

**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE CIENCIAS FÍSICAS**

DEPARTAMENTO DE ESTRUCTURA DE LA MATERIA, FÍSICA TÉRMICA Y  
ELECTRÓNICA



**TRABAJO DE FIN DE GRADO**

Código de TFG: [Código TFG]

Sistema de comunicaciones inalámbricas

Wireless communication system

Supervisor/es: Javier Olea Ariza

**Jose Luis Gutiérrez Moreno**

Grado en Ingeniería Electrónica de Comunicaciones

Curso académico 2023-2024

Convocatoria XXXX

[Título extendido del TFG (si procede)]

## Resumen:

### **Abstract:**

Nota: el título extendido (si procede), el resumen y el abstract deben estar en una misma página y su extensión no debe superar una página. Tamaño mínimo 11pto.

Extensión máxima 50 páginas sin contar portada, contraportada y declaración responsable (sí se incluye índice, introducción, conclusiones y bibliografía)



UNIVERSIDAD  
COMPLUTENSE  
MADRID

**Declaración Responsable sobre Autoría y Uso Ético de**  
**Herramientas de Inteligencia Artificial (IA)**

Yo, **APELLIDOS Y NOMBRE**

Con DNI/NIE/PASAPORTE:

declaro de manera responsable que el/la presente:

- Trabajo de Fin de Grado (TFG)
- Trabajo de Fin de Máster (TFM)
- Tesis Doctoral

Titulado/a

**CAMPO PARA TÍTULO**

es el resultado de mi trabajo intelectual personal y creativo, y ha sido elaborado de acuerdo con los principios éticos y las normas de integridad vigentes en la comunidad académica y, más específicamente, en la Universidad Complutense de Madrid.

Soy, pues, autor del material aquí incluido y, cuando no ha sido así y he tomado el material de otra fuente, lo he citado o bien he declarado su procedencia de forma clara -incluidas, en su caso, herramientas de inteligencia artificial-. Las ideas y aportaciones principales incluidas en este trabajo, y que acreditan la adquisición de competencias, son mías y no proceden de otras fuentes o han sido reescritas usando material de otras fuentes.

Asimismo, aseguro que los datos y recursos utilizados son legítimos, verificables y han sido obtenidos de fuentes confiables y autorizadas. Además, he tomado medidas para garantizar la confidencialidad y privacidad de los datos utilizados, evitando cualquier tipo de sesgo o discriminación injusta en el tratamiento de la información.

En Madrid a **FECHA**

**FIRMA**

# Índice

<b>1. Introducción</b>	<b>4</b>
1.1. Objetivos y motivación . . . . .	4
<b>2. Marco Teórico</b>	<b>4</b>
2.1. Sistemas de comunicación y Transmisión ASK . . . . .	4
2.2. Transistores y sus parámetros característicos . . . . .	6
2.3. Realimentación . . . . .	8
2.4. Sistemas integrados: Atmega328p . . . . .	10
<b>3. Desarrollo</b>	<b>13</b>
3.1. Parte analógica . . . . .	14
3.1.1. Diseño de la parte analógica de RF del transmisor . . . . .	14
3.1.2. Diseño de la parte analógica de RF del receptor . . . . .	23
3.2. Parte digital . . . . .	30
3.2.1. Diseño del demodulador digital para la recepción . . . . .	32
3.2.2. Diseño del codificador digital para la transmisión . . . . .	36
<b>4. Cronología del proyecto y diagrama de Grant</b>	<b>40</b>
4.1. transmisor FM a varactor . . . . .	40
4.2. receptor superheterodino FM . . . . .	41
4.3. Máquina de estados digital . . . . .	42
4.4. Alternativa viable: conversión directa . . . . .	44
4.5. Diagrama de Grant . . . . .	45
<b>5. Resultados y conclusiones</b>	<b>46</b>
<b>6. Bibliografía</b>	<b>46</b>
<b>7. Indice de figuras</b>	<b>46</b>

## 1. Introducción

La radiofrecuencia tiene sus raíces en el descubrimiento de las ondas electromagnéticas. Este hallazgo abrió las puertas a las primeras aplicaciones prácticas de la transmisión inalámbrica. Desde entonces, la tecnología de radiofrecuencia ha evolucionado exponencialmente, convirtiéndose en un pilar de las comunicaciones modernas.

Este proyecto se trata de un sistema de comunicaciones de radio digital que pretende ser un circuito de control inalámbrico para aplicaciones generales. El objetivo principal de este proyecto es dar solución a un problema real, en este caso la reparación de un sistema calefactor, mediante un diseño propio de un sistema de comunicaciones elaborado desde cero. El sistema de comunicaciones se compone de emisor y receptor. Estos poseen una estructura similar: una etapa analógica que trabaja con las señales de radio, y una etapa digital, basada en un microcontrolador programable. Este microcontrolador implementa la codificación del sistema y trabaja con las señales en banda base.

### 1.1. Objetivos y motivación

La elección de este proyecto surge de mi interés tanto por las bases teóricas de las comunicaciones inalámbricas como por la posibilidad de aplicar estos conocimientos en un contexto práctico. Además, valoro la libertad sujeta al desarrollo del proyecto, ya que me permite realizar un trabajo creativo y personal, para llegar a resolver un problema real. Los objetivos del proyecto son los siguientes:

1. Recorrer íntegramente el proceso de diseño de un sistema de comunicaciones de radiofrecuencia al más bajo nivel posible.
2. Resolver los problemas adjuntos al proceso de diseño.
3. Poner en práctica los conceptos teóricos adquiridos.
4. Relacionar los conceptos de los diferentes campos teóricos estudiados en el grado.
5. Proponer soluciones sencillas y funcionales para dar al proyecto una aplicación real.

