Co-funded by the European Union

6GSNS

SUPERIOT

D4.4 Demonstrator 4: Large-area IoT Node

Project number	**101096021**
Project name	**Truly Sustainable Printed Electronics-based IoT Combining Optical and Radio Wireless Technologies**
Project acronym	**SUPERIOT**
Call	**HORIZON-JU-SNS-2022**
Deliverable No	**D4.4**
Deliverable Name	**Demonstrator 4: Large-area IoT Node**
Status	**Final**
Dissemination level	**Public**
Due date of deliverable	**2025-10-31 (M34)**
Actual submission date	**2025-10-31 (M34)**
Resubmission date	**2026-05-22**
Work package	**WP4 "Test Bed/Demonstrators"**
Lead beneficiary	**INESC TEC**
Authors	Nuno Paulino (INESC TEC), Francisco Ribeiro (INESC TEC), Guido Dolmans (IMEC), Sofia Inácio (INESC TEC), Mohamed Ghatas (INESC TEC), Eva Campos (INESC TEC)
Reviewers	João Coelho (NOVA.id.FCT), Emanuel Carlos (NOVA.id.FCT), André Branquinho (WAVECOM)

The SUPERIOT project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101096021, including top-up funding by UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee.

Executive Summary

This deliverable, D4.4 – Demonstrator 4: Large-area IoT Node, presents the design, implementation, and experimental validation of a Reconfigurable Intelligent Surface (RIS) integrated into the SUPERIOT multi-technology IoT platform. The work demonstrates how RIS technology can be combined with localization and message-based control frameworks to enable adaptive and sustainable connectivity in dynamic indoor environments.

The demonstrator integrates three main functional components: 1) a BLE localization subsystem, which continuously tracks the position of a mobile IoT node and publishes its coordinates via MQTT; 2) an embedded RIS controller, implemented on an ESP32-S3 microcontroller, which computes phase-control maps and reconfigures a hardware RIS tile in real time; and 3) a monitoring and visualization interface, which aggregates localization and control data, computes steering angles, and visualizes the complete system state through a MATLAB Graphical User Interface (GUI).

Together, these components demonstrate dynamic, software-defined reconfiguration of the radio environment through real-time RIS beam steering driven by external sensing information. In the deployed BLE-assisted scenario, the RIS configuration is continuously updated based on the estimated position of a moving IoT node (using the previously developed SUPERIOT core node in WP2 and the BLE-based localization subsystem in WP3). Complementarily, a second vision-assisted scenario demonstrates closed-loop RF validation through real-time S21 monitoring using horn antennas and a vector network analyzer (VNA).

The demonstrator was deployed and validated at imec's Vitality Hub, where it operated as part of the shared SUPERIOT IoT testbed, and was additionally showcased at ICASSP 2026 through a vision-assisted beam-steering demonstrator with closed-loop RF measurements. Experimental runs confirmed correct real-time integration across the SUPERIOT MQTT backbone, reliable beam-steering updates at sub-second latency, stable operation of the RIS tile hardware under continuous reconfiguration, and successful RF beam adaptation validated through real-time S21 measurements. The complete system highlights the feasibility of RIS-based repeater nodes as part of future programmable and sustainable IoT infrastructures.

Beyond the demonstrator, the deliverable also reports on novel RIS hardware technologies that enable future scalability. Namely, 1) glass-based transmissive metasurfaces for optically transparent RIS integration into building facades and windows, achieving >90% optical transparency and full 360° phase coverage at 10 GHz; and 2) solution-based memristor-controlled unit cells for ultra-low-power reflective RISs, achieving 180° phase tuning at 6 GHz with sub-nanojoule switching energy and no static power consumption.

Together, these results confirm the SUPERIOT vision of reconfigurable, sustainable, and material-integrated IoT surfaces, bridging traditional RF design with emerging printed and smart-material technologies.

Table of contents

Editions

Version	Date	Modified by	Modification
0.1	2025-10-20	Nuno Paulino, INESC TEC	First document outline
0.2	2025-10-21	Francisco Ribeiro, INESC TEC	GUI related content
0.3	2025-10-22	Nuno Paulino, INESC TEC	Core content and structural revision
0.4	2025-10-23	Nuno Paulino, INESC TEC Francisco Ribeiro, INESC TEC	Core content and structural revision
0.5	2025-10-24	Nuno Paulino, INESC TEC Francisco Ribeiro, INESC TEC Guido Dolmans, IMEC	Core content and structural revision; Integrated IMEC content; Integrated Chapter 3.
0.6	2025-10-27	Nuno Paulino, INESC TEC	Ready for internal review.
0.7	2025-10-30	Nuno Paulino, INESC TEC	Fixes after internal review.
0.8	2025-10-30	Nuno Paulino, INESC TEC	Fixes after internal review.
1.0	2025-10-30	Luis Pessoa, INESC TEC	Final quality assurance.
2.0	2026-05-18	Eva Campos, INESC TEC Nuno Paulino, INESC TEC Luis Pessoa, INESC TEC	Modifications addressing the request for revision: Added a section 2.3 with a second scenario enabling closed-loop RF validation, and a new section describing how project components have been used. Updated the executive summary.

Acronyms

Table 1: List of Acronyms

6G	The sixth-generation mobile communication system
AP	Access Point
BLE	Bluetooth Low Energy
CS	Channel-Sounding
DK	Development Kit
GUI	Graphical User Interface
HG2	Handle Graphics 2 – MATLAB's modern UI rendering system
IoT	Internet-of-Things
JSON	JavaScript Object Notation
MQTT	Message Queuing Telemetry Transport
MRM	Multiple Reflection Method
NLoS	Non-Line-of-Sight
OT	Optical Transparency
PIN Diode	Positive-Intrinsic-Negative Diode
RF	Radio-Frequency
RIS	Reconfigurable Intelligent Surface
SPI	Serial Peripheral Interface
SUPERIOT	Truly Sustainable Printed Electronics-based IoT Combining Optical and Radio Wireless Technologies
VLC	Visible-Light Communication

1 Introduction

1.1 Motivation

The emergence of the Internet of Things (IoT) has driven a fundamental transformation in how the physical environment interacts with the digital world. Reconfigurable Intelligent Surfaces (RIS) have recently emerged as a promising solution to tackle the density of connected objects which continues to grow, helping to maintain the reliable, sustainable, and adaptive wireless connectivity which is key to future 6G systems.

While conventional wireless infrastructures struggle to deliver uniform coverage and efficient spectrum reuse in complex indoor or urban environments, RIS technology enables the dynamic shaping of the radio environment, redirecting or focusing electromagnetic waves in real time. Such a capability transforms ordinary walls, ceilings, or furniture into components of the communication network, forming so-called smart surfaces that enable novel communication techniques.

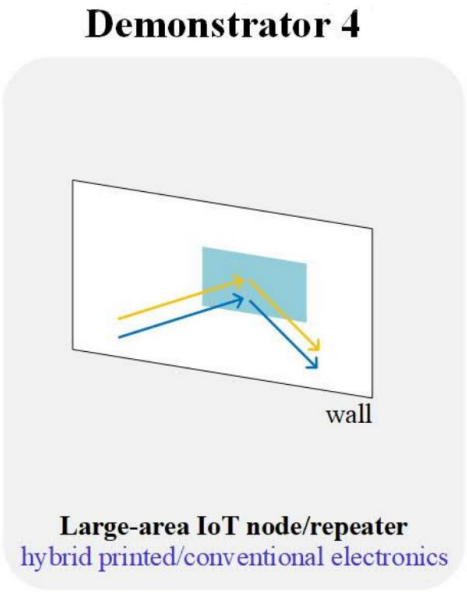


Figure 1. General concept of Demo 4.

Research on concrete technologies and implementations that fulfill this potential are aligned with several strategic objectives and their respective benefits:

- **Sustainability and scalability:** exploring the feasibility of RIS structures that can ultimately be fabricated using environmentally friendly substrates, conductors, and printed active devices, reducing both cost and ecological impact.
- **Integration into built environments:** enabling *large-area deployment* on everyday structures such as walls, doors, or windows, providing communication coverage and sensing functions without dedicated antennas.
- **Multi-functionality:** combining connectivity, localization, and perception into a single reconfigurable platform that interacts seamlessly with the wider IoT ecosystem.
- **Contribution to 6G research:** validating one of the enabling technologies for the envisioned sustainable, intelligent, and programmable radio environment.

This forms the general motivation of Demonstrator 4 in SUPERIOT, which aims to show how a large-area, reconfigurable surface can act as a smart IoT node or repeater, enhancing

coverage, improving link reliability, and enabling context-aware communication and sensing. This demonstrator therefore bridges the gap between theoretical RIS concepts and their integration into sustainable large-scale IoT infrastructures, complementing SUPERIOT's broader goal of merging printed electronics with advanced communication technologies.

1.2 Structure of the document

The remainder of this Deliverable is organized as follows: Chapter 2 presents the detailed description of Demonstrator 4, including its requirements, architecture, hardware and software design, monitoring interface, and deployment setup at imec's Vitality Hub; Chapter 3 presents fundamental research towards alternative technologies for RIS design, which will enable lower-cost scalability; finally, Chapter 4 concludes the document by summarizing the demonstrator's results, lessons learned, and future directions toward scalable and sustainable RIS deployments.

2 Demonstrator 4: Adaptive Beam Steering Scenarios

Demonstrator 4 explores the use of Reconfigurable Intelligent Surfaces (RIS) as adaptive radio components capable of dynamically steering wireless signals toward moving targets. By electronically controlling the phase response of an array of unit cells, the RIS enables software-defined manipulation of the radio environment, allowing wireless links to be redirected in real time. Within SUPERIOT, the goal of Demonstrator 4 was to validate how RIS technology can integrate with heterogeneous sensing and communication systems through a shared MQTT-based infrastructure. To this end, two complementary scenarios were developed around a common RIS hardware and embedded software platform:

- a communication scenario where **beam steering toward a target IoT node** (the SUPERIOT node core) is assisted by a BLE-based localization system, deployed at imec's Vitality Hub; and
- a **vision-assisted scenario**, presented at ICASSP 2026, where computer-vision algorithms detect targets in real time, enabling automatic beam steering and closed-loop RF validation.

Both scenarios share the same modular 6.5 GHz RIS tile, ESP32-S3 controller, MQTT communication layer, and embedded beamforming software. External sensing systems provide target coordinates, which are converted into steering angles and translated into RIS phase-control patterns applied in real time to the hardware. Together, these scenarios demonstrate real-time RIS beam steering, interoperability through MQTT middleware, embedded phase-pattern computation, and integration of RIS technology into adaptive and context-aware IoT environments.

The following sections first describe the common RIS hardware and embedded software platform shared by both scenarios, including the RIS tile architecture, firmware structure, and MQTT-based communication interfaces. Subsequently, each scenario is detailed individually, focusing on its respective sensing subsystem, visualization software, deployment setup, and experimental operation.

