Metasurface Mask Designer (MetaMaDe) Manual

This application allows users to explore the behavior of Reconfigurable Intelligent Surfaces (RISs) in both Far Field and Near Field configurations. The users can adjust the angles of incidence and reflection, choose whether to apply randomization or not, and visualize key outputs such as phase mask colourmaps, beam patterns, and vector fields. In addition, the users can design codebooks, containing multiple phase masks. This feature allows for spatial scanning in both near field and far field, using either the mmWave or Sub-6 RIS models, that are currently available at the THz Lab of Arizona State University (ASU).

FAR FIELD SIMULATION

This section of the MetaMaDe application demonstrates the redirection feature of the RIS. In this case, both the Base Station (BS) and the User Equipment (UE) are located at the far field region of the metasurface, meaning that the incident on the RIS Electromagnetic (EM) wave, as well as the received from the UE EM wave are considered planar (constant phase and amplitude of the wave front).

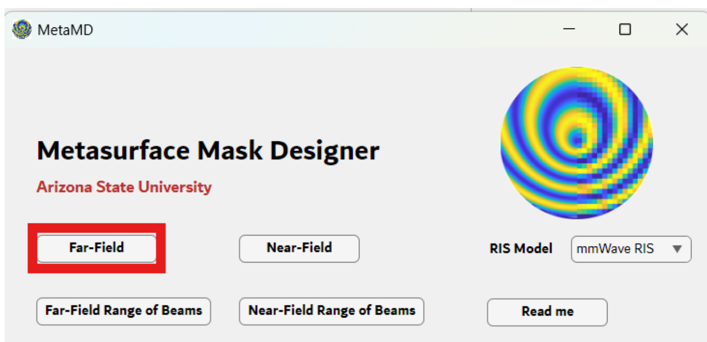

Figure 1: MetaMaDe app homepage (Far-Field)

The user can define the elevation (θ_i) and azimuth (ϕ_i) angles of the incident on the RIS EM wave and choose the elevation (θ_r) and azimuth (ϕ_r) angles of reflection from the left panel, as shown in figure 3. The aforementioned angles correspond to the coordinate system shown in figure 2.

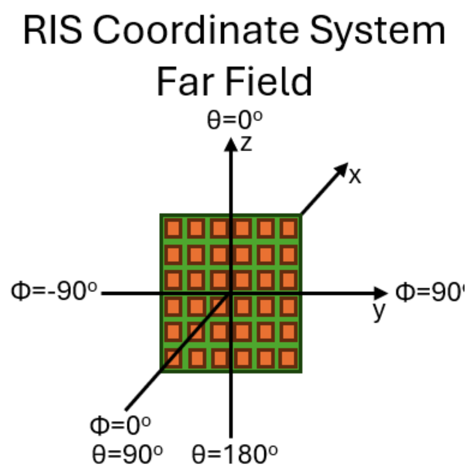

Figure 2: The coordinate system used to define the angles of incidence and angles of reflection for the Far Field Modes.

