

WIRELESS

(Wireless Intelligent Remote Environment Lighting & Electrical-appliance Smart System)

IOT house appliance central hub and controller

**Group 22 Authors:**

|  |  |  |  |
| --- | --- | --- | --- |
| [Yekusiel Paley](https://webcourses.ucf.edu/groups/836460/users/4445661) | [Ridley Piniarski](https://webcourses.ucf.edu/groups/836460/users/4374742) | [Vy Nguyen](https://webcourses.ucf.edu/groups/836460/users/4377418) | [Nicholas Newman](https://webcourses.ucf.edu/groups/836460/users/4278253) |
| Electrical Engineer | Computer Engineer | Computer Engineer | Computer Engineer |

**Review Committee:**

|  |  |  |
| --- | --- | --- |
| Mark Maddox | Electrical Engineer | Adjunct Professor |
| Xun Gong | Electrical Engineer | Professor |
| Saleem Sahawneh | Electrical Engineer | Lecturer |

**Advisor:**

|  |  |  |
| --- | --- | --- |
|  | Dr. Sonali Das |  |
|  | Ph. D Electrical Engineering |  |
|  | Indian Institute of Engineering Science and Technology |  |

Table of Contents

[2.1 Project background and motivation 3](#_Toc222307992)

[Project Background and Motivation 3](#_Toc222307993)

[2.2.0 Goals 3](#_Toc222307994)

[Basic Goals 3](#_Toc222307995)

[Advanced Goals 4](#_Toc222307996)

[Stretch Goals 4](#_Toc222307997)

[2.2.1 Objectives 4](#_Toc222307998)

[Basic Objectives 4](#_Toc222307999)

[Advanced Objectives 4](#_Toc222308000)

[Stretch Objectives 5](#_Toc222308001)

[2.3 Features and functionalities 5](#_Toc222308002)

[2.4 Existing products 6](#_Toc222308003)

[2.5 Engineering Specifications 7](#_Toc222308004)

[MCU Selection 10](#_Toc222308005)

[Comparing MCUs 10](#_Toc222308006)

[Final Decision: ESP32-S3 for Main Hub and ESP32-C3 for Distributed Nodes 11](#_Toc222308007)

[Touchscreen Selection 11](#_Toc222308008)

[Comparing Touchscreens 11](#_Toc222308009)

[Graphics Library Selection 12](#_Toc222308010)

[Basic RF Transmitter and Receiver Design 12](#_Toc222308011)

[Basic Transmitter Design 12](#_Toc222308012)

[RF Transmitter Circuit Design: 18](#_Toc222308013)

[RF Receiver 19](#_Toc222308014)

[2.6 General Hardware Block Diagram 21](#_Toc222308015)

[2.7 General Software Block Diagrams 23](#_Toc222308016)

[2.8 Prototype illustration / blueprint 27](#_Toc222308017)

[2.9 House of Quality 28](#_Toc222308018)

[10.1 Budget and Financing 29](#_Toc222308019)

[10.2 Project milestones 32](#_Toc222308020)

[Appendix A: Reference 35](#_Toc222308021)

[Appendix C: Etc. 36](#_Toc222308022)

# 2.1 Project background and motivation

## Project Background and Motivation

The evolution of the Internet of Things (IoT) has fundamentally reshaped how we interact with our environment, moving from isolated mechanical systems to a world of pervasive connectivity. While the early 20th century focused on the electrification of the home, the 21st century has shifted toward the "smart" integration of these systems. Today, IoT applications span from automated industrial floors to energy-efficient smart homes, aiming to provide seamless oversight of our physical world. However, many current solutions rely heavily on proprietary, "black-box" off-the-shelf components, which can obscure the underlying engineering complexities and limit customization for specific high-performance applications.

The complexity of modern wireless ecosystems necessitates a deeper understanding of the hardware-software interface. Our senior design project aims to develop a bespoke IoT system featuring a central wireless control hub and a network of controllable appliance nodes. Unlike consumer-grade products that abstract away the hardware layer, our system emphasizes the ground-up design and implementation of analog and RF circuitry. By building the wireless communication link and embedded firmware internally, this senior design project will be able to give the group members a deep understanding of many different engineering principles, from RF circuit design to microcontroller control and digital signal processing.

This project serves mainly as a critical bridge between theoretical coursework and the multidisciplinary demands of the engineering industry. By integrating RF communication, embedded systems, and analog circuit design into a single system-level architecture, the team gains firsthand experience in how these distinct domains interact in a real-world environment. This collaborative effort allows each member to specialize in their intended career path—whether in hardware design or software architecture—while ensuring the seamless integration of all components. Ultimately, the project fosters technical proficiency and interdisciplinary problem-solving skills essential for success in modern engineering roles.

# 2.2.0 Goals

## Basic Goals

* **Centralized Architecture:** Design and implement a primary control hub that serves as the gateway for the network, managing data traffic between the user and peripheral nodes.
* **Multi-Node Connectivity:** Successfully demonstrate bidirectional wireless communication between the hub and three distinct remote appliance nodes.
* **Hardware-Level Integration:** Develop custom-designed analog and RF circuitry to facilitate signal transmission.
* **Unified Control Interface:** Develop hardware for control for three different devices, including turning them on or off, controlling quantized output like brightness, and receiving status commands.
* **System Robustness:** Enhance the reliability of the wireless link to ensure consistent performance in non-ideal environments, with proper encoding and modulation schemes.

## Advanced Goals

* **System Longevity:** Focus on design choices that favor durability, security, and power efficiency for long-term deployment.
* **Network Scalability:** Develop an architecture capable of supporting additional nodes without significant hardware or software overhauls.
* **Performance Benchmarking:** Elevate system operation beyond basic functionality by optimizing data throughput, and response latency, and including node command acknowledgment.

## Stretch Goals

* **Real-World Optimization:** Implement sophisticated features that mirror commercial-grade IoT ecosystems.
* **AI encryption implementation:** Use AI to make a constantly changing key encryption system, for high device security.

# 2.2.1 Objectives

## Basic Objectives

* **Spectrum Utilization:** Establish reliable wireless communication within the **902–928 MHz ISM band** utilizing the **GMSK** modulation scheme, for system robustness to noise, low power usage, and a lower bandwidth usage.
* **Firmware Development:** Design and implement embedded firmware for both the central hub and remote nodes to manage peripheral logic.
* **Command Execution:** Transmit and receive discrete control commands, such as toggle states (on/off), status requests, or multi-state control, for three different nodes.
* **Analog Circuit Design:** Design and fabricate required analog circuitry for stable power regulation and signal interfacing between components.
* **Operational Validation:** Demonstrate successful end-to-end system operations within a controlled laboratory environment.

## Advanced Objectives

* **Data Integrity:** Implement error detection and packet validation techniques, such as **Cyclic Redundancy Checks (CRC)**, checksums, or parity bits.
* **Link Reliability:** Improve communication stability through the implementation of automated retransmission requests and packet acknowledgments.
* **Characterization:** Quantitatively measure and document system performance metrics, including effective communication range, data rate, and power consumption.
* **Firmware Optimization:** Utilize hardware interrupts and low-power sleep modes to increase system efficiency and stability.

## Stretch Objectives

* **Security Implementation:** Integrate device-level addressing and basic encryption to prevent unauthorized access or "spoofing" of the control hub.
* **EMI Mitigation:** Demonstrate functional operation and signal recovery in a moderately noisy RF environment with active interference.
* **Physical Prototyping:** Design and manufacture custom enclosures/housing for the central hub and remote nodes to ensure structural integrity and professional presentation.
* **Wall Plug in Functionality:** Design the central hub and nodes to be able to be powered by the 120V 60Hz output from the wall.

# 2.3 Features and functionalities

The proposed system consists of a central wireless control hub and one or more remote devices capable of controlling or monitoring appliances. The hub is responsible for coordinating communication, issuing control commands, and in future developments, receiving status updates from remote nodes. Remote devices respond to commands from the hub and interface with appliances or simulated loads. Core functionalities include:

* Wireless transmission and reception of control commands within the 902–928 MHz ISM band.
* Will have a touchscreen central hub interface.
* Addressing and identification of individual remote devices.
* Basic command execution, such as turning a device on or off or reporting status.
* Embedded firmware handling communication, timing, and control logic.
* Supporting analog circuitry for power regulation, signal conditioning, and RF interfacing.

The system is designed to be modular and plug and play, allowing additional remote devices to be added without major changes to the central hub, with easy pairing to the central hub of new devices. The feature set of this project is informed by commonly available IoT and smart home products such as smart plugs, wireless switches, and home automation hubs. These products typically employ a central controller that communicates wirelessly with multiple end devices using unlicensed RF bands. Key features such as device addressing, bidirectional communication, and reliable command delivery are standard expectations in comparable commercial systems. Additionally, many prior academic senior design and hobbyist IoT projects demonstrate the effectiveness of star-topology wireless networks, where a central hub manages multiple nodes. These projects often prioritize simplicity, reliability, and low power consumption, which influenced the design decisions for this system.

The decision to design the wireless communication, embedded firmware, and analog circuitry at a low level—rather than relying entirely on off-the-shelf IoT modules—was made to emphasize learning outcomes while still reflecting the structure of real-world IoT systems. The central hub will use a higher powered MCU for control and the touchscreen interface, while the receivers will use ultra-low powered MCUs. Other electronic decisions, like the regulators will be designed according to already existing, high efficiency designs that can be customized on the TI website, as the focus of this group’s learning isn’t necessarily on designing analog electronics, but on designing digital and RF solutions.

# 2.4 Existing products

Commercial smart home systems such as Amazon Echo/ Alexa, Google Nest, Samsung SmartThings, and Philips Hue are designed to maximize reliability, ease of use with smooth integration within a closed ecosystem. These systems rely on standardized and proprietary wireless protocols such as Zigbee, Z-Wave, and Thread.

* **Zigbee:** A low-power, low-data-rate mesh networking protocol operating primarily in the 2.4 GHz ISM band, designed for short-range device-to-device communication with support for large node counts and self-healing networks.
* **Z-Wave:** A proprietary sub-GHz protocol (typically around 900 MHz in North America) optimized for smart home applications, offering improved wall penetration and reduced interference at the cost of lower data rates and vendor lock-in.
* **Thread**: An IPv6-based, low-power mesh networking protocol built on IEEE 802.15.4, enabling secure, scalable, and vendor-agnostic device communication while natively supporting direct internet connectivity without reliance on centralized hubs.