2.1 RIS Tile and Embedded Software

2.1.1 RIS Tile Hardware

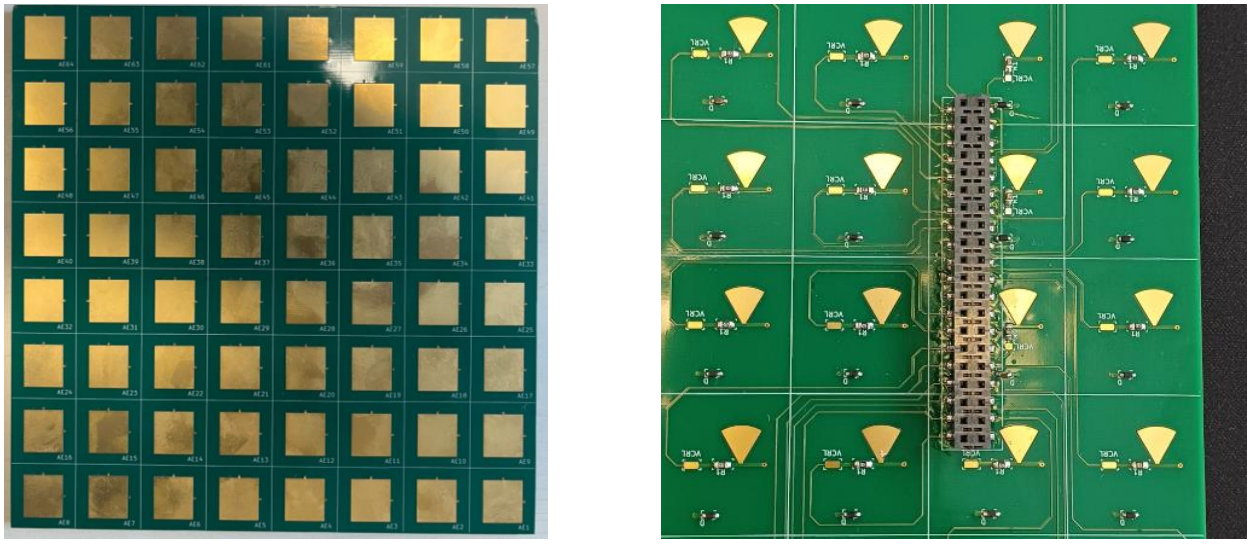
The physical implementation shared by both scenarios is based on a modular 6.5 GHz RIS developed within the SUPERIOT project [1] and refined for the present demonstrators. The system consists of the reflective panel of unit cells itself, and a rear-mounted control board hosting the micro-controller as well as auxiliary hardware for unit-cell reconfiguration and power delivery.

2.1.1.1 RIS Tile

The RIS tile (Figure 2a) integrates 64 reconfigurable unit cells arranged in an 8×8 array, separated by a 20mm pitch, which is approximately half of the wavelength of the central frequency. Each cell operates at 6.5 GHz, achieving a 180° phase shift between its two states through the biasing of a PIN diode. The tiles are fabricated on a dual-layer PCB stack consisting of 3 mm F4B and 0.4 mm FR4 substrates, ensuring mechanical stability and low dielectric loss.

Whilst the first version of the RIS tile (shown in MS7) relied on a Skyworks SMP1331-079LF PIN diode, and a 23mm unit cell which was not symmetric within its bounding box (i.e., a rectangular path which was offset by from the centre of the unit cell) as presented in [1]. Based on the experimental results, the current version relies on a square patch centred within

the cell and an Infineon BAR64-02V PIN diode, achieving an improved response at the target frequency. The redesigned square patch is centred within the cell, improving symmetry which promotes easier scalability by composition of multiple tiles.



(a) (b)

Figure 2. Front layer (a) and back layer (b) of re-designed 8x8 RIS tile for 6.5GHz, with 1-bit unit cells.

2.1.1.2 Control Board

The RIS tile connects mechanically and electrically to its dedicated control board through two 40-pin SMD connectors on the rear side (Figure 2b). This design choice ensures clean front-facing radiation and simplifies tile replacement or scaling to multi-tile configurations. That is, by acting as a physical support for the tile which does not require compromising the unit cell pitch for adjacently placed tiles, several control boards held by a singular backing structure (i.e., rigid acrylic plate), allow for tiles to be joined edge-to-edge while maintaining phase alignment between neighbouring cells, creating a seamless large-area RIS array.

Figure 3. Control board with 8 shift registers totalling 64 bits for the PIN diode-based cells.

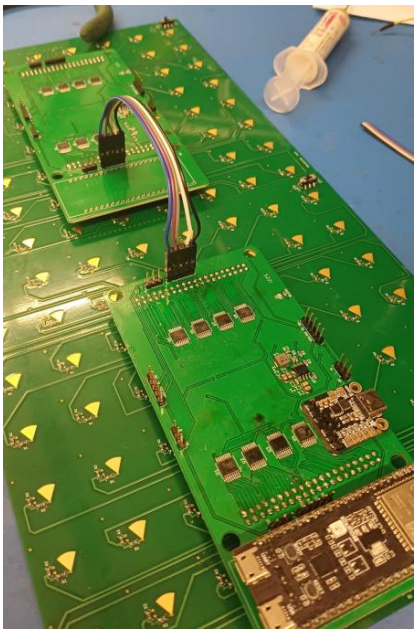


Figure 4. Two control boards forming a series of 16 shift registers to control two tiles.

The control board (Figure 3) itself hosts eight 74HC595 shift registers, providing 64 individually addressable digital outputs, one for each RIS element. These are daisy-chained in series and driven through a Serial Peripheral Interface (SPI) link. To drive these shift registers, the control board also hosts an ESP32-S3-DevKitC-1 microcontroller. The ESP32 provides integrated Wi-Fi connectivity for communication with the MQTT broker, a dedicated SPI interface, a 240MHz dual-core processor and close to 1Mb of program memory, which proved sufficient for all required asynchronous services and real-time beamforming calculations. Power is provided by a USB-C power delivery module, followed by a 3.3V buck regulator. More than one control board can be chained, each attached to its own tile, to form larger assemblies in a scalable way (Figure 4).

2.1.2 Embedded RIS Software

2.1.2.1 Description

The embedded software is implemented in C/C++ using the ESP-IDF framework and a VSCode environment for compilation onto the ESP32-S3. It operates as a Wi-Fi-enabled MQTT client, subscribing to position updates published by the BLE localization subsystem. Whenever new coordinates are received, the RIS firmware computes a reflection pattern and updates the hardware configuration through the SPI interface. The pattern generation is based on a geometric beamforming model, which calculates the required phase compensation for each unit cell to focus the reflected energy toward the target node. For the 1-bit RIS used in this demonstrator, the computed continuous phase values are quantized into two discrete states (0° and 180°), corresponding to the OFF and ON states of the PIN diodes.

2.1.2.2 Software Structure

The firmware running on the ESP32-S3 present on the controller board is responsible for establishing MQTT communication with our monitoring interface, in order to receive target angle updates. It then computes the required phase pattern on-chip, and establishes this RIS control state by updating the values held by the shift-register daisy chain. The code is structured around object-oriented encapsulation: each subsystem (Wi-Fi, MQTT, SPI, RIS array) is abstracted into a class with clear initialization and update interfaces. This enables the

firmware to be extended for multiple tiles, different tile sizes and operating frequencies, or additional protocols with minimal modification.

Table 2. Software structure overview of RIS controller firmware.

File / **Class**	Purpose
main.cpp	Entry point. Initializes Wi-Fi, MQTT, SPI, and the array controller. Defines the main event loop that reacts to MQTT messages.
WiFiInterface	Encapsulates Wi-Fi setup, connection handling, and reconnection logic. Provides callbacks to notify other components (e.g., MQTT) when the device successfully connects or disconnects.
MqttClient	Manages MQTT communication by acting as a wrapper to the *AsyncMqttClient* library for the ESP32. Handles broker connection, topic subscriptions, and dispatches received messages to the appropriate handlers in the control logic.
SpiInterface	Abstracts the low-level SPI driver controlling the 74HC595 shift registers. Provides simple methods for writing 64-bit bitmaps to the RIS array, ensuring atomic updates and low latency.
TileParams / Tile / Array	Three classes defining the geometric and physical properties of the RIS tile (size, element pitch, reference coordinates). Contain the beamforming equations that map a desired (θ, ϕ) steering direction into a binary ON/OFF matrix for the unit cells. The *Array* class aggregates multiple Tile for scalability. All tiles share the same *TileParams* instance of information regarding physical design of the tile, from which other data is derived (e.g., relative and absolute positions of the unit cell in the assembly and overall coordinate system).
ArrayController	Top-most class encapsulating an instance of *WiFiInterface*, *MqttClient*, *SpiInterface*, and an *Array*. Exposes only the required public methods to perform beamforming and query on status. It receives a pointer to a control loop function defined in *main.cpp,* where the logic for this particular demo is implemented, meaning other control loops could be implemented easily without modifying core functionality.

Furthermore, the entire functionality of the firmware has been generalized into abstract C/C++ classes encapsulating most of the functionality, apart from low-level target specific features (e.g., Wi-Fi setup). This means the software can currently be compiled for ESP32 targets as well as x86/x64 targets with ease, allowing for a tile assembly to be controlled from a conventional laptop without loss of MQTT functionality or SPI control of the shift-registers.

2.1.2.3 MQTT Messaging

The firmware also exposes several MQTT command interfaces that allow remote control and diagnostics. These include topics for pattern updates, system resets, and telemetry reporting. A compact JSON-based message format is used, allowing seamless interoperability with other SUPERIOT subsystems and visualization tools. Namely, the software subscribes to the following MQTT topics:

- **"beamform/steer"**: which indicates a new target angle for the RIS; this message receives as payload the string *"theta=<arg0>,phi=<arg1>"* where the arguments are expected to be two floating point numbers, e.g., *"theta=45.0,phi=0.0";*
- **"beamform/print"**: a command with a single boolean flag as argument, which enables or disables all serial port output, i.e., calls to *printf* or similar functions. This greatly reduces overhead (and therefore increases refresh rate), if the control board is not connected to a laptop/computer and thus this status/debug information output is not required. If enabled, the control software outputs sequences of function calls, boot and status information, as well as the current phase pattern status;
- **"beamform/zero"**: sets all shift-register outputs to logical zero. Since no target angle produces a status where all unit cells are set to an "OFF" state, this is means to enforce a "soft reset" on the shift-register chain;

- **"beamform/feed"**: changes the position of the feedhorn relative to the RIS surface, which has implications on how the beamforming is performed. Allows for changes in physical setup to be reflected on the control software without need of recompilation and reprogramming of the ESP32-S, which would greatly compromise ease of deployment;
- **"beamform/report"**: requests that the control software reply with the current phase control status, which is sent also via MQTT to topic **"beamform/output"**, to which our monitoring software subscribes to display system status. This output topic is automatically published too after steering is performed, but the **"beamform/report"** exists to allow for asynchronous querying of status.

After establishing a Wi-Fi connection, the ESP32 automatically connects to the project's MQTT broker and subscribes to all above topics. It then updates the RIS status uninterruptedly, until powered down or reset. If MQTT or Wi-Fi connectively is lost, the firmware automatically soft resets to re-establish connectivity.

