

## Article

# Broadband RF Phased Array Design with MEEP: A 3D-Printed Open-Source RF Horn in the multi-GHz Bandwidth

Jordan C. Hanson<sup>1</sup> \*, Adam Wildanger<sup>1</sup>

<sup>1</sup> Whittier College, Whittier, CA, USA ([www.whittier.edu](http://www.whittier.edu))

\* Correspondence: [jhanson2@whittier.edu](mailto:jhanson2@whittier.edu)

## Abstract

Radio-frequency (RF) antenna design traditionally proceeds in three phases. First, the design performance is modeled with expensive and proprietary computational electromagnetism (CEM) software. Second, the design is fabricated using intricate metal machining. Third, the fabricated design is characterized using benchtop RF measurement tools. The traditional process can include machine learning algorithms for optimization. We have developed an open-source alternative process that utilizes the MIT Electromagnetic Equation Propagator (MEEP) CEM package for design and simulation, and 3D printing with conductive filament for fabrication. Using this process, we designed and fabricated an exponentially curved RF horn antenna. To characterize our design, we show that the E-plane and H-plane radiation patterns, the VSWR, and the cross-polarization ratios match our CEM calculations. These results indicate that our design is a linearly polarized, broadband RF horn antenna in the [5.5–6] GHz bandwidth. The bandwidth of our instrumentation is limited to 6 GHz, and our CEM calculations predict good performance above 6 GHz. We therefore conclude the bandwidth of our printed design extends above 6 GHz. Future work will include expanding to lower bandwidth by 3D printing larger antennas, and constructing a broadband RF phased array.

**Keywords:** Computational Electromagnetism (CEM), Additive Manufacturing, MEEP, RF Engineering, Open-Source Design

## 1. Introduction

Broadband RF antennas are ubiquitous tools within scientific instrumentation and communication applications. Traditionally, RF antennas are designed using expensive, proprietary software packages, like XFDTD and HFSS [1,2]. Designs are fabricated by cutting and shaping metal with precision machine tools. This technique is sometimes called *subtractive manufacturing*, as opposed to *additive manufacturing*. Compared to *additive manufacturing* techniques, subtractive manufacturing can be costly [3,4]. Open-source additive manufacturing boosts cost efficiency in both the design and fabrication of RF antennas, provided that the 3D printing filament has sufficient conductivity.

Development of new RF antenna designs requires exploration of conductor shapes that set the boundary conditions for radiated signals. Developing new conductor shapes that meet design requirements has been aided by machine learning (ML) strategies [5–7]. While mixing new ML algorithms into open CEM code is straightforward, incorporating them into proprietary software is often challenging and time-consuming. Ideally, RF engineers would keep the functionality found in proprietary software to compute standard RF antenna

Received:

Revised:

Accepted:

Published:

**Citation:** Hanson, J.C.; Wildanger, A.

*Electronics* **2025**, *1*, 0.

<https://doi.org/>

**Copyright:** © 2026 by the authors.

Submitted to *Electronics* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

metrics, while retaining the flexibility to incorporate new ML techniques. Creating a design process based on open CEM tools achieves this goal.

Previously, J. C. Hanson (2021) demonstrated that the open-source MEEP software package may be used as an RF antenna design tool [8]. MEEP operates via the FDTD algorithm for Maxwell's Equations on a Yee lattice [9]. Specifically, the author of [8] produced RF phased array designs in two and three dimensions using MEEP code. The designs included broadband RF horn elements in the [0.5-5] GHz bandwidth. Further, J. C. Hanson demonstrated that open-source CAD software may be used to create complex designs that can be 3D-printed [10].

In this work, we present the first open-source broadband RF horn designed with MEEP and fabricated entirely from conductive 3D-printer filament. In Sec. 2, we explain the CEM design, open-source CAD, and fabrication techniques. In Sec. 3, we show that the E-plane and H-plane radiation patterns, the VSWR, and the cross-polarization ratios match predictions. In Sec. 5, we discuss the limitations of our analysis, summarize key results, and discuss applications and future directions for this research.