On the academic level there was a previous senior design team doing a smart home system on wireless control. It was group 5 in the summer 2015 semester; their project was called “Wireless Home Control System” designing a smart home system that focuses on energy efficiency. That project used store bought RF transmitter and receiver components to enable communication between a central controller and distributed sensor and actuator nodes. The wireless link was treated as a supporting subsystem for device control, with protocol framing and addressing implemented at a basic level to demonstrate functional communication. While the project successfully showcased low-power operation and modular system design, the physical layer RF behavior and communication protocol were largely abstracted by the chosen RF hardware. However, our project will focus less on energy efficiency, and more on our custom implementation of the GMSK protocol, proving our ability to design and implement efficient RF data transmission.

In contrast, the WIRLESS project avoids these premade communication stacks and instead implements the wireless system from the physical layer upward. By designing the RF subsystem and communication protocol us, the project focuses on implementing baseband encoding, waveform shaping, and modulation at the physical layer, along with framing, addressing, and basic reliability at the digital communication layers. Instead of treating wireless communication as a black box, WIRLESS breaks down and rebuilds the structure used in commercial smart home systems, giving the team hands on experience with both RF design and protocol development while still supporting a scalable hub and node architecture.

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Commercial Systems (Echo, Nest, Hue)** | **2015 Senior Design (Group 5)** | **Our Project (WIRLESS)** |
| **Primary Goal** | Reliability, ease of use, and ecosystem lock-in. | Energy efficiency and modular device control. | **Custom RF design & GMSK protocol implementation.** |
| **Wireless Stack** | Standardized/Proprietary (Zigbee, Z-Wave, Thread). | Off-the-shelf RF transmitter/receiver modules. | **Custom-built from the Physical Layer upward.** |
| **RF Protocol** | "Black Box" (Pre-implemented by manufacturers). | Abstracted by hardware; basic framing. | **Hand-coded GMSK, waveform shaping, & baseband encoding.** |
| **Hardware Focus** | Consumer-grade integration & aesthetics. | Store-bought components for subsystem support. | **Custom RF subsystem design and signal integrity.** |
| **Network Topology** | Mesh (Self-healing, large node counts). | Central Controller to Node (Functional link). | **Scalable Hub and Node (Custom framing/addressing).** |
| **Key Advantage** | "It just works" for the average consumer. | Demonstrated low-power operation. | **Deep technical mastery of RF & digital comms layers.** |

# 2.5 Engineering Specifications

|  |  |  |
| --- | --- | --- |
| **System Level Specifications** | | |
| **Specification** | **Target value (range)** | **Technical Justification** |
| Max operating frequency range | 902-928MHz | FCC Part 15 ISM band, low enough band that cheap FR-4 boards can still be used with minimal loss, without needing expensive Roger’s boards. The 2.4Ghz band so there will be less band hopping required, due to many other devices using the 2.4GHz band. |
| Communication range | >=20m | Measured indoor line-of-sight, to show functionality within average home distances |
| Number of supported nodes | Three Nodes: RGB strip, fan, door lock | This demonstrates scalability, plug and play functionality, central node and multistate control. |
| End-to-End command latency | <=1 second | This is the maximum time a user would be willing to wait for a command to be implemented, as any slower would be considered “laggy” or “slow.” |
| Data rate | 1Mbps | The data rate will be 1Mbps, so in the future the system will use less power, and if need be, will be able to do more complex control with the higher bitrate. |
| System power supply | Wall power AC to 12V DC (24W), with 12V,5V,3.3V,etc. regulators | Large enough battery for power, and regulators for different required functions |
| Max RF output power | 1W | From FCC: "§ 15.247 (a)(3):  For systems using digital modulation in the 902-928 MHz, 2400-2483.5 MHz, and 5725-5850 MHz bands: 1 Watt.” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Component-Level Specifications** | | | |
| **Component** | **Specification** | **Target value** | **Technical Justifications** |
| IF Bandpass filter capacitors | Capacitor Type | C0G (NP0) Ceramic | Extremely stable across temperature. |
| IF Bandpass filter Inductors | Inductor Type | Wire Wound, Ceramic Core | Higher Q at 50Mhz (less equivalent series resistance), compared to other inductor types. |
| RF Receiver Switch | Switch type | Transistor controlled biasing for a diode | The switch will use a diode to turn on and off RX when the node or hub is sending TX, so the TX signal won’t fry the delicate RX components. Using diodes is simplest, and DC transistors will control it’s biasing, so it will act as a switch, and not just a power limiter. |
| FPGA | Clock Speed | >=70Mhz | If in the future a band hopping feature is implemented, the output bitrate will need to be between 27-53Mbps. Having 20Mhz extra room will be necessary extra legroom. |
| DAC | Samples per second | >=159MSPS | Even though the Nyquist rate is 2\*f (106MSPS), however then the other DAC created images will be very close to the center frequency of output information, therefore making the DAC’s speed 3\*x, gives a lot more room between the center frequency and spurious images, for easier filtering. |
| Central Hub User Interface Hardware type | Touchscreen | ILI9341 (2.8’’) | Suitable resolution, screen size, touch input support, and a good price-to-performance ratio. |
| TX Upconverter Mixer Oscillator | Frequency | 875MHz | Looking at Mouser and DIGIKEY for cheap oscillators at the required frequency range (850-900Mhz) gave the two following cheap oscillator options: 850Mhz and 875Mhz. 875Mhz was chosen since then the FPGA won’t have to work as fast when creating the IF. 900Mhz wasn’t chosen, since then the IF and RF frequencies would be extremely close, making filtering almost impossible for a senior design project, and having a 27-53Mhz difference between IF and RF allows for future band hopping, and a large enough difference between IF and RF that a simple elliptic filter can filter away. |
| TX Upconverter Mixer Oscillator | Maximum LO Power input | 10dBm | The data sheet for the MAMXSS0012 passive mixer states in its data sheet that the input 1dB compression happens typically at 15dBm, meaning that the input wave starts exhibiting nonlinear properties (top rounding). Therefore, to give plenty of legroom, a maximum input power of 10dBm has been instantiated. |
| DAC | Output Frequency Range | 27-53Mhz | Since the ISM band that will be operated in will be 902-928MHz, and the oscillator is 875Mhz. 928 – 875 = 53Mhz, 902 – 875 = 27Mhz. |
| RF Bandpass Filter Capacitor | SRF | >2GHz | The self-resonant frequency of the capacitors must be much higher than the operating frequency, so the capacitor won’t act as a short circuit in the operating frequency. |

## MCU Selection

### Comparing MCUs

Four microcontrollers were evaluated for this project based on system requirements, development complexity, and long-term maintainability:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **ESP32-S31** | **ATmega328P2** | **ESP32-C33** | **MSP4304** |
| CPU Architecture | 32-bit Xtensa | 8-bit AVR | 32-bit RISC-V | 16-bit RISC |
| Core Count | Dual core | Single core | Single core | Single core |
| Max Clock Speed | Up to 240 MHz | 16 MHz | Up to 160MHz | 1-25 MHz |
| SRAM | ~512 KB | 2 KB | ~400 KB | 2-64KB |
| Touchscreen Support | Yes | No | Limited | No |
| Internal RF Component | Yes | No | Yes | No |
| Coding Environment | Arduino/ESP-IDF | Arduino | Arduino/ESP-IDF | CSS |

The design requirements are to have two different types of PCB. The first one will be the main hub where the MCU will be responsible for managing the touchscreen user interface, system control logic, and coordination of communication with the distributed nodes. This board serves as the primary point of interaction for the user and must support real-time display updates, touch input processing, and packet framing for data transmission to the RF subsystem.

The second type of PCB will be the distributed node, which is designed for simpler functionality, and will have a simpler MCU to do the digital tasks on the PCB. Each node is responsible for generating and sampling digital bit streams for the custom RF subsystem, as well as handling basic input and output operations such as buttons and LEDs. These tasks require reliable timing and GPIO control but do not demand significant computational resources or graphical capabilities.

### Final Decision: ESP32-S3 for Main Hub and ESP32-C3 for Distributed Nodes

The ESP32-S3 is selected as the primary microcontroller for the central hub. While the internal wireless capabilities of the ESP32-S3 are not used for RF transmission in this project, the device remains a strong choice due to its system-level capabilities. Key reasons for selecting the ESP32-S3 include:

* Sufficient processing power and memory to support a graphical touchscreen interface
* High-speed SPI and DMA support for efficient display updates
* Dual-core architecture enabling separation of user interface tasks and communication logic
* Strong development ecosystem and long-term support

The touchscreen interface requires real-time rendering, touch event handling, and state management, which are not practical on low-resource microcontrollers. Even without utilizing the internal RF peripherals, the ESP32-S3 provides the necessary performance and flexibility to act as the system controller for the hub.

The ESP32-C3 was selected for the distributed nodes, since it was proven as a simplified, single-core alternative within the same microcontroller family. Although the nodes only require bitstream generation and sampling for the custom RF subsystem, using the ESP32-C3 provides several advantages:

* Consistent toolchain and development environment across hub and nodes
* Simplified debugging and firmware maintenance due to shared architecture
* Adequate GPIO and timer resources for precise bit-level signaling
* Lower power consumption compared to higher-end ESP32 variants

Using microcontrollers from the same manufacturer reduces integration risk and allows firmware development to follow a unified design approach.

## Touchscreen Selection

### Comparing Touchscreens

It has been determined the ESP32-S3 will be used for the Main Hub, and a viable display controller must be looked for. Recall that this display is the primary way the user interacts with the system.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Controller** | **GC9A01** | **ST7735** | **ST7789** | **ILI9341** |
| **Screen Size** | 1.3” | 0.9”-1.8” | 1.0”-4.3” | 2.0”-3.5” |
| **Screen Shape** | Circular | Rectangular | Rectangular | Rectangular |
| **Resolution** | 240x240 | 160x128 | 320x240 | 320x240 |
| **Price** | ~$10-$30 | ~$5-$15 | ~$10-$40+ | ~$16-$20 |

These controllers were primarily chosen for consideration due to their compatibility with TFT\_eSPI and LVGL, which are two popular graphics libraries created for 32-bit microcontrollers. Ultimately, the conclusion was that the 2.8” model of the ILI9341 will be used due to having a suitable resolution, screen size, touch input support, and a good price-to-performance ratio. The GC9A01 and ST7735-supported displays would be difficult for the end user to see and ultimately interact with and use. The ILI9341 also gets a slight edge over the ST7789-supported displays due to the fact most ST7789’s use ribbon cables as connectors, which are more difficult to prototype with, while the ILI9341’s has pre-soldered pin headers, which can be connected directly into breadboards.