## 2. Marco Teórico

### 2.1. Sistemas de comunicación y Transmisión ASK

Una comunicación inalámbrica tiene como objetivo el intercambio de información a través de un medio de propagación no guiado. En este trabajo, se realizará la comunicación inalámbrica por medio de radiofrecuencia. Esta técnica consiste en acoplar la señal eléctrica que contiene la información a transmitir, a una señal de alta frecuencia. La señal de información se denomina moduladora, mientras que la señal de radiofrecuencia es llamada portadora. La acción de separar la señal portadora de la moduladora se denomina demodulación.

Los elementos que realizan la comunicación son emisor y receptor. La calidad de estos elementos viene definida por las siguientes características<sup>1</sup>:

#### Receptor

---

<sup>1</sup>American Radio Relay League (ARRL). (1970). *The Radio Amateur's Handbook* (p. 94).

Las características principales que definen a un buen receptor son: **sensibilidad**, propiedad de recibir señales débiles; **selectividad**: propiedad de distinguir entre señales muy próximas en frecuencia, y **estabilidad**: propiedad de mantener de manera fiable una comunicación a lo largo del tiempo.

Cabe mencionar que, por la forma de diseño, los receptores se pueden clasificar en función del tipo de detección utilizada: regenerativos y super-regenerativos, que normalmente utilizan una conversión directa, o heterodinos y super-heterodinos, los cuales convierten la señal de radiofrecuencia recibida en una señal de frecuencia intermedia, favoreciendo el grado de selectividad principalmente. En general, los receptores super-heterodinos presentan mejores prestaciones a costa de una complejidad y coste mayor.

## Transmisor

La característica principal que define a un buen transmisor es la eficiencia de radiación. Esta medida es la relación entre la potencia transmitida a la antena y la potencia total consumida por el mismo. Idealmente este parámetro es:  $\eta = \frac{P_{rad}}{P_{in}} = 1$ . La potencia radiada, en esencia, es la potencia que se emite al canal de comunicación. Si se mantiene la eficiencia de radiación, y se aumenta la potencia del transmisor, se consigue un aumento lineal de la potencia radiada. Como resultado, se hace llegar la comunicación a mayor distancia.

También existen otros parámetros que se pueden considerar heredados, ya que son más propios de las antenas, como por ejemplo, la directividad. La mejora de estos parámetros es sustancial a la hora de diseñar un buen transmisor.

## Modulación ASK

La modulación ASK es un tipo de modulación digital que se basa en la transmisión de una señal digital en función de la emisión conmutada de una señal portadora, donde la recepción de esta señal representa un símbolo lógico, mientras que su ausencia representa otro. El esquema de la comunicación ASK se representa en la figura 1.

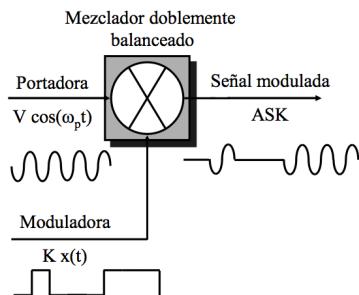


Figura 1: Esquema de una posible modulación ASK

## 2.2. Transistores y sus parámetros característicos

Un transistor es un dispositivo semiconductor de tres terminales: entrada, salida y terminal común, capaz de amplificar la corriente que circula a través de él. Las numerosas técnicas de fabricación de estos dispositivos dan lugar a los distintos tipos de transistores que existen: BJT, MOSFET, JFET, entre otros. En este trabajo se utilizarán transistores bipolares BJT, tanto NPN como PNP, para la realización del enlace de radio frecuencia, mientras que el microcontrolador utilizado en el apartado digital utiliza principalmente transistores de efecto campo MOSFET.

Los transistores BJT logran la amplificación gracias a dos uniones p-n interconectadas entre sí, donde un pequeño flujo de corriente de entrada regula un gran flujo de corriente de salida. Por otro lado, en un transistor MOSFET la amplificación se logra por medio de la regulación del estrechamiento del canal por donde circula la corriente, aplicando una tensión inversa.

**Polarización:** Para lograr que un transistor realice la función de amplificación, es necesario proporcionar al dispositivo las tensiones de trabajo adecuadas para que funcione de forma deseada. Esta acción es conocida como polarización del transistor. En función de la polarización aplicada, el transistor puede trabajar de diferentes formas, algunas de las cuales son: activa directa, donde se trabaja en amplificación de señales; o corte y saturación, que son utilizadas principalmente en circuitos digitales. Existen numerosas técnicas para lograr la polarización deseada, estas serán expuestas en los apartados de desarrollo correspondientes.

**Modelo de gran señal del transistor bipolar:** Existen numerosos modelos matemáticos para definir el comportamiento de un transistor, en este trabajo se utilizará el modelo SPICE del transistor caracterizado por el siguiente modelo circuital y sus ecuaciones características:

$$\begin{aligned} I_E &= \frac{I_{be}}{B_F} + I_{be} - I_{bc} & I_{be} &= I_s \cdot \left( e^{\left( \frac{V_{BE}}{N_T \cdot V_t} \right)} - 1 \right) \\ I_C &= I_{be} - \frac{I_{bc}}{B_R} - I_{bc} & I_{bc} &= I_s \cdot \left( e^{\left( \frac{V_{BC}}{N_T \cdot V_t} \right)} - 1 \right) \\ I_B &= \frac{I_{be}}{B_F} + \frac{I_{bc}}{B_R} \end{aligned}$$

Considerando una polarización en activa directa, donde la amplificación de señales se realiza de manera óptima, las ecuaciones quedan simplificadas al despreciar  $I_{bc}$ , ya que  $V_{CB} < 0$ . El resultado de las ecuaciones, a las cuales se aplica el efecto Early, que no se puede considerar despreciable, queda de la siguiente forma.