2.1.2.4 Example

As an example, consider the following output of the firmware executed on an x86 machine. For these runs the MQTT server runs locally, and the SPI output is redirected to a pipe file descriptor consumed by a separate process. A small C application is used to publish an update to the "beamform/steer" topic to trigger the beamforming behaviour by the firmware. Both examples run with status/debugging prints enables to the terminal (which on the ESP32 would be the serial-over-USB port). The output in Figure 5 shows the graphical representation of the phase control status at boot, which steers towards (θ=0, φ=0), and Figure 6 shows the output for a 2x2 tile assembly after boot and after a command requesting (θ=15º, φ=0º).

```
Init SPI dummy to file...
DummyInterface created for file/device: /tmp/riscontrol
DummyInterface init: Attempting to create/open FIFO /tmp/riscontrol...
DummyInterface init successful. FIFO file descriptor: 3
Init MQTT (Via mosquitto)...
MqttClientBase constructed with config: localhost:1883
🚀 MQTT client initialized and connected (x86)
📡 Subscribing to: beamform/steer
✅ Subscribed to topic: beamform/steer
Build array classes...
Initing array with 1 x 1 size
Initing tile at offset (x, x) = (0.000000, 0.000000) size
Initializing Tile #0 at grid position (x, y) = (0.000000, 0.000000)
  Tile cell dimensions: (8 x 8) = 64
  Unit cell size: 23.077 mm
  Tile size (mm): (184.616 x 184.616)
  Tile center offset from array center (mm): (0.000 x 0.000)
Total number of tiles: 1
Total number of unit cells: (8 x 8)
Total array size in mm: (184.62 x 184.62)
Set Zero state...
 0 0 0 1 1 0 0 0
 0 1 1 1 1 1 1 0
 0 1 1 1 1 1 1 0
 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1
 0 1 1 1 1 1 1 0
 0 1 1 1 1 1 1 0
 0 0 0 1 1 0 0 0

Test MQTT Loop...
MqttClientX86: 🔗 MQTT connected (x86)
MqttClientX86: entering on_message event
MqttClientX86: exiting on_message event
message = beamform/steer
payload = theta=15.0,phi=0.0
 1 0 0 0 1 1 1 1
 0 0 0 1 1 1 1 1
 0 0 0 1 1 1 1 1
 0 0 0 1 1 1 1 1
 0 0 0 1 1 1 1 1
 0 0 0 1 1 1 1 1
 0 0 0 1 1 1 1 1
 1 0 0 0 1 1 1 1

DummyInterface::send64 successful.
```

Figure 5. Example for one 8x8 tile receiving a beamforming command.

```
Init SPI dummy to file...
DummyInterface created for file/device: /tmp/riscontrol
DummyInterface init: Attempting to create/open FIFO /tmp/riscontrol...
DummyInterface init successful. FIFO file descriptor: 3
Init MQTT (Via mosquitto)...
MqttClientBase constructed with config: localhost:1883
🚀 MQTT client initialized and connected (x86)
📡 Subscribing to: beamform/steer
✅ Subscribed to topic: beamform/steer
Build array classes...
Initing array with 2 x 2 size
Offsetting x center...
Offsetting y center...
Initing tile at offset (x, x) = (-0.500000, -0.500000) size
Initializing Tile #0 at grid position (x, y) = (-0.500000, -0.500000)
  Tile cell dimensions: (8 x 8) = 64
  Unit cell size: 23.077 mm
  Tile size (mm): (184.616 x 184.616)
  Tile center offset from array center (mm): (-92.308 x -92.308)

<...>

Total number of tiles: 4
Total number of unit cells: (16 x 16)
Total array size in mm: (369.23 x 369.23)
Set Zero state...
 1 1 1 1 1 0 0 0   0 0 0 1 1 1 1 1
 1 1 1 0 0 0 0 0   0 0 0 0 0 1 1 1
 1 1 0 0 0 0 0 0   0 0 0 0 0 0 1 1
 1 0 0 0 0 0 0 0   0 0 0 0 0 0 0 1
 1 0 0 0 0 0 0 1   1 0 0 0 0 0 0 1
 0 0 0 0 0 1 1 1   1 1 1 0 0 0 0 0
 0 0 0 0 0 1 1 1   1 1 1 0 0 0 0 0
 0 0 0 0 1 1 1 1   1 1 1 1 0 0 0 0

 0 0 0 0 1 1 1 1   1 1 1 1 0 0 0 0
 0 0 0 0 0 1 1 1   1 1 1 0 0 0 0 0
 0 0 0 0 0 1 1 1   1 1 1 0 0 0 0 0
 1 0 0 0 0 0 0 1   1 0 0 0 0 0 0 1
 1 0 0 0 0 0 0 0   0 0 0 0 0 0 0 1
 1 1 0 0 0 0 0 0   0 0 0 0 0 0 1 1
 1 1 1 0 0 0 0 0   0 0 0 0 0 1 1 1
 1 1 1 1 1 0 0 0   0 0 0 1 1 1 1 1

MqttClientX86: 🔗 MQTT connected (x86)
Test MQTT Loop...
MqttClientX86: entering on_message event
MqttClientX86: exiting on_message event
message = beamform/steer
payload = theta=15.0,phi=0.0
 1 1 0 0 1 1 1 0   0 0 0 0 0 0 0 0
 1 0 0 0 1 1 0 0   0 0 0 0 1 1 1 1
 1 0 0 1 1 1 0 0   0 0 1 1 1 1 1 1
 1 0 0 1 1 0 0 0   0 1 1 1 1 1 1 1
 0 0 1 1 1 0 0 0   1 1 1 1 1 1 1 1
 0 0 1 1 0 0 0 1   1 1 1 1 1 1 1 1
 0 0 1 1 0 0 0 1   1 1 1 1 1 1 1 1
 0 0 1 1 0 0 0 1   1 1 1 1 1 1 1 1

 0 0 1 1 0 0 0 1   1 1 1 1 1 1 1 1
 0 0 1 1 0 0 0 1   1 1 1 1 1 1 1 1
 0 0 1 1 0 0 0 1   1 1 1 1 1 1 1 1
 1 0 0 1 1 1 0 0   1 1 1 1 1 1 1 1
 1 0 0 1 1 1 0 0   0 1 1 1 1 1 1 1
 1 0 0 1 1 1 0 0   0 0 1 1 1 1 1 1
 1 0 0 0 1 1 0 0   0 0 0 1 1 1 1 1
 1 1 0 0 1 1 1 0   0 0 0 0 0 0 0 0

DummyInterface::send64 successful.
```

Figure 6. Example for four 8x8 tiles receiving a beamforming command.

2.2 Scenario 1: Localization-Assisted Beam Steering

The goal of this scenario was to showcase the integration of reconfigurable radio hardware with the project's multi-technology IoT platform. Its purpose is to illustrate how a Reconfigurable Intelligent Surface (RIS) can operate as a smart IoT repeater node, extending wireless coverage and enhancing link reliability in complex indoor environments through adaptive beam steering toward a tracked IoT target.

This scenario builds upon technology developed within the project and previously reported in Milestone MS7 – "Large-Area Node Design Ready", including both the hardware and software components for scalable RIS assemblies. For the present work, the RIS hardware and corresponding control software were further revised and improved. The RIS surface design was refined at the unit-cell level to achieve an improved response at the target frequency, while addressing design issues identified in earlier revisions of the tile and control boards. The embedded software was extended to support on-chip calculation of phase-control patterns for an arbitrary number of sequentially connected tiles, as well as MQTT-based communication for remote RIS control and integration with external sensing systems.

Through these improvements, beam steering toward a mobile IoT node is assisted by a BLE-based localization system deployed at imec's Vitality Hub (2nd floor of High Tech Campus 85, Eindhoven, the Netherlands) during the project meeting on 23–25 September 2025. The SUPERIOT sub-systems interact through the project's common Message Queuing Telemetry Transport (MQTT) infrastructure. BLE localization data is shared, allowing for the RIS controller and visualization software to dynamically adapt the reflected beam direction in real time.

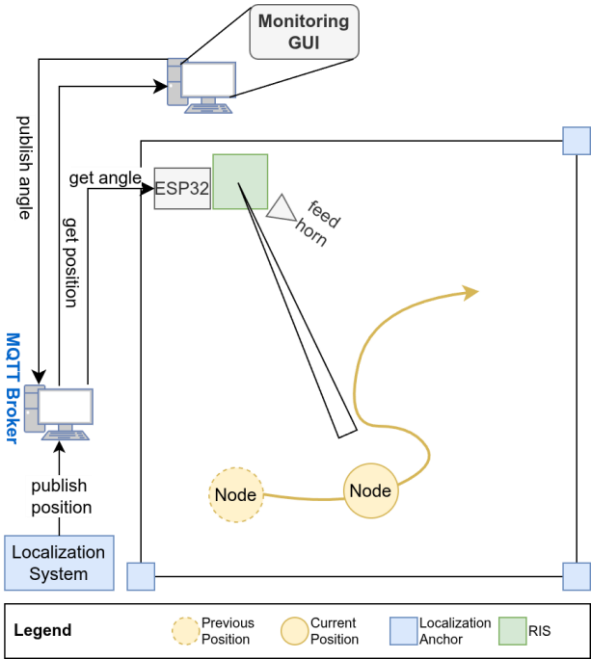


Figure 7. Demonstrator implementation deployed at imec's Vitality Hub.

Figure 7 shows how the RIS-based repeater node can automatically adjust its reflective properties to redirect an incoming Wi-Fi signal toward a moving IoT device. The setup exploits a Bluetooth Low Energy (BLE) localization subsystem available at the Vitality Hub as the source of localization data. This system continuously publishes the position of target IoT nodes within a region of interest to an MQTT server with sub-meter precision. The RIS controller subscribes to this information to compute and apply the optimal reflection pattern in real time, demonstrating how dynamic Non-Line-of-Sight (NLoS) links can be established through integration with external tracking systems.

This integration of hardware-level beam steering and cloud-based system information, as well as the status visualization, highlights the potential of RIS technology as an intelligent, adaptive component. It demonstrates not only the beam-steering and reconfigurability of the RIS, but also coordination between distributed IoT components—combining sensing, communication, and control into a unified adaptive system. It shows how the underlying technology enables several application scenarios where connectivity or sensing can be improved in dynamic environments. For example, by achieving coverage extension in complex indoor spaces, demonstrating sensing-assisted communication, and interoperability of technologies to achieve intelligently managed links. A video was recorded of this demo, to be shown in the final meeting with the PO.

The following sections describe the architecture and data flow of the scenario, followed by the BLE localization subsystem, visualization and monitoring software, and the demonstrator deployment and operation.

2.2.1 Architecture and Dataflow

The scenario shows how an incident signal, from a near-field illumination antenna, is focused as a pencil beam onto a target IoT node located several meters away. The node moves in arbitrary fashion within a region of interest, and its real-time location is provided by a supplementary system with sub-meter precision. Our direct contribution lies in the RIS tile hardware, its embedded software, and a centralized visualization software, as well as the integration with the localization system.

This demonstration scenario integrates three main functional blocks:

- The **RIS hardware and embedded software** explained in the previous section, composed of a single 8x8 tile of 6.5GHz unit cells each controlled by a single PIN diode, and the embedded software which is hosted by a ESP32-S3 based micro-controller board. This software implements all connectivity and MQTT services necessary to retrieve the target reflection angle, compute the respective phase patter, and reconfigure the RIS. To retrieve the target angle, the software also communicates via MQTT with the visualization component which provides system status.
- The **BLE Localization Subsystem**, which was readily available for use in the imec Vitaliy Hub. It is responsible for estimating the position of a mobile BLE node, which we considered as our target end-point for signal focusing. The localization system continuously posts the position of the node within the Vitality Hub showroom floor to an MQTT broker. We exploit integration with this service to implement the following two components.
- A **Monitoring Interface**, which we implemented to provide visualization of the system status, including the location of the target node within the region of interest, as well as the control status of the RIS and resulting beam. This component is responsible for fetching the target position via MQTT, converting this absolute position to position relative to the normal surface of the RIS, and computing the respective angle between both points. By posting this to the MQTT server, the RIS software receives angle updates and redirects the beam accordingly.

Together, these components demonstrate how two edge components, i.e., the localization system and RIS controller, interact with server and middle layer processes, i.e., the MQTT broker and monitor interface to achieve novel communication capabilities. We will explain each component in detail below, but the overall flow of the demo functionality follows this sequence of steps:

1. After establishing a Wi-Fi connection, the ESP32 automatically connects to the project's MQTT broker and subscribes to the topics carrying localization updates;
2. The target node moves in an arbitrary fashion in the region of interest. The localization system posts the location of the target node as a JSON payload to the MQTT broker;

3. The monitoring software is subscribed to the localization data of the node, and when any update is posted, this localization is used to update visualization components, and to compute an angle which is posted back to the MQTT broker under another topic;
4. The embedded firmware receives any angle updates, computes a new reflection phase map and updates the RIS control accordingly. It then posts the current phase control status to the MQTT broker;
5. The monitoring software updates this component of the visualization by fetching it from the broker, and computes the estimated pencil beam shape and direction based on the control state;
6. Repeat from steps 2 to 5.