## 2. Methods and Materials

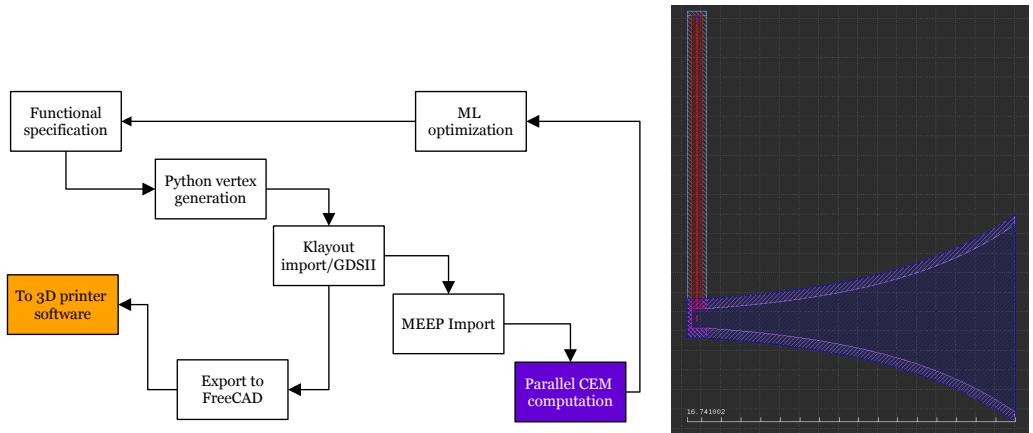
The goal of this work was to use open CEM tools to design, model, and fabricate a linearly polarized, broadband RF horn antenna in the multi-GHz bandwidth. We selected MEEP for our open CEM tools. Over the past decade, MEEP has most often been used to design and model photonics systems [11,12]. However, due to the scale-invariance of Maxwell's equations, MEEP may be used to model electromagnetic propagation with wavelengths at the cm-scale [8,10]. Further, antenna designs created with open CAD tools like kLayout can be imported into MEEP. Using parametric CAD design, we can specify antenna shapes using analytic functions that are then translated into kLayout and MEEP. Designs in kLayout can also be translated into 3D printer files, ensuring we are simulating the same device we are fabricating.

Our open fabrication technique hinges on the existence of commercially available conductive 3D printer filament. Multi3D<sup>1</sup> has shown that a copper-doped thermoplastic serves as a 3D printer filament with conductivities of  $\approx 10^{-1} \Omega \text{ cm}$ , depending on the printing pattern. Using a ruby-tipped 3D printing head and Electrifi filament from Multi3D, we can complete antenna prints on the scale of 16 cm in length in several hours. The printed structures are fitted with RF connectors and tested with a network analyzer, standardized metal antennas, and custom mounts. Figure 1 (left) contains a flow diagram for the design and fabrication process.

### 2.1. CEM Design and Open-Source CAD

The kLayout design of the RF horn is shown in Figure 1. The open-source CEM design begins with choosing parametric design parameters for the exponential RF horn. The horn *cavity* is the rectangular volume where the coaxial cable attaches. The horn *surfaces* are the curved structures that connect the cavity to open space. The *opening* is the area where the *surfaces* stop and radiation exits the horn. Let the origin of an *xy* coordinate system refer to the center of the outside of the cavity. Let  $c_l$  refer to the cavity length in the *x*-direction,  $c_w$  refer to the cavity width in the *y*-direction,  $s_l$  refer to the surface length in the *x*-direction, and  $w$  refer to the opening width in the *y*-direction. Using these variables, Equations 1 and 2 specify the shape of the RF horn surfaces in the *xy*-plane:

<sup>1</sup> See <https://www.multi3d.com>



**Figure 1.** (Left) Flow diagram for the open-source design and fabrication process. (Right) The kLayout CAD design for the RF horn (top view), with a coaxial cable attached. The  $x$ -direction is to the right, the  $y$ -direction is up, and the  $z$ -direction is out of the page.

**Table 1.** Design parameters for the RF horn antenna.

Parameter	Variable Name	Value [cm]
Cavity Length	$c_l$	0.77
Cavity Width	$c_w$	1.00
Surface Length	$s_l$	16.52
Opening Width	$w$	9.59
Height	$z$	2.0

$$f(x) = \frac{c_w}{2} \exp(k(x - c_l)) \quad (1)$$

$$k = (s_l - c_l)^{-1} \ln\left(\frac{w}{c_w}\right) \quad (2)$$

The function  $f(x)$  in Equation 1 describes the upper surface in Figure 1, while  $-f(x)$  describes the lower surface. Table 1 contains the values for  $c_l$ ,  $c_w$ ,  $s_l$ , and  $w$  corresponding to our first 3D printed RF horn. The horn is designed to be linearly polarized in the  $y$ -direction. When we are using a three-dimensional CEM model, we give the RF horn a height  $z$ . The  $z$ -direction is out of the page in Fig. 1 (right).

