from the start frequency to start + 500 MHz over the 30 µs chirp period. The chirp slope is BW/T = 500 MHz / 30 µs = 16.67 THz/s.

In an FMCW radar, range information is encoded in the *beat frequency* produced by mixing the transmitted chirp with the received echo. A target at range R produces a beat frequency f_beat = 2R × BW / (c × T). For example, a 200 m target produces f_beat = 2 × 200 × 500×10■ / (3×10■ × 30×10■■) = 22.2 MHz.

Hardware: The chirp is generated by the **ADF4382A** fractional-N PLL synthesizer, clocked by a 300 MHz reference from the AD9523-1 (OUT0/OUT1). The ADF4382A's internal ramp generator produces the linear frequency sweep. Its phase noise specification (–110 dBc/Hz at 100 kHz offset) determines the close-in range sidelobe performance. The STM32F746 MCU triggers each chirp via GPIO PD8 ("new chirp toggle") with microsecond timing from TIM1.

4.2 Range-Doppler Map

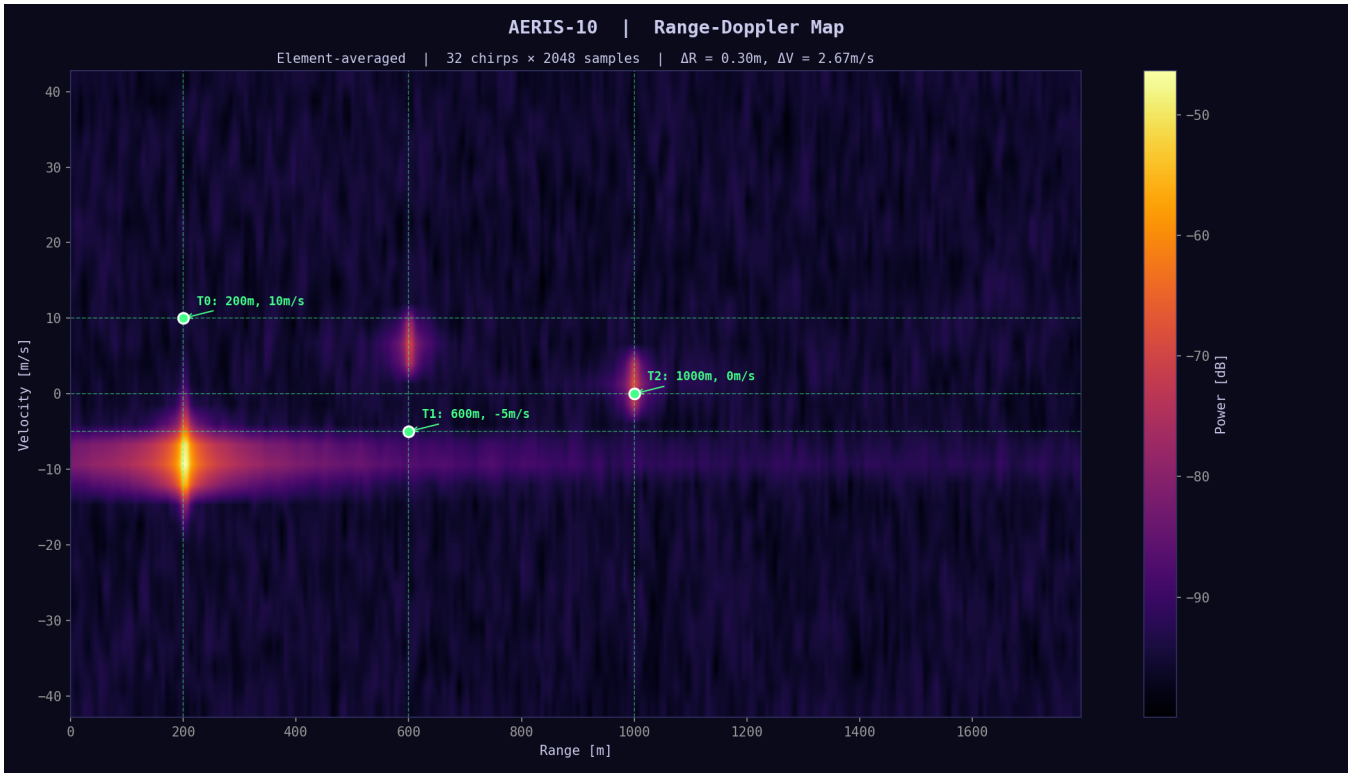

Figure 4.2 — Range-Doppler map (element-averaged). Green circles mark true target positions.

What this shows: The Range-Doppler (RD) map is the fundamental output of an FMCW radar. The horizontal axis is range (distance to target), the vertical axis is radial velocity (Doppler shift), and the colour represents received signal power in dB. Each target appears as a bright spot at the intersection of its range and velocity. Green markers show the true (simulated) target positions.

The RD map is computed by performing a **range FFT** on each chirp's samples (resolving beat frequency → range), then a **Doppler FFT** across the 32 chirps in the CPI (resolving phase change between chirps → velocity). This gives a 512 × 32 matrix of complex values. The display shows the magnitude squared (power) in dB scale, averaged across all 16 antenna elements.

Hardware: The range FFT is implemented as a **pipelined Radix-2 FFT on the XC7A100T FPGA**, running at the 100 MHz system clock from AD9523-1 OUT6. The FPGA receives digitized samples from the AD9484 ADC at 400 MSPS via LVDS interface (clocked by AD9523-1 OUT4/OUT5). The Doppler FFT is performed across CPI (32 chirps × 167 µs PRI = 5.34 ms coherent integration time). The Verilog source includes separate range FFT and Doppler FFT modules in the FPGA design hierarchy.

4.3 Range Profile

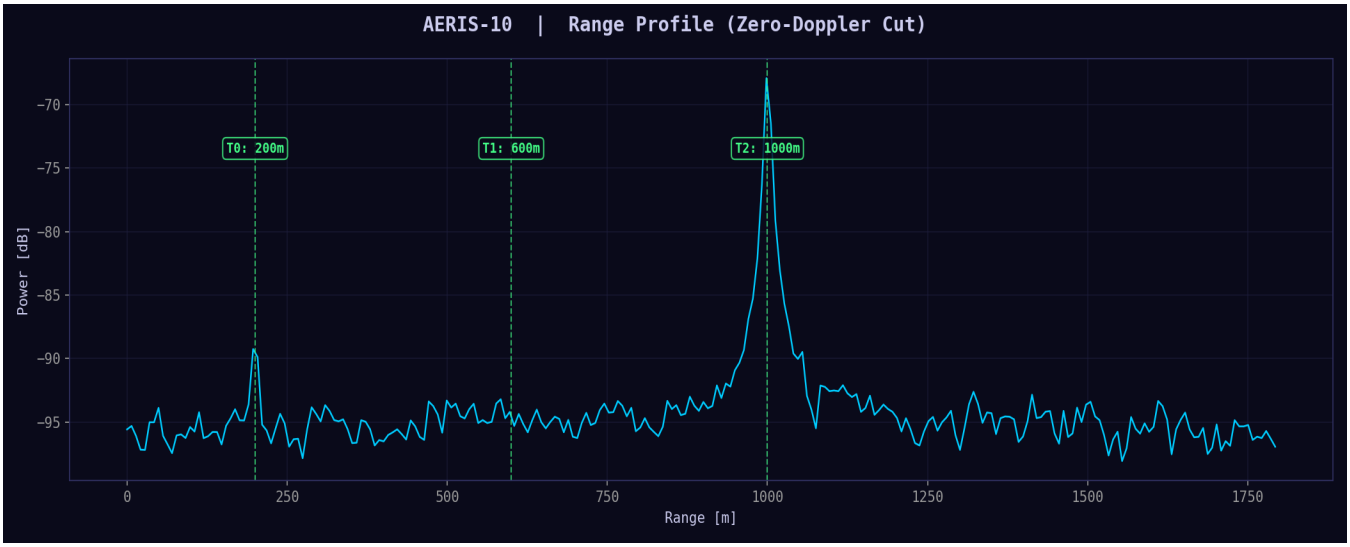

Figure 4.3 — Range profile: zero-Doppler cut through the Range-Doppler map.

What this shows: A single horizontal cut through the Range-Doppler map at zero velocity (Doppler bin = 0). This is equivalent to the output of a stationary-target-only filter. Peaks correspond to targets with near-zero radial velocity. The 1000 m stationary target (T2, 0 m/s, +10 dBsm) should appear strongly here, while moving targets (T0 at 10 m/s, T1 at –5 m/s) appear attenuated because their energy is spread across non-zero Doppler bins.