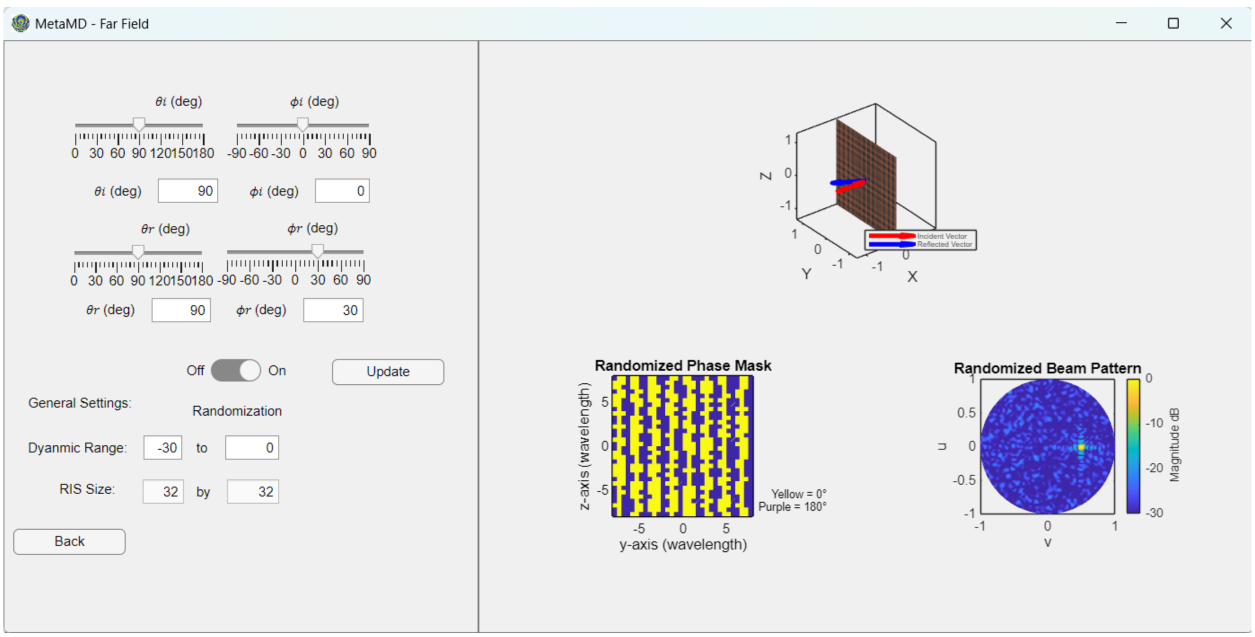

Figure 3: Far Field Mode.

Moreover, the user can choose whether to use the pattern of randomized phases or not. These randomized phases are utilized to suppress the quantization lobe, which appears symmetrically to the main beam due to the binary nature of the phase mask. Finally, the Dynamic range refers to the U-V plot, which will appear at the right panel after the completion of the simulation.

After pressing the update button, the results of the simulation will appear on the right panel of the window. The results include a visualization of the vectors of the incident (red arrow) and reflected (blue arrow) EM waves, a colourmap of the phase mask that needs to be applied, in order to redirect the wave to the requested direction and finally a U-V plot. The U-V plot presents the radiation pattern of the reflected wave from the point of view of someone looking towards the metasurface.

This mode is only for demonstration purposes and cannot be used for the generation of a codebook. The Range of Beams mode, which is explained in the following section, was designed to cover this need.

FAR-FIELD RANGE OF BEAMS SIMULATION

This section of the MetaMaDe application was designed to generate a set of different phase masks (codebook), in .txt format, which in turn can be copied and pasted into the C-code provided by the THz Lab, which controls the microcontroller, that is attached to the RIS motherboard.

Similarly to the Far Field simulation, both the BS and UE are placed at the far field region of the metasurface.

As shown in figure 5, the user can define the elevation (θ_i) and azimuth (ϕ_i) angles of the incident, on the RIS, EM wave. The difference with the Far Field mode is that here the user can simulate a set of different elevation (θ_r) and azimuth (ϕ_r) angles of reflection, by setting a starting angle, the step and the final angle. The angles correspond to the coordinate system shown in figure 2.

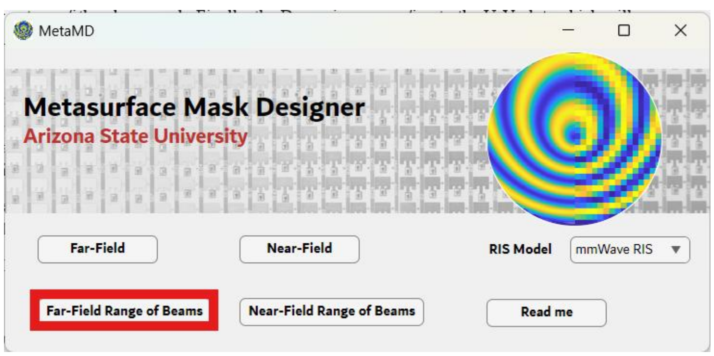

Figure 4: MetaMaDe app homepage (Far-Field Range of Beams).

Moreover, the user can choose whether to use the pattern of randomized phases or not. These randomized phases are utilized to suppress the quantization lobe, which appears symmetrically to the main beam due to the binary nature of the phase mask. Finally, the Dynamic range refers to the U-V plot, which will appear at the right panel after the completion of the simulation.