The use of MQTT as a universal communication layer ensures that the RIS subsystem can interact seamlessly with other components or nodes in the SUPERIOT ecosystem, regardless of their physical communication medium – Radio Frequency (RF), Visible-Light Communication (VLC), or Bluetooth Low Energy (BLE). This architectural choice exemplifies the project's emphasis on interoperability across heterogeneous IoT technologies.

2.2.2 BLE Localization Subsystem

The Demo 4 experiments were carried out at imec's Vitality Hub. The Vitality Hub is a collaborative test and exhibition space jointly operated by TU/e, Fontys, and imec NL, designed to host research pilots in health and vitality technologies. The layout of the space is shown in Figure 8a.

Its modular office and corridor layout provides realistic conditions for indoor localization, wireless communication, and multi-sensor experimentation. Within this space, a dedicated SUPERIOT working area was established along the L-shaped section, specifically at the top-right area of the room, allowing for flexible placement of access points (APs), light sources, and sensor nodes.

The localization setup consisted of multiple Bluetooth 6.0 Channel-Sounding (CS) access points (Figure 10), shown in Figure 8c as A1-A6, and corresponding mobile SUPERIOT core nodes (Figure 9) equipped with visible-light communication (VLC), a Nordic BLE transceiver, electronic-paper displays, and environmental sensors. Each of the six anchor beacons consisted of a Nordic BLE Development Kit (DK), a power bank, and a directional antenna. Dedicated firmware was programmed for the SUPERIOT nodes, and for the 6 anchors.

A mobile SUPERIOT core node, mounted on a movable trolley, served as the tag under test. Localization and sensor data were streamed via MQTT to the SUPERIOT broker, and visualized in real-time (Figure 8b), showing dynamic position updates and sensor readouts on a live floor-plan interface.

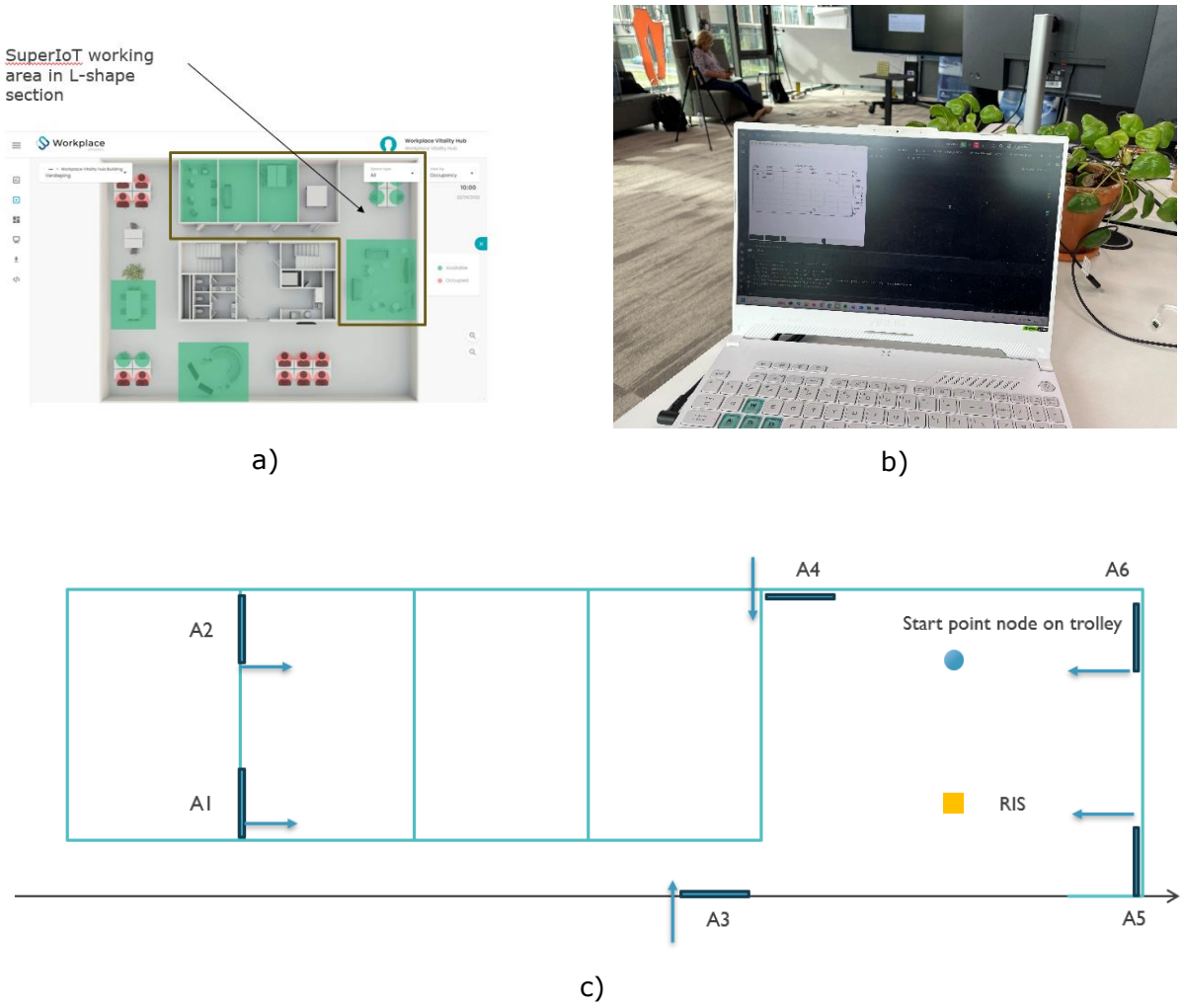


Figure 8. a) Layout of the Vitality Hub (demo located in top right corner); b) station displaying the real-time position data of the node moving on the trolley; and c) placement of the six (A1-A6) beacons, and the target node (blue dot) and RIS (yellow square).

Figure 9. SUPERIOT core node (left: front view, right: back view).

Figure 10. BLE channel sounding anchor (directional antenna, Nordic DK, and power bank).

The setup provided a robust and accurate localization framework, with sub-meter precision and seamless integration into the MQTT-based control and visualization infrastructure described below.

2.2.3 Visualization and monitoring interface

The visualization and monitoring interface provides an operator with a consolidated, real-time view of the demonstrator. It was developed to integrate seamlessly with the embedded RIS firmware and the BLE-based localization subsystem through the shared MQTT backbone. Its main purposes are to (i) display the BLE-based target position within a scaled map of the room, (ii) compute steering angles from localization data and publish them to the controller and (iii) display the resulting RIS state as a live 8×8 configuration grid alongside an indicative reflection/radiation view computed from the current steering geometry.

The interface is thus a middle-layer component between the MQTT broker and the RIS control software. It gathers information from three sources: 1) the localization system for real time position of the target node, 2) the RIS control state, i.e., the phase control pattern, and 3) direct inputs to the GUI itself, such as changing the feed horn position (which is propagated to the RIS firmware via MQTT also), and the position of the RIS tile itself within the region of interest. With this information, the interface publishes steering commands to the RIS firmware, implementing real-time beam tracking of a target IoT node. Furthermore, it serves as a live graphical representation of the RIS operation for experimental and dissemination purposes.

2.2.3.1 MQTT Messaging

The interface operates as an MQTT client to both consume live telemetry and publish control messages, connecting to the remote broker. Payloads rely on lightweight ASCII and JSON payloads for interoperability. The localization system can publish the location of more than one target node, and thus this monitoring layer can be configured to fetch one of the possible nodes for tracking. In this demo, tracking of a single node was presented. The MQTT topics employed by the monitoring layer are as follows:

- **Subscriptions** (inputs)
  - *"demo1/imec/#"* — BLE position events in JSON. The parser accepts messages containing at least {"x": <cm>, "y": <cm>}, optionally {"device": <string>}. A device filter can be applied so that only one named tag drives the UI; otherwise the first valid payload is used. On receipt, the GUI updates the BLE marker and flags the map for re-draw.
  - *"beamform/output"* — RIS configuration as a stream of ASCII '0'/'1' characters. The payload is reshaped into an 8×8 logical matrix (row×col) and rendered in the RIS grid. Reception also triggers a map and radiation refresh.
- **Publications** (outputs)
  - *"beamform/steer"* — Steering command as ASCII: theta=<deg>,phi=<deg>. The GUI computes θ, φ from the BLE position transformed into the RIS local frame and republishes only if either angle changes by more than a small epsilon in order to minimize traffic to significant positions updates only. θ is clamped to [0, 180] and φ wrapped to [0, 360) to match the downstream controller's expectations.

If the MQTT broker becomes unreachable, the GUI gracefully degrades to an offline visualizer, displaying the last known data or empty placeholders, and will automatically resume online operation once connectivity is reestablished.

2.2.3.2 Implementation

The GUI (Figure 11) is implemented in MATLAB, using the HG2 UI stack (*figure*, *uicontrol*, and *axes* elements). This choice ensured rapid development and tight integration with existing

MATLAB toolboxes for 3-D visualization and geometric transformation. The interface comprises a left-hand Map View and a right-hand stack with RIS Configuration, Radiation View, and Pose/Publish Controls. All spatial quantities are expressed in centimeters.

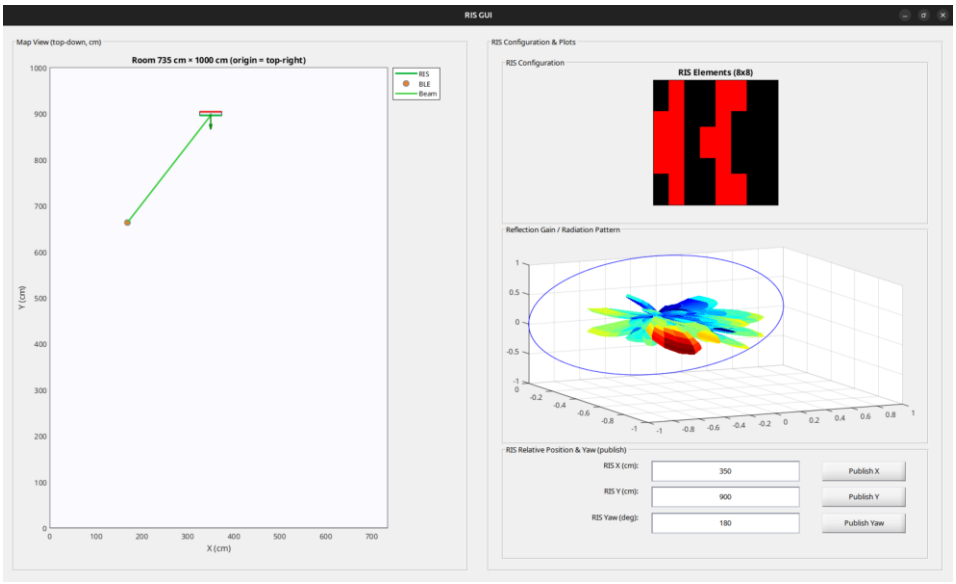


Figure 11. Example Graphical Interface Visualization.