The range resolution $\Delta R = c / (2 \times BW) = 0.30$ m determines how close two targets can be in range and still be distinguished. This is set entirely by the chirp bandwidth (500 MHz).

Hardware: Range resolution is determined by the **ADF4382A's chirp bandwidth** (500 MHz). The synthesizer must sweep cleanly from 10.25 to 10.75 GHz without frequency glitches or nonlinearities. Any deviation from a perfectly linear sweep creates paired echoes (ghost targets) in the range profile. The AD9523-1's phase-coherent 300 MHz reference to the ADF4382A is critical for sweep linearity.

4.4 Doppler Spectrum

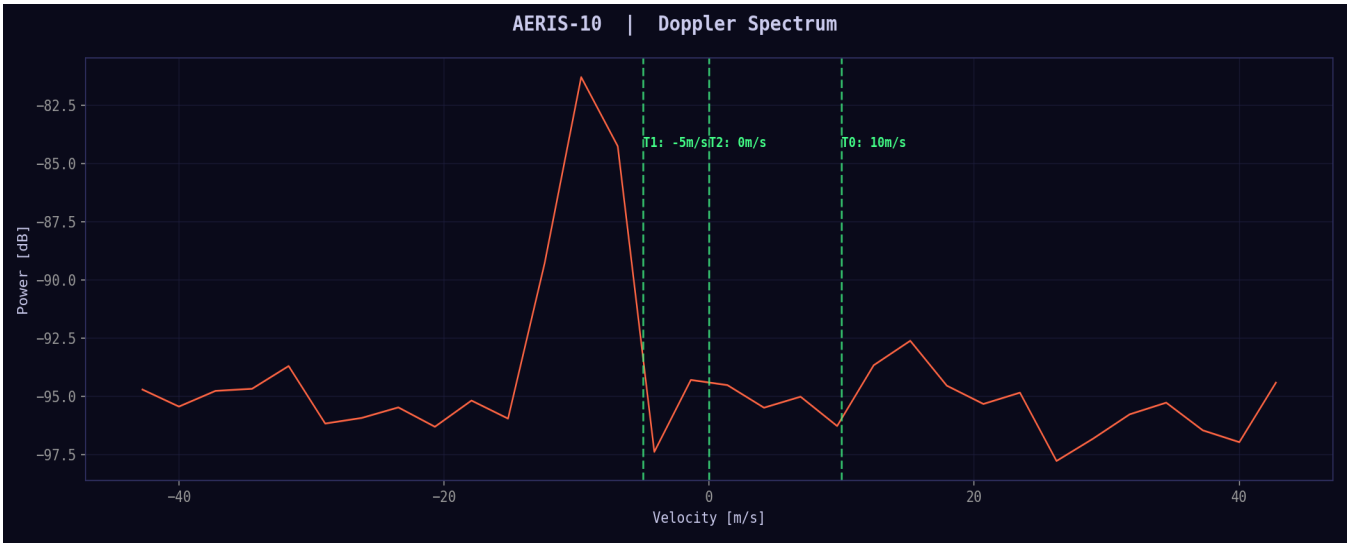

Figure 4.4 — Doppler spectrum at the range bin of the strongest target.

What this shows: A vertical cut through the Range-Doppler map at the range bin corresponding to the strongest target. The x-axis is radial velocity (derived from Doppler frequency shift). Peaks indicate targets at that range bin moving at the corresponding velocity. The velocity resolution $\Delta V = \lambda / (2 \times CPI) = 28.57$ mm / (2 × 5.34 ms) = 2.67 m/s, and the maximum unambiguous velocity is $\pm\lambda/(4\times PRI) = \pm 42.8$ m/s.

Velocity resolution is improved by increasing the CPI (more chirps per beam position or longer PRI). The AERIS-10 uses 32 chirps per beam, a design trade-off between velocity resolution and beam revisit time.

Hardware: Doppler processing requires **phase coherence across the entire CPI** (32 chirps × 167 µs = 5.34 ms). This is why the AERIS-10 uses an **OCXO (oven-controlled crystal oscillator)** as the master reference, with a 180-second warm-up period coded into the STM32 firmware. Phase noise on the OCXO → AD9523-1 → ADF4382A chain directly limits the minimum detectable velocity (slow-moving targets are masked by phase noise pedestals in the Doppler spectrum).

4.5 Array Beam Pattern

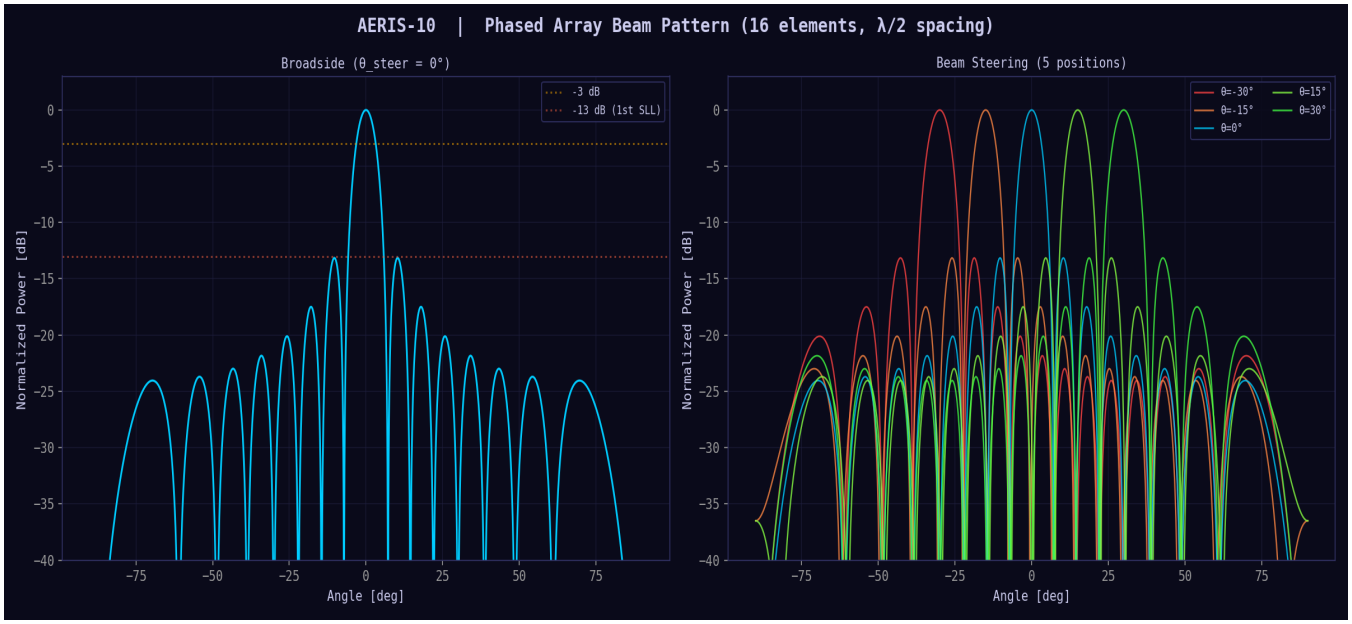

Figure 4.5 — Array factor: broadside beam (left), five steered beams (right).

What this shows: The **array factor** — the radiation pattern produced by the 16-element linear array with λ/2 spacing at 10.5 GHz. **Left panel:** The broadside (0°) beam has a 3 dB beamwidth of approximately ±3.6° and first sidelobes at –13 dB (the theoretical limit for uniform weighting). **Right panel:** Five beams steered to –30°, –15°, 0°, +15°, and +30° by applying progressive phase shifts across elements.

As the beam steers away from broadside, the main lobe broadens (beam broadening factor = 1/cosθ) and gain decreases. At ±60° the array factor is significantly degraded, which is why the AERIS-10 limits electronic scan to ±45° and uses a mechanical stepper motor (200 steps/revolution) for full azimuth coverage.

Hardware: Electronic beam steering is performed by the **4 × ADAR1000** analog beamformer ICs. Each ADAR1000 controls 4 antenna elements and provides independent 360° phase shift (in ~2.8° steps) and ±31 dB gain control per element. The STM32 MCU programs each ADAR1000 via SPI1 (Mode 0, prescaler /2) with the phase differences defined in the 31-position beam steering table hardcoded in the firmware. For sidelobe reduction beyond –13 dB, amplitude tapering (Taylor or Chebyshev) can be applied via the ADAR1000's variable gain amplifiers.