$$I_C = I_s \cdot \left( e^{\left( \frac{V_{BE}}{N_T \cdot V_t} \right)} - 1 \right) \cdot \left( 1 + \frac{V_{CE}}{V_{AF}} \right) \quad (1)$$

$$I_B = \frac{I_s}{B_F} \cdot \left( e^{\left( \frac{V_{BE}}{N_T \cdot V_t} \right)} - 1 \right) \quad (2)$$

**Pequeña señal:** Como se ha mencionado anteriormente, si se dispone de un transistor que trabaja en activa directa, se consigue la amplificación de las señales. Debido a que el transistor es un dispositivo no lineal, la señal introducida debe ser suficientemente pequeña para poder aproximar al dispositivo como un elemento lineal. Por tanto, a la hora de trabajar con pequeña señal, se modelará al transistor como un cuadripolo lineal de dos puertos. En función de las variables de entrada o salida que se elijan, el modelo circuital del cuadripolo variará. El modelo principal con el que se trabajará será el de parámetros híbridos, el cual se caracteriza por usar como variables

independientes  $i_1, v_2$  y dependientes  $i_2, v_1$ . Cabe mencionar la existencia de otras configuraciones de parámetros característicos como son el modelo de admitancias, cuyas variables independientes son  $v_1, v_2$ , o el modelo de impedancias, cuyas variables independientes son  $i_1, i_2$ . El modelo general del cuadripolo lineal así como sus parámetros de entrada y salida se muestran en la figura 2 a continuación:

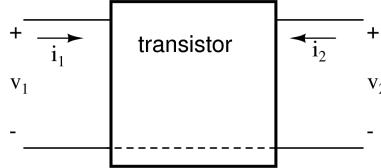


Figura 2: Representación de un transistor como cuadripolo lineal.

En general, las ecuaciones que describen el conjunto de los modelos son:

$$dY_1 = \frac{\partial f_1}{\partial X_1} \cdot \partial X_1 + \frac{\partial f_1}{\partial X_2} \cdot \partial X_2 \quad (3)$$

$$dY_2 = \frac{\partial f_2}{\partial X_1} \cdot \partial X_1 + \frac{\partial f_2}{\partial X_2} \cdot \partial X_2 \quad (4)$$

Donde  $X_1$  y  $X_2$  corresponden a las variables independientes y  $Y_1$  e  $Y_2$  a las variables dependientes

Las ecuaciones que definen al modelo de parámetros híbridos en concreto son por tanto:

$$v_1 = h_{11} \cdot i_1 + h_{12} \cdot v_2$$

$$i_2 = h_{21} \cdot i_1 + h_{22} \cdot v_2$$

El valor de los parámetros  $h_{nm}$  se deriva de las ecuaciones del modelo de gran señal, simplificadas para una polarización activa directa. Al derivar dichas ecuaciones se obtiene un modelo de parámetros en admitancias, fácilmente transformable al modelo de parámetros híbridos. Además, en función de la configuración del terminal común del transistor, el valor de los parámetros característicos cambiará, en este caso, se supondrá una configuración en emisor común. Los valores de los parámetros para las distintas configuraciones pueden ser deducidos utilizando relaciones matemáticas sencillas. A continuación se expone la obtención de los valores de los parámetros híbridos, para la configuración de emisor común, a partir de las ecuaciones de gran señal, ecuaciones 1 y 2.

$$\begin{cases} i_b = \frac{\partial I_B}{\partial V_{BE}} \cdot v_{be} + \frac{\partial I_B}{\partial V_{CE}} \cdot v_{ce} \\ i_c = \frac{\partial I_C}{\partial V_{BE}} \cdot v_{be} + \frac{\partial I_C}{\partial V_{CE}} \cdot v_{ce} \end{cases} \Rightarrow \begin{cases} i_b = y_{11} \cdot v_{be} + y_{12} \cdot v_{ce} \\ i_c = y_{21} \cdot v_{be} + y_{22} \cdot v_{ce} \end{cases}$$

$$\begin{cases} v_{be} = h_{ie} \cdot i_b + h_{re} \cdot v_{ce} \\ i_c = h_{fe} \cdot i_b + h_{oe} \cdot v_{ce} \end{cases} \Rightarrow \begin{cases} v_{be} = \frac{1}{y_{11}} \cdot i_b + \left( \frac{-y_{12}}{y_{11}} \right) \cdot v_{ce} \\ i_c = \frac{y_{21}}{y_{11}} \cdot i_b + \left( y_{22} - \frac{y_{12} \cdot y_{21}}{y_{11}} \right) \cdot v_{ce} \end{cases}$$

Teniendo en cuenta las relaciones obtenidas, se puede establecer la relación de los parámetros híbridos con su valor numérico como se muestra a continuación.

$$h_{ie} = \frac{1}{y_{11}} = \frac{N_F \cdot V_t}{I_B} \quad h_{re} = \frac{-y_{12}}{y_{11}} \approx 0 \quad (5)$$

$$h_{fe} = \frac{y_{21}}{y_{11}} = \frac{I_C}{I_B} \quad h_{oe} = y_{22} - \frac{y_{12} \cdot y_{21}}{y_{11}} \approx \frac{I_C}{V_{AF}} \quad (6)$$

Seguidamente, se establecen las relaciones de los parámetros híbridos en emisor común para las demás configuraciones de terminal común. Las relaciones se obtienen de manera similar a las anteriores dadas:

$$\begin{array}{ll} h_{ic} = h_{ie} & h_{rc} = 1 - h_{re} \\ h_{fc} = -(1 + h_{fe}) & h_{oc} = h_{oe} \end{array} \quad \left| \quad \begin{array}{ll} h_{ib} = \frac{h_{ie}}{1+h_{fe}} & h_{rb} = \frac{h_{ie} \cdot h_{oe}}{1+h_{fe}} - h_{re} \\ h_{fb} = -\frac{h_{fe}}{1+h_{fe}} & h_{ob} = \frac{h_{oe}}{1+h_{fe}} \end{array} \right. \quad (7)$$

Por último, cabe mencionar que los transistores, por su construcción física, poseen elementos denominados parásitos. Estos elementos se modelan, de manera circuital principalmente como condensadores, y pueden influir en los valores de las impedancias de entrada, salida o realimentaciones. Estos efectos se incluirán cuando sean necesarios en el desarrollo del proyecto<sup>2</sup>.