In previous studies [8], antenna structures were instantiated as native MEEP objects. For example, the following Python3 code with the Meep library imported as `mp` adds  $N$  2D exponential horns to the overall geometry via the `mp.Block()` method:

```

for j in range(0,n_antenna):
    y = j*d_y+y0
    #Describe the cavity
    cav_back_location = mp.Vector3(x0,y)
    cav_location_pos = mp.Vector3(c_l/2.0+x0,c_w/2.0+y)
    cav_location_neg = mp.Vector3(c_l/2.0+x0,-c_w/2.0+y)
    backplate = mp.Block(backplate_size,center=cav_back_location,material=mp.metal)
    cav_upper = mp.Block(side_plate_size,center=cav_location_pos,material=mp.metal)
    cav_lower = mp.Block(side_plate_size,center=cav_location_neg,material=mp.metal)
    geometry.append(cav_upper)
    geometry.append(cav_lower)
    geometry.append(backplate)
    #Describe the horn surfaces

```

```

    size_upper = mp.Vector3(dx,thickness)          100
    size_lower = mp.Vector3(dx,thickness)          101
    for i in range(0,n_slices):                  102
        center_upper = mp.Vector3(i*dx+c_l+x0,c_w/2*np.exp(k*(i*dx-c_l))+y) 103
        center_lower = mp.Vector3(i*dx+c_l+x0,-c_w/2*np.exp(k*(i*dx-c_l))+y) 104
        geometry.append(mp.Block(size_upper,center=center_upper,material=mp.metal)) 105
        geometry.append(mp.Block(size_lower,center=center_lower,material=mp.metal)) 106

```

With appropriate constants defined, this code creates a one-dimensional phased array of 2D RF horns with MEEP conductors. There are two advantages to this *parametric design*: is that it is fast and simple to code, and no other tools or programs outside of MEEP are required. Structures can be quickly modified by exploring the parameter space defined by the constants in Table 1. The disadvantage is that complex structures must be assembled from simple ones with little intuition or visualization. Using open CAD programs like kLayout, however, allow the RF engineer to design antenna structures visually before importing them into MEEP. By introducing open-source CAD, the same structures given to the CEM modeling can be given to the 3D printing system.

Ideally, the advantages of parametric design and open-source CAD should be preserved. First, we print the vertices of the antenna structures from our Python3 code. Second, we create polygons in kLayout, with initial lists of vertices. Finally, we copy the vertices from Python3 code into the kLayout polygons. This procedure results in 2D structures like those shown in Figure 1 (right). Figure 1 (right) is a design of the RF horn in the  $xy$ -plane, and we give the design height and thickness in the  $z$ -direction after importing it to MEEP. MEEP accepts the GDSII file format produced by kLayout, and GDSII files can be converted to STL files suitable for PrusaSlicer and other 3D printing software.

### 2.1.1. Radiation Pattern Calculations

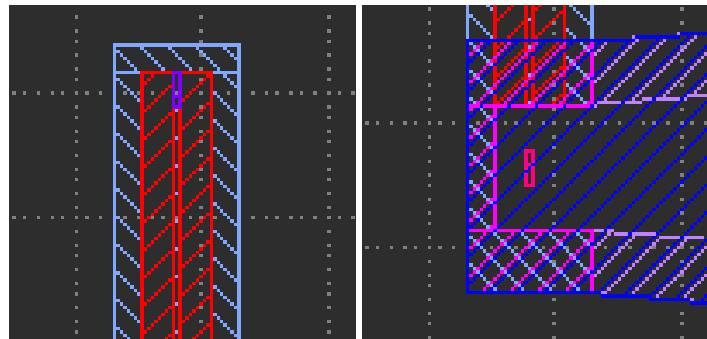
We compute the antenna radiation pattern in MEEP using near-to-far field projection<sup>2</sup>. First, a `NearToFarRegion` is created to fully enclose the structure, similar to a Gaussian surface in classical electrodynamics. Second, the `get_farfield()` function computes the **E** and **H** fields a distance  $r$  from the origin. In Figure 1, the  $xy$ -plane is the **E**-plane, and the  $xz$ -plane is the **H**-plane. We chose  $r = 10$  meters in our calculations, given the dimensions of the horn (Table 1) and  $\approx 1$  cm wavelengths. Third, we compute the normalized Poynting vector along the **E** and **H**-planes. The following code produces the normalized radiation pattern in dB, with MEEP as `mp`, NumPy as `np`, and `proj_box` as the `NearToFarRegion`:

```

def calculate_E_plane_rad_patt(sim,proj_box):          133
    r = 1000                                         134
    npts = 360                                         135
    E = np.zeros((npts,3),dtype=np.complex128)          136
    H = np.zeros((npts,3),dtype=np.complex128)          137
    angles = 2*np.pi/npts*np.arange(npts)              138
    for n in range(npts):                            139
        x = r*np.cos(angles[n])                      140
        y = r*np.sin(angles[n])                      141
        ff = sim.get_farfield(proj_box,mp.Vector3(x,y,0)) 142
        E[:,:] = [ff[j] for j in range(3)]           143
        H[:,:] = [ff[j+3] for j in range(3)]          144
    Px = np.real(np.conj(E[:,1])*H[:,2]-np.conj(E[:,2])*H[:,1]) 145
    Py = np.real(np.conj(E[:,2])*H[:,0]-np.conj(E[:,0])*H[:,2]) 146
    Pr = np.sqrt(np.square(Px) + np.square(Py))        147