## Graphics Library Selection

Choosing between TFT\_eSPI and LVGL requires consideration as to what the overall goals of the project are. In terms of feature support, LVGL is significantly more robust than TFT\_eSPI. It offers dozens of pre-made widgets, supporting 3D textures, animations, tables, sliders, buttons, etc. However, this comes at the cost of a heavier burden on the CPU and RAM. The worst-case scenario for Core 0 of the ESP32-S3 must be considered, which will primarily be responsible for sending out control commands and receiving acknowledgements. The purpose of the GUI that will be made with either TFT\_eSPI or LVGL, will be to easily control the appliances using a touchscreen interface. However, if the CPU is too busy working on the graphics, it might delay the execution time for the actual commands, defeating the purpose of the system. In the end, the decision heavily favored using LVGL due to LVGL being more pleasing to look at, and the UI is not so complex that TFT\_eSPI’s relative ease with many functions would need to be used.

## Basic RF Transmitter and Receiver Design

### Basic Transmitter Design

#### First stage: Baseband shifting and Modulation

First the before transmission of any digital signal, the signal must be baseband shifted. The reason for this is mainly because, sharp edges require a larger bandwidth than smooth edges. Since a one-bit digital signal is made up of usually a “high” and a “low” with a sharp transition between them (a square wave). The bandwidth of such a wave reduces at the speed of 1/n in the frequency domain with respect to the center frequency. This is an extremely large bandwidth, and wave shifting is used to greatly reduce the bandwidth by smoothening out the sharp edges from the digital signal. Currently the two following band shifting methods are being considered:

|  |  |  |
| --- | --- | --- |
| **Filter Type** | **RC Exponential Shaping** | **Gaussian Pulse Shaping** |
| **Ease to implement (analog)** | Easy, RC ladder, low pass. | Harder, can only approximate such a filter response, with Bessel type filters. |
| **Spectral Decay** | Faster initial band reduction, but slower bandwidth reduction at far out spectral splatter | Better for reducing total occupied bandwidth. () where is the standard deviation. |
| **Wave Characterization** | rising edge, falling edge (digital high).rising edge, falling edge (digital low). | for digital high, for digital low. |
| **Side lobes** | Present | Nonexistent |
| **Potential for Ringing and Overshoot** | Yes | None |
| **Phase Response** | Nonlinear (distorts digital pulses) | Linear (preserves digital pulses) |

Originally the plan was to implement this baseband shifting using the Bessel approximation of the Gaussian Pulse Shaping, since doing the shaping via analog would be easy to implement with basic tried and true analog filter topologies, like Sallen-Key or MSB. That was true when the group wanted to send the bits at a rate of 100kbps. However, recently the group decided to change the bit rate to a much larger bit rate, 2Mbps. This decision was made based on a few factors, mainly that designing at this higher frequency will allow for less power usage if proper high-side power gating is used, and the difficulty would prove our skills and abilities, and would allow control and computations of higher bit rate peripherals like face sensors, fingerprint sensors, etc. Ideally this wouldn’t be a problem implementing using analog filters however, since the capacitors used would be on such a small scale as 1-100pF, such small capacitance is very close to the inherent capacitance of the PCB board, so the analog design became much more complex. Therefore, the band shaping would have to be done another way; digitally. Here is the original plan:

* **Phase 1: Signal Encoding and Modulation** The initial signal is output using **Manchester Encoding**, where a logic '1' is represented by '10' and a logic '0' by '01'. This encoded signal is then processed through an **XOR gate** with a clock signal running at the same rate as the output. This ensures constant switching, facilitating the use of high-pass filtering by preventing long strings of identical bits.
  + **Frequency Consideration:** Due to the Manchester encoding process, the original signal frequency effectively doubles.
  + **Zero-Centering:** An **op-amp subtractor** is utilized to shift the signal, centering it at 0.
* **Phase 2: Filtering and Bandwidth Optimization:** To refine the signal and eliminate unnecessary harmonics, the system employs an **elliptical Low-Pass Filter (LPF)** and **High-Pass Filter (HPF)**.
  + **Waveform Shaping:** By creating a modified sine wave, the system approximates a Gaussian baseband shift (effective down to approximately -60dB).
  + **Alignment:** The "humps" at the end of the passbands for both the LPF and HPF are aligned to match the required bandwidth, ensuring maximum efficiency for low data-rate transmission.
  + **Phase Management:** Potential phase shifts will be addressed during the final integration and testing phase.

However, there were a few issues with this strategy. The first being as explained before, analog filtering can’t be used anymore and would therefore need a digital solution. The second problem being that later it was realized that since an FSK/PSK modulation strategy is being used, Manchester encoding wouldn’t be needed to ensure constant change in the signal, since any frequency modulation by its very nature forces the signal to continuously change. GMSK will be used instead of simple FSK due to the following reasons:

|  |  |  |
| --- | --- | --- |
| **Concept** | **Simple FSK** | **GMSK (using I/Q)** |
| **Transition** | Abrupt frequency jump | Smooth Gaussian glide |
| **Waveform** | Discontinuous | Continuous and smooth |
| **Spectrum** | Wide/Messy | Tight/Clean |

Finally, the concluded baseband shaping is as follows:

* **Abstract Model:** While analog components are viable, the preferred implementation is primarily digital to leverage the specific expertise of the Computer Engineering (CPE) team; and not analog due to the reasons above. This is following the modulation called GMSK, since this modulation technique proved to be the most efficient with low power usage, high noise robustness and low bandwidth usage, compared to other modulation techniques, and why FSK was not explained above. All the following steps will technically be taken at once using a lookup table, however they will be elaborated on as if being done sequentially.
  + **Quantization and Sampling:** All the 1’s and 0’s will be quantized and sampled. All 1’s will later be a positive voltage output and the 0’s will be a negative voltage output.
  + **Band shaping:** The quantized stages will then be sent through a look-up-table (LUT) that will output the “1” as a function and a “0” as a function.
  + **Integration:** The Gaussian band shaped signal will then be integrated to a function called “. (This is the standard name of the integrated Gaussian wave.)
  + **Formation of I(t) and Q(t) (PM):**  and . That simple. However, these two generated signals must be generated sequentially onto two separate wires. This is the phase modulation (PM) part of this modulation technique, since the signal’s phases are changing continuously. The reason for this is that slow phase changes allow for a smaller bandwidth than quick phase changes in the following frequency modulation (FM) step.
  + **AM:**  and . At this stage this is basically AM, as the signal is just being multiplying by an envelope (which was phase modulated.) These are still on separate wires. These multiplied signals will have the frequency of the IF, which will be 10-70MHz.
  + **Summing:** Now the two signals are summed together, . To see what the ending signal looks like:

As one can deduct from the final equation, this signal is a form of phase modulation, since the only changing variable in this signal is the phase. However, since the signal was integrated, the signal can be considered a type of FM as well, therefore this type of modulation is said to be a hybrid of both modulation techniques.

* **Realized Implementation:** The following is the plan to actualize this idea. The quantization and sampling, band shaping, integration, formation of I(t) and Q(t), AM and summing stages will all be done by the **Intel Cyclone IV** FGPA development board (or the **AMD Artix 7** dev board if another FPGA is needed). The FPGA do all math using math function, since these two dev boards don’t have enough memory for a giant math LUT, however does have enough LUTS and DSP slices for math. The input to the FPGAs will be 1’s and 0’s from the MCU, and the output from the FPGAs will be 10 bits in parallel to the 10 bit DAC **THS8136.** The IC has 3 DACs, so in the future if a higher sample rate is desired, it can do it in parallel (with very precise tuning.)

As shown, the conversion, modulation, and encoding will happen all at the same time in the FPGA, and the DAC will output the signal as an analog signal. A higher frequency IF signal will make the later RF stages easier, but will make the FPGA have to work faster, and the DAC as well, which might cause timing problems in the future, and make the signal act closer to RF signals, causing potential mismatches, ringing, loss, etc. For initial breadboard testing, the DAC will have its own custom designed surface mount to through hole converter, since the DAC is surface mount only. All wires will be stripped to the bare minimum, and decoupling capacitors, and required resistors will be added.

#### Second Stage: Low-Pass Filter

A low pass filter must be made after the DAC output, since for every DAC output there are spurious images that are formed, with their frequencies given by the following equation:

Where is the spurious frequency, is an integer representing the harmonic of the clock, is the sampling rate (180MSPS), and is the fundamental frequency (10-50MHz). The normalized amplitude (normalized to the center main frequency) is given as the following equation:

To ensure spectral purity and operational flexibility for the 915 MHz ISM band transmission, the system architecture utilizes a dual-stage filtering strategy to mitigate unwanted images generated by the frequency translation process. An Elliptic low-pass reconstruction filter is implemented at the output of the **THS8136 DAC**. This specific topology was chosen for its rapid roll-off and stopband nulls, which are essential for suppressing primary DAC sampling images before they reach the mixer stage to be up-converted. By maintaining a wide, flat passband between 27 MHz and 53 MHz, the design provides the flexibility to shift the carrier frequency within the 902–928 MHz band via software without requiring hardware modifications.

Without this filtering, up-converted spurious images would fall within the restricted bands defined in **47 CFR § 15.205**. Consequently, the system is designed to ensure these emissions adhere to the general radiated emission limits of **47 CFR § 15.209(a)**, which mandates a maximum field strength of **500 µV/m** (measured at 3 meters) for frequencies above 960 MHz. The original elliptic filter topology was created using the Ansoft filter designer wizard, then the values were changed to match standardized capacitor and inductor values. In front of the filter is a 100nF capacitor, to function as a DC blocker, and a parallel Zobel Network, to help attenuate the reflected high frequencies, since even though the first image created by the DAC is -8dBc, that still might be problematic. In the end a 2-stage high pass filter was used as the Zobel Network, giving a -0dB to -1dB in the passband, and around -11dB at the first potential spurious image that is 127Mhz. Since the first image is already -8dBc, having that image come back to the DAC as -19dB, is perfectly fine. Figures 1,2,3 do not consider Q factors of capacitors or inductors, and when bought, those Q-factors must be sufficiently high enough so in-lab conditions will closely match the simulated ones.