### 2.3. Realimentación

La realimentación es un concepto ligado a la amplificación. La realimentación, en los amplificadores, consiste en tomar una muestra de tensión o de corriente a la salida y reenviarla a la entrada a través de una red apropiada. Gracias a la realimentación se consigue estabilizar la ganancia, la resistencia de entrada, la resistencia de salida y el ancho de banda, aunque el fin más importante es la estabilización de la ganancia<sup>3</sup>.

En la figura 3 se muestra el esquema general de un sistema realimentado, donde  $A$  se corresponde con la ganancia en lazo abierto del amplificador, y  $f$  la red de realimentación correspondiente. Dependiendo de la naturaleza del amplificador y del tipo de realimentación, la ganancia en lazo abierto puede ser de tensión, corriente, transimpedancia o transconductancia.

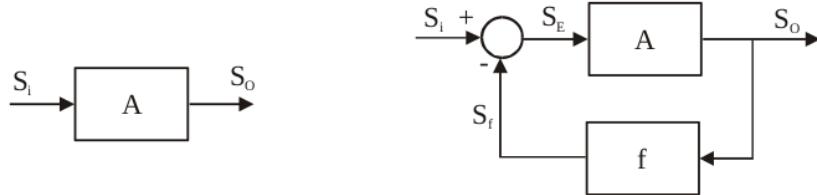


Figura 3: derecha: Amplificador sin realimentación, izquierda: Sistema realimentado

En la figura 3 se calculan las ganancias propias de cada sistema. La nomenclatura  $S_n$  se refiere a una señal que bien puede ser de corriente o tensión. Por un lado, la ganancia en lazo abierto del amplificador se calcula como:

$$A = \frac{S_o}{S_i}$$

Por otro lado, se calcula la ganancia en lazo cerrado  $A_f$ :

$$\begin{cases} S_f = S_o \cdot f \\ S_E = S_i - S_f \\ S_o = S_E \cdot A \end{cases} \Rightarrow \begin{aligned} \frac{S_o}{A} &= S_i - S_o \cdot f \\ S_i &= S_o \cdot \left( \frac{1}{A} + f \right) = S_o \cdot \frac{1}{A} \cdot (1 + f \cdot A) \\ A_f &= \frac{S_o}{S_i} = \frac{A}{(1+f \cdot A)} \end{aligned}$$

<sup>2</sup>Franco Peláez, F. J., González Díaz, G., & Mártil de la Plaza, I. (n.d.). *Apuntes de electrónica analógica*.

<sup>3</sup>Revista Todoeléctronica. (n.d.). Fascículo segundo.

Se señalan las ecuaciones que serán de utilidad:

$$A_f = \frac{S_o}{S_i} = \frac{A}{(1 + f \cdot A)} \quad (8)$$

$$A = \frac{S_o}{S_E} \quad (9)$$

$$A_l = A \cdot f \quad (10)$$

Es útil definir el parámetro ganancia en lazo abierto,  $A_l = A \cdot f$ , para poder analizar el comportamiento del sistema en lazo cerrado cuando  $A_l$  varía. Esto se conoce como el criterio de Barkhausen, el cual se expone considerando el criterio de signos de la ecuación 8:

- **Si  $A_l \gg 1$ :** en este caso se obtiene una ganancia total del sistema  $A_f = \frac{1}{f}$ . Esta realimentación se conoce como negativa.
- **Si  $A_l \ll 0$ :** en este caso se tiene que  $S_E = S_i + S_f$ , es decir, las señales se encuentran en fase y se suman en lugar de restarse. Esta suma es amplificada una y otra vez dando lugar a un sistema inestable. Esta realimentación se conoce como positiva.
- **Si  $A_l = -1$ :** en este caso el sistema se encuentra en la frontera entre la estabilidad y la inestabilidad. Por lo que idealmente, el sistema responderá a la función impulso o delta de Dirac con una oscilación continuada. Este caso es conocido como el criterio de Barkhausen y se trata de la condición necesaria para encontrar oscilaciones<sup>4</sup>.

Por último, se han de mencionar los diferentes tipos de realimentación que se dan en los amplificadores prácticos. Como se ha mencionado anteriormente, las señales de trabajo pueden ser de tensión, corriente o incluso una combinación de ambas. De esta forma, se pueden clasificar los diferentes tipos de realimentación en función de la señal de trabajo tanto a la entrada como a la salida. Algunos de estos tipos se exponen en las figuras 4 y 5<sup>5</sup>.

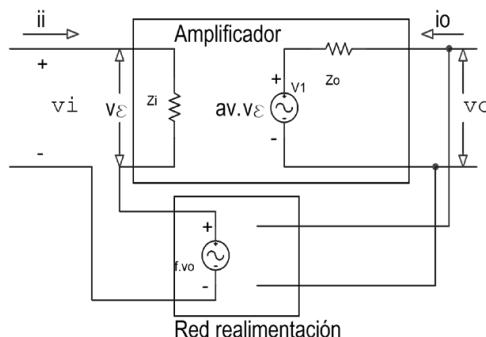


Figura 4: Amplificador de tensión con realimentación serie paralelo ideal.