Table 3. Components of the GUI of the monitoring layer

GUI Component	Description
Map View	The plan represents a top-down scaled room with real dimensions of 735 × 1000 cm room. A light-blue polygon shows the room's boundary; axes are in centimeters. The calculations involving the map assume a top-right origin (x increases right to left on screen, y increases top to bottom, despite axis labels). The RIS tile is rendered to scale (48 × 8 cm) with front/back edges coloured (front = green, back = red) and a normal arrow indicating the current yaw. A RIS local frame, used for computation of relative positions between RIS and target node, is defined at the tile's center with x = right when facing the RIS, y = down, and z = outward surface normal. The conversion from Cartesian to spherical coordinates used for beam steering follows the standard relations: $x = r \sin\theta \cos\varphi$, $y = r \sin\theta \sin\varphi$, $z = r \cos\theta$, with θ ∈ [0°, 180°] and φ ∈ [0°, 360°). The BLE node appears as a filled marker. A beam line joins the RIS front-edge midpoint to the BLE marker, visually indicating the intended steering direction in the horizontal plane. Whenever positions or pose change, the GUI also computes the BLE coordinates in the RIS frame and refreshes dependent views.
RIS Configuration Grid	A compact 8x8 matrix plot displaying per-element states from the most recent *"beamform/output"* payload. The colormap is binary (0 = black, 1 = red) sine the RIS employs 1-bit quantization. The plot is updated locally (i.e., without need to refresh the entire GUI) as messages are received. If the incoming matrix dimensions change, the plot reinitializes to accommodate the new size, supporting visualization of tiles with different dimensions.
Radiation View (3D)	A 3D surface presents an indicative reflection gain pattern based on the applied phase control. The surface is normalized and color-mapped to aid interpretation; a reference ring lies on the RIS plane (z = 0). The camera is set to a stable azimuth/elevation for readability (≈ −25° / +20°). This is a model-based visualization meant to support intuitive interpretation rather than provide a calibrated measurement.
Pose and Publish Controls	Three fields control the RIS pose used by all transformations: X (cm), Y (cm), and Yaw (deg). Each field has a Publish button that sends a complete pose snapshot to *"room/ris/offset"*. Editing any field immediately updates the on-screen tile and beam. Steering is typically auto-published by the GUI as the BLE target moves; pose updates are manual to avoid accidental shifts of the logical

anchor.

On each map refresh, the GUI (1) transforms the BLE position from room to RIS frame, (2) converts it to spherical coordinates (θ, φ), (3) updates the radiation view, and (4) publishes new steering angles when the change exceeds a small epsilon.

Beyond its operational role, the interface was designed as an intuitive visualization tool for demonstrations. Through its integration with the common MQTT infrastructure and its real-time graphical feedback, the GUI effectively bridges the embedded hardware layer and the system-level logic, validating the concept of adaptive, software-defined radio environments envisioned in SUPERIOT.

2.2.4 Demonstrator Deployment

The final deployment of Demonstrator 4 was carried out at imec's Vitality Hub in Eindhoven, where the demonstrator was integrated with the existing SUPERIOT testbed network. The objective of the deployment was to validate the interoperability between the RIS controller, the BLE localization subsystem, and the MQTT-based orchestration layer in a real indoor environment representative of a multi-technology IoT space.

2.2.4.1 Setup

The floorplan of the Vitality Hub is shown again in Figure 12, where RIS reflector demonstrator was placed, specifically within the sub-area at the top-right of the floor plan. Therefore, our target node would be able to move within an area of approximately 14.73m x 7.35m.

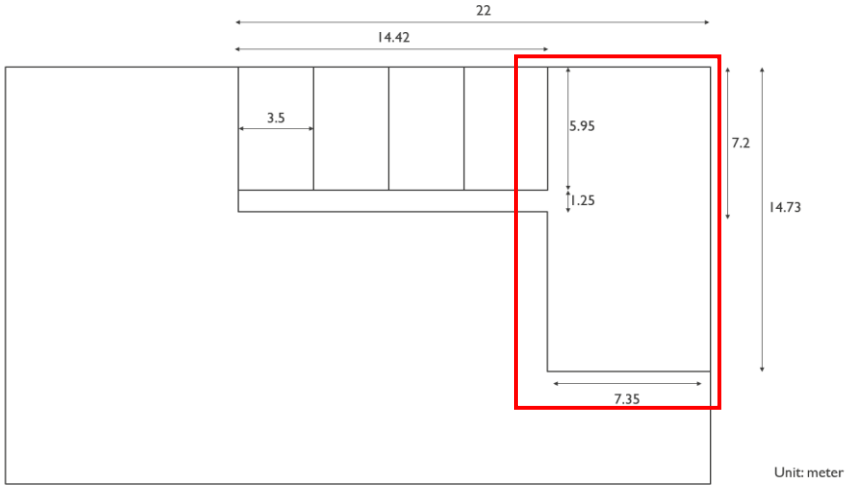


Figure 12. Vitality Hub area with annotated dimensions.

The showroom provides a semi-controlled environment, with arbitrary movement of people and other wireless devices, including our target IoT node. The localization anchors were readily installed on-site, and MQTT broker was available remotely. For the purpose of this demonstrator, the 8 × 8 RIS tile and control board were mounted on a table in the demonstrator booth, approximately at the top center of the highlighted area, and oriented perpendicular to the floor.

Power and data were supplied via a single USB-C link, and the ESP32-S3's integrated Wi-Fi interface provided direct connection to the broker through the local access point. Before demonstration, a short calibration sequence was executed to: 1) confirm correct orientation and coordinate alignment between the room frame and RIS frame, 2) verify MQTT

connectivity, and 3) test the end-to-end command path by sending synthetic steering commands.

2.2.4.2 Operation

During the demonstration, the target node moved arbitrarily while the GUI displayed its real-time position and the corresponding steering beam computed by the system. Each position update triggered an automatic recalculation of the phase map, immediately reflected on steering. The audience could observe both the graphical beam motion in the GUI and the changing radiation plot of the beam, illustrating how the RIS dynamically "tracks" the target device.

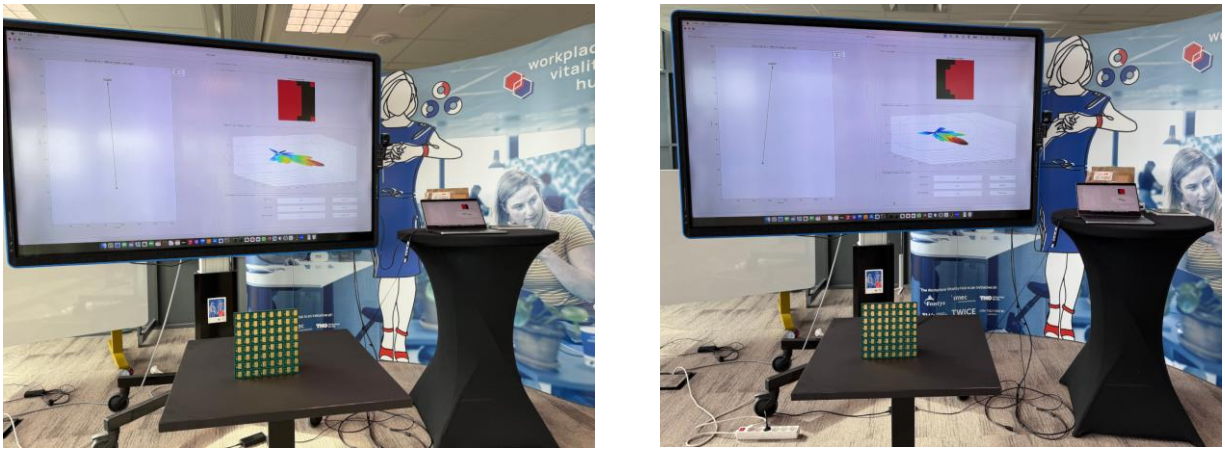


Figure 13. Deployed RIS tile and control software of SUPERIOT Demo 4.

Figure 13 illustrates two control states for the demonstrator, and respective estimated beam shape, position of the node, and phase pattern control state. The demonstration thereby validated: service integration via MQTT, on-chip calculation of phase control patterns in real-time, and status reporting for control and visualization.

2.2.4.3 Reproducibility

To reproduce the demonstrator in another environment, the following elements are required:

- A Wi-Fi network with MQTT broker accessible to both controller and host PC.
- A RIS implementation with any number of unit cells and central frequency, with firmware capable of exchanging and reacting to the MQTT messages outlined above. Due to the implementation of the GUI, the unit cells should be 2-state unit cells.
- Any kind of secondary system capable of determining the position of a target end-point (e.g., video tracking or UWB localization, etc), and of publishing the coordinates via MQTT.
- A middleware layer responsible for fetching target position, and computing the corresponding steering angle, which should be published also via MQTT, allowing for the RIS firmware to react. The visual component of this middle-layer is optional.

The above points require configuration of MQTT broker, and knowledge regarding the dimensions of the region of interest, coordinate systems, and internet connectivity for the RIS firmware and middleware layer.

2.3 Scenario 2: Vision-Assisted Closed-Loop Beam Steering

The goal of this scenario was to demonstrate how computer-vision systems can assist adaptive wireless communication through real-time RIS beam steering. Its purpose is to illustrate how a Reconfigurable Intelligent Surface (RIS) can dynamically redirect an incoming RF beam toward a moving human target identified and tracked within the environment. In contrast with Scenario 1, where target localization is provided by a BLE-based positioning system, this scenario relies on vision-based sensing and closed-loop RF measurements to validate beam steering experimentally.

This scenario builds upon the same RIS hardware and embedded software platform described previously, including the 6.5 GHz RIS tile, ESP32-S3 control board, and MQTT-based communication infrastructure. The demonstrator further integrates a ZED 2 AI stereo camera connected to a Jetson Orin Nano, enabling real-time target detection and localization through depth-sensitive computer vision. Steering information is distributed through MQTT to both the RIS controller and a MATLAB-based beamforming visualization interface, allowing the reflected beam direction and corresponding RIS phase pattern to adapt dynamically in real time.

The scenario was presented as a "Show and Tell Demo" at ICASSP 2026 in Barcelona. It demonstrates how a Reconfigurable Intelligent Surface (RIS) can dynamically redirect an incoming RF beam toward a target identified through computer vision. The demonstrator combines sensing, embedded beamforming control, visualization software, and RF instrumentation into a unified closed-loop adaptive system.

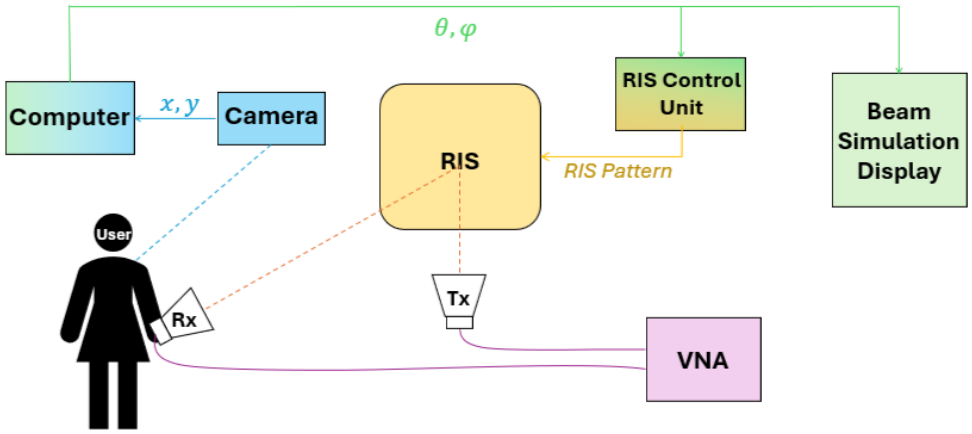


Figure 14. Architecture and data flow of the vision-assisted RIS beam steering scenario, including MQTT-based control and S21 validation through the VNA and horn antennas.