```

<sup>2</sup> See [https://mEEP.readthedocs.io/en/master/Python\\_Tutorials/Near\\_to\\_Far\\_Field\\_Spectra](https://mEEP.readthedocs.io/en/master/Python_Tutorials/Near_to_Far_Field_Spectra)



**Figure 2.** (Left) The upper left region of 1. The purple region is the radiating source used for VSWR analysis, the red regions are dielectric material, and the light blue regions are conductors. The combined structure represents a coaxial cable. (Right) The lower left region of 1. The coaxial cable reaches the horn cavity, shown in light blue. The dark blue regions are the beginning of the horn surfaces. The pink structure in the center is the radiation source used for radiation pattern analysis.

```
directivity = 10.0*np.log10(Pr/max(Pr))  
return (angles,directivity)
```

To obtain the radiation pattern from near-to-far field projection, the MEEP simulation must be run such that a sufficient amount of the radiation has passed through the NearToFarField region.

### 2.1.2. VSWR Calculations

The voltage standing wave ratio (VSWR) is a standard figure of merit in RF antenna characterization. The VSWR may be expressed in terms of the complex reflection coefficient,  $\Gamma$ , between coaxial line and RF antenna input port:

$$VSWR = \frac{1 + |\Gamma|}{1 - |\Gamma|} \quad (3)$$

Our strategy to calculate the VSWR is to send a pulse signal down a model for a coaxial cable, and measure  $\Gamma$ . This strategy is based on MEEP FluxRegion objects, similar to the NearToFarRegion objects in the previous section<sup>3</sup>.

Figure 2 (left) shows the upper left region of the RF horn CAD, depicting the radiation source within the structure representing the coaxial cable. The coaxial cable model is built from outer conductors, dielectric materials, and an inner conductor with a radiating source polarized radially. While the radiation pattern analysis uses a continuous-wave (CW) source, the source in our VSWR analysis is given a custom Gaussian signal with a broadband spectrum. In the middle of the cable, a MEEP FluxRegion is instantiated. Similar to a NearToFarRegion, the FluxRegion records electromagnetic flux through a geometric region, similar to a Gaussian surface in classical electrodynamics. Both MEEP classes can record flux at multiple frequencies. In our CW radiation pattern analysis, we set the NearToFarRegion frequency to the source frequency. In our VSWR analysis, we chose 1024 frequency bins ranging from [0-10] GHz and passed them to the FluxRegion.

The reflection coefficient  $\Gamma$  is measured in two stages. First, the MEEP simulation is run for a time corresponding to the length of the coaxial cable, accounting for the speed of propagation given the dielectric material. The input flux is recorded, and the simulation is reset. The simulation is run a second time, but the input flux is multiplied by  $-1$  and pre-loaded into the FluxRegion. The simulation is run for a factor of 2 longer, so that any reflections from the RF antenna have time to reach the FluxRegion a second time.

<sup>3</sup> See [https://mEEP.readthedocs.io/en/master/Python\\_Tutorials/GDSII\\_Import](https://mEEP.readthedocs.io/en/master/Python_Tutorials/GDSII_Import).



**Figure 3.** (Left) The Prusa XL 3D printer beginning a print with the Electrifi filament. (Middle) The finished RF horn with SMA connector. (Right) The finished RF horn viewed with opening visible.