<sup>4</sup>Gray, P. R., Hurst, P. J., Lewis, S. H., & Meyer, R. G. (2001). *Analysis and Design of Analog Integrated Circuits*. Wiley. Capítulo 8: Feedback.

<sup>5</sup>González Díaz, G. (n.d.). *Apuntes de electrónica analógica*. Capítulo 7: Realimentación negativa

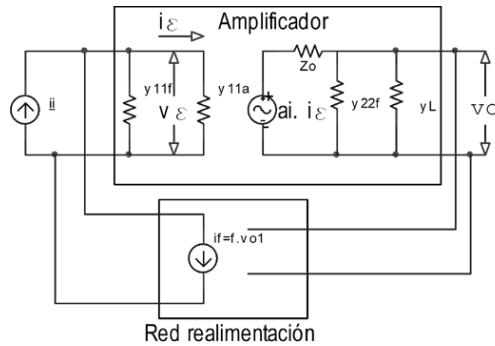


Figura 5: Amplificador de transresistencia con realimentación paralelo paralelo real.

## 2.4. Sistemas integrados: Atmega328p

Un sistema integrado o embebido es un sistema digital complejo, compuesto principalmente por CPU, memoria, buses y periféricos, entre otros. La conjunción específica del sistema se denomina arquitectura. El conjunto de los elementos del sistema también es conocido como SOC (*System On Chip*).

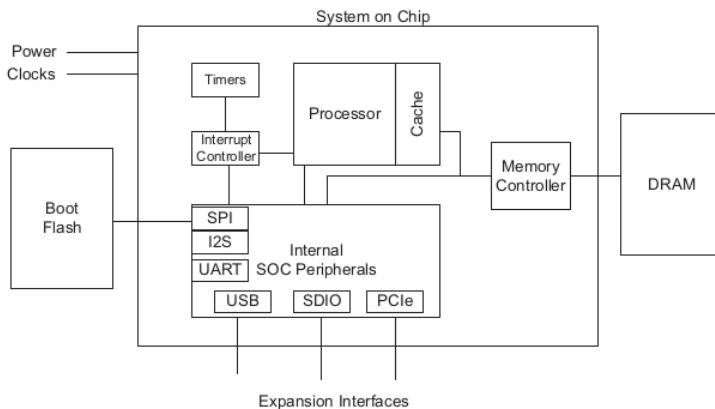


Figura 6: Esquema general de un SOC.

A continuación se realiza una descripción de los elementos principales que componen el SOC:

- **CPU:** Un sistema integrado posee al menos una CPU, la cual se encarga de la ejecución de los programas, operando con datos e instrucciones. Dependiendo del diseño de la CPU, se tiene una arquitectura de  $n$  bits, lo cual implica que el tamaño de los registros y de las direcciones es de  $n$  bits. El diseño de la CPU también especifica el uso de un repertorio de instrucciones concreto. Algunos ejemplos de repertorio de instrucciones son ARM, MIPS o CISC. La CPU se consolida como el principal elemento de la arquitectura.

A pesar de que la CPU constituya el elemento principal del sistema integrado, para que todo su procesamiento de datos resulte en trabajo útil, es necesario el soporte de hardware externo. Dentro de este conjunto de hardware se pueden distinguir:

- **Subsistema de memoria:** La memoria en general se encarga de almacenar y servir los datos e instrucciones utilizadas por el procesador. El sistema de memoria se puede descomponer

en varios módulos, como por ejemplo: la memoria caché, una memoria físicamente al lado del procesador con una velocidad de trabajo de unos pocos ciclos de procesador; la memoria DRAM, un dispositivo de memoria con un espacio de almacenamiento mayor que la caché, aunque normalmente un orden de magnitud más lenta, u otras memorias externas de diferentes tipos como pueden ser SRAM, FLASH o ROM.

- **Controlador de interrupciones:** Este mecanismo gestiona los requerimientos de atención del procesador por parte de los dispositivos, sin necesidad de que este tenga que estar pendiente de la falta de atención continuamente.
- **Timers:** El objetivo de estos dispositivos es generar una frecuencia de onda cuadrada estable. Estos dispositivos son imprescindibles para el funcionamiento de la CPU, ya que controlan la frecuencia de trabajo del procesador, o incluso otras tareas como las interrupciones periódicas, programación de eventos o la fecha y hora.

**Mapa de memoria:** El mapa de memoria es la lista de direcciones accesibles de todos los elementos del sistema: DRAM, controlador de interrupciones... El tamaño total del mapa de memoria dependerá del tipo de arquitectura del procesador y se calcula como  $2^n$ . Cuando el procesador ejecuta una instrucción de lectura o escritura, la dirección es decodificada por los decodificadores y finalmente enrutada hacia el correspondiente elemento del sistema<sup>6</sup>.

**AVR Atmega328p:** Como ejemplo de un sistema integrado, se introduce el procesador Atmega328p, el cual he utilizado en el proyecto. En la figura 7 se muestra un esquema general del sistema integrado Atmega328p. El Atmega328p es un sistema integrado tipo RISC con un procesador de 8 bit, el cual es capaz de ejecutar una instrucción por ciclo. Esto es posible gracias al diseño de una arquitectura tipo harvard, la cual se caracteriza por disponer de memorias separadas para datos y para las instrucciones del programa. Las instrucciones son ejecutadas con un nivel de segmentación, lo que permite que, mientras una instrucción está siendo ejecutada, la siguiente instrucción está siendo buscada en la memoria de programa. Es necesario aclarar que la CPU es capaz de trabajar con registros dobles, siendo capaz de direccionar un total de  $2^{16}$  posiciones de memoria.