As illustrated in Figure 14, the setup is centred around the 8 × 8 RIS tile and associated control board, mounted vertically on a 3D-printed support structure. The RIS was configured to automatically adjust its reflective properties according to the real-time position of a tracked user. To achieve this tracking, the demonstrator exploits a vision-based localization subsystem composed of a StereoLabs ZED stereo camera connected to a Jetson Orin Nano. The depth-sensitive camera, positioned below the RIS, continuously monitors the environment and a lightweight YOLO-based detection algorithm continuously identifies users within the scene. The detected target coordinates are processed by the Jetson, which computes the corresponding steering angles (θ, φ) required to redirect the reflected beam toward the selected user. These steering coordinates are transmitted via MQTT to the RIS control unit which computes and applies the corresponding phase pattern to the tile. The resulting phase pattern is then communicated to a second computer running a MATLAB-based beam simulation interface to display the RIS configuration and associated 3D radiation pattern. All communication between the demonstrator components occurs through MQTT over a local wireless access point, with the Jetson computer simultaneously hosting the MQTT broker.

To validate beam steering experimentally, the RIS is illuminated by a transmitting horn antenna connected to a Keysight FieldFox N9914A handheld VNA, while a second horn antenna positioned near the tracked user acts as the receiver and closes the RF measurement loop. Variations in the measured S21 response provide direct confirmation of successful beam steering as the selected target moves within the scene, illustrating how the reflected beam direction and corresponding RIS phase pattern adapt dynamically in real time.

The resulting proof of concept highlights the potential of RIS technology as an intelligent adaptive component for future sensing-assisted and dynamically managed wireless environments. The following sections describe the architecture and data flow of the scenario, followed by the visualization software and demonstrator deployment and operation.

2.3.1 Architecture and Dataflow

The scenario demonstrates how an incident signal, generated by a transmitting horn antenna, can be dynamically redirected by the RIS toward a target identified through computer vision. The target moves freely within the region of interest, while a depth-sensitive stereo camera continuously estimates the target position in real time. The resulting system combines computer vision, embedded beamforming control, visualization software, and RF instrumentation into a closed-loop adaptive communication environment.

This demonstration scenario integrates four main functional blocks:

- **The RIS hardware and embedded software**, described in Section 2.1, consisting of a single 8×8 RIS tile operating at 6.5 GHz and controlled through an ESP32-S3-based control board. The embedded software retrieves steering commands via MQTT, computes the corresponding phase-control pattern, and reconfigures the RIS in real time.
- **The vision-based localization subsystem**, implemented using a ZED 2 AI stereo camera connected to a Jetson Orin Nano. The Jetson executes the target detection and tracking algorithms, continuously estimating the position of users within the field of view. When multiple users are detected, the closest target is selected automatically. The corresponding steering coordinates are then published to the MQTT broker.
- **A beamforming visualization interface**, implemented in MATLAB, which displays both the RIS phase configuration and the estimated 3D radiation pattern. The interface subscribes to steering and RIS-state information via MQTT, providing live visualization of the expected beam direction and phase distribution. The RIS tile itself also provides direct visualization through per-element LEDs indicating the active phase state.
- **A closed-loop RF measurement subsystem**, composed of a Keysight FieldFox N9914A handheld VNA and two horn antennas. One horn statically illuminates the RIS, while the second horn, positioned near the tracked user and representing the receiver location, measures the received signal power. Variations in the measured S21 parameter provide experimental validation of the beam steering operation.

Together, these components demonstrate how sensing, embedded beamforming, visualization, and RF instrumentation can be combined to realize a closed-loop adaptive radio environment. The overall operation of the demonstrator follows this sequence of steps:

1. The Jetson Orin Nano receives image and depth information from the ZED 2 camera and detects the target user within the scene;
2. The position of the selected target is converted into steering coordinates and published to the MQTT broker;

3. The RIS controller subscribes to these updates, computes the corresponding reflection phase map, and updates the RIS configuration accordingly;
4. The MATLAB visualization interface receives the same steering information and displays both the RIS phase pattern and estimated 3D beam shape;
5. The VNA continuously measures the S21 response between the transmitting and receiving horn antennas, allowing the effect of beam steering to be observed experimentally as the target moves;
6. Steps 1–5 repeat continuously in real time.

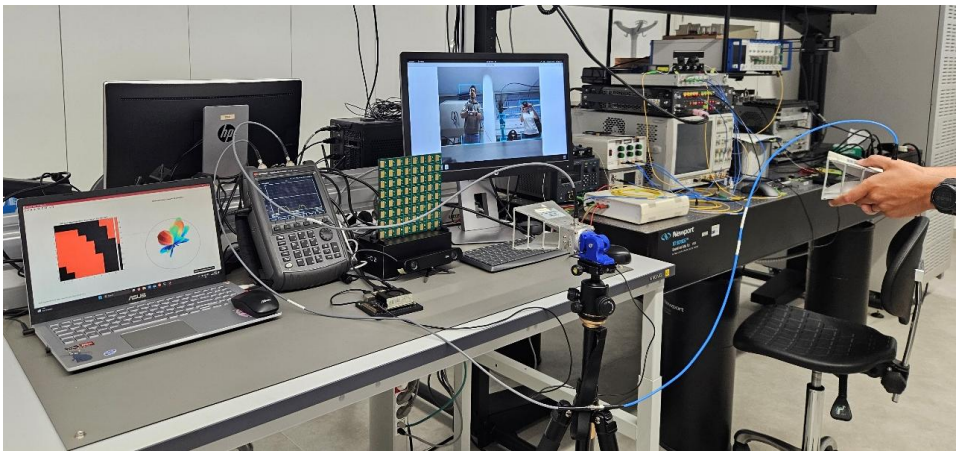


Figure 15. Displays the laboratory demonstrator setup – the camera and RIS are positioned to align their respective frames of reference as closely as possible.

Figure 15 shows a photograph of the demo setup being validated at INESC TEC's communications laboratory. The setup includes the 8 × 8 RIS tile and control board, the StereoLabs ZED camera aligned with the RIS frame of reference, the MATLAB beamforming visualization interface, and the Keysight FieldFox VNA with transmitting and receiving horn antennas used for closed-loop S21 validation.

2.3.2 Visualization and monitoring interface

The visualization and monitoring interface used in this scenario, shown in Figure 16, is conceptually similar to the one described for Scenario 1, but operates in a simplified and fully passive manner. In this case, the interface does not participate in the steering-control loop itself and is not required for operation of the RIS demonstrator. Instead, it acts solely as a visualization tool for monitoring the current RIS configuration and the corresponding estimated beam pattern. This provides an intuitive representation of both the active RIS phase state and the expected reflected beam direction during operation, complementing the RF measurements obtained through the VNA measurement loop.

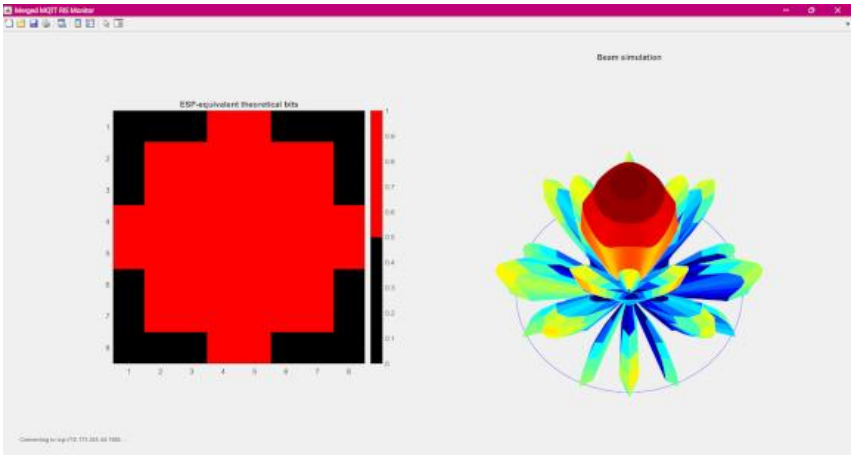


Figure 16. Passive display of the RIS steering status

2.3.2.1 MQTT Messaging

In this scenario, the visualization software subscribes exclusively to the `beamform/output` MQTT topic, which publishes the current RIS configuration as a stream of ASCII `'0'` and `'1'` characters. The received payload is reshaped into an 8 × 8 logical matrix corresponding to the RIS element states and rendered in the RIS configuration grid. Reception of a new payload also triggers an update of the beamforming and radiation-pattern visualizations.

2.3.3 Demonstrator Deployment

The final deployment of Scenario 2 was carried out as a standalone "Show and Tell" demonstrator in the ICASSP 2026 conference showroom, within a single exhibition booth. The objective of the deployment was to validate the complete closed-loop operation of the RIS beam steering system in a compact, self-contained setup, combining vision-based target detection, MQTT-based control, RIS reconfiguration, MATLAB visualization, and RF validation through the VNA measurement loop.

2.3.3.1 Setup

The demonstrator was deployed within a dedicated booth area in the ICASSP 2026 showroom, where users interacted with the setup by moving within the field of view of the ZED camera. Given the camera's aperture, the interaction region was approximately 1.5–2 meters in front of the RIS.

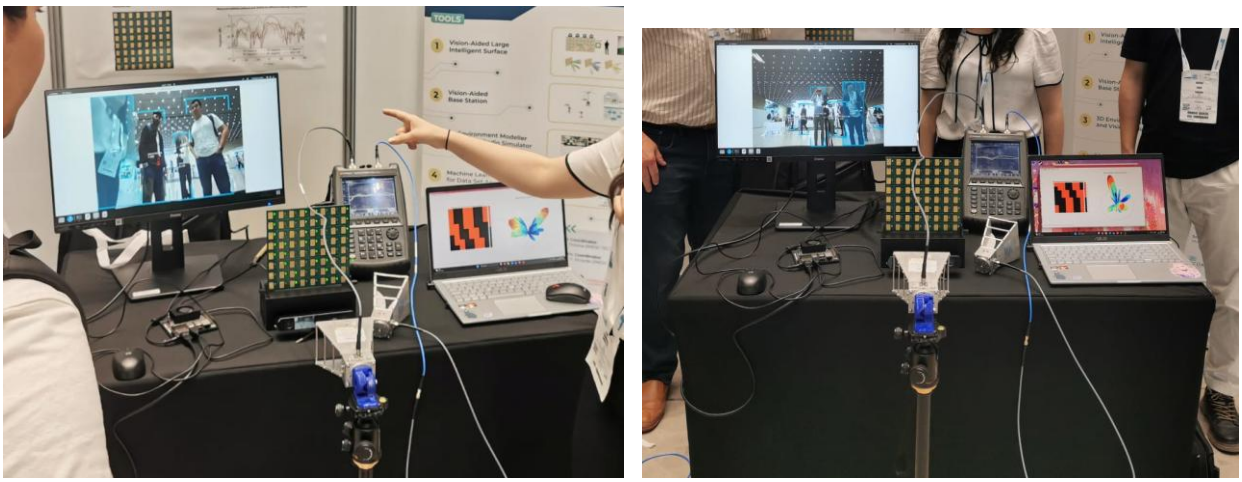


Figure 17. Vision-assisted RIS beam steering demonstrator showcased at ICASSP 2026.

As per the laboratory preparation, the 8 × 8 RIS tile and control board were mounted vertically on a 3D-printed support structure, with the camera positioned below the RIS to approximately align both frames of reference. A transmitting horn antenna illuminated the RIS, while a second horn antenna near the user position acted as the receiver for the VNA measurement loop. The ESP32-S3 controller connected to the local MQTT broker hosted on the Jetson Orin Nano through its integrated Wi-Fi interface. Prior to operation, a short calibration sequence verified camera alignment, MQTT connectivity, and end-to-end RIS steering operation.