Now for the oscillator choice that will be the LO to the up-conversion mixer. The cheapest oscillator frequencies that were found were: 850Mhz, 875Mhz, 900Mhz. If 850Mhz was chosen, then the IF would have to be extremely high, which can strain the FPGA, and 900MHz would be almost impossible to filter, therefore the Renesas **XUL535875** oscillator will be used to generate stable output, that’s why the pass band before entering the mixer is 27-53MHz. The Renesas XUL535875 is a current output oscillator, so after putting a 100-ohm resistor between the differential output (differential output must be combined with a balun), the impedance coming out of the oscillator will be around 50 ohms. The oscillator provides a differential output power of approximately **-8 to -9 dBm**, which must be amplified to a level between **10 and 15 dBm** to properly drive the **MAMXSS0012** passive diode mixer. This power range ensures a strong LO (local oscillator) drive while maintaining enough headroom to avoid the mixer’s typical **15 dBm** input 1 dB compression point. Consequently, the chosen amplifier must feature an output **P1dB of at least 15 dBm**. The amplifier chosen that fits under these requirements was the **BGA616H6327XTSA1**, since this amplifier has 19dB of gain, might not even need to attenuate, and a P1dB of 18dBm. The amplifier needs some setup since the output comes from the power pin, but it is completely manageable.

#### Third Stage: IF-RF Mixer

As stated above, the **MAMXSS0012** diode mixer will be used, with the **XUL535875** oscillator as the LO. The IF and RF ports of the mixer are matched to 50 ohms; however, the LO port isn’t matched. There is a functional schematic given in the data sheet, however the data sheet didn’t specify if the given impedances are exact, or if the impedances are the impedances of discrete components on a specific PCB. Since this is the case, a specialized test board must be fabricated, to test this component in the lab. This will be tested either using a VNA with an RF probe and using Port Extension (also called Electrical Delay) to take out the impedance from measuring it (and the solder if there is), or measuring using a SMA soldered port, with a pi network in between, finding impedance, and changing accordingly. Using dial capacitors and inductors would be ideal. After the output of the RF port, there will be a matching network, and an amplifier/buffer to increase the power to around 20dBm, to allow for far transit of signal, then finally an SMA connector to an antenna. Afterwards there will be another bandpass filter to remove all spurious created frequencies from the mixer. If the Qualcomm / RF360 B39921 series or the Abracon AFII-SW-0029 saw filters are used, which are specifically tuned for the 902-928MHz ISM passband, then:

**At the Lowest Channel (902 MHz Output):**

* **IF Generated:** 27 MHz
* **Desired RF:** 875 + 27 = **902 MHz:** **PASSES**
* **Image Frequency:** 875 - 27 = **848: BLOCKED** (52 MHz away from the passband edge)
* **LO Leakage:** **875 MHz**: **BLOCKED** (25 MHz away from the passband edge)

**At the Highest Channel (928 MHz Output):**

* **IF Generated:** 53 MHz
* **Desired RF:** 875 + 53 = **928 MHz**: **PASSES**
* **Image Frequency:** 875 - 53 = **822 MHz**: **BLOCKED** (78 MHz away from the passband edge)
* **LO Leakage:** **875 MHz**: **BLOCKED** (25 MHz away from the passband edge)

Do more research on PCB design for saw filters, Gemini AI said something about needing matching (of course) and that there need to be multiple thermal/ground vias underneath the ground pads of the filter.

In the end, the Qualcomm **B39921B4301F210** SAW filter will be used, as it has -40dB in stop band, and maximum -2dB attenuation in the passband. It is matched on both sides to 50 ohms, so no matching circuit is necessary, unless the chip is very far from the amplifier and mixer output. The maximum

#### Fifth Stage: Buffer Stage / Amplifier stage

This time the **GRF4003** amplifier will be used with possibly an attenuation network, since this amplifier has a relatively small gain of 13dB, when comparing to other similar amplifiers of this class, with a P1dB at 25dBm, giving a comfortable linear range to work with. There will be a matching network before and after the amplifier, with a choke next to the antenna to not allow any signal to come in.

#### Sixth Stage: Antenna

The antenna will be either a PCB-integrated on-board antenna or an SMA-connected wire dipole, with the on-board antenna as the primary choice. The target RF power will be 10-13dBm. There are a few types of on-board antenna designs as well:

|  |  |  |
| --- | --- | --- |
| **Antenna Type** | **Pros** | **Cons** |
| **Printed Inverted-F Antenna (IFA / PIFA)** | Compact (≈ λ/4), ground-referenced and predictable, easier impedance matching, tolerant to enclosure and hand effects, widely used with many reference designs | Requires shorting via to ground, needs antenna keep-out region, post-fabrication tuning typically required |
| **Meandered Monopole** | Simple geometry, no shorting via required, easy PCB routing | Strong dependence on ground plane, sensitive impedance matching, easily detuned by nearby objects, typically lower efficiency than IFA |
| **Printed Loop Antenna** | Compact footprint, reduced sensitivity to nearby metal, can be suitable for noisy environments | Narrow bandwidth, lower radiation efficiency at sub-GHz, difficult impedance matching, highly layout-dependent |
| **Patch Antenna** | Directional radiation, high efficiency when properly designed, stable impedance | Large size at 915 MHz, narrow bandwidth, requires controlled substrate parameters, generally unsuitable for compact designs |
| **SMA-Connected Wire Dipole** | High efficiency, well-defined radiation pattern, minimal PCB tuning, ideal for validation and testing | Requires external connector, increased cost and mechanical complexity, not suitable for fully integrated products |

Again, a lot of components are already being designed, as well as a lot of HDL work and programming, and therefore, buying an SMA connected antenna will be best.

All filtering and impedance matching will most probably be done with discrete components, since at the 902-928MHz frequency range, matching and filtering would take up a lot of PCB area. Unless very little area is required, most of this will be done with passive RLC components.

## RF Transmitter Circuit Design:

#### Abstract:

First and foremost, the circuit will be designed in as many separate PCB’s as possible, so if one breaks or is problematic, less will have to be thrown away. Therefore, the DAC and the output DAC filter will be designed on a single PCB. The input will be male header pins which will connect to the FPGA output pins, power connections and the output will be a 50-ohm SMA connector. The circuit will be designed on a 4-layer FR-4 PCB, so the traces can be smaller. The DAC has 4 sides, each with 12 pins of pins, 3 for RGB digital input, and the other side with other pins. The green input side will be used, since that side also has the clock, as to allow for a single male header pin strip to be used. All unused digital input pins will be grounded, as to avoid possible interference.

A 50-ohm resistor must first be chosen, so the output impedance of the DAC will be 50-ohms. Since the DAC is current output, all that is needed to be chosen is a resistance value as output, and that will be the output resistance. A thin film resistor must also be chosen instead of a thick film one, as to reduce parasitic inductance. A 50-ohm resistor will also be places on the unused red and blue outputs of the DAC, as to match the circuit, as it is unknown if the circuit requires all outputs to have the same impedance, therefore the same resistance will be added to the unused output ports, straight to ground. Since 50-ohms is not as widely available at 49 or 51, a 51-ohm resistor was chosen, as a 1-ohm difference is negligible. The Yageo **RT0805FRD0751RL** resistor was chosen since it passes all criteria: it is surface mount, thin film 51-ohms, large enough to place on the PCB with tweezers (case code – in: 0805), 1% tolerance, and a power rating of 1/8W, much larger than the maximum possible power from the DAC which is a maximum output of 45mW.

# RF Receiver

The recipient will be slightly different from the transmitter. It will first and foremost have an LNA (low noise amplifier) instead of a PA (power amplifier). The general idea s

#### First Stage: Switch

There are 2 types of switches that can be used, either mechanical relays or solid-state switches. Here are the pros and cons of each:

|  |  |  |
| --- | --- | --- |
| **Feature** | **Mechanical Relay** | **Solid-State RF Switch IC** |
| **Switching speed** | Slow: milliseconds12 | Fast: nanoseconds13 |
| **Control voltage/current** | Often higher coil current (10–100 mA) | Logic-level control (3.3–5 V, few mA) |
| **Isolation** | Excellent14 (~60–80 dB typical) | Good15 (~30–50 dB typical) |
| **Insertion loss** | Lower16 | Low |
| **Power handling** | High17 (can handle several watts easily) | Lower18 (typically mW to <1 W unless specialized device) |
| **Linearity / distortion** | Excellent (almost perfect, purely passive contact) | Good, but may produce nonlinearities at high RF power |
| **Lifetime** | Limited (~1–10 million cycles) | Very high19 (billions of cycles) |
| **Size / PCB integration** | Bulky, often requires space for coil | Tiny, surface-mount, PCB-friendly |
| **Cost** | Moderate to high for RF-rated relay | Varies; low-cost ICs available (~$2–6) |
| **Susceptibility to vibration/shock** | Can be affected, since it is mechanical | Very robust |
| **EMI** | Can generate contact bounce transients | Minimal, clean switching |

Since our receivers will switch very frequently, solid state switches must be bought. However, these switches are very expensive, over $20.

In the end, it was concluded that a diode would be used, and some DC controlled transistors to change the bias of the diode. A diode to handle the RF instead of a transistor since diodes have a larger linear range than transistors in RF frequencies with less complex circuits.20 More about these diodes is on the website quoted in footnote 20. Then most probably a DC powered transistor will be used to change the bias of the diode, allowing for a cheap, efficient, and easily MCU controlled receiver protection circuit. Exactly which diode will be figured out later. When the diode is forward biased, it acts like a short circuit to ground, reflecting RF energy, and when reversed biased it acts like a small capacitor, allowing small RF signals to pass. Make sure to have RF chokes to allow DC bias to reach the diode but blocks RF energy going to the DC power supply.

#### Second Stage: LNA

The design will either utilize the THS/SPF5189Z or a custom high-speed RF amplifier implemented using discrete transistors. Designing a discrete RF amplifier is challenging; therefore, the remaining RF components will be designed and validated first. To verify correct operation of the overall RF chain, a commercially available amplifier IC will initially be used for this stage. After system-level performance has been confirmed, this amplification stage will be redesigned using a custom transistor-based solution.