Como se puede apreciar en la figura 7, el bloque *Flash program memory* (memoria flash) se corresponde con el espacio de almacenamiento donde se encuentran las instrucciones de nuestro software. Seguidamente se encuentra *Instruction register*, que permite el nivel de segmentación de las instrucciones. Continuando el esquema, La instrucción a ejecutar es decodificada siguiendo el mapa de memoria y correctamente enrutada al dispositivo correspondiente mediante *Instruction decoder*.

Una vez la instrucción es decodificada, pasa a ser ejecutada, entrando en escena la parte de datos. El bloque de los registros, *General purpose Registers* sirven de operandos y junto al bloque *ALU*, se encargan de realizar las operaciones que se requieran. Una vez generados los datos, se vuelcan en el bus de datos y serán recogidos por el dispositivo interesado, gobernados por *Control Lines*. La lectura de los operandos, la operación con la *ALU* y la escritura del resultado en el banco de registros, se realiza en un solo ciclo de reloj. El *Status Register* es un registro que se actualiza en cada operación aritmética con las particularidades de dicha operación: bit *zero*, *carry*, *overflow*, incluso la habilitación de las interrupciones globales.

---

<sup>6</sup>Harris, D. M., & Harris, S. L. (2012). *Digital Design and Computer Architecture* Capítulo 4.

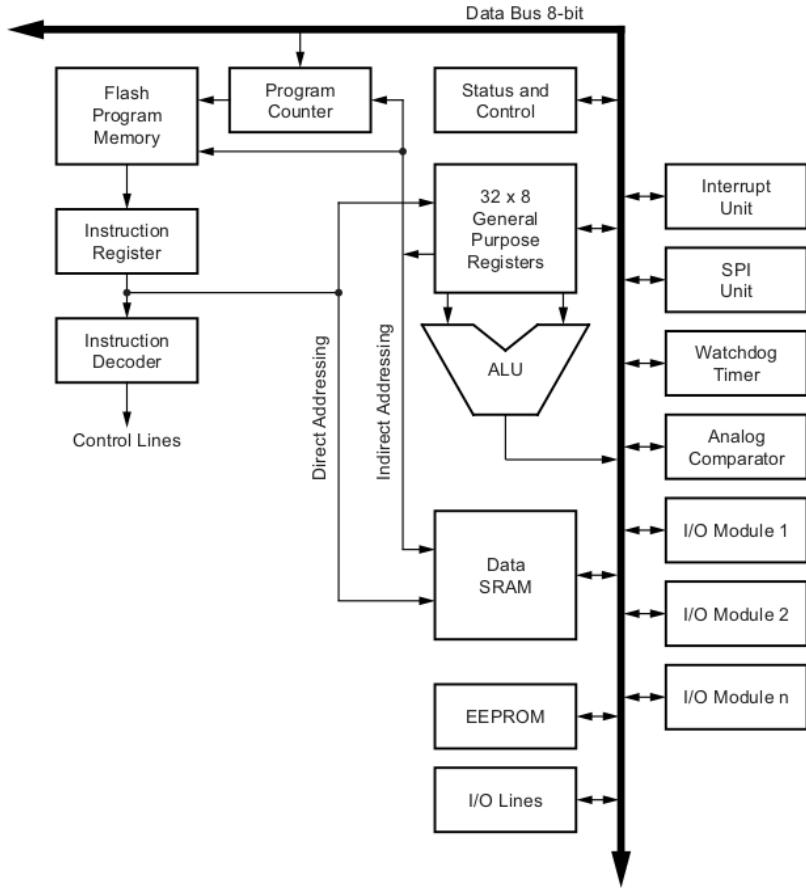


Figura 7: Diagrama de bloques del Atmega328p, en detalle la CPU

Como se ha comentado, existen dos mapas de memoria bien diferenciados: instrucciones y datos. La memoria de instrucciones es la flash, la cual tiene una capacidad de  $2^{15}B \approx 32kB$ , por lo que todas las direcciones de memoria están dedicadas a este elemento. Por otro lado, se tiene el mapa de memoria de datos, estructurado de la forma mostrada en la figura 8. Se puede observar en la

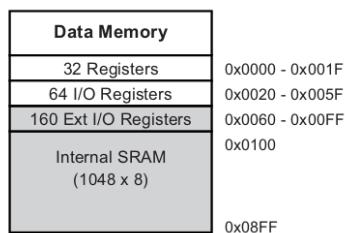


Figura 8: Mapa de memoria de datos direccionada por bytes

figura 8 que el mapa de memoria de datos está separado en distintas regiones: para el banco de registros, para los dispositivos de entrada salida y para la *SRAM*. La *SRAM* es la memoria física de datos y posee una capacidad de 2 kB. Al igual que la *FLASH*, esta memoria es accedida mediante registros dobles. El soporte para lidiar con datos de 16 bits se realiza por medio de unos registros especiales, nombrados como X, Y y Z<sup>7</sup>.

<sup>7</sup>Atmel. (n.d.). ATmega328P Automotive Microcontrollers Datasheet (Atmel-7810). Recuperado de <https://www.microchip.com/wwwproducts/en/ATmega328P>.

### 3. Desarrollo

En este apartado se desarrollarán los diferentes módulos que componen el sistema. Como se ha introducido en el apartado 1, el sistema de comunicaciones consta de un emisor y receptor que comparten la misma estructura de diseño: una parte analógica que actúa como la interfaz de comunicación de radiofrecuencia y una parte digital que trabaja con las señales banda base. Además, la parte digital también se encarga de la codificación y decodificación del enlace.

En cuanto a la parte analógica de radiofrecuencia nos encontramos las siguientes características generales que se desarrollan en los consiguientes apartados. La comunicación que se llevará a cabo consiste en un enlace digital mediante radio frecuencia de dos canales. La frecuencia de trabajo es de 30 MHz sobre una modulación es ASK. Esta modulación implica que el sistema analógico de radio trabaja recibiendo tonos sintonizados a la frecuencia de trabajo.