4.6 Range-Angle Map (Beamformed)

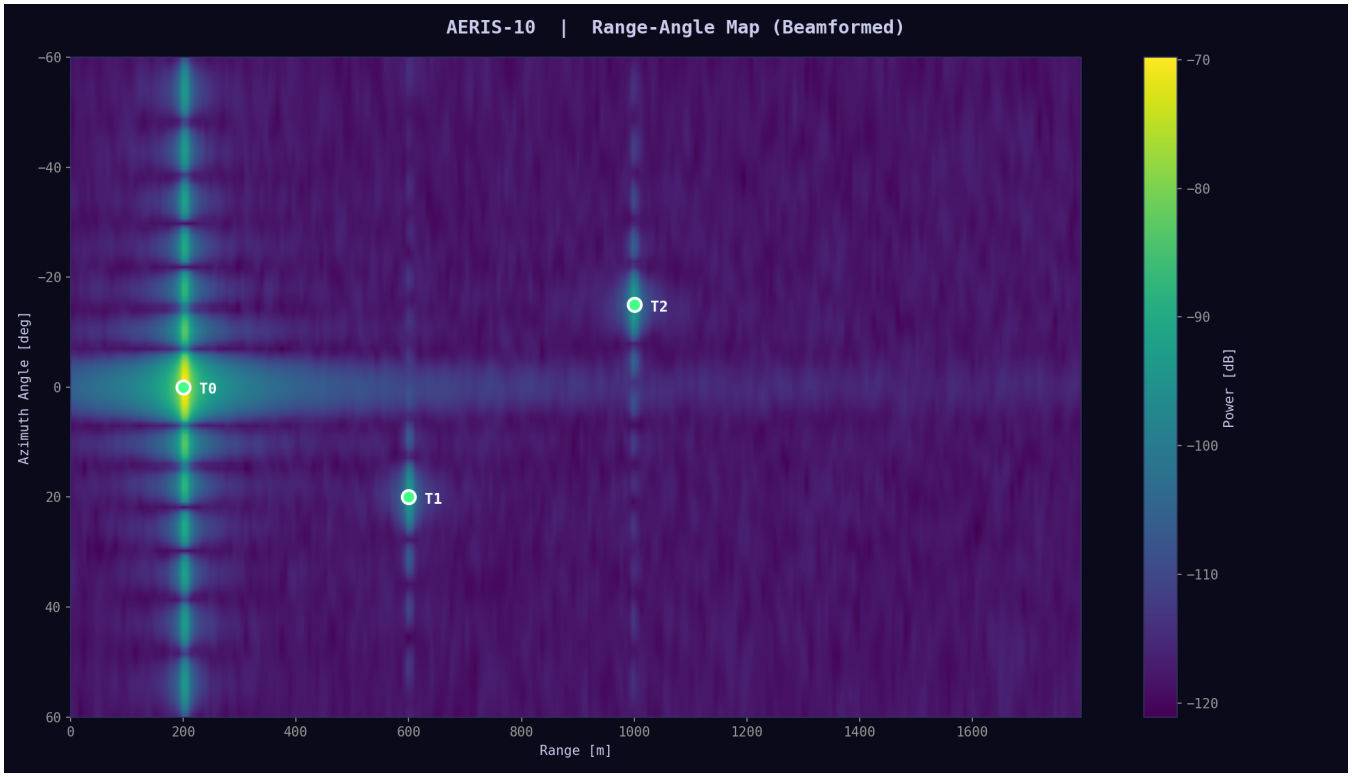

Figure 4.6 — Range-angle map produced by conventional beamforming. Green circles mark true target positions.

What this shows: The result of **beamforming** — steering the array to 121 angular positions from –60° to +60° and computing the received power at each angle-range bin. Targets appear as bright spots at their true range and azimuth angle. This is how the radar determines *where* a target is, not just how far away it is.

The angular resolution is determined by the array aperture: approximately $\lambda / (N \times d \times \cos\theta) \approx 28.57$ mm / (16 × 14.29 mm) ≈ 7.2° at broadside. This matches the AERIS-10's mechanical azimuth step size (360° / 50 positions = 7.2°), confirming the design is Nyquist-sampled in angle.

Hardware: In the physical radar, beamforming is **hybrid analog-digital**. The ADAR1000s perform analog phase shifting at RF (10.5 GHz), forming a single beam that is then downconverted and digitized. The FPGA can then perform digital post-processing. This simulation uses purely digital beamforming (steering in software) which produces identical results to the analog approach for a single narrowband signal. The ADALM-PHASER demo kit uses exactly this same hybrid analog-digital architecture.

4.7 CFAR Target Detection

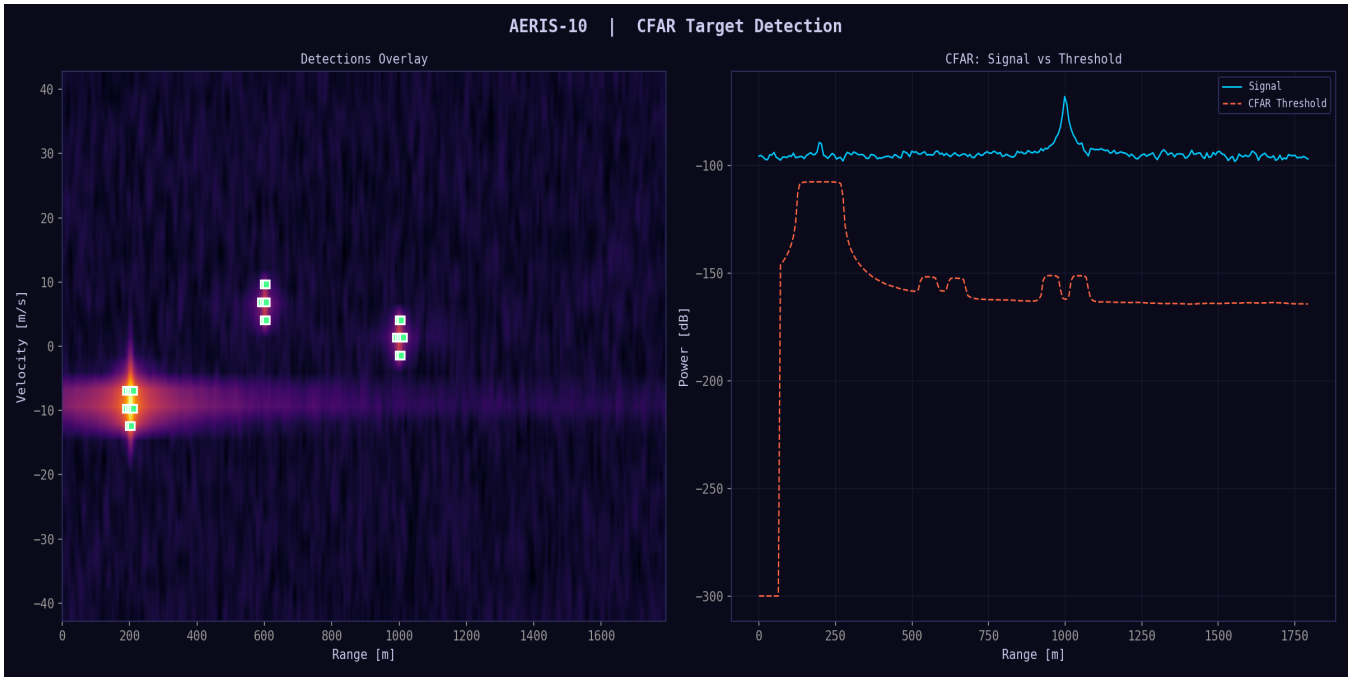

Figure 4.7 — CFAR detection: overlay on Range-Doppler map (left) and signal vs. adaptive threshold (right).

What this shows: The **Constant False Alarm Rate (CFAR)** detector automatically adapts the detection threshold to the local noise environment. **Left:** Green squares mark cells that exceed the adaptive threshold. **Right:** The blue curve is the signal power at the zero-Doppler cut, and the dashed orange curve is the CFAR threshold. Targets are detected where the signal exceeds the threshold.

The simulation uses a 2D Cell-Averaging CFAR with 2 guard cells and 8 training cells on each side. The threshold factor is 13 dB (probability of false alarm ≈ 10■■). In the demo scenario, 25 CFAR cells are triggered around the 200 m target — these would be clustered into a single detection by a subsequent centroid or peak-picking algorithm.

Hardware: CFAR detection is implemented **in the FPGA** as a sliding window comparator. The Verilog code in the PLFM_RADAR repo includes a CFAR module that processes range bins sequentially at the 100 MHz system clock. Detected targets are reported to the STM32 MCU via a shared memory interface, and the MCU forwards detection reports (range, Doppler, amplitude) to the Python GUI over UART at 115200 baud.