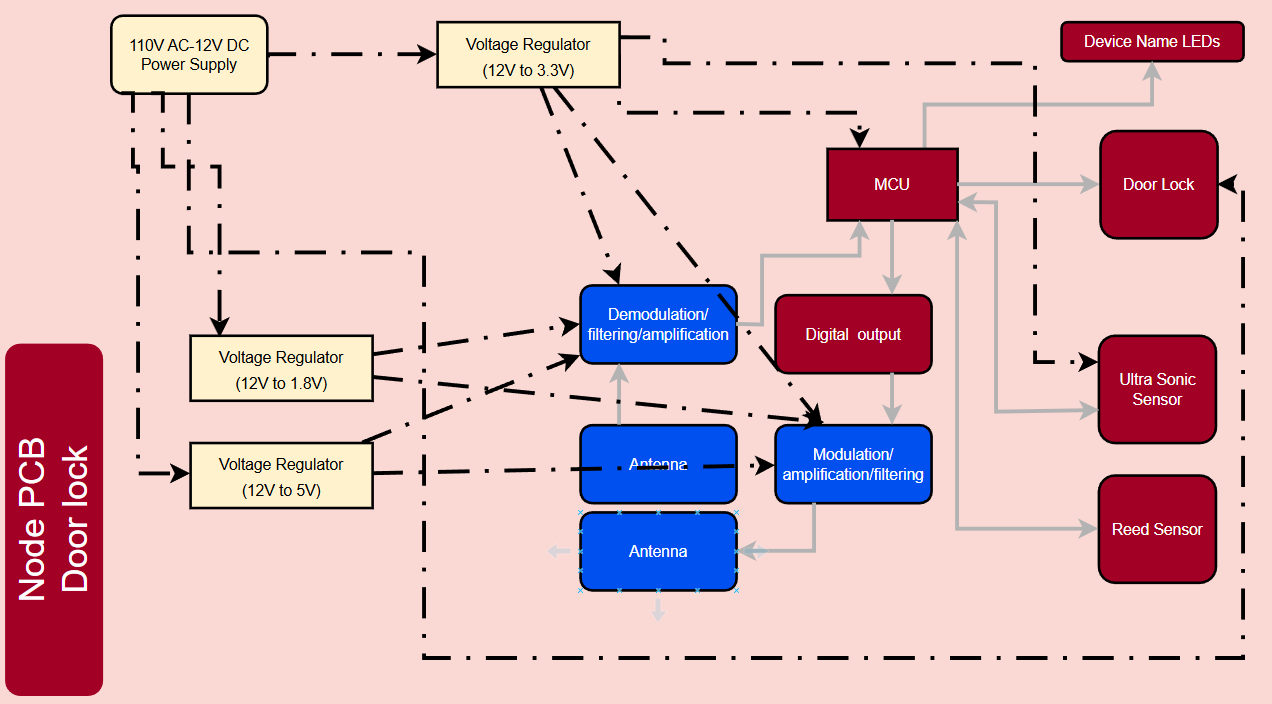
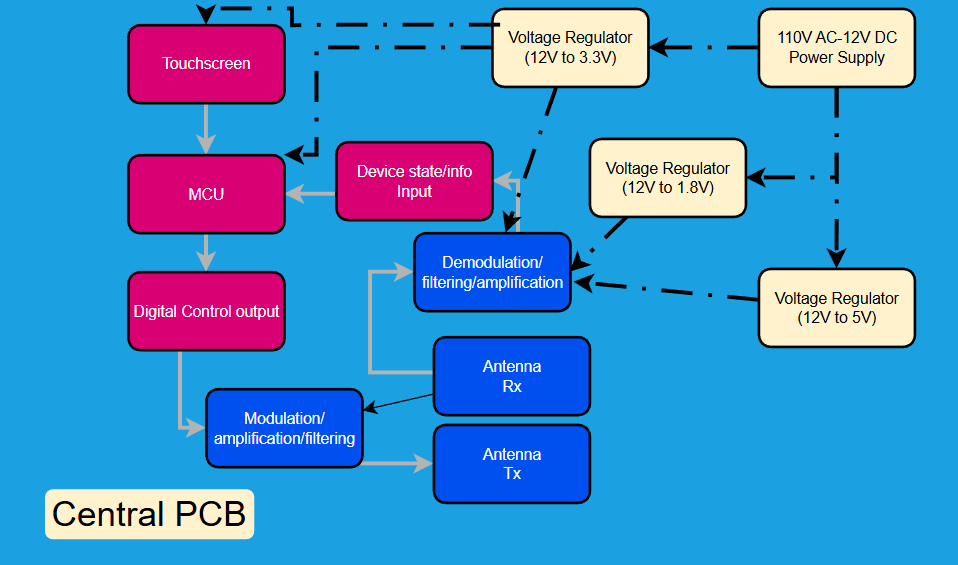
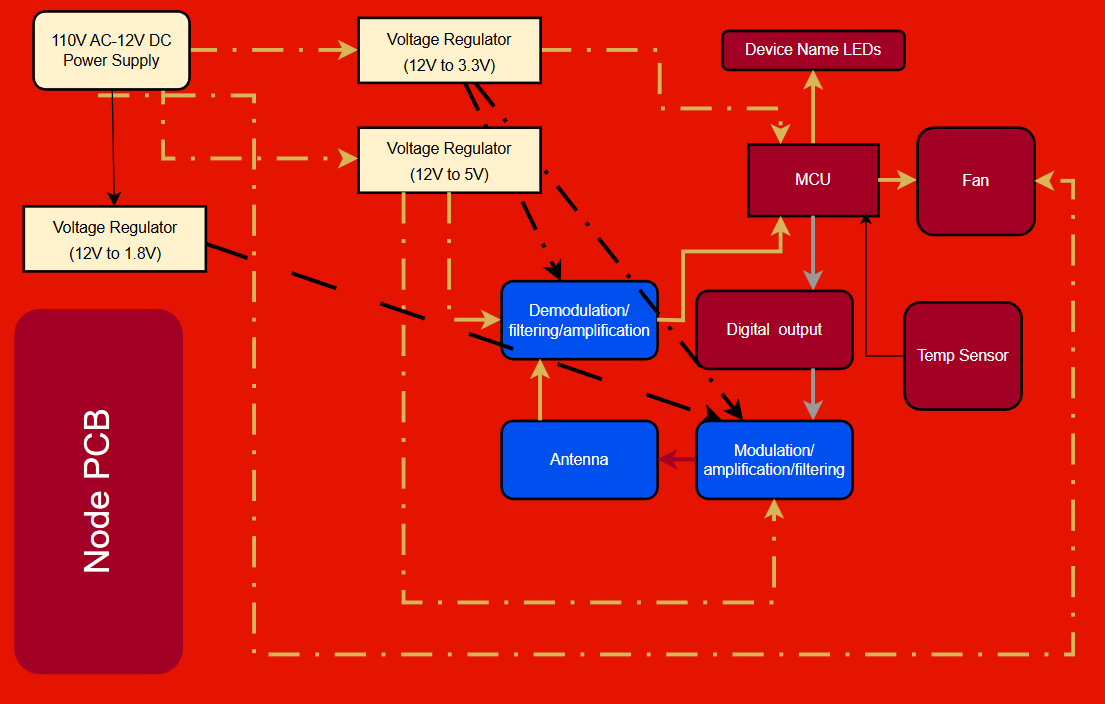
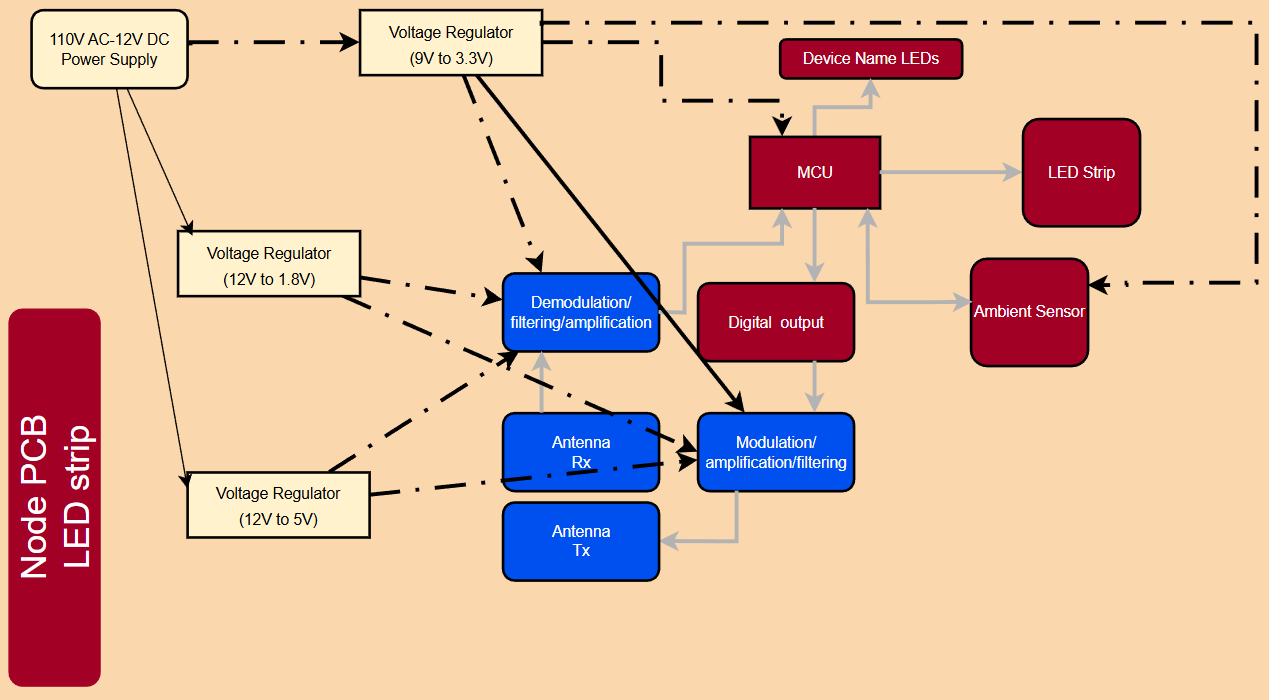
#### Third Stage: Downconverter

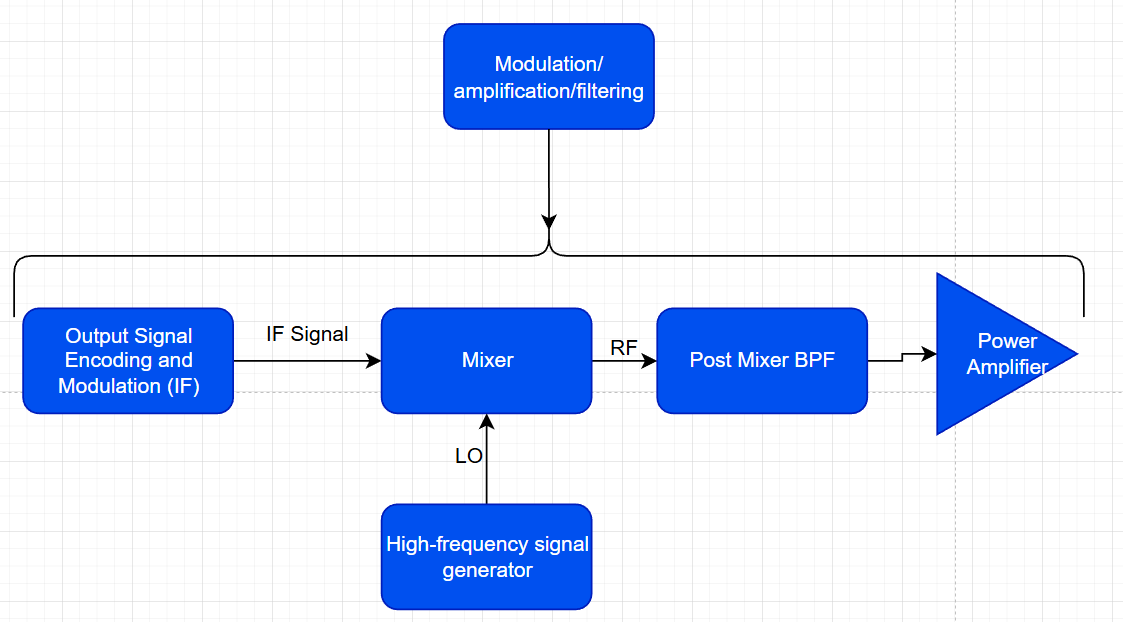
At first the idea was to take the input quadrature signal then down convert to 10-50MHz again, and then use an ADC, however ADCs in that speed range are very expensive. The new idea will be to use something like the ADL5380 chip, which is a quadrature demodulator, which will demodulate the quadrature signal down to 1MHz, where ADC’s are much less expensive.