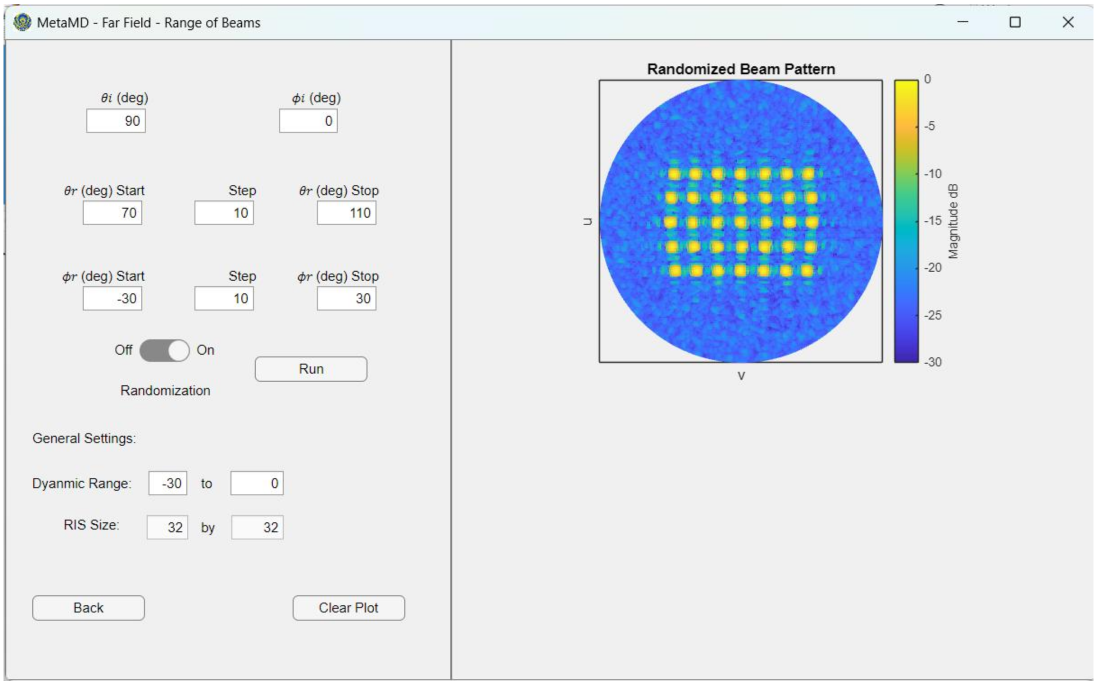

Figure 5: Far-Field Range of Beams Mode.

After pressing the Run button, the U-V plot appears on the right panel of the window, showing the simulated beams from the point of view of someone looking towards the RIS. Moreover, a .txt file will appear on the window, where the codebook is stored.

Notes:

- **The step should always be greater than zero.**
- **Simulating large number of angles at once can slow down the process.**
- **The randomization option should remain on, when the codebook is going to be used for the mmWave RIS.**
- **The U-V plot should be cleared using the clear plot button, before each simulation.**
- **Each time the update button is pressed, a window appears, prompting the user to store the .txt file. The file can be stored anywhere, <u>except</u> for the Program Files folder.**

To utilize the generated codebook, using the C-Code provided by the THz Lab, Copy and paste the generated codewords from the .txt file to the microcontroller's code inside the **uint8_t bitList[][256]** vector, under the **//Mask 0 Dummy** codeword.

NEAR FIELD SIMULATION

This section of the MetaMaDe application demonstrates the near field focusing feature of the RIS. In this case, both the Source of Illumination (SI) and the Focal Point (FP) are located at the near field region of the metasurface, meaning that the phase and the amplitude of the wavefront, of the incident on the RIS Electromagnetic (EM) wave are not constant. Thus, the phase mask needs to be calculated, taking into consideration the power contribution, to the focal point, of each element, of the RIS, separately.

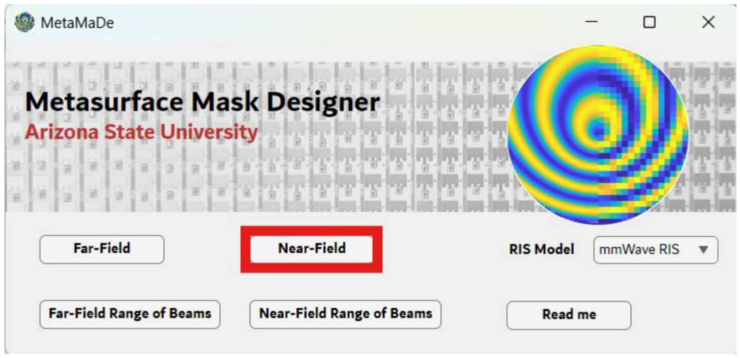

Figure 6: MetaMaDe app homepage (Near-Field)