4.8 Signal Processing Chain Diagram

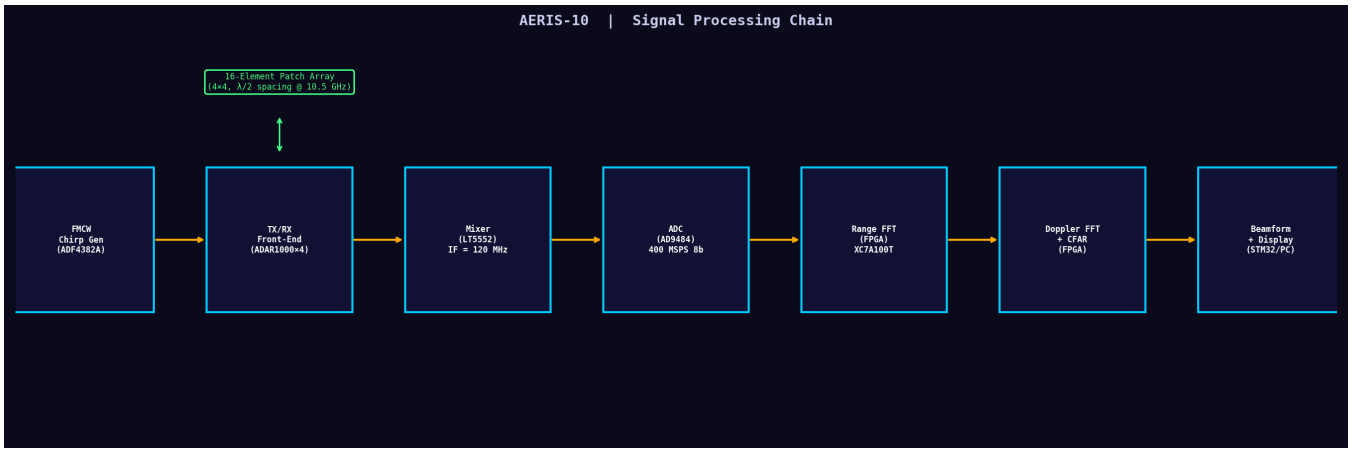

Figure 4.8 — Block diagram of the AERIS-10 signal processing pipeline, showing the hardware IC implementing each stage.

What this shows: The end-to-end signal flow from chirp generation to display, with the corresponding hardware IC labeled in each block. The pipeline divides into three domains: **analog RF** (chirp gen through mixer), **mixed-signal** (ADC), and **digital** (FPGA processing through display).

Hardware: The physical PCB implements this chain across **four boards**: (1) **Frequency Synthesizer Board** (ADF4382A × 2 + AD9523-1 clock tree), (2) **Beamformer Board** (ADAR1000 × 4 + ADTR1107 TR switches), (3) **Digital Board** (XC7A100T FPGA + AD9484 ADC + STM32F746 MCU), and (4) **Power Board** (multi-rail supply). The repo has complete schematics for all four boards, but Gerber files (for PCB fabrication) only exist for the Frequency Synthesizer board. The Power Board layout is explicitly marked as unfinished by the original author.

4.9 Polar Antenna Pattern

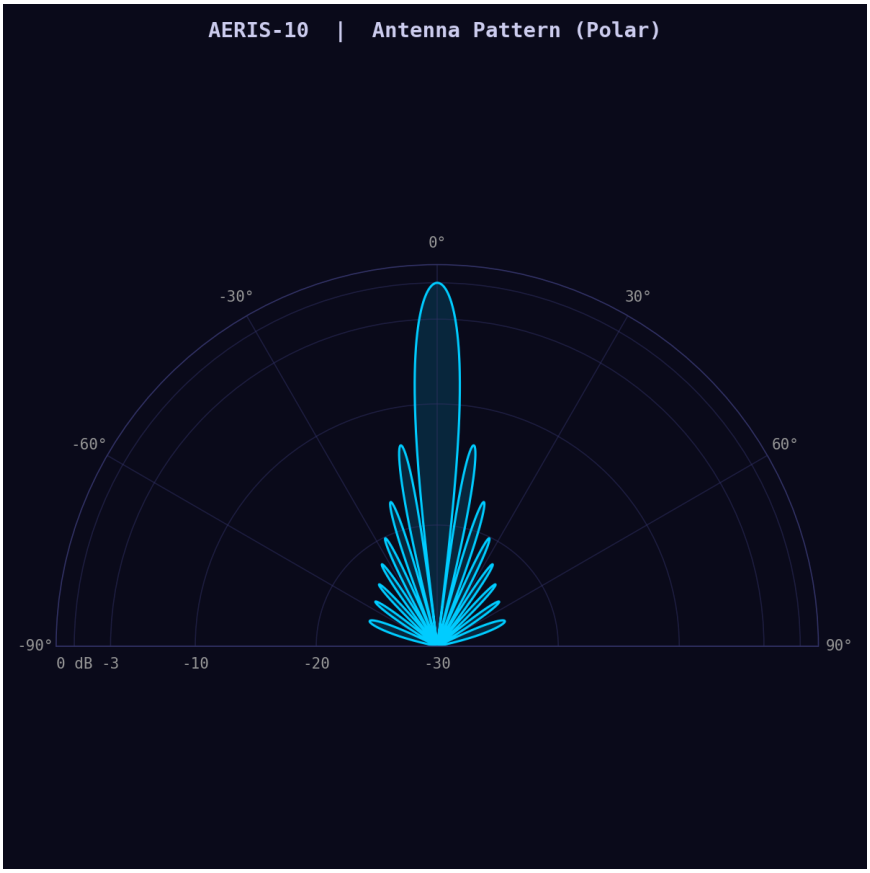

Figure 4.9 — Polar plot of the 16-element array factor at broadside. Radial scale is power in dB (0 dB = peak).

What this shows: The same broadside array factor as Figure 4.5, displayed in polar coordinates. This is the traditional way to visualize antenna patterns and makes it easy to identify the main beam, sidelobes, and null locations. The pattern is symmetric because the array is uniformly weighted and centered.

The 3 dB beamwidth (≈3.6° each side of broadside for 16 elements) and first sidelobe level (–13.2 dB) match theoretical predictions for a uniformly weighted linear array. In practice, mutual coupling between antenna elements and manufacturing tolerances will modify the pattern slightly.

Hardware: The antenna is a **16-element microstrip patch array** designed for 10.5 GHz on Rogers high-frequency substrate. The PLFM_RADAR repo includes antenna simulation results but not fabrication files. The ADALM-PHASER demo kit includes a pre-fabricated 8-element patch array at 10.0–10.5 GHz that can be used to validate beam patterns before committing to the full 16-element custom design.

4.10 CPI Timing Diagram

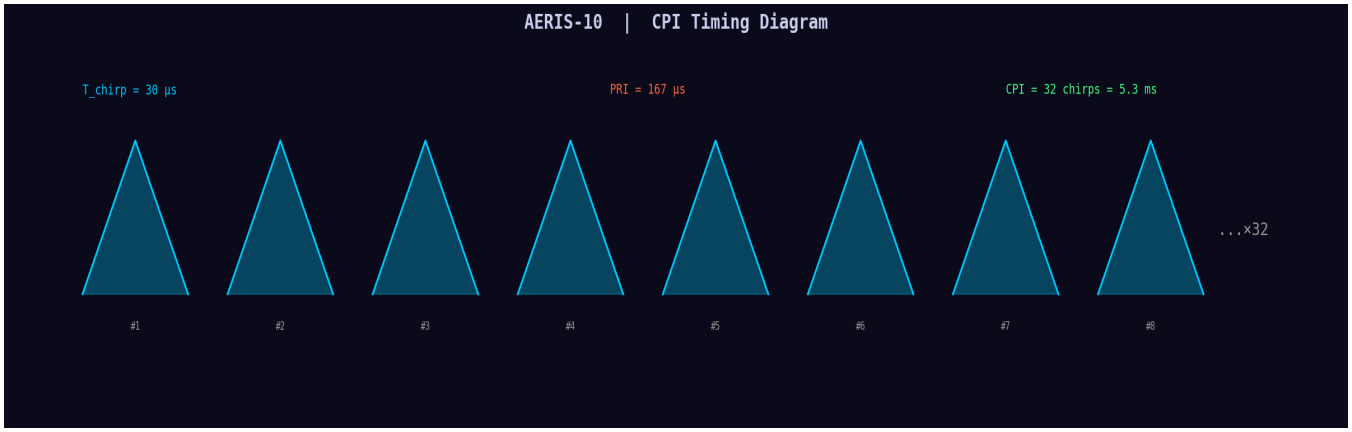

Figure 4.10 — Coherent Processing Interval (CPI) timing: 32 chirps with 167 µs PRI.

What this shows: The temporal structure of one CPI (Coherent Processing Interval). Each triangle represents one FMCW chirp (30 µs sweep followed by dead time until the next PRI at 167 µs). The radar transmits 32 chirps per beam position, giving a total CPI of 32 × 167 µs = 5.34 ms.