The FluxRegion records the output flux, having already cancelled the input flux from the pre-loaded opposite input flux. Thus, using one FluxRegion, the ratio of output to input flux,  $\Gamma$ , can be calculated. The VSWR ratio (Equation 3) is formed and converted to dB. 177  
178  
179

## 2.2. Fabrication Technique

This section is for Adam to contribute

## 3. Results

Using the techniques described in Sec. 2.2, we fabricated an RF horn with the parameters in Table 1. Figure 3 contains photos of the results. As listed in Table 1, the RF horn is 16.5 cm long (see ruler in Figure 3, middle). The dimensions of the opening are xx by yy cm. The SMA connector is mounted to the cavity, with a coaxial pin extending into the chamber such that the initial radiation is linearly polarized. The radiation pattern measurements were made in a co-polarized S21 configuration, and VSWR measurements are made with respect to a  $50 \Omega$  RF coaxial cable connected to printed RF horn. 180  
181  
182  
183  
184  
185  
186  
187  
188  
189

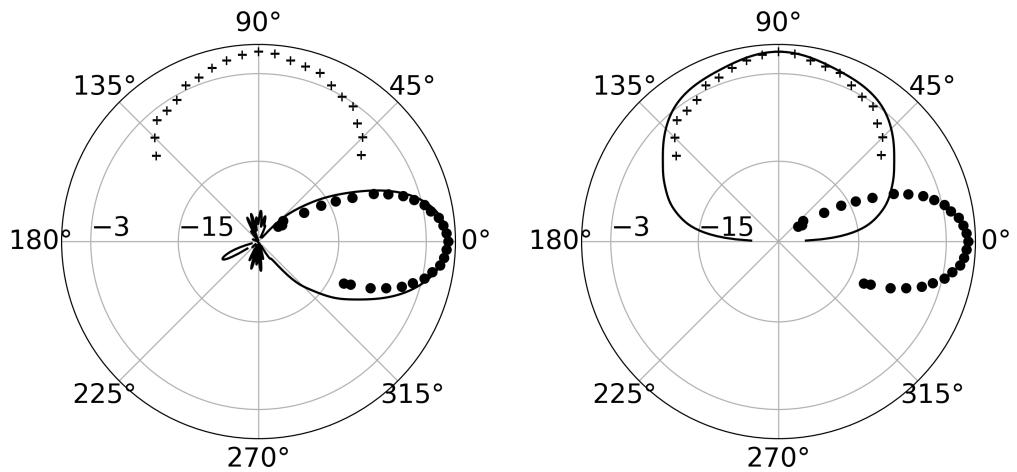
### 3.1. Radiation Patterns

To measure the radiation patterns of the RF horn, we bolted the RF horn on a camera mount, and configured it as the receiver into a network analyzer with [0-6] GHz bandwidth. For a transmitter, we used a rear-fed log-periodic dipole array (LPDA). The netork analyzer was calibrated with a short element, open element, and  $50 \Omega$  terminator. After calibration, the S21 measurement mode was activated, and the cables were connected together to form a closed loop without the antennas. The measured S21 showed a 0 dB loss across the 6 GHz bandwidth. The receiver and transmitter where then inserted, and the S21 measured versus frequency. The transmitter power was set to maximum, 20 dBm, meaning no amplifier was necessary. The distance between transmitter and receiver varied between 1-2 meters. Received power was measured above the noise floor in the bandwidth [5.5-6] GHz 190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200



**Figure 4.** The co-polarized configuration to measure the radiation pattern of the printed RF horn.

We first verified qualitatively that the S21 power varied appropriately with E-plane angle, H-plane angle, and co-polarization angle. Next, the receiver and transmitter were 201  
202



**Figure 5.** E-plane radiation pattern (circles, 5.967 GHz) and H-plane radiation pattern (crosses, 5.85 GHz). (Left) Line: calculated E-plane pattern from our 3D MEEP model of the RF horn. (Right) Line: calculated H-plane pattern from our 3D MEEP model of the RF horn.