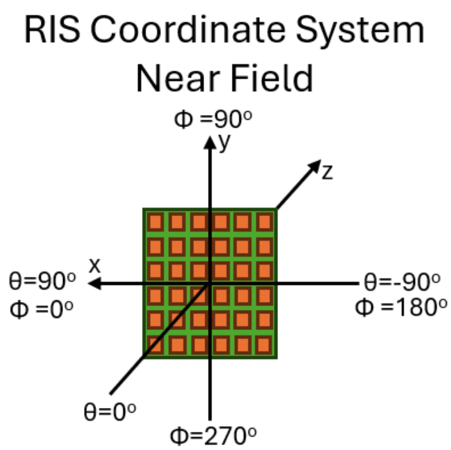

Figure 7: The coordinate system used to define the angles of incidence and angles of reflection for the Near Field Modes.

The user can input the exact coordinates of the IS (x_i, y_i, z_i) and the FP (x_r, y_r, z_r) in mm and notice the azimuth and elevation (θ_i, ϕ_i) and (θ_r, ϕ_r) of the incident and reflection angles respectively. There is also the option to either use or not the randomization scheme. The Z-Cut variable defines the distance from the RIS that the X-Y plane radiation plot is located. Finally, the user can set the dynamic range for the Illumination Magnitude, X-Y Cut, and Y-Z Cut plots.

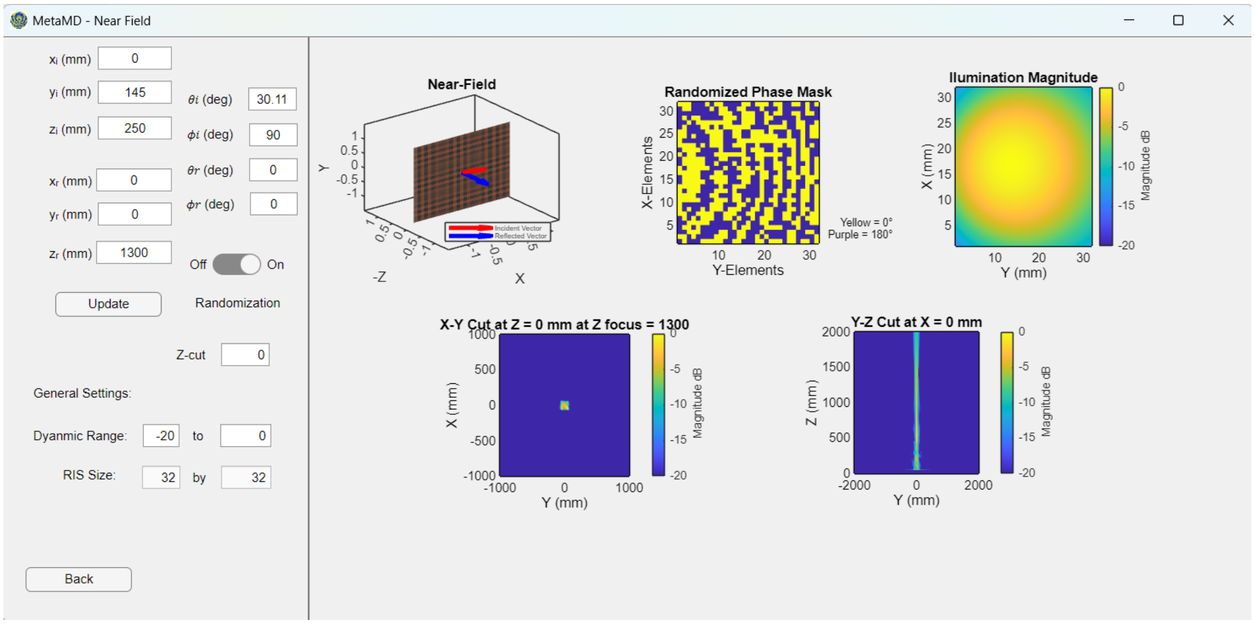

Figure 8: Near-Field Mode.

After pressing the update button, the results of the simulation will appear on the right panel of the window. The results include a visualization of the vectors of the incident (red arrow) and reflected (blue arrow) EM waves, a colourmap of the phase mask that needs to be applied, in order to focus the wave on the requested focal point. The illumination magnitude plot refers to the amplitude distribution of the wavefront of the incident wave on the surface of the RIS. The X-Y Cut plot presents a vertical slice showing the wave behavior on the xy plane in different distances z-cut in front of the RIS. The Y-Z cut plot presents a horizontal slice showing the behavior of the reflected wave along the axis of propagation.