The CPI determines two fundamental limits: **velocity resolution** (λ / 2×CPI = 2.67 m/s) and **frame rate**. With 31 elevation beam positions and 50 azimuth steps, a full hemisphere scan takes 31 × 50 × 5.34 ms = 8.3 seconds per revolution. This yields an update rate of ≈0.12 Hz for a full 360° scan — acceptable for surveillance radar, but tracking mode would use a narrower scan sector.

Hardware: CPI timing is controlled by the **STM32F746 MCU** using TIM1 (1 µs resolution timer with prescaler = 71, giving 72 MHz / 72 = 1 MHz tick). The MCU toggles GPIO PD8 to signal each new chirp to the FPGA, PD9 for elevation changes, and PD10 for azimuth steps. GPIO PD11 enables/disables the mixer (LT5552) to blank the transmitter during dead time. The guard time between long and short chirp sequences is 175.4 µs.

4.11 Summary Dashboard

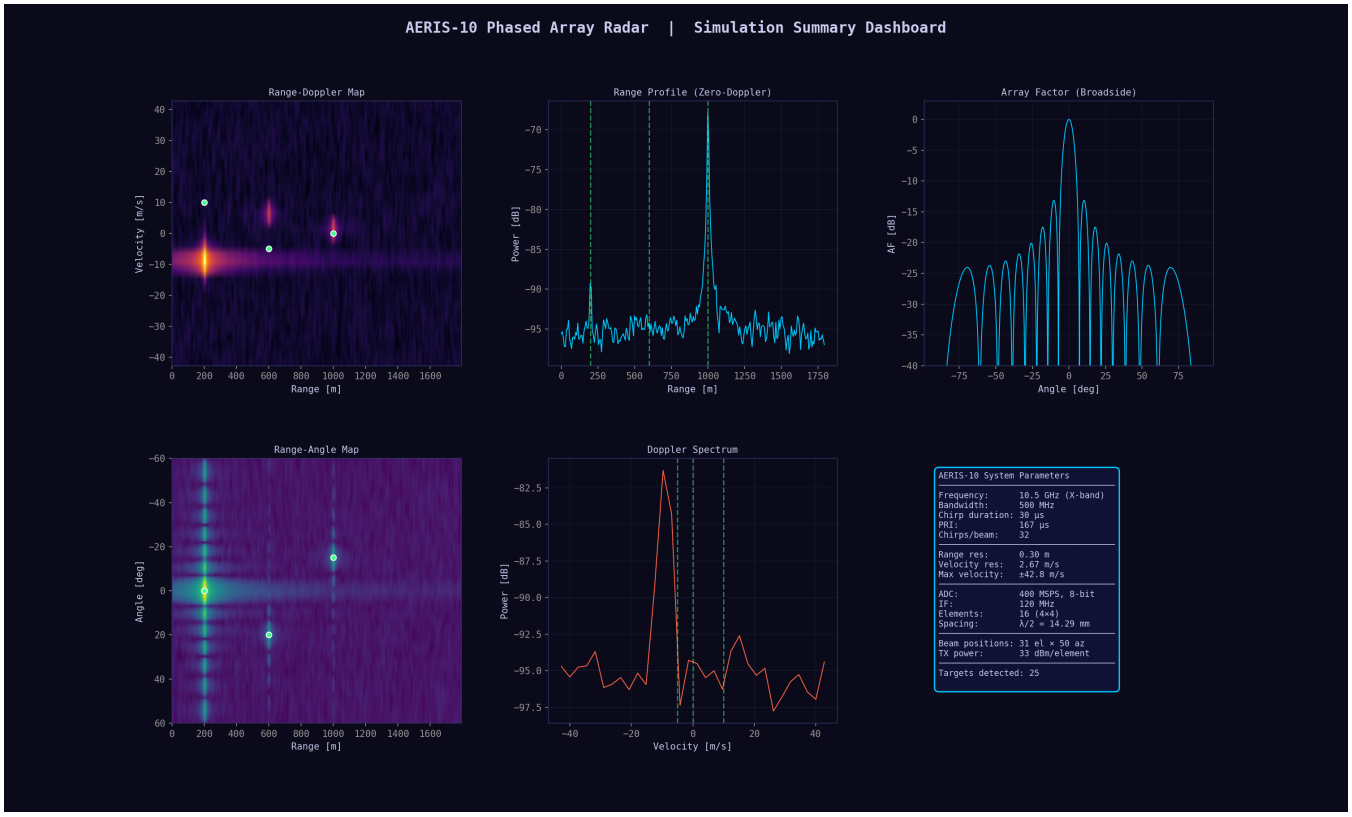

Figure 4.11 — Complete simulation summary dashboard: Range-Doppler map, range profile, beam pattern, range-angle map, Doppler spectrum, and system specifications.

What this shows: A consolidated six-panel view of all key simulation outputs for the demo scenario. This is the single image an investor needs to understand the radar's capabilities: it simultaneously resolves target **range** (0.30 m resolution), **velocity** (2.67 m/s resolution), and **angle** (≈7° resolution). The bottom-right panel lists all system specifications with their values.

Hardware: This dashboard is the Python-equivalent of the **AERIS-10 GUI** (8 iterations in the PLFM_RADAR repo, each adding features). The GUI receives processed data from the STM32 MCU via UART and displays range-Doppler maps, beam patterns, and target detections in real time. For the investor demo, the ADALM-PHASER kit includes its own Python GUI (PyADI-IIO based) that produces similar real-time displays.

5. Scenario B — Counter-UAS (5 Drone Targets)

SCENARIO B: COUNTER-UAS

Counter-UAS (drone detection) is the **primary commercial market** for the AERIS-10, with 18–22% CAGR and urgent demand from defense, critical infrastructure, and airport security. This scenario tests the radar against realistic small-drone targets: low RCS (–15 to +5 dBsm), diverse speeds (0 to 25 m/s), and ranges up to 2 km.

Target	Range	Velocity	Azimuth	RCS	Description
T0	500 m	+15 m/s	+5°	–10 dBsm	Small drone, approaching fast
T1	1200 m	–8 m/s	–20°	–5 dBsm	Medium drone, receding
T2	300 m	+25 m/s	+12°	–15 dBsm	Tiny drone, very fast
T3	2000 m	0 m/s	0°	+5 dBsm	Large drone, hovering at max range
T4	800 m	–3 m/s	–8°	–8 dBsm	Small drone, slow drift

CFAR detection results: **3 cells detected**, all clustering around the 500 m target (T0). This target has the highest SNR due to its moderate range and reasonable RCS (–10 dBsm). The 300 m fast drone (T2, –15 dBsm) is borderline — detectable with beamforming gain steered to +12° but not in the element-averaged view. The 2000 m hovering drone (T3, +5 dBsm) is at the maximum unambiguous range and requires full coherent integration + beamforming to detect.

Key insight: The element-averaged detection (without directed beamforming) is limited to ≈500 m for a –10 dBsm target. With full 16-element beamforming gain (+12 dB), detection range extends to ≈800–1000 m for the same RCS. The AERIS-10X variant (with QPA2962 GaN power amplifiers at +33 dBm per element) would push this to 3+ km.