2.3.3.2 Operation

During operation, participants could observe themselves in the camera preview interface, where detected users were indicated through bounding boxes and the currently selected target was highlighted. Simultaneously, the RIS tile displayed the active phase configuration through its per-element LEDs, while the MATLAB visualization interface presented both the corresponding phase map and the estimated 3D radiation pattern.

The VNA continuously displayed the magnitude of the S21 parameter of the two-horn system over a frequency range of approximately 0.3 GHz centred around the operating frequency (6.12 GHz). When the selected user positioned the receiving horn approximately within the reflected beam direction and aimed it toward the RIS, a significant increase in received power could be observed. Conversely, noticeable reductions in received power occurred when the horn orientation or user position changed, or when another participant entered the scene and became the newly selected target of the reflected beam.

Figure 17 shows photographs of the demo booth, where the setup is shown to include the ZED camera and Jetson-based YOLO target detection interface (left display), the 8 × 8 RIS tile with embedded controller and per-element LED visualization (centre), the Keysight FieldFox VNA for closed-loop S21 monitoring, and the MATLAB beamforming interface (right display) showing the active RIS phase pattern and estimated 3D radiation pattern. The horn antenna mounted on the tripod is the feed antenna of the measurement loop.

2.3.3.3 Reproducibility

To reproduce the demonstrator in another environment, the following elements are required:

- A Wi-Fi network with an MQTT broker accessible to the RIS controller, visualization software, and vision-processing system.
- A RIS implementation with firmware capable of receiving steering commands via MQTT and applying the corresponding phase-control patterns.
- A vision-based localization system capable of detecting and tracking users in real time, and publishing steering coordinates or target positions through MQTT.
- (Optional) A visualization or middleware layer capable of displaying the RIS state and estimated beam pattern based on the received RIS configuration.

The demonstrator additionally requires RF instrumentation for closed-loop validation, namely a transmitting source illuminating the RIS and a receiving antenna connected to a VNA or equivalent measurement system. Proper alignment between the camera frame of reference, RIS orientation, and receiver position is also required to ensure consistent steering behaviour.

3 Towards Alternative Technologies for Scalability

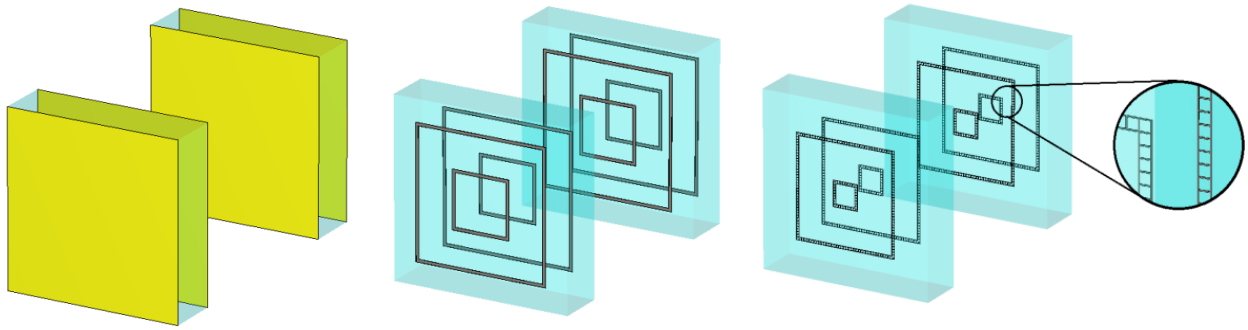
As part of ongoing efforts to explore scalable and sustainable alternatives to conventional printed circuit board–based RIS implementations, we also explored 1) glass-based metasurface structures that combine radio transparency with optical clarity, and 2) on the use of memristors as switching elements, allowing for continuous low-power control versus other devices such as PIN diodes or varactors.

3.1 Glass Substrates for Transmissive Transparent Metasurfaces

Glass surfaces commonly used in architectural designs cause strong attenuation of radio signals. The proposed concept transforms such glass elements into active electromagnetic components, embedding a transmissive metasurface directly into standard double-glazed windows. This preserves the purpose of glass surfaces, i.e., for windows and other fixtures allowing for natural light, while allowing for enhancing radio transmission and beam control for indoor connectivity.

3.1.1 Design

We targeted a design for an example frequency of 10 GHz, representative of mid-band 6G frequencies. The design employs two 4 mm glass substrates, separated by a 15 mm air cavity, emulating the physical characteristics of commercial double-glazed windows, as shown in Figure 18a. Each outer surface of the glass stack is patterned with a metallic mesh layer (Figure 18c), forming a 2-bit transmissive metasurface capable of four distinct phase states (−135°, −45°, +45°, +135°).



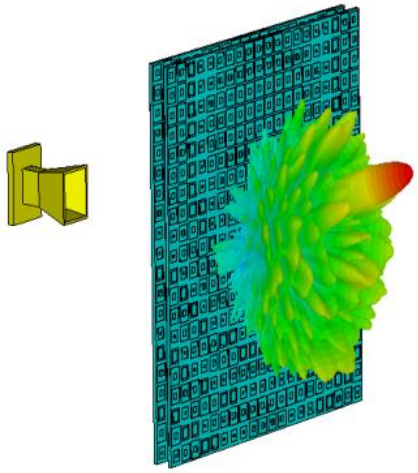
a) b) c)

Figure 18. a) Unit cell structure; b) Double square loop unit cell, and c) Meltal-mesh unit-cell (for $S21 = 45^{\circ}$).

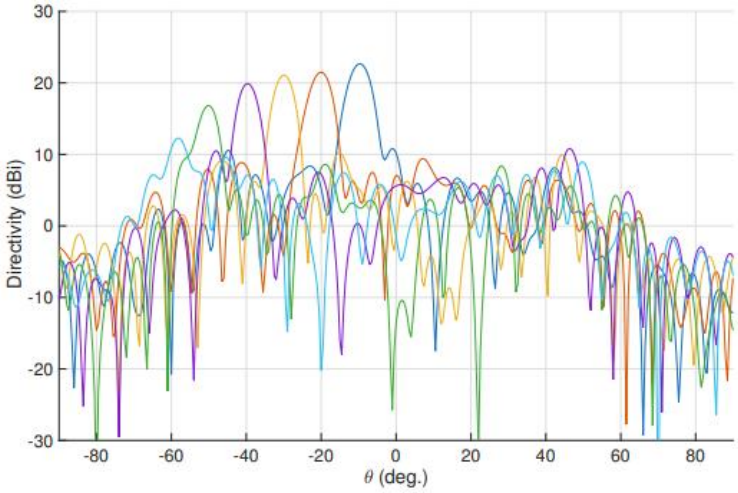
The structure was modeled analytically using the Multiple Reflection Method (MRM) and validated through full-wave electromagnetic simulations in CST Microwave Studio. The ABCD matrix formalism was applied to characterize multilayer propagation, while a double square loop unit cell served as the basic geometric element (Figure 18b). The unit cell's electromagnetic response was captured through an equivalent circuit model, allowing efficient mapping of impedance to geometry parameters.

3.1.2 Key Results

To validate the performance of the proposed structure, a 2-bit transmit array comprising 20 × 20 elements was designed and simulated. The array was illuminated by a 10 dBi reference horn antenna placed at a focal distance of 240 mm (F/D = 0.8) and aligned with the centre of



the metasurface to ensure uniform excitation. ilustrates the simulated transmit-array setup and the applied phase distribution corresponding to a 20°



steering direction, while presents the 2-D directivity patterns at 10 GHz for multiple beam directions (10°–60°). A summary of key metrics is as follows:

- The double-glass configuration achieved a full 360° transmission phase range with insertion losses below 1 dB for all phase states;
- Each unit cell maintains over 90% optical transparency (OT), with transparency ranging from 90.2% to 94.3% across the four phase states;
- The 2-bit metasurface array (20×20 elements) demonstrated efficient beam steering up to 60°, achieving a peak gain of 22.6 dBi and aperture efficiency of 14.4% at 10 GHz;
- The optically transparent metal-mesh pattern, with 25 μm line width and 0.2 mm spacing, provides a good trade-off between electrical conductivity and visual transparency, suitable for integration in real glass windows.

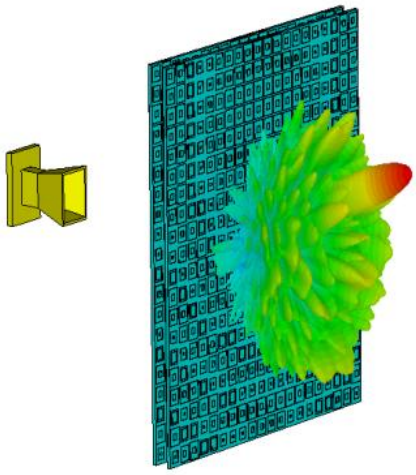


Figure 19. Transmit array setup with 20º beam direction

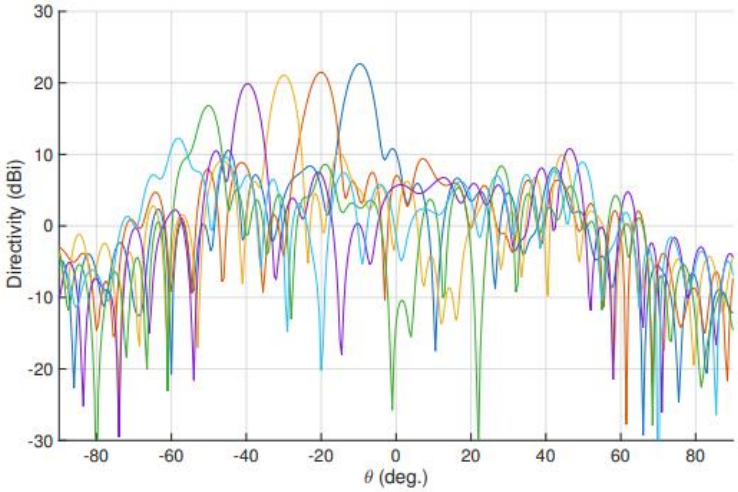


Figure 20. Directivity for different beam direction angles obtained in CST.

3.1.3 Conclusions

This approach represents a significant step toward scalable and architecturally compatible RIS technology. By using glass as both structural and electromagnetic substrate, the metasurface can be integrated into existing infrastructure, supporting the goals of sustainable, invisible connectivity. Future work will focus on switchable or tunable materials (e.g., printed memristors or transparent conductive oxides) to enable active beam control.

3.2 Memristor-Controlled Unit Cells for Low-Power Reconfigurable Surfaces

To further advance scalability and energy efficiency, ongoing work within SUPERIOT has investigated memristor-based switching elements as a replacement for conventional PIN diodes or varactors, which we have reported in a recently submitted paper for EuCAP 2026 entitled [2]. We explore how non-volatile memristors can drastically reduce control power and enable compact, multi-layer RIS designs compatible with future low-cost fabrication technologies.

Biasing element such as PIN diodes require continuous bias currents to maintain their state, leading to non-negligible static power consumption, which hinders scaling RIS panels into large arrays. Memristors provide a non-volatile resistive switching mechanism. Once programmed to either a high-resistance (OFF) or low-resistance (ON) state, a memristor retains its state without any continuous biasing power being applied. This evidently has the potential to enable RIS designs at large scale with much lower energy consumption.

3.2.1 Design

The proposed RIS unit cell uses a memristor as a reconfigurable element. This memristor consists of MoS_2 (molybdenum disulfide) sandwiched between Cr/Au and Ti/Au electrodes (Figure 21a). When short DC pulses of 0–2 V are applied, conductive filaments form or dissolve within the dielectric, switching the device between two resistive states. The memristor has been characterized in terms of S-parameters at frequencies up to 10 GHz. The characterization demonstrates an insertion loss < 1 dB in the low-resistance state (LRS) and an isolation >30 dB in the high-resistance state. The intrinsic parameters are ON-Resistance RON = 6.5Ω, OFF-Resistance ROFF = 14.1KΩ and-OFF Capacitance COFF = 4.3 fF.