This mode is only for demonstration purposes and cannot be used for the generation of a codebook. The Range of Beams mode, which is explained in the following section, was designed to cover this need.

NEAR-FIELD RANGE OF BEAMS SIMULATION

This section of the MetaMaDe application was designed to generate a set of different phase masks (codebook), in .txt format, which in turn can be copied and pasted into the C-code provided by the THz Lab, that controls the microcontroller, that is attached to the RIS motherboard.

Similarly to the Near Field simulation, both the IS and FP are located at the near field region of the metasurface.

As shown in figure 10, the user can define the exact coordinates of the IS (x_i, y_i, z_i). In order to create the codebook for a set of different FPs the user needs to define the coordinates (x_r, y_r) of the starting point and those of the final point and choose the scanning resolution by appropriately defining a step.

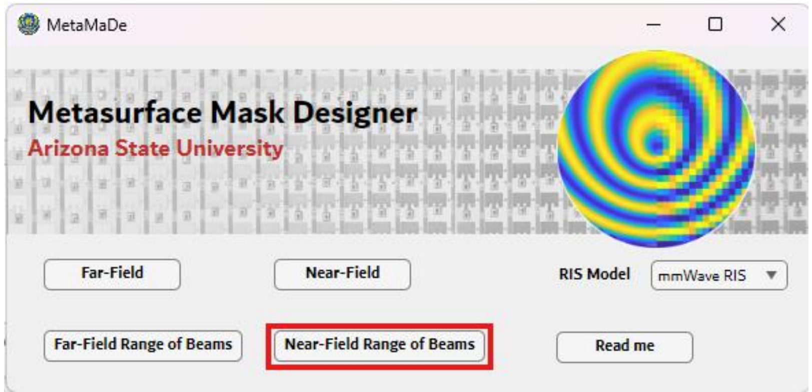

Figure 9: MetaMaDe app homepage (Near-Field Range of Beams)