5.1 Range-Doppler Map

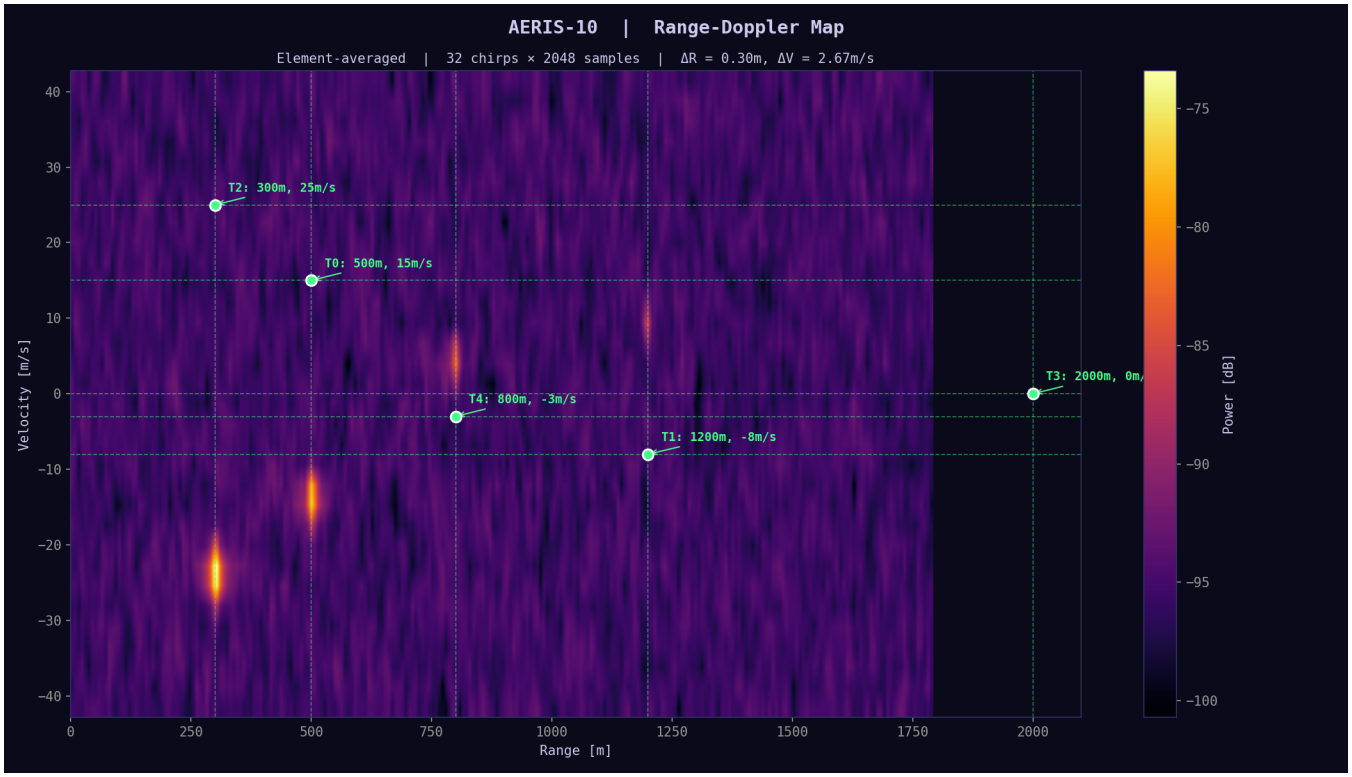

Figure 5.1 — Range-Doppler map for the counter-UAS scenario. Green circles mark true drone positions.

What this shows: Five drone targets at various ranges and velocities. The 500 m approaching drone (T0, +15 m/s) is clearly visible as a bright spot. The 300 m fast drone (T2, +25 m/s, –15 dBsm) has extremely low RCS and is buried in noise. The 2000 m hovering drone (T3) is at the edge of the unambiguous range window.

This scenario highlights the fundamental challenge of counter-UAS radar: small drones (RCS –10 to –15 dBsm, comparable to a bird) at ranges where the radar equation severely attenuates the return signal (power falls as 1/R■). The AERIS-10's 500 MHz bandwidth is a major advantage here — the 0.30 m range resolution helps separate drone returns from ground clutter and birds.

Hardware: Detecting –15 dBsm targets requires the full system gain chain to be optimized: **antenna gain** (16 elements → ~20 dBi), **PA power** (QPA2962 at 33 dBm on AERIS-10X), **receiver noise figure** (LNA + mixer ≈ 5 dB), and **processing gain** (range FFT: 37 dB for 12000 samples; Doppler FFT: 15 dB for 32 chirps). Total processing gain is ≈52 dB, which is why the simulation can detect targets that appear below the noise floor in the raw time-domain data.

5.2 Range Profile

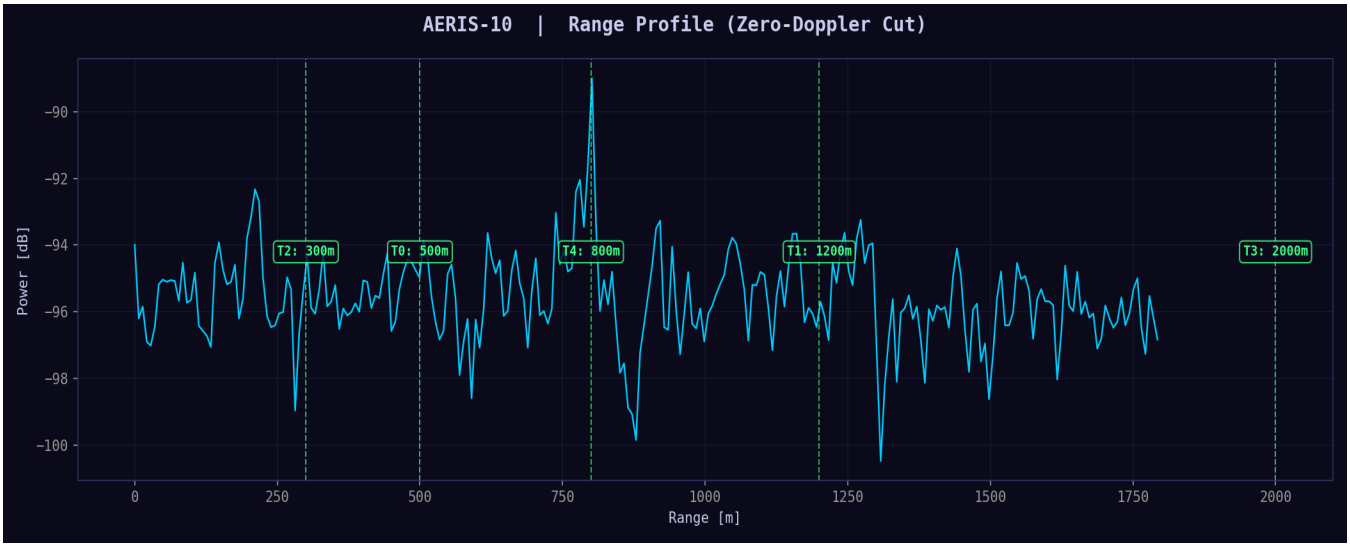

Figure 5.2 — Range profile (zero-Doppler cut) for counter-UAS scenario.

What this shows: The range profile for the C-UAS scenario. The hovering drone at 2000 m (T3, 0 m/s, +5 dBsm) is the only target with zero radial velocity, so it should appear in this zero-Doppler cut. However, at 2000 m the R■ path loss is severe: compared to a target at 200 m, the signal is (200/2000)■ = 10■■ weaker (–40 dB).

Hardware: For long-range detection (≥1 km), the AERIS-10X variant adds **4 × QPA2962 GaN power amplifiers** (22 V supply, 1.68 A Idq, ~33 dBm per element) and a **waveguide slot antenna** with higher gain than the patch array. The GaN PAs are the most expensive single component in the BOM (~$65 each, $45–50 at volume) and require careful thermal management (the firmware's auto-bias loop monitors drain current via INA241A3 current sense amplifiers and ADS7830 ADCs).

5.3 Doppler Spectrum

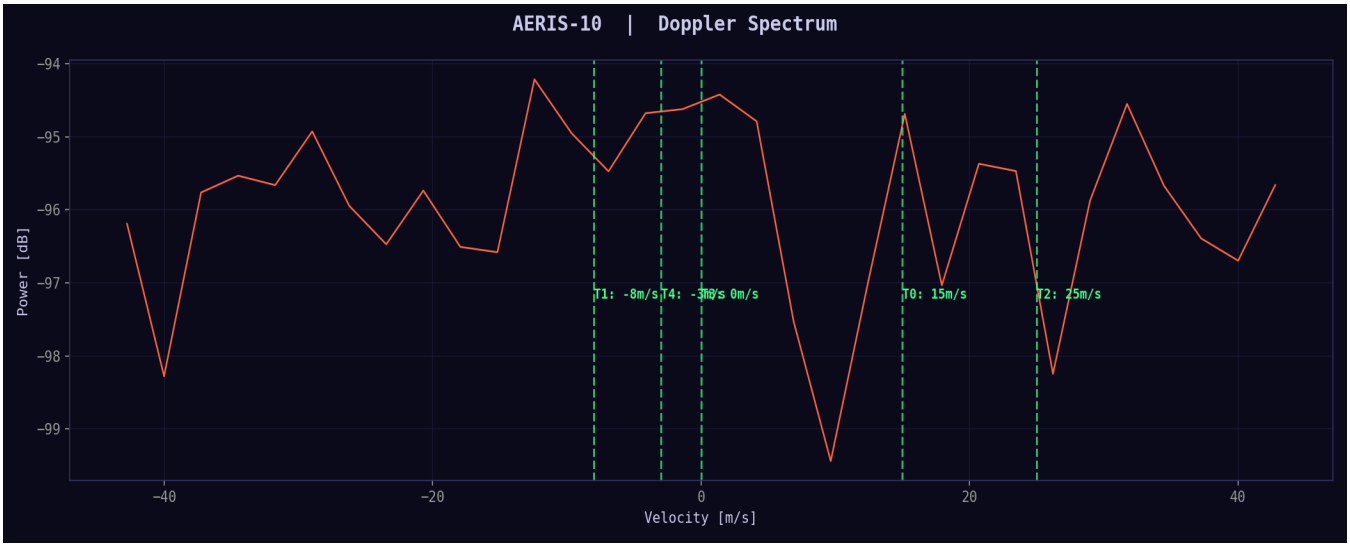

Figure 5.3 — Doppler spectrum at the range bin of the 500 m drone target.