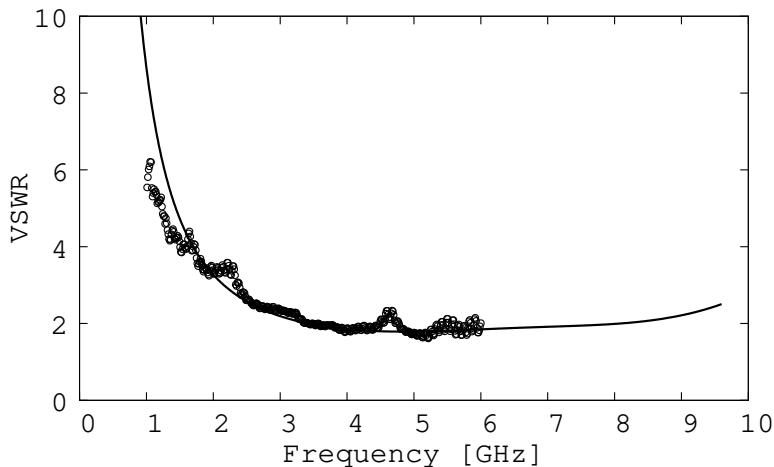
mounted in a vertically polarized configuration so that E-plane angle could be measured with a smartphone level tool (Figure 4). The S21 power was measured versus E-plane angle. Next, the antennas were mounted in a horizontally polarized configuration, so that H-plane angle could be measured with the same smartphone tool. The S21 power was normalized to the maximum received power, when E-plane and H-plane angle were both 0 degrees. The results are shown in Figure 5.

The polar plot ranges from -26 dB to 1 dB in normalized power. In the spherical coordinate system for a linearly polarized RF antenna radiating in the  $x$ -direction, the E-plane angle is the azimuthal angle as measured from the  $x$ -axis. Thus, in Figure 5 (left) the data and CEM model peak at 0 dB when the angle is 0 degrees. There is good agreement between the CEM model and the measured S21 power. In the same coordinate system, the zenith angle is the H-plane angle. Thus, in Figure 5 (right) the data and CEM model peak at 0 dB when the angle is 90 degrees (when the zenith gives a position vector in the  $xy$ -plane). As with the E-plane, there is good agreement between the CEM model and S21 power.

### 3.2. The VSWR

To measure the VSWR of the RF horn, we bolted the RF horn to the camera mount, and configured it as the transmitter on the network analyzer. Once again, the network analyzer was calibrated on [0-6] GHz. After calibration, the S11 measurement mode was activated, and we verified that we measured  $50\ \Omega$  resistance in the bandwidth of interest. Finally, we extracted the S11 reflection coefficient in dB from the network analyzer. We took two additional post-processing steps. First, we converted S11 in dB to linear S11, and then to VSWR using Equation 3. Second, we smoothed the data with an 11-point running average. We chose the 11-point filter to remain symmetric about the sample of interest (5 points to either side). Though the S11 fluctuated, the 11-point running average revealed a trend that matches results from CEM modeling. The results are shown in Figure 6.

After the two-stage post-processing, the VSWR from CEM modeling matched the measured VSWR once we shifted the measured VSWR down by a factor of 4. The reason for this is the impedance mismatch between the line impedance from the coaxial cable, and the load impedance from the RF horn. For example, if  $Z_{\text{ant}} = 200\Omega$ , and  $Z_L = 50\Omega$ , then the VSWR is a factor of 4 higher than it would be if the antenna were matched to the line ( $Z_{\text{ant}} = 50\Omega$ ). Though the measured  $Z_{\text{ant}}$  fluctuated between 50-200  $\Omega$ , after converting



**Figure 6.** Circles: measured VSWR data, smoothed with a running average filter. Line: VSWR from CEM modeling. The black circles have been shifted down to account for the impedance mismatch between 3D printed RF horn and coaxial cable. See text for details.

to VSWR and scaling the data, we find a match to CEM. Two additional observations are  
234  
important to note. First, our VSWR calculation in MEEP was performed in 2D. This makes  
235  
the calculation simpler, but it is not necessary. Second, the bandwidth of the network  
236  
analyzer was limited to 6 GHz. Though our measured data stops at 6 GHz in Figure 6, the  
237  
simulated VSWR in Figure 6 predicts good performance above 6 GHz.  
238

### 3.3. Cross-Polarization Ratios

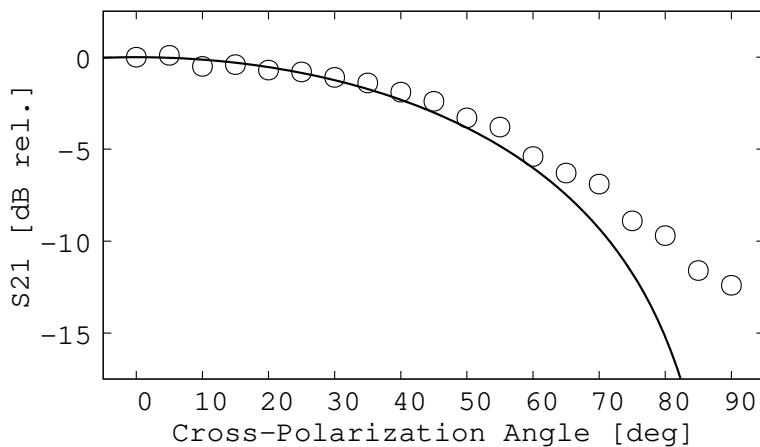
The final measurement we performed was the cross-polarization ratio. The 3D printed  
240  
RF horn and LPDA were arranged in the same S21 configuration that was used to measure  
241  
the radiation patterns. The RF horn was the receiver, and bolted to the mount such that an  
242  
inclinometer would measure cross-polarization angle. The prediction for the dependence  
243  
of received power versus cross-polarization angle is known from optics and antenna theory.  
244  
Let  $P_r$  be the received power,  $P_t$  be the transmitted power, and  $\theta$  be the cross-polarization  
245  
angle. These are related by  
246