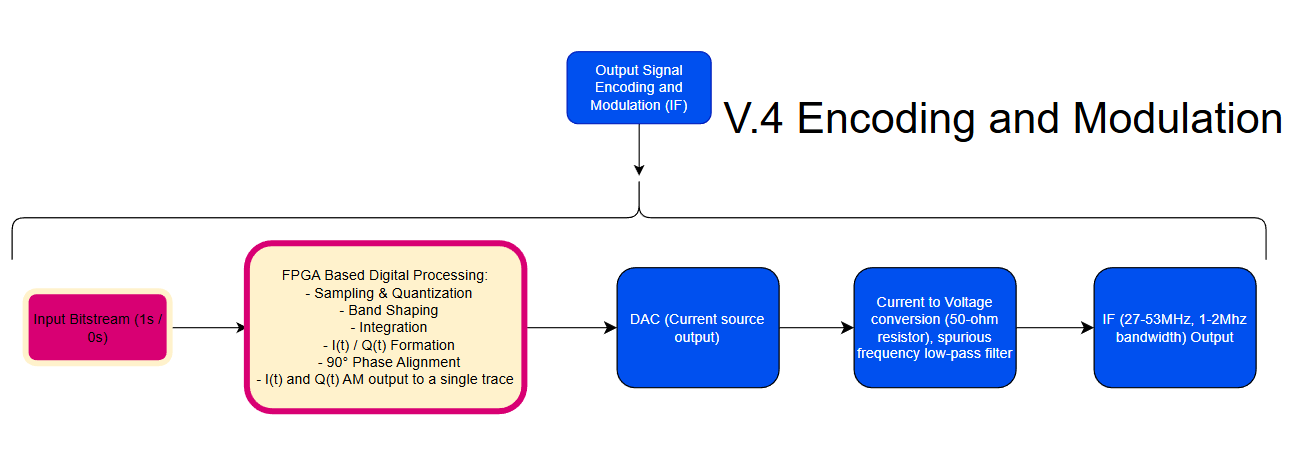
#### Output Power Limits

There are 2 different rulesets for output power, one ruleset is if your bandwidth is under 500kHz, and one is when your output power is above 500kHz. If the bandwidth is under 500kHz, you must implement a band hopping algorithm, which seems very time consuming, and a bit over scope for our project. Therefore, our bandwidth must be 500kHz or more. The transmission requirements at that frequency bandwidth are as follows: "§ 15.247 (a)(2): Systems using digital modulation techniques may operate in the 902-928 MHz ... bands. **The minimum 6 dB bandwidth shall be at least 500 kHz.21**" Since that bandwidth is being operated on in this project, our system is classified as a “system using digital modulation” and **not** a “frequency hopping system.” The maximum allowed output power of our transmitter is defined as: "§ 15.247 (a)(3): For systems using digital modulation in the 902-928 MHz, 2400-2483.5 MHz, and 5725-5850 MHz bands: 1 Watt. As an alternative to a peak power measurement, compliance with the one-Watt limit can be based on a measurement of the maximum conducted output power. Maximum Conducted Output Power is defined as the total transmit power delivered to all antennas and antenna elements averaged across all symbols in the signaling alphabet when the transmitter is operating at its maximum power control level.”

