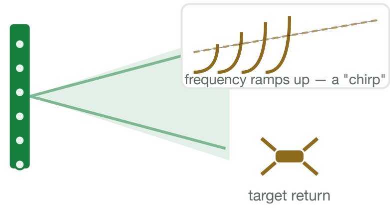
KASHAF كشّاف

What KASHAF is — and the technology that makes it work

What is KASHAF?

TECHNICAL BRIEF

KASHAF is a **10.5 GHz (X-band) phased-array radar** that detects, tracks, and measures the speed of airborne targets — including the small, low, slow drones that are hardest to see. It transmits a **Pulsed Linear-Frequency-Modulated (PLFM)** waveform: short pulses whose frequency ramps up linearly, called a **chirp**. On receive, **pulse compression** squeezes each long chirp back into a sharp spike, giving both long range *and* fine range resolution from modest transmit power.



Four ideas behind the radar

WAVEFORM
Chirp & pulse compression
Long, low-power pulses become sharp echoes — long range with fine resolution.

ANTENNA
Phased array
The beam is steered electronically by shifting each element's phase — no moving dish.

DOPPLER
Coherent processing
Phase-locked oscillators let it measure a target's velocity and boost weak returns.

BAND
X-band, 10.5 GHz
A ~2.85 cm wavelength — the sweet spot for spotting small aerial targets.

Key specifications

10.5 GHz X-band carrier	**Pulsed LFM** Chirp waveform	**FPGA** Real-time signal processing	**3 / 20 km** Two range variants

System at a glance

REFERENCE	**OCXO** master clock	→	**Clock gen** AD9523-1	→	**FPGA** XC7A200T	→	**STM32** MCU control		
TRANSMIT	**FPGA chirp** DDS	→	**DAC**	→	**Synth TX** ADF4382	→	**GaN PA**	→	**Array + ADAR1000** beam steer
RECEIVE	**Array**	→	**Mixers** 2× LT5552	→	**Synth RX** ADF4382	→	**ADC**	→	**FPGA DSP** → GUI

From power-on to first detection

6 STAGES

1

Power & clock discipline STM32 · OCXO · AD9523

The STM32 MCU sequences the power rails, then waits for the OCXO — an oven-controlled crystal oscillator — to reach thermal stability, since it is the master frequency reference. The MCU manages voltage sequencing and configures the AD9523 clock generator, fanning out a phase-aligned clock network that synchronizes the FPGA and every RF component.

2

Bring up sensors & RF front-end GPS · IMU · BARO · ADF4382 · ADAR1000

The system initializes the GPS, IMU, and barometer, then the two ADF4382 frequency synthesizers — one for transmit, one for receive — and the four ADAR1000 beamforming chips that steer the array.

3

Operator start & GPS lock PYTHON GUI ↔ STM32

When the operator presses **Start** in the Python GUI, the interface sends a handshake packet and the radar settings to the STM32. The MCU verifies the configuration, monitors the GPS stream, and waits for a lock before returning position data — which the GUI uses to center its map on the radar's location, giving the operator spatial context for every target.

4

Orientation & north reference IMU + BARO → STEPPER

While the GPS updates, the STM32 reads the IMU and barometer to compute pitch, roll, yaw, and altitude. The yaw value commands the stepper motor to turn the antenna to a known reference bearing, so every scan starts from the same direction — keeping the radar's angular measurements consistent.

5

Phase-coherent RF 2× ADF4382 · SYNC

The RF chain now enters phase-coherent operation. The MCU sets the two ADF4382 synthesizers and runs a synchronization step that keeps the transmit and receive oscillators locked in phase. This coherence is required for **Doppler processing** and **coherent pulse compression**, sharpening both target detection and velocity measurement.

6

FPGA fast signal processing ARTIX-7 XC7A200T

The FPGA handles the fast signal processing: it generates the chirps, captures the echoes, and down-converts them to baseband using two LT5552 microwave mixers. It then applies filtering, pulse compression, a Doppler FFT, moving-target indication, and CFAR detection before streaming the results to the GUI.

Inside the FPGA: the signal-processing pipeline

Chirp gen	→	Echo capture	→	Down-convert	→	Pulse compress	→	Doppler FFT	→	MTI + CFAR
Transmit waveform		ADC samples		2× LT5552		Matched filter		Velocity bins		Detect targets

Why the boot order matters

Every stage exists to protect one thing: **phase coherence**. Stable clocks, a locked GPS position, a known north reference, and synchronized oscillators are what let the FPGA turn faint echoes into confident range, bearing, and velocity — the difference between seeing a drone and missing it.

Want the full engineering package?

Write to us: **alshehhi@ruyatechai.com** · Ruya — Dubai, UAE