$$P_r = P_t \cos^2 \theta \quad (4)$$

When normalized, the ratio  $P_r/P_t$  in dB is equivalent to normalized S21. The results are  
247  
shown in Figure 7, and show good agreement between data and CEM modeling.  
248

## 4. CAD Methods

The CAD design for the model was made using Autodesk Inventor utilizing Au-  
250  
todesk's free Education Plan [1]. Inventor was chosen for ease of access and its Equation  
251  
Curve command, which was essential in precisely and accurately reproducing the exponen-  
252  
tial curve of the horn. Inventor is also a parametric modeling system [2], which was needed  
253  
in order to swiftly make revisions to the model. Design challenges were encountered using  
254  
the equation curve command and ensuring the accurate curve was reproduced. This was  
255  
later found to be verifiable by plotting the expected end point on the same sketch as the  
256  
equation curve. Another challenge was translating the exponential curve to a correctly un-  
257  
derstandable format for the Equation Curve command. This was solved through trial and  
258  
error. Later versions used a parameterized version of the equation produced by ChatGPT.  
259  
This AI generated equation, like many AI generated answers [3], had subtle errors which  
260  
had to be corrected manually, namely the exp and log . The equation was also translated to  
261  
Inventor's parametric equation format with unitless versions of the parameters created.  
262



**Figure 7.** Circles: measured cross-polarization ratio in dB relative to maximum power. Line: Prediction from Equation 4.

This part goes in a table

Other challenges that are unique to this process involve keeping constraints consistent. It was observed that projections would often break, so adjusting base parameters would break sketches, which would have to be repaired. This was further complicated by the equation curves not consistently responding to constraints. Sometimes, the program could not solve a constraint for no apparent reason. The equation curve may need to be remade in these cases. Other times, connecting the offset of the equation curve could have difficulties. This can be solved by building the sketch without using the offset command and potentially either weakening the structure or using more filament than necessary. Closing the sketch could sometimes show as improperly defined but would refuse to allow itself to be defined. This problem was not predictable and there was difficulty in attempting to replicate or correct it consistently.

The workflow for manufacturing has some difficulties which require careful consideration. The filament that was used is Multi3D Electrifi. This filament was found to be more difficult to print than other filaments, such as PLA (Polylactic Acid) or PETG (Polyethylene Glycol). The process for printing is to use a layer of masking tape on a cool print bed with an application of glue, such as spray glue or glue sticks. The tall and narrow nature of the model produced issues with build plate adhesion. This was solved by adding buttress wings to prevent wobbling. The filament was printed between 135–145 °C with no bed heating. The printer used was a Prusa XL with two printheads. A tungsten tipped nozzle was used to reduce abrasion. Crucially, a custom gcode command was required to allow the printer to print below its minimum safety temperature. The command is simple to implement using Marlin Firmware code M302 [4] appended to the end of the starting gcode for printing. PrusaSlicer, the slicer that was used, has a method for automatically appending custom gcodes to the beginning or end of a file.

## 5. Conclusions

Here we summarize key results, discuss the limitations of our analysis, and describe applications and future directions for this research.

### 5.1. Summary of Key Findings

We have shown that the properties of our 3D printed RF horn align with open-source CEM modeling performed using MEEP. The E-plane and H-plane radiation patterns match 3D CEM models created in kLayout and imported to MEEP (Figure 5). The H-plane results

begin to deviate by 1 dB from our CEM model at large angles. ... xxx remember to measure  
the opening of the RF horn.