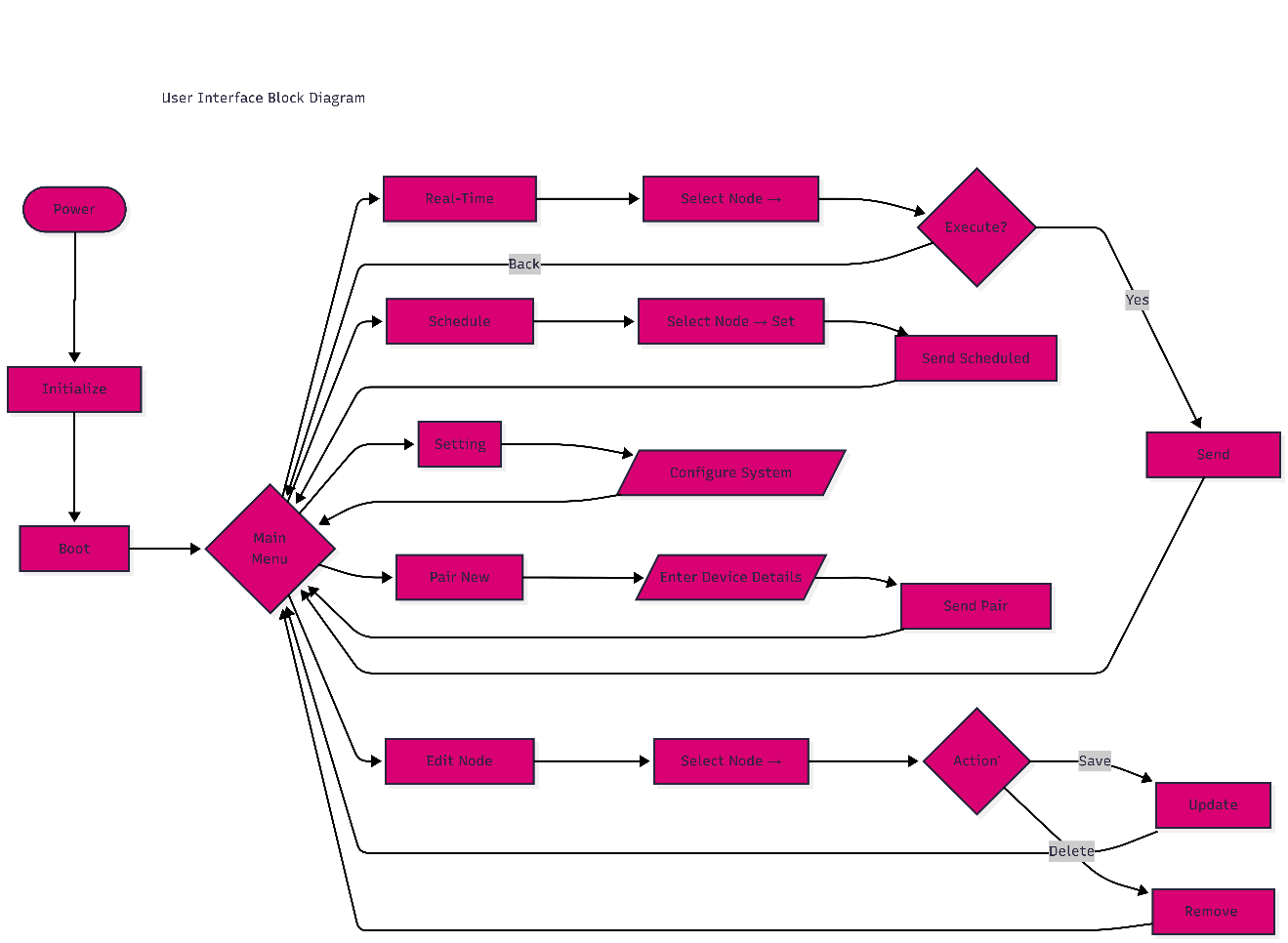
# 2.6 General Hardware Block Diagram

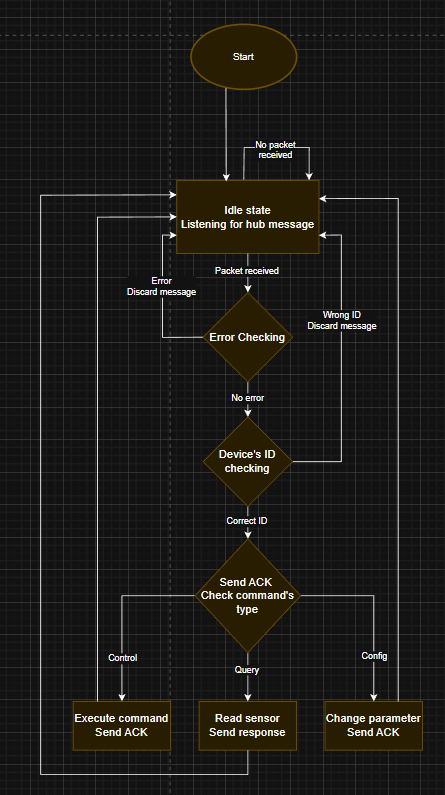






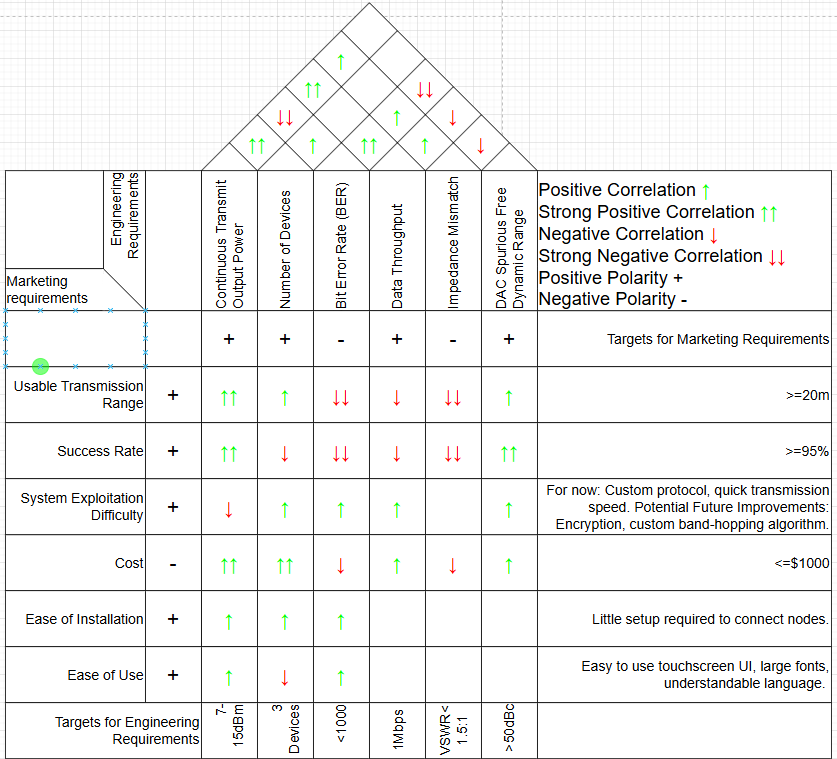
# 2.7 General Software Block Diagrams





# 2.8 Prototype illustration / blueprint

# 2.9 House of Quality



# 10.1 Budget and Financing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item** | **Vendor Source** | **Qty** | **Unit Cost (USD)** | **Subtotal (USD)** | **Notes** |
| ESP32-S3 DevKitC-1 | Mouser / DigiKey | 1 | $13.30 | $13.30 | Central hub MCU dev |
| ESP32-C3 Dev Board | Amazon / AliExpress | 2 | $7.00 | $14.00 | Node MCU dev |
| AD9850 DDS Eval Module | Amazon / Walmart | 1 | $16.20 | $16.20 | IF + FM generation |
| ADF4351 PLL Eval Module | Amazon / Newegg | 1 | $44.00 | $44.00 | LO generation |
| RF PA Eval Module | Amazon | 1 | $20.00 | $20.00 | Transmit amplifier |
| RF LNA Eval Module | Amazon | 1 | $20.00 | $20.00 | Receive front end |
| ILI9341 2.8” Touchscreen | Amazon | 1 | $18.00 | $18.00 | Hub UI |
| **Eval Board Subtotal** |  |  |  | **$165.50** |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item** | **Vendor** | **Qty** | **Unit Cost (USD)** | **Subtotal (USD)** | **Notes** |
| AD9850 DDS IC | DigiKey | 1 | $9.50 | $9.50 | IF generation |
| ADF4351 PLL IC | DigiKey | 1 | $22.00 | $22.00 | LO synthesis |
| RF Power Amplifier IC | DigiKey | 1 | $12.00 | $12.00 | TX stage |
| RF Low-Noise Amplifier IC | DigiKey | 1 | $10.00 | $10.00 | RX stage |
| ESP32-S3 MCU (bare) | DigiKey | 1 | $6.00 | $6.00 | Hub controller |
| ESP32-C3 MCU (bare) | DigiKey | 2 | $5.00 | $10.00 | Node controllers |
| RF diodes (RX switch) | DigiKey | — | $5.00 | $5.00 | PIN / fast diodes |
| Bias transistors | DigiKey | — | $3.00 | $3.00 | Diode control |
| RF passives (R/L/C) | DigiKey | — | $20.00 | $20.00 | Filters, matching |
| SMA connectors (PCB-mount) | DigiKey | 4 | $3.00 | $12.00 | RF I/O |
| **Bare Components Subtotal** |  |  |  | **$109.50** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **PCB Type** | **Qty** | **Cost (USD)** | **Notes** |
| Digital PCB (Hub + Nodes) | 3 | $30.00 | FR-4 |
| Analog / Baseband PCB | 2 | $20.00 | FR-4 |
| RF PCB (FR-4, tuned) | 3 | $45.00 | Iteration expected |
| **PCB Subtotal** |  | **$95.00** |  |

|  |  |  |
| --- | --- | --- |
| **Item** | **Cost (USD)** | **Notes** |
| 900 MHz Antennas | $18.00 | Hub + nodes |
| RF cables & adapters | $12.00 | SMA interconnect |
| Regulators & power modules | $10.00 | 3.3 V rails |
| Breadboards, headers, wire | $10.00 | Prototyping |
| Debug / replacement buffer | $25.00 | Component failures |
| **Misc Subtotal** | **$75.00** |  |

|  |  |
| --- | --- |
| **Category** | **Cost (USD)** |
| Evaluation Boards | $165.50 |
| Bare Components | $104.50 |
| PCB Fabrication | $95.00 |
| Miscellaneous | $75.00 |
| **Total Estimated Budget** | **$440.00** |

# 10.2 Project milestones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Description** | **Responsible** | **Start Date** | **End Date** |
| System Block Diagram | Partition digital, analog, and RF subsystems | All | Jan 27 | Jan 31 |
| Interface Definition | Define Digital–Analog–RF electrical interfaces | EE, CPEs | Jan 29 | Feb 4 |
| Communication Protocol | Packet structure, addressing, timing | CPEs | Jan 29 | Feb 4 |
| RF Link Budget | Power levels, bandwidth, FCC limits | EE | Jan 27 | Feb 3 |
| Hub MCU Bring-Up | ESP32-S3 boot, SPI, touchscreen | CPE | Jan 27 | Feb 6 |
| Node MCU Bring-Up | ESP32-C3 GPIO timing tests | CPE | Jan 29 | Feb 6 |
| DDS Evaluation | AD9850 SPI control and output testing | EE | Feb 1 | Feb 9 |
| System Block Diagram | Partition digital, analog, and RF subsystems | All | Jan 27 | Jan 31 |
| Interface Definition | Define Digital–Analog–RF electrical interfaces | EE, CPEs | Jan 29 | Feb 4 |
| Communication Protocol | Packet structure, addressing, timing | CPEs | Jan 29 | Feb 4 |
| RF Link Budget | Power levels, bandwidth, FCC limits | EE | Jan 27 | Feb 3 |
| Hub MCU Bring-Up | ESP32-S3 boot, SPI, touchscreen | CPE | Jan 27 | Feb 6 |
| Node MCU Bring-Up | ESP32-C3 GPIO timing tests | CPE | Jan 29 | Feb 6 |
| DDS Evaluation | AD9850 SPI control and output testing | EE | Feb 1 | Feb 9 |
| Hub Firmware Core | Task scheduling, SPI, RF control | CPE | Feb 10 | Feb 24 |
| Node Firmware Core | TX/RX finite state machine | CPE | Feb 10 | Feb 24 |
| Encoding Logic | Manchester / baseband encoding | CPE | Feb 17 | Mar 2 |
| Reliability Layer | ACKs, retries, timeouts | CPE | Feb 24 | Mar 9 |
| GUI Framework | Touchscreen UI development | CPE | Feb 17 | Mar 9 |
|  |  |  |  |  |
| Baseband Filter Design | RC / active filter simulation | EE | Feb 10 | Feb 23 |
| Signal Scaling & Biasing | Level shifting and buffering | EE | Feb 17 | Mar 2 |
| RX Signal Conditioning | Comparator and recovery circuitry | EE | Feb 24 | Mar 9 |
| Analog Schematic | Complete baseband schematic | EE | Mar 2 | Mar 16 |
| Bench Validation | Breadboard testing | EE | Mar 9 | Mar 16 |
| IF FM/FSK Stage | AD9850 modulation testing | EE | Feb 17 | Mar 2 |
| LO Generation | ADF4351 configuration | EE | Feb 24 | Mar 9 |
| Mixer Design | Passive diode ring mixer | EE | Mar 2 | Mar 16 |
| RX Protection Switch | Diode-based RX switch | EE | Mar 2 | Mar 16 |
| RF Front-End Planning | PA, LNA, filtering | EE | Mar 9 | Mar 16 |
| Digital → Analog Integration | MCU to baseband interface | All | Mar 17 | Mar 23 |
| Analog → RF Integration | IF injection and matching | EE | Mar 20 | Mar 30 |
| TX Chain Validation | Digital to RF output | All | Mar 23 | Apr 2 |
| RX Chain Validation | RF to digital recovery | All | Mar 23 | Apr 2 |
| Multi-Node Testing | Control of ≥2 nodes | CPEs | Mar 30 | Apr 6 |
| **Section** | **Responsible** | **Start Date** | **End Date** | **Page Target** |
| Digital Firmware | CPE | Mar 17 | Apr 28 | ~30 |
| Analog Baseband Design | EE | Mar 17 | Apr 28 | ~30 |
| RF System Design | EE | Mar 17 | Apr 28 | ~30 |
| GUI & Integration | CPE | Mar 17 | Apr 28 | ~30 |
| Editing & Review | All | Apr 14 | Apr 28 | — |
| **PCB** | **Task** | **Responsible** | **Start Date** | **End Date** |
| Digital PCB | Hub and node schematics | EE, CPE | May 1 | May 10 |
| Analog PCB | Baseband layout | EE | May 1 | May 10 |
| RF PCB | RF layout and grounding | EE | May 11 | May 20 |
| Design Review | ERC/DRC checks | All | May 21 | May 23 |
| Fabrication Order | FR-4 board order | EE | May 24 | May 31 |
| **Task** | **Description** | **Responsible** | **Start Date** | **End Date** |
| Power Validation | Verify all rails | EE | Jun 1 | Jun 5 |
| Firmware Programming | Flash hub and nodes | CPEs | Jun 3 | Jun 7 |
| Analog Debug | Signal integrity testing | EE | Jun 6 | Jun 15 |
| RF Debug | Spectrum and power testing | EE | Jun 10 | Jun 25 |
| Full System Debug | End-to-end testing | All | Jun 15 | Jun 30 |
| Final Assembly | All PCBs connected | All | Jul 1 | Jul 5 |
| Validation Testing | Range and latency | All | Jul 3 | Jul 9 |
| Demo Preparation | Final demo readiness | All | Jul 7 | Jul 11 |