Por otro lado, la parte digital, trabaja tanto con la codificación en la parte emisora, como con demodulación de las señales digitales en la parte receptora. La codificación y demodulación se realiza con ayuda de un microcontrolador Atmega328p, el cual, mediante las señales producidas por dos distintos pulsadores, codifica la señal digital de manera NRZ apolar. Esta señal es posteriormente demodulada y decodificada en el receptor.

En la figura 9, se muestra un diagrama de bloques como esquema general de ambas partes del proyecto, transmisor y receptor. En ambos casos se diferencian la parte digital y analógica en cada caso.

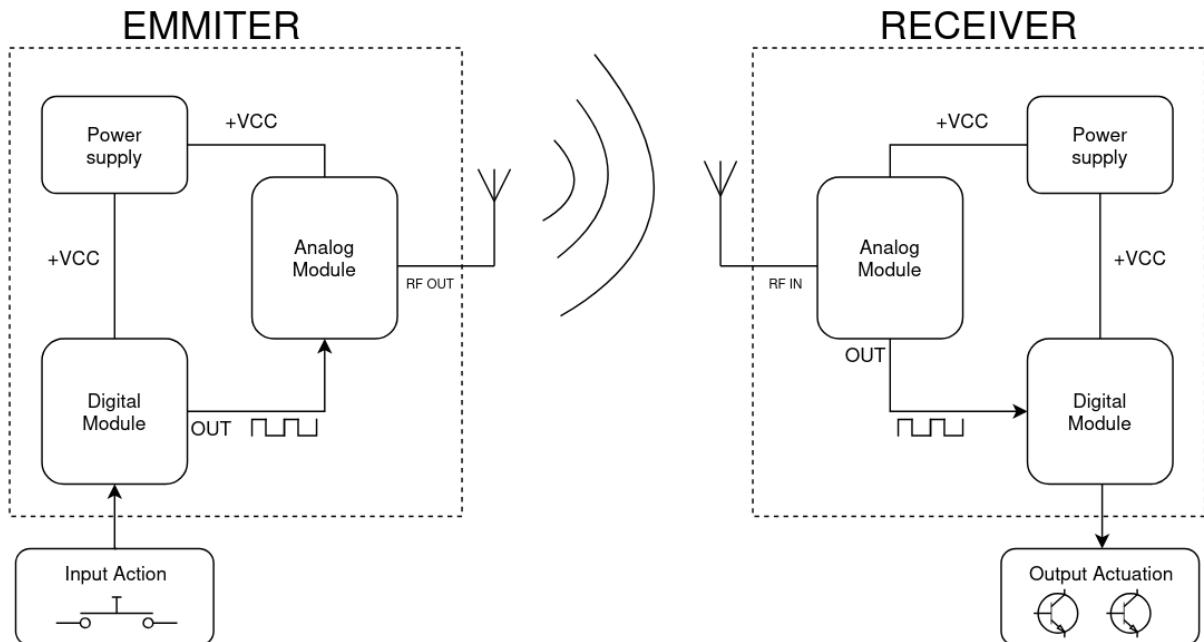


Figura 9: Diagrama de bloques general del proyecto

### 3.1. Parte analógica

En este apartado se desarrolla todo lo referente a la interfaz de radiofrecuencia (RF). En primer lugar, se explica de manera analítica el funcionamiento del transmisor y receptor, para posteriormente realizar un análisis cuantitativo, llevando a cabo los cálculos correspondientes a los valores de los componentes del sistema.

**Transmisor:** El transmisor está basado en un oscilador de un solo transistor en base común sintonizado por un circuito  $LC$ , conocido como circuito tanque. Los valores del condensador e inductancia del circuito tanque poseen una frecuencia de resonancia, la cual se amplificará por medio de la realimentación positiva. La oscilación es cortada eléctricamente a conveniencia por medio de otro transistor, produciendo una modulación AM-ASK. El transmisor se diseña de forma que radie la mayor potencia posible, y en consecuencia, propagar la señal a la mayor distancia posible.

**Receptor super-regenerativo:** El receptor super-regenerativo, diseñado en 1920, se basa en el concepto de realimentación positiva. Sin embargo, su antecesor, el receptor regenerativo, consiste en diseñar un bucle de realimentación cuyo Al sea  $A_l = 1$  (explicación en el apartado 2.3), dotando a este receptor de gran sensibilidad a la frecuencia de resonancia. El receptor regenerativo, en la práctica, es muy complicado de llevar a su condición de trabajo,  $A_l = 1$ , pues las mínimas variaciones harán que el circuito comience a oscilar o no ser tan sensible. Por este hecho se desarrolla el receptor super-regenerativo, que se basa en este mismo concepto de realimentación positiva, con la diferencia de que, en este caso,  $A_l > 1$  permitiendo en consecuencia, la oscilación. Pasado un determinado tiempo, el circuito corta la oscilación, permitiendo que el ciclo comience de nuevo. Esta señal de reinicio y paro se denomina *quench-signal*. En cada inicio del periodo de la señal *quench-signal*, momento en el cual la oscilación se está generando, el circuito atraviesa un periodo de sensibilidad máxima a las señales con frecuencia igual a la de resonancia. Si una señal es detectada, la oscilación del circuito se producirá de forma más rápida, aumentando así la frecuencia de la *quench-signal*, obteniendo como salida una señal con modulación FM con frecuencia de la *quench-signal*.