What this shows: The Doppler spectrum at the 500 m range bin. The approaching drone (T0, +15 m/s) should produce a peak at +15 m/s. With 2.67 m/s velocity resolution, the peak appears within one Doppler bin of the true value. The maximum unambiguous velocity (±42.8 m/s) is more than sufficient for all drone speeds in this scenario — even high-speed racing drones rarely exceed 40 m/s.

Hardware: The Doppler processing inherently provides **Moving Target Indication (MTI)** — stationary clutter (ground, buildings) appears in the zero-Doppler bin and can be suppressed by the FPGA's clutter filter. This is critical for C-UAS: distinguishing a hovering drone from a tree requires micro-Doppler analysis (detecting rotor blade signatures). The AERIS-10's 2.67 m/s velocity resolution can detect blade modulation from drones whose rotors create periodic velocity variations.

5.4 Range-Angle Map

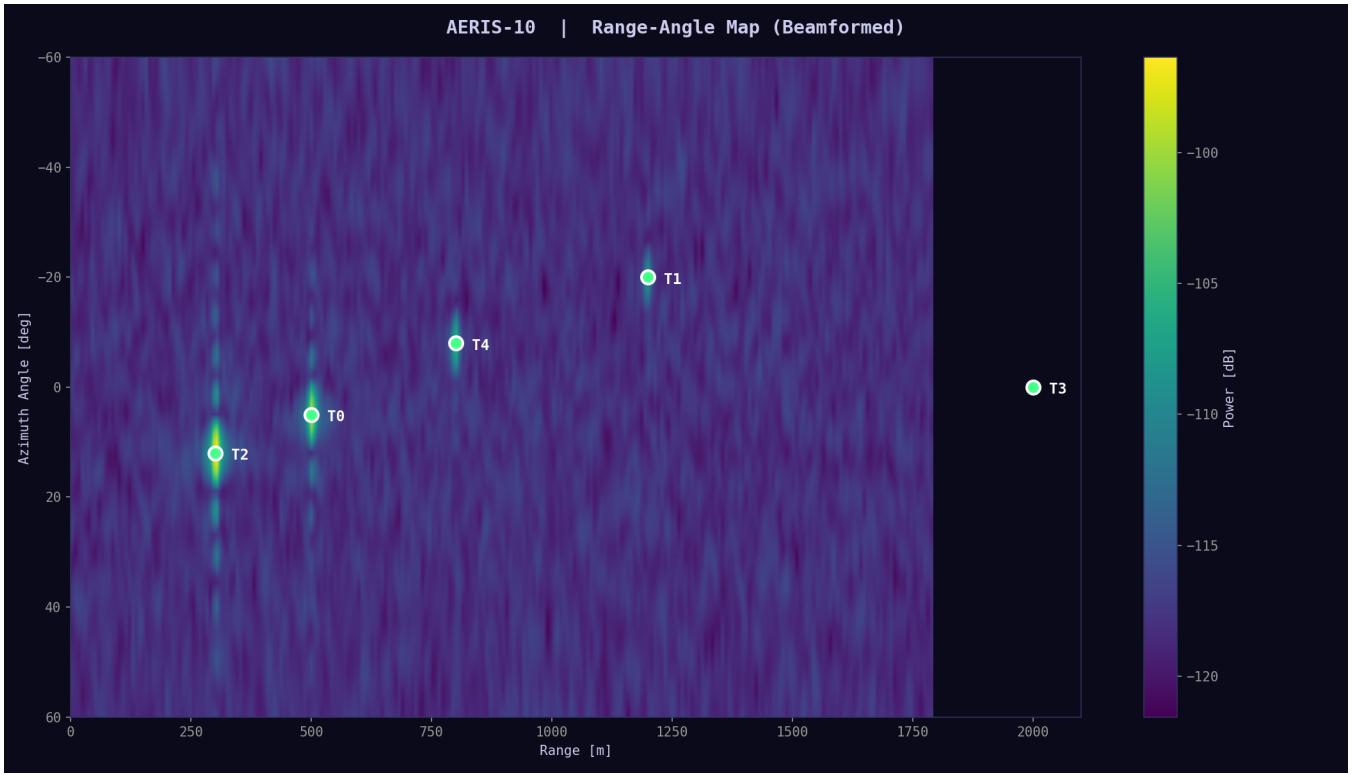

Figure 5.4 — Range-angle map for counter-UAS scenario. Green circles mark true drone positions.

What this shows: The beamformed range-angle map reveals the angular positions of all five drones. The 500 m target at +5° is the brightest. Note how targets at wider angles (–20° for T1, +12° for T2) are dimmer because the beam steering gain drops as sin(θ) increases the effective element spacing.

The angular resolution (~7°) means that two drones at the same range but separated by less than 7° in azimuth cannot be distinguished. This is adequate for early warning but insufficient for precision tracking. The AERIS-10's monopulse angle estimation (sum/difference beams) can refine the angle estimate to ≈1° accuracy within the beamwidth.

Hardware: The ADAR1000 supports **monopulse beamforming** by independently controlling the amplitude and phase of each element. The ADALM-PHASER demo kit specifically includes a monopulse tracking example that forms sum (Σ) and difference (Δ) beams to estimate target angle with sub-beamwidth precision. This is the same technique used by missile seeker heads and fire-control radars.

5.5 CFAR Detection

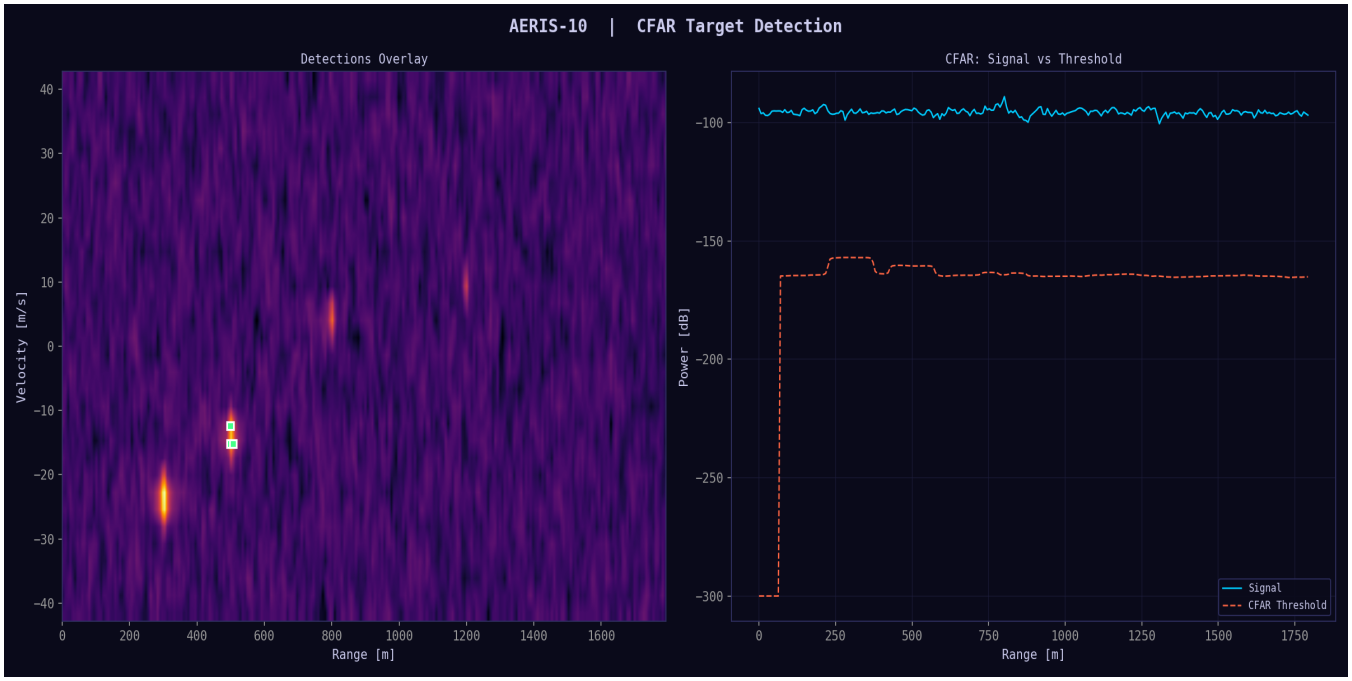

Figure 5.5 — CFAR detection for counter-UAS: fewer detections due to low-RCS targets at longer ranges.