# Appendix A: Reference

1. <https://documentation.espressif.com/esp32-s3_datasheet_en.pdf>
2. <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf>
3. <https://documentation.espressif.com/esp32-c3_datasheet_en.pdf>
4. <https://www.ti.com/product-category/microcontrollers-processors/mcus/msp430/overview.html?utm_source=chatgpt.com>
5. <https://www.electronics-tutorials.ws/rc/rc_3.htmls>
6. <https://en.wikipedia.org/wiki/Elliptic_filter>
7. <https://en.wikipedia.org/wiki/Manchester_code>
8. [https://www.biolinscientific.com/blog/does-the-qcm-fundamental-frequency-matter#:~:text=1)%20Crystal%20thickness&text=For%20example%2C%20a%205%20MHz,a%20thickness%20of%20~167%20um](https://www.biolinscientific.com/blog/does-the-qcm-fundamental-frequency-matter" \l ":~:text=1)%20Crystal%20thickness&text=For%20example%2C%20a%205%20MHz,a%20thickness%20of%20~167%20um).(“For example, a 5 MHz QCM crystal will have a thickness of ~334 um….”)
9. <https://www.dynamicengineers.com/content/how-a-pll-phase-locked-loop-works-a-detailed-overview> (“...a base station may use a reference frequency from a highly stable crystal oscillator and lock its output frequency using a PLL to ensure precise carrier generation…”)
10. <https://www.alldatasheet.com/datasheet-pdf/view/539256/AD/AD9850SLASHCGPCB.html>
11. <https://books.google.com/books?id=6SBTAAAAMAAJ&focus=searchwithinvolume&q=Marchand+balun>
12. <https://www.everythingrf.com/search/relays> “An electro-mechanical RF Relay has its switching time in milli-seconds(ms)...”
13. <https://www.everythingrf.com/search/relays> “...and a solid-state RF Relay switching time is expressed in nanoseconds (ns).”
14. <https://www.everythingrf.com/search/relays> “Electromechanical RF Relays exhibit excellent isolation…”
15. <https://www.everythingrf.com/search/relays> “...whereas solid-state RF Relays have good isolation.”
16. <https://www.everythingrf.com/search/relays> “Electromechanical RF Relays usually provide the lowest possible loss as compared to solid state relays.”
17. <https://www.everythingrf.com/search/relays> “...RF Relay switches hence have a higher power handling capability….”
18. <https://www.everythingrf.com/search/relays> “...solid-state Relay has a low power handling profile”
19. <https://www.everythingrf.com/search/relays> “…as their cycles are usually much higher so we don’t consider it a factor.”
20. <https://www.analog.com/en/resources/technical-articles/cmos-switches-offer-high-performance-in-low-power-wideband-applications.html> “PIN diodes have commonly been used for switching RF signals, as they are highly linear when conducting and exhibit very good distortion characteristics.”
21. <https://www.ecfr.gov/current/title-47/chapter-I/subchapter-A/part-15/subpart-C/subject-group-ECFR2f2e5828339709e/section-15.247>

# Appendix C: Etc.

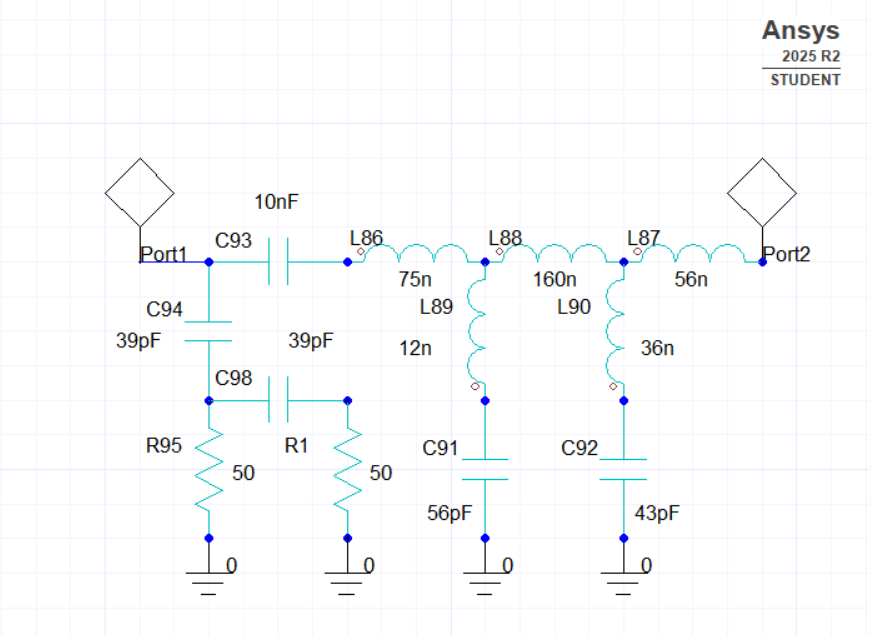


Figure 1

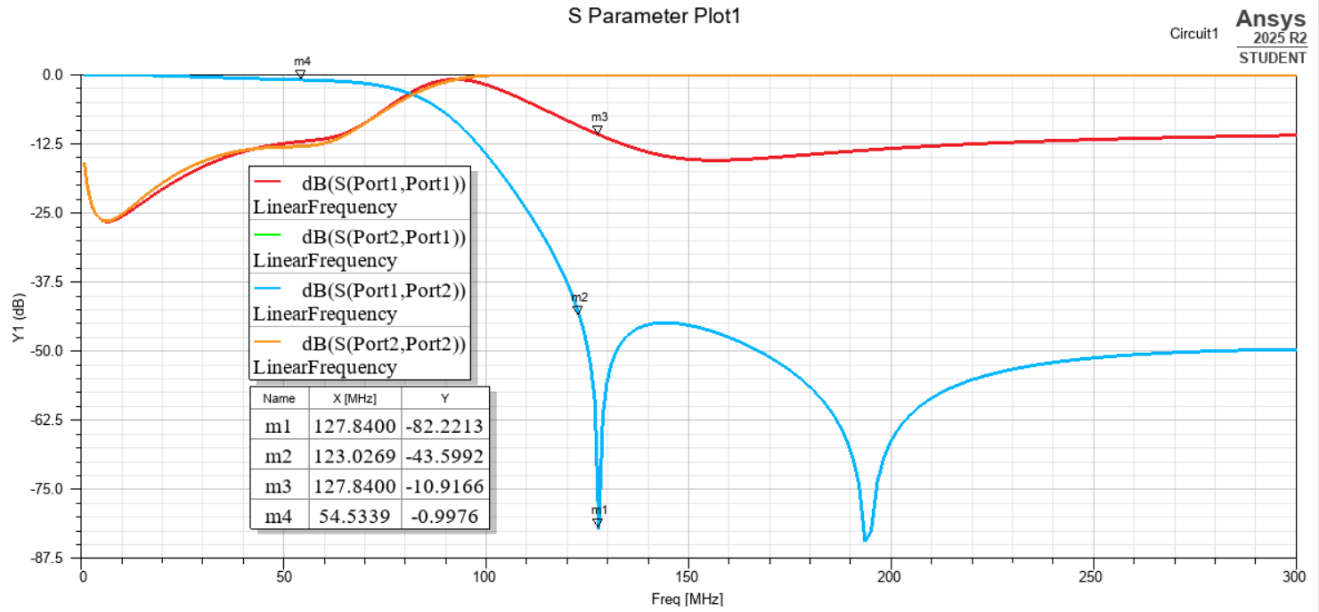


Figure 2

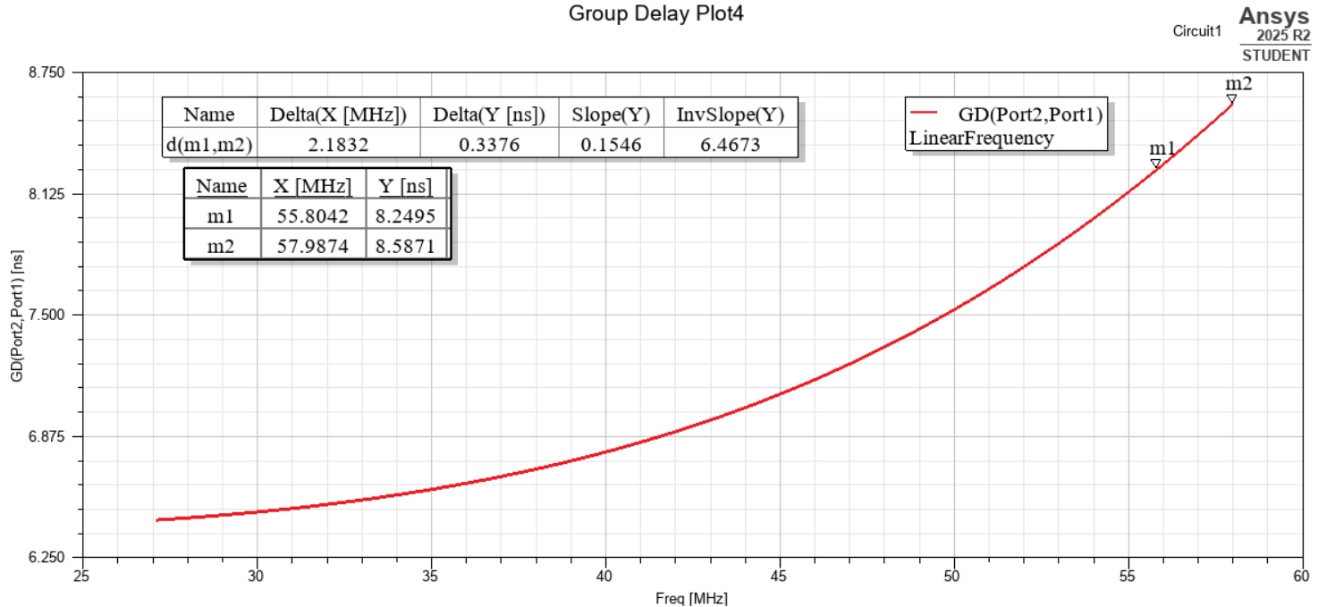


Figure 3

**Figure 1:** This figure shows the elliptic bandpass filter that will be used between the DAC and the mixer IF port. All components have been chosen to fit withing E24 passive element size standard. This network is made of firstly a second order high pass filter, to lower S11, allowing for most of the reflected power coming back from the low pass elliptic filter to be attenuated, instead of being absorbed by the DAC. The circuit also has a 10nF DC blocking capacitor, and the last part is a third order elliptic low pass filter.

**Figure 2:** This figure shows the S parameters of the circuit shown in figure 1. Originally S11=S22 since this was a passive reciprocal network, however after adding the 2nd order high pass filter, S11 greatly dropped in the stopband.

**Figure 3:** This figure shows the group delay. Even though it looks that the group delay has a large difference, this is the total group delay for all possible frequency inputs, from 27-53Mhz. Since the bandwidth of any of these input frequencies will be at maximum 2Mhz, the highest possible group delay difference would be what was highlighted on the table that is on the figure, which actually is a group delay difference of .3376ns, so group delay shouldn’t be a problem in this circuit.