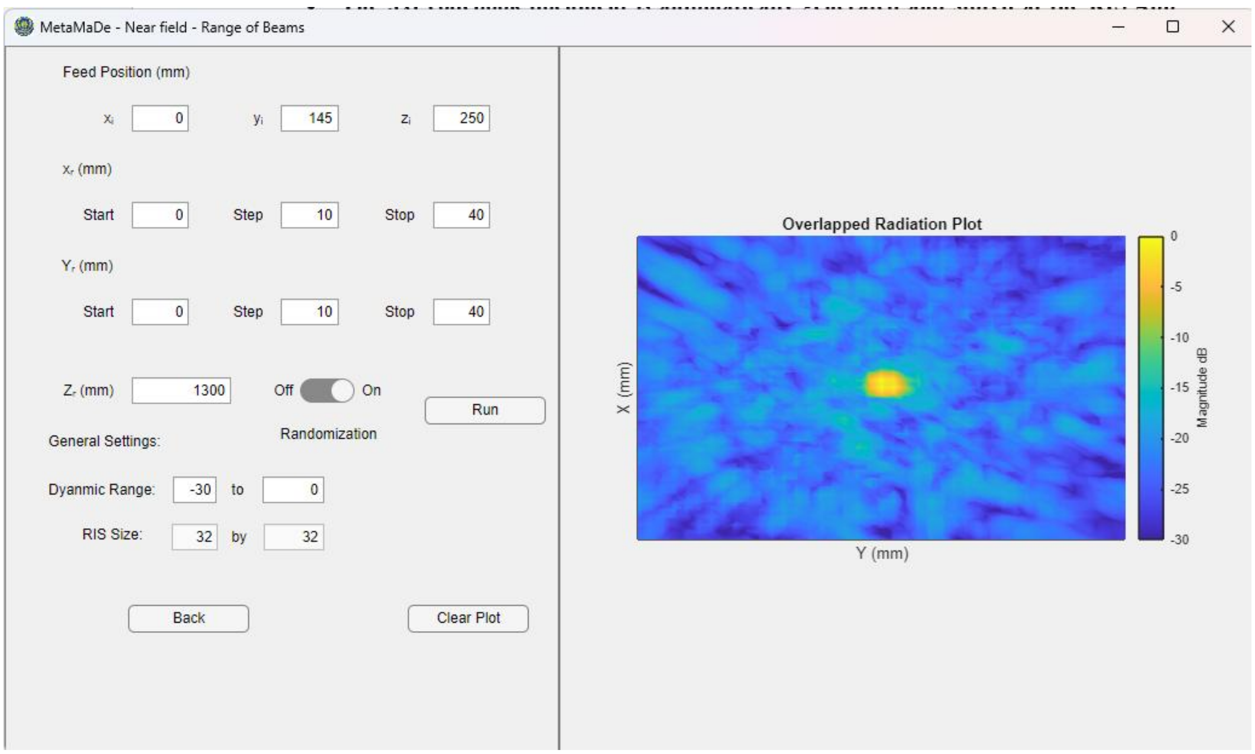

Figure 10: Near-Field Range of Beams Mode.

After pressing the Run button, a plot showing the different simulated focal points appears on the right panel of the window, showing the simulated beams from the point of view of someone looking towards the RIS. Moreover, a .txt file will appear on the window, where the codebook is stored.

Notes:

- **The step should always be greater than zero.**
- **Simulating large number FPs at once can slow down the process.**

- **The randomization option should remain on, when the codebook is going to be used for the mmWave RIS.**
- **The radiation plot should be cleared using the clear plot button, before each simulation.**
- **Each time the update button is pressed, a window appears, prompting the user to store the .txt file. The file can be stored anywhere, <u>except</u> for the Program Files folder.**

To utilize the generated codebook, using the C-Code provided by the THz Lab, Copy and paste the generated codewords from the .txt file to the microcontroller's code inside the **uint8_t bitList[][256]** vector, under the **//Mask 0 Dummy** codeword.

RIS MODEL

On the right side of the MetaMaDe application's homepage the user can find a dropdown menu with the metasurfaces that are currently available at the THz lab of ASU.

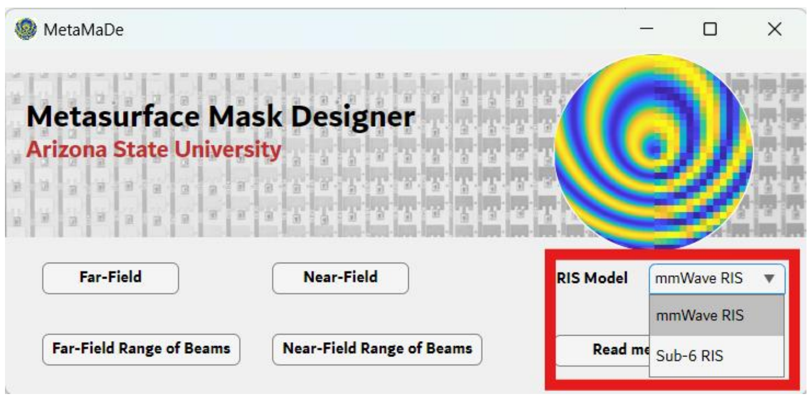

Figure 11: MetaMaDe app homepage (Near-Field Range of Beams)

Sub-6 RIS

The Sub-6 RIS shown in figure 12 consists of 4 Tiles, with each tile having 14×19 elements. In this case each element consists of a radiating patch, which is connected to a smaller parasitic patch through a PIN diode, thus achieving the required reconfigurability. The center frequency is 5.8 MHz with a bandwidth of 150 MHz. The phase mask is distributed to the respective elements using a circuit of shift registers