What this shows: Only 3 CFAR cells trigger in this scenario, all around the 500 m target. This is a realistic result: small drones at long range are genuinely difficult to detect. The CFAR threshold (orange dashed line in the right panel) adapts to the noise floor, and only the 500 m target's peak exceeds it.

Implication for the product: The AERIS-10N (without GaN PAs) has a practical C-UAS detection range of ≈500–800 m for small drones. The AERIS-10X (with GaN PAs providing +10 dB additional TX power) extends this to ≈1.5–3 km. Competitors like Echodyne EchoGuard detect small drones at 2–3 km using similar X-band phased array technology but at 10–20× the price.

Hardware: To improve detection performance beyond this simulation, three hardware paths exist: (1) **More elements** — doubling from 16 to 32 elements adds +3 dB array gain and halves the beamwidth; (2) **Longer CPI** — increasing from 32 to 128 chirps per beam adds +6 dB processing gain but slows the scan rate; (3) **GaN PAs** — the QPA2962 adds ~10 dB TX power per element. The AERIS-10 architecture supports all three upgrades without redesigning the digital backend.

5.6 Summary Dashboard

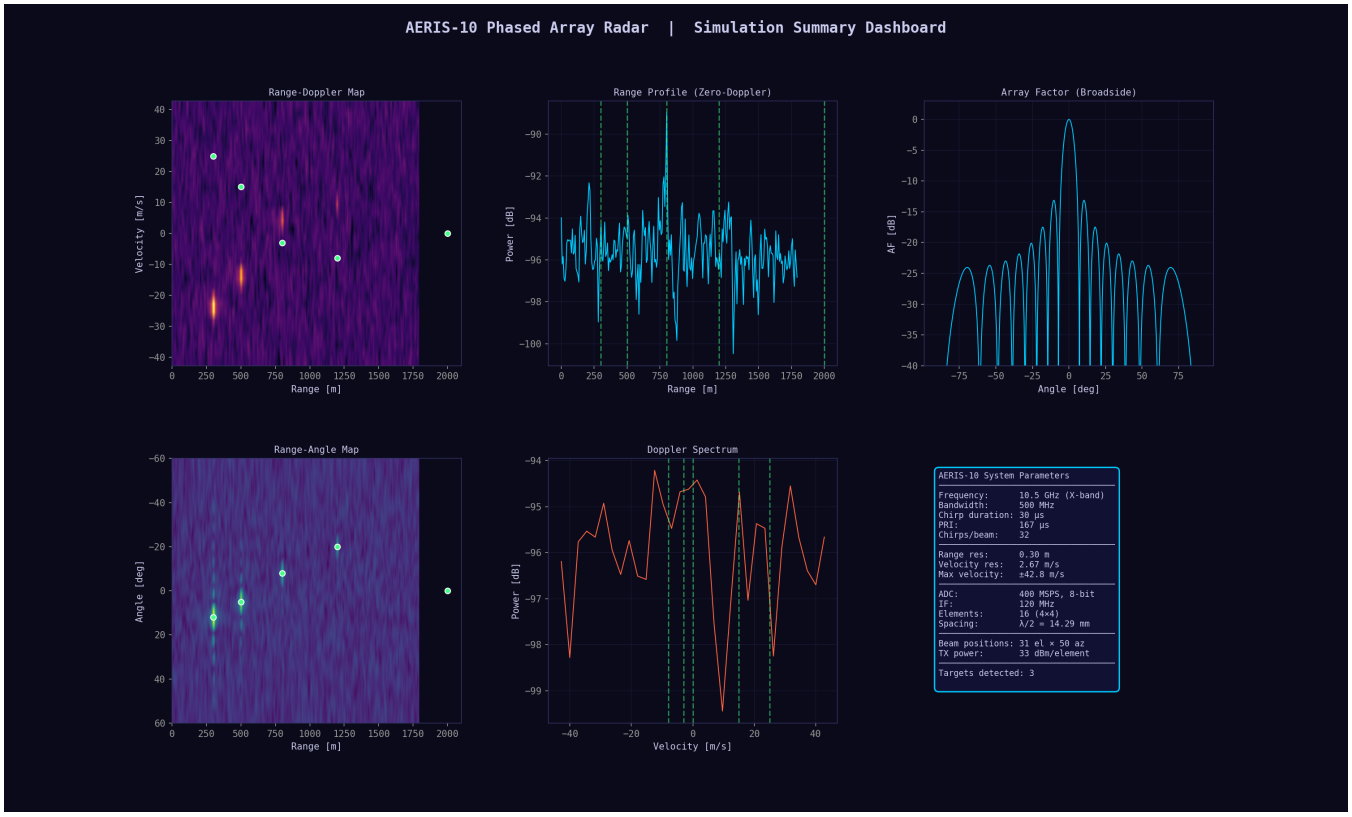

Figure 5.6 — Counter-UAS simulation summary dashboard.

What this shows: The six-panel summary for the counter-UAS scenario. Compared to the demo scenario (Figure 4.11), the targets are dimmer and the noise floor is more prominent, reflecting the greater challenge of detecting small drones at longer ranges. The beam pattern and system specifications are identical between scenarios (same hardware), only the targets differ.

This dashboard would be presented to defense/security investors as evidence that the AERIS-10 can detect realistic C-UAS targets, alongside the ADALM-PHASER live demo showing actual beam steering at 10.5 GHz.

6. Hardware Correlation Summary

The table below summarizes how each simulation output maps to physical hardware on the AERIS-10 board, what the simulation validates, and what it cannot validate (requiring actual hardware testing).

Simulation Output	Hardware Component(s)	Validates	Requires HW Test
TX chirp waveform	ADF4382A + AD9523-1	Chirp linearity, timing, BW	Phase noise, spectral purity, spurs
Range FFT / profile	AD9484 ADC + FPGA	Range resolution, bin mapping	ADC SFDR, quantization noise, clock jitter
Doppler FFT / spectrum	FPGA + OCXO	Velocity resolution, max vel.	Inter-chirp phase coherence, OCXO stability
Beam pattern	ADAR1000 × 4 + antenna	Array factor, steering range	Mutual coupling, element mismatch, scan loss
Range-Angle map	ADAR1000 + FPGA	Angular resolution, beam gain	Real antenna patterns, near-field effects
CFAR detection	FPGA + STM32	Detection threshold, Pfa	Clutter statistics, multipath, interference
CPI timing	STM32 TIM1 + GPIO	Frame rate, update time	Jitter, latency, ISR overhead
Noise model	LNA + mixer + ADC	SNR estimates, range limits	Actual NF, gain flatness, I/Q imbalance

The rightmost column ("Requires HW Test") defines the specific measurements needed from the ADALM-PHASER demo kit and eventual AERIS-10 prototype. These are the risks that simulation alone cannot retire.

7. Demo Readiness Assessment

The following table assesses readiness for an investor demo, combining this simulation with the ADALM-PHASER hardware demo and the TI IWR1443 radar evaluation module.

Demo Element	Source	Status	Investor Impact
FMCW signal processing theory	This simulation	READY	Shows deep technical understanding of the radar chain
Range-Doppler-Angle resolution	Simulation figures	READY	Quantifies what the radar can measure and its limits
Live beam steering at 10.5 GHz	ADALM-PHASER kit	PENDING HW	HIGH — live demo is the strongest pitch element
Live target detection	ADALM-PHASER + IWR1443	PENDING HW	HIGH — detecting a real person/drone is visceral
FPGA real-time processing	Arty A7-100T + Verilog	PENDING HW	MEDIUM — shows the design runs on real silicon
Counter-UAS detection range analysis	This simulation	READY	Shows honest assessment of capabilities and limits
Hardware cost validation	Business proposal BOM	READY	Proves the 5–10× price advantage claim
System architecture walkthrough	Block diagram + schematics	READY	Shows a real, complete hardware design exists

Bottom line: The simulation package (this report) plus the business proposal PDF provide the *analytical foundation* for an investor pitch. The ADALM-PHASER demo kit, once procured, provides the *visceral proof* that the technology works in hardware. Together, they form a complete pre-seed investor package.

AERIS Radar Systems | Simulation Technical Report | March 2026 | All parameters derived from github.com/NawfalMotii79/PLFM_RADAR (MIT License)

Simulation code: aeris10_radar_sim.py | PDF generator: generate_sim_report_pdf.py | Environment: Python 3.14 + NumPy + Matplotlib + ReportLab