**Author Contributions:** J. C. Hanson conceived of the open-source design process and produced the initial MEEP calculations that led to the RF horn design, including a previous publication in *Electronics Journal*. J. C. Hanson identified the conductive 3D printer filament with sufficient conductivity to fabricate RF antennas. J. C. Hanson demonstrated that open-source CAD can be incorporated into the process. Finally, J. C. Hanson produced fully three-dimensional CEM models using MEEP to compute the VSWR and radiation patterns for the fabricated designs. A. Wildanger sourced materials, and imported CAD designs into 3D printing format. A. Wildanger successfully completed 3D prints to produce the designs. Together, J. C. Hanson and A. Wildanger collected data using the RF measurement tools. J. C. Hanson used the data to show that the MEEP calculations match the lab measurements. A. Wildanger explored new models to be printed using open-source and free CAD programs, including larger RF horn models.

**Funding:** This research was funded by NAME OF FUNDER grant number XXX.

**Data Availability Statement:** We encourage all authors of articles published in MDPI journals to share their research data.

**Acknowledgments:** Acknowledge Lisa Newton and Sal Johnston

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

CEM	Computational Electromagnetism
CW	Continuous Wave
MEEP	MIT Electromagnetic Equation Propagator
NEEC	Naval Engineering Education Consortium
NSWC Corona	Naval Surface Warfare Center, Corona Division
RF	Radio-frequency
LPDA	Log-periodic Dipole Array

## References

1. Ansys HFSS: Best-In-Class 3D High Frequency Structure Simulation Software. <https://www.ansys.com/products/electronics/ansys-hfss>. Accessed: 2025-12-02.
2. XFDTD Software for 3D Electromagnetic Simulation. <https://www.remcom.com/xfDTD-3d-em-simulation-software>. Accessed: 2025-12-02.
3. Segovia-Guerrero, L.; Baladés, N.; Gallardo-Galán, J.J.; Gil-Mena, A.J.; Sales, D.L. Additive vs. Subtractive Manufacturing: A Comparative Life Cycle and Cost Analyses of Steel Mill Spare Parts. *Journal of Manufacturing and Materials Processing* **2025**, *9*, 138. <https://doi.org/10.3390/jmmp9040138>.
4. Additive Manufacturing vs. Subtractive Manufacturing: A Cost-Benefit Analysis. <https://plentifulchoices.com/production-line/additive-manufacturing-cost>. Accessed: 2025-12-02.
5. Gajbhiye, P.A.; Singh, S.P.; Sharma, M.K. A comprehensive review of AI and machine learning techniques in antenna design optimization and measurement. *Discover Electronics* **2025**, *2*, 46. <https://doi.org/10.1007/s44291-025-00084-9>.
6. Goudos, S.K.; Kalialakis, C.; Mittra, R. Evolutionary Algorithms Applied to Antennas and Propagation: A Review of State of the Art. *International Journal of Antennas and Propagation* **2016**, *2016*, 1–12. <https://doi.org/10.1155/2016/1010459>.
7. Linden, D.; Altshuler, E. Evolving wire antennas using genetic algorithms: a review. *Proceedings of the First NASA/DoD Workshop on Evolvable Hardware* **1999**, pp. 225–232. <https://doi.org/10.1109/eh.1999.785457>.
8. Hanson, J.C. Broadband RF Phased Array Design with MEEP: Comparisons to Array Theory in Two and Three Dimensions. *Electronics* **2021**, *10*, 415, [2102.04585]. <https://doi.org/10.3390/electronics10040415>.
9. Oskooi, A.F.; Roundy, D.; Ibanescu, M.; Bermel, P.; Joannopoulos, J.; Johnson, S.G. Meep: A flexible free-software package for electromagnetic simulations by the FDTD method. *Computer Physics Communications* **2010**, *181*, 687–702. <https://doi.org/10.1016/j.cpc.2009.11.008>.
10. MeepCon 2022. *Broadband RF Phased Array Design with MEEP*, MIT, 2022. MIT.

- 
11. Hammond, A.M.; Oskooi, A.; Chen, M.; Lin, Z.; Johnson, S.G.; Ralph, S.E. High-performance hybrid time/frequency-domain topology optimization for large-scale photonics inverse design. *Optics Express* **2022**, *30*, 4467. <https://doi.org/10.1364/oe.442074>. 338
12. Majumder, A.; Shen, B.; Polson, R.; Menon, R. Ultra-compact nanophotonic devices designed by computational metamaterials. *Imaging and Applied Optics 2017 (3D, AIO, COSI, IS, MATH, pcAOP)* **2017**, p. IM2E.2. <https://doi.org/10.1364/isa.2017.im2e.2>. 339
- 340
- 341
- 342
- 343
- 344

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.