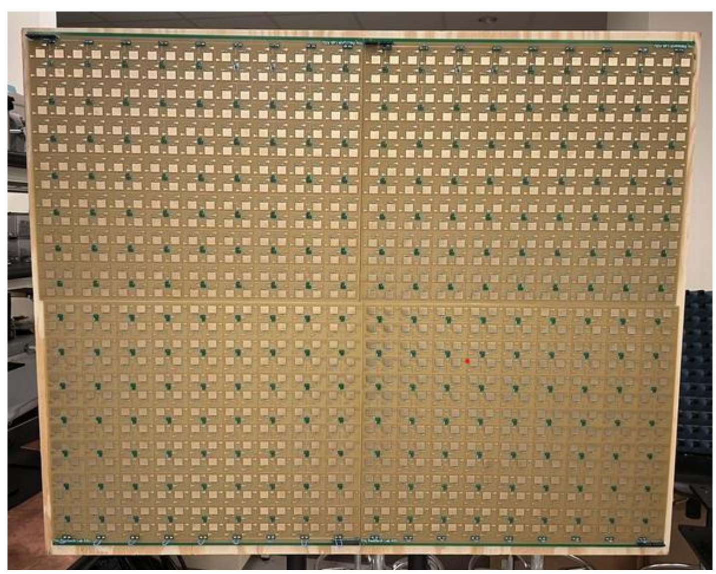

Figure 12: Sub-6 RIS model

(SR) placed on the top layer of each one of the 4 Tiles. These SRs are controlled by a microprocessor (Arduino DUE or Teensy 4.1), attached to the custom motherboard, which was designed to host 4 Tiles. The system's average power consumption of the mmWave RIS is 8 W, and the required DC voltage supply of the system is 2.7V, provided to the RIS by a voltage regulator attached to the back of the RIS.

mmWave RIS

The mmWave RIS shown in figure 13a consists of 4 square Tiles, with each tile having 16×16 patch antennas. The center frequency is at 27.2 GHz, with a bandwidth (180º±20º) of 1.8GHz. Each patch is connected to the ground through a PIN diode achieving in this way the required reconfigurability. The phase mask is distributed to the respective elements using a circuit of shift registers (SR) placed on the bottom layer of each one of the 4 Tiles. These SRs are controlled by a microprocessor (Arduino DUE), attached to the custom motherboard shown in figure 13b, which was designed to host 4 Tiles. The system's average power consumption of the mmWave RIS is 8 W, and the required DC voltage supply of the system is 2.7V, provided to the RIS by a voltage regulator attached to the back of the RIS.

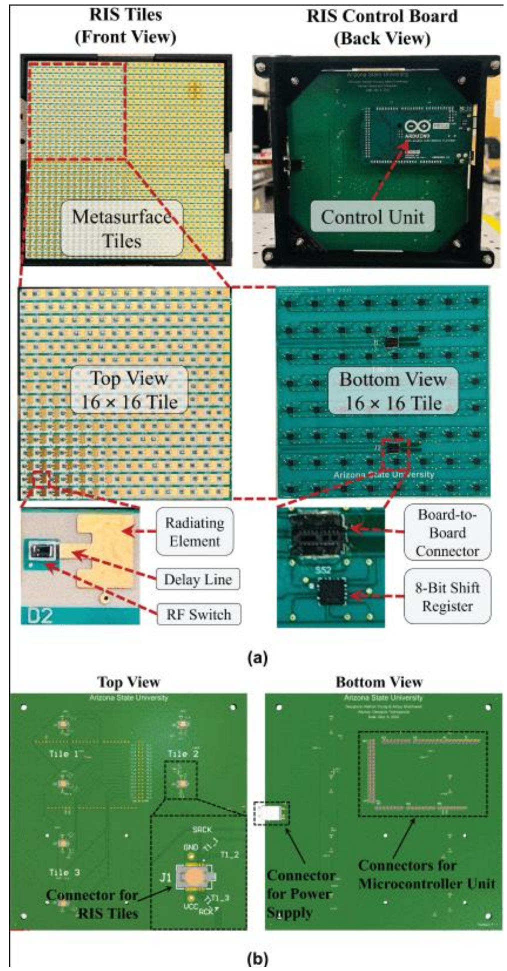

Figure 13: mmWave RIS model

Scalable Biasing Circuit

Due to the limited number of the MCU output pins a circuit of daisy chained Shift Registers is equipped to route the bits to the respective elements as shown in figure 14. Each tile consists of 8 chains of 8 shift registers. Each shift register has 8 output pins, however only every other pin is used, meaning that only 4 out of 8 outputs are currently in use. The unused pins are fed with dummy bits (0s). Each shift register chain is used to bias two rows of elements; thus, the bits of the mask need to be placed in the correct position depending on their destination. Figure 15 shows how each bit of the phase mask is mapped to the outputs of the Shift Register circuit. Each line of the codeword consists of 4 Bytes, and each one of them corresponds to 1 of the 4 Tiles. Each one of the 8 bits is fed to one of the 8 chains of serially connected Shift Registers. Every other row of bytes contains Dummy bits that are fed to the unused output pins.