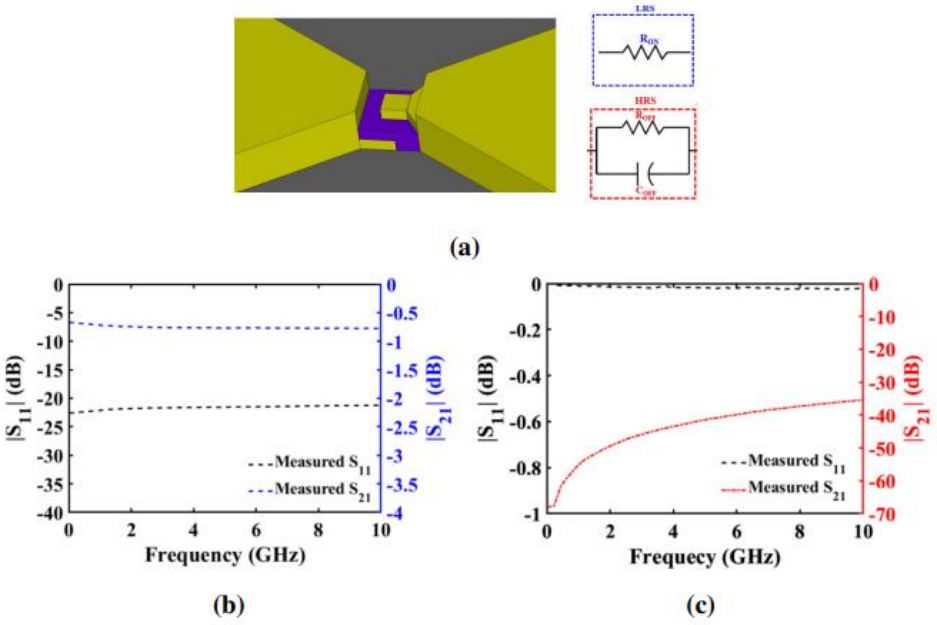


Figure 21. Memristor (a) The 3D structure and its equivalent circuit, (b) and (c) RF performance of the fabricated memristors.

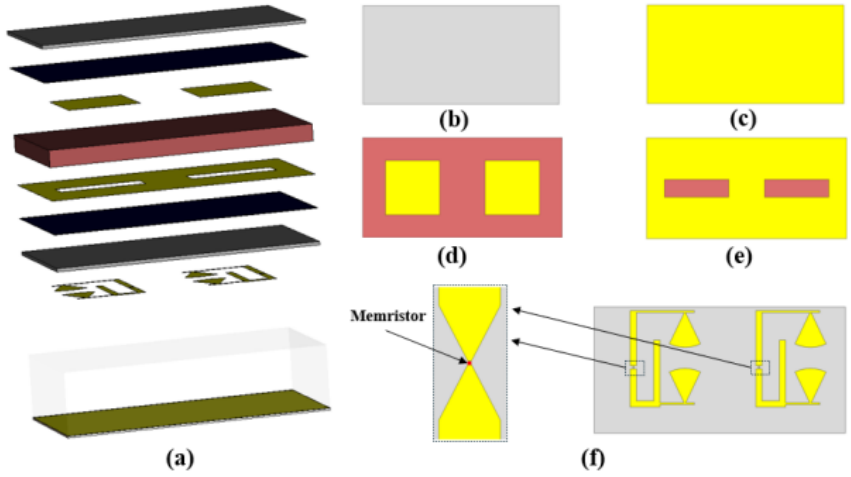


Figure 22. Stack-up layers of the 2 RIS-UCs (a) 3D structure (b) Top view (c) Bottom view (d) Square patches (e) Slotted ground plane layer (f) Active delay line with memristor.

To interface the memristor with the RIS structure, a custom low-voltage driver circuit was designed using standard components. The circuit generates controlled ramp and pulse signals for the SET (LRS) and RESET (HRS) operations through a combination of a 555 timer, voltage-controlled modulator (LTC6992), and PMOS logic network.

The proposed RIS unit cell (Figure 22) adopts an aperture-coupled configuration based on 3 dielectric layers: two glass substrates S_1, S_3 (ε_r = 3.78, t = 0.5 mm) separated by Rogers substrate S_2 (ε_r = 2.1, t = 2.54 mm). The glass substrate provides surface smoothness and fabrication compatibility for memristors. A square patch is printed on the top of Rogers, while a ground plane, with an aperture, is located underneath. This allows the mutual coupling to the J-shaped delay line which is printed on the underside of the S_3 substrate.

3.2.2 Key Results

Full-wave electromagnetic simulations in CST Microwave Studio confirmed the expected behaviour of the memristor-controlled unit cell. The reflection coefficient and surface-current distributions (Figure 23), illustrate the phase-shifting mechanism. The unit cell exhibits a reflection loss < 1.3 dB across the 5.6–6.3 GHz band, and a phase difference Δφ ≈ 180° ± 20° between the two memristor states, across an operational bandwidth of approximately 0.67GHz (Figure 24). The RIS-UC performance is also confirmed using the waveguide simulator. The total energy required to toggle between states is approximately 330 pJ per transition. In contrast, a comparable PIN-diode-based RIS element consumes around 10 mW continuously and requires 36 J to maintain its ON state for one hour.

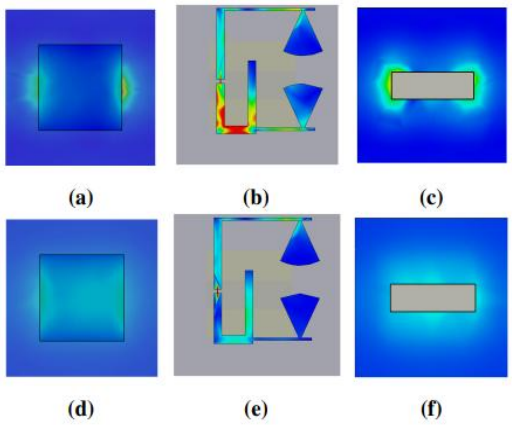


Figure 23. Current distribution over all layers at 6 GHz (up -> OFF state; down -> ON

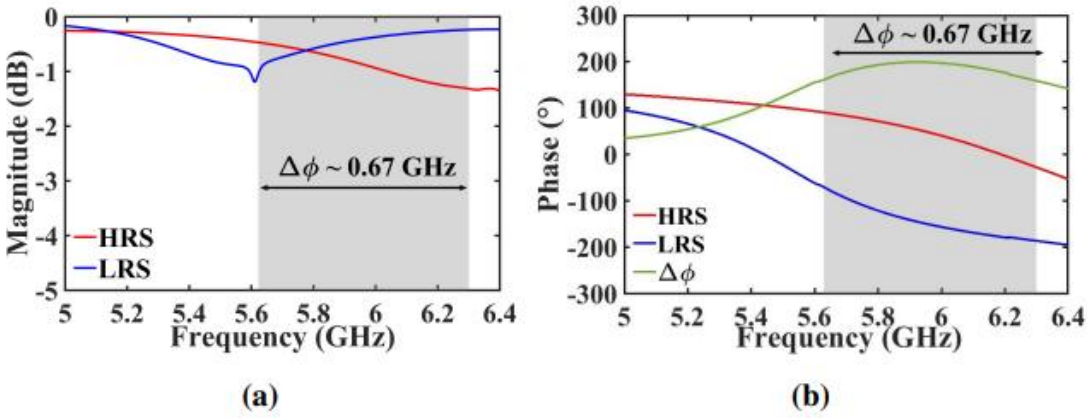


Figure 24. The PBC simulation results at normal incidence (θ = 0◦) (a) Reflection magnitude (b) Reflection Phase.

3.2.3 Conclusions

The presented memristor-based RIS unit cell demonstrates a feasible, low-power alternative to conventional electronically reconfigurable surfaces. Memristor-based switching can achieve the same electromagnetic performance as PIN-diode-controlled RIS elements while reducing power consumption. The combination of glass-compatible substrates, via-less fabrication, and non-volatile switching creates a platform suitable for scalable, sustainable, and hybrid integration. Future work will focus on extending the concept to multi-bit phase control and large-area

assemblies, as well as exploring transparent conductive oxides and printed memristor integration for fully sustainable smart-surface applications.

4 Usage of project components in Demonstrator 4

Demonstrator 4 integrates several technologies, algorithms, and hardware components developed throughout the SUPERIOT project into a unified RIS-assisted adaptive communication platform. In particular, the demonstrator employs:

- The MQTT-based interoperability framework and system architecture defined in WP1, enabling real-time communication and coordination between heterogeneous sensing, control, and visualization subsystems.
- The BLE-based localization system developed within WP3 for target-position estimation.
- The SUPERIOT core node developed in WP2, used as the IoT node within the BLE-based localization system.

Furthermore, although not directly integrated into the main demonstrator deployment, the present deliverable also reports additional RIS technologies aligned with the broader SUPERIOT technology portfolio and sustainability vision, namely:

- The memristor devices developed within WP2, which were considered as part of the proposed memristor-based RIS concept.
- The optically transparent metal-mesh structures considered for the proposed glass-based transmissive metasurface concept, whose characteristics were derived from WP2 printed-electronics developments.

These additional technologies establish a direct link with the sustainability methodologies and analyses reported in Deliverable D1.5, particularly regarding sustainable electronics, printed technologies, and low-power IoT infrastructure concepts.

5 Conclusions

The Large-area IoT Node Demonstrator successfully validated the integration of reconfigurable radio hardware, embedded control, and system-level software within the SUPERIOT IoT architecture. The work demonstrated that a single-tile RIS operating at 6.5 GHz can be dynamically reconfigured in real time based on localization data and MQTT-based coordination with other IoT subsystems.

The combination of embedded beamforming computation, low-latency message exchange, and real-time visualization illustrates the potential of RIS technology as a flexible and intelligent communication component. Demonstrator 4 confirmed the interoperability of the SUPERIOT architecture, where different sensing and communication technologies (BLE, RF, and control middleware) coexist under a unified protocol framework.

In parallel, the exploratory work lays out the groundwork for the next generation of RIS hardware, where sustainability and scalability are key drivers. The glass-based transmissive metasurface concept demonstrates the feasibility of embedding RIS functionality into architectural materials, while the memristor-based reflective cell establishes a pathway toward non-volatile, ultra-low-power operation.

The demonstrator validates a functional system prototype and provides a platform for future research on integrated reconfigurable surfaces, and the fundamental research on unit cell substrates and semiconductor devices for low cost and low power unit cell control contribute towards technologies that merge radio, optical, and printed-electronic domains.

6 Bibliography

[1] N. Paulino, F. M. Ribeiro, L. Outeiro, P. A. Lopes, S. Inácio and L. M. Pessoa, "Design and Implementation of Scalable 6.5 GHz Reconfigurable Intelligent Surface for Wi-Fi 6E," 2025 19th European Conference on Antennas and Propagation (EuCAP), Stockholm, Sweden, 2025, pp. 1-5, doi: 10.23919/EuCAP63536.2025.10999788.

[2] Ghatas et al., "Reflective Intelligent Surface Unit Cell Based on Memristor for Mid-Band 5G Communications, 2026, EuCAP 2026 20th European Conference on Antennas and Propagation (Submitted for review)

7 List of figures

8 List of tables

9 List of contributors

Contributors	**Short name**	**Country**
INESC TEC - INSTITUTO DE ENGENHARIADE SISTEMAS E COMPUTADORES, TECNOLOGIA E CIENCIA	INESC TEC	Portugal
STICHTING IMEC NEDERLAND	IMEC-NL	Netherlands