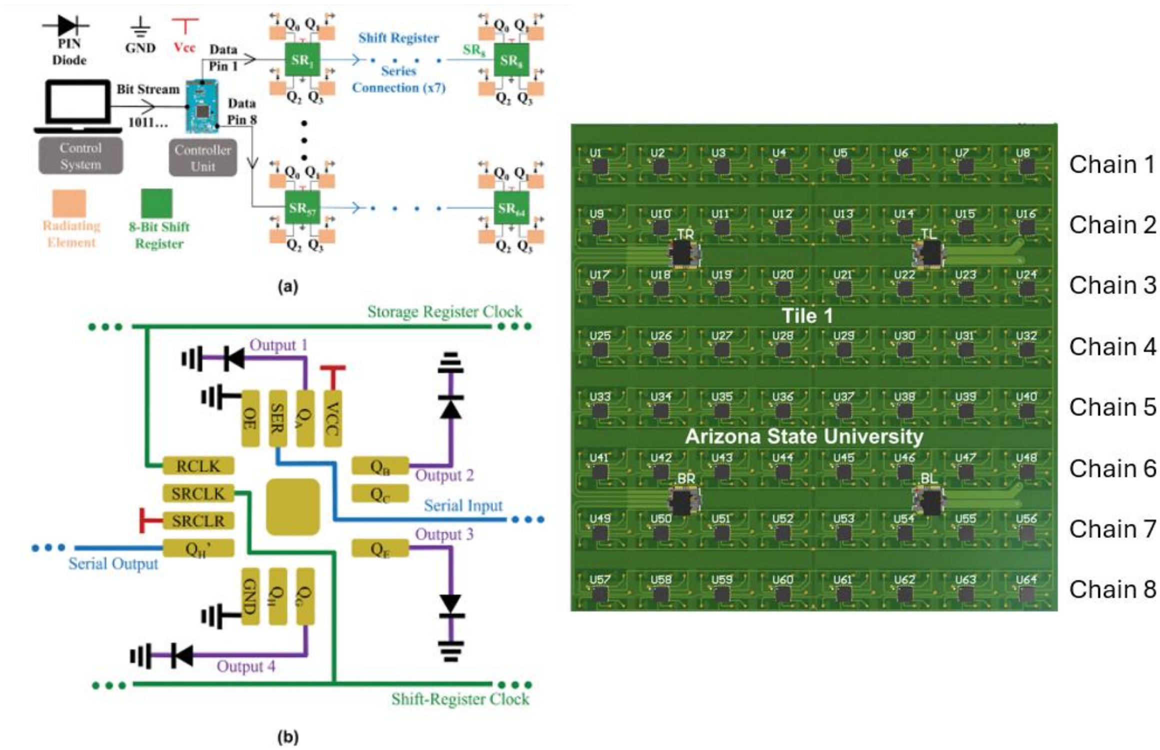

Figure 14: Biasing Circuit Topology

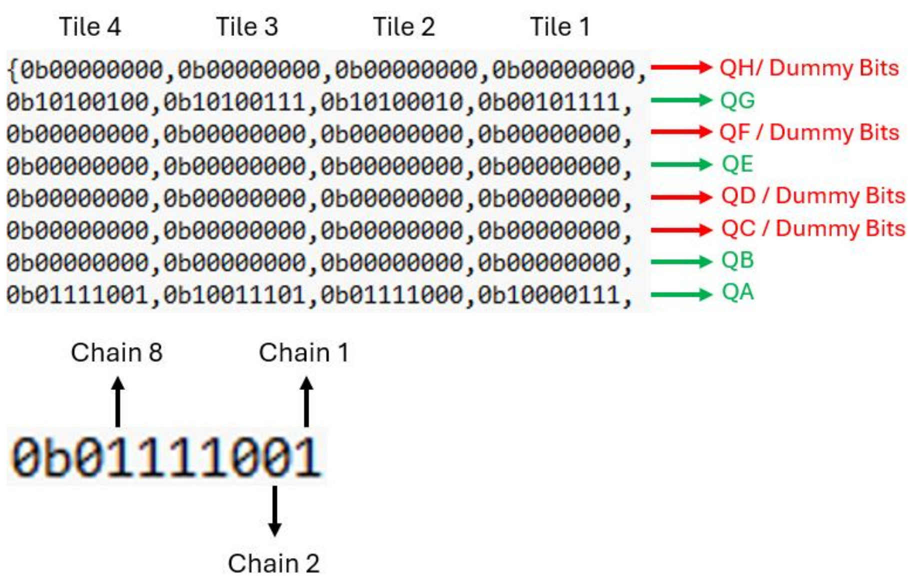

Figure 15: Bits mapped to the circuit.