#### 3.1.1. Diseño de la parte analógica de RF del transmisor

En este apartado se expondrá el diseño del transmisor, los cálculos matemáticos necesarios, la simulación por ordenador y los resultados prácticos. El transmisor está diseñado para generar una modulación ASK y emitir a una frecuencia de unos 30 MHz. La frecuencia de emisión es sintonizable con la del receptor por medio de un condensador de capacidad variable.

En la figura 10 se puede observar el esquema eléctrico del transmisor. El principio de funcionamiento del transmisor es un oscilador basado en un par resonante  $LC$ , el cual fija la frecuencia de emisión. La oscilación se genera realizando un bucle de realimentación positiva, donde el transistor NPN juega el papel de elemento activo de amplificación. El circuito tanque  $LC$  es el filtro que permite que en cada iteración del bucle se amplifique la frecuencia deseada. A su vez, el condensador de realimentación genera la realimentación positiva, sumando una fracción de la salida con la señal de entrada, que en este caso es el propio ruido generado por el circuito. El circuito se diseña de forma que se disipe la mayor potencia posible, con el fin de que la señal recorra la mayor distancia alcanzable.

Por otra parte, el circuito permite modular la señal portadora eléctricamente cortando y produciendo la oscilación en función de las variaciones de la señal moduladora. Esto es posible gracias al

segundo transistor PNP, el cual trabaja en corte y saturación y corta el paso de corriente general del circuito. El esquema completo del transmisor se expone en la figura 10.

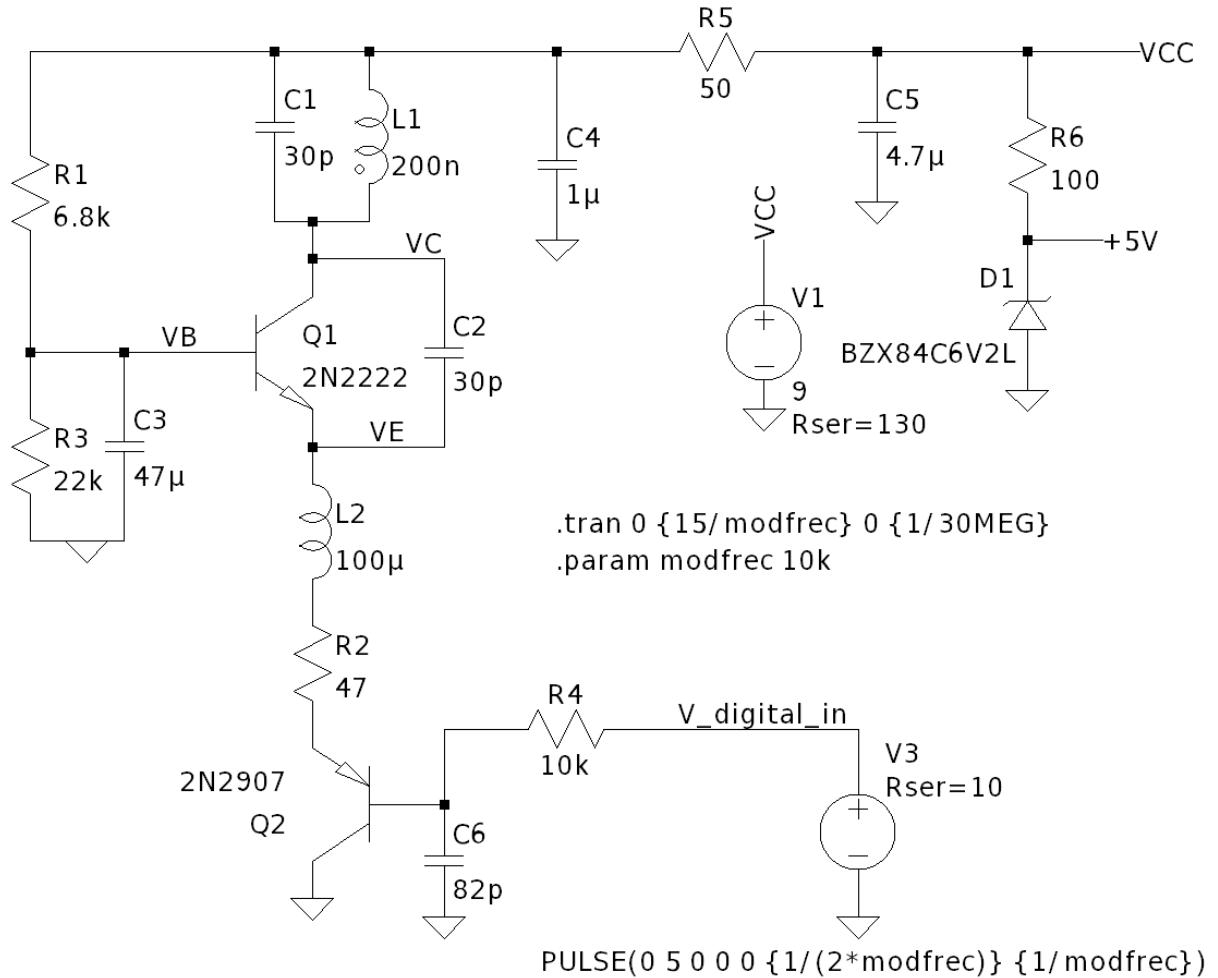


Figura 10: Esquema eléctrico del transmisor

## Diseño del oscilador

### Polarización

Primeramente, se debe fijar el punto de operación deseado. Se deben tener en cuenta dos cosas: la zona de trabajo del transistor y la potencia del circuito. La zona de trabajo debe ser activa directa, pues para producir la oscilación, el bucle de realimentación positiva debe tener la etapa de amplificación proporcionada por el transistor. La potencia del circuito, junto a la frecuencia de diseño, acotan el modelo de transistor que se ajusta a las necesidades del circuito.

En primer lugar, se necesita un transistor con una frecuencia de transición  $f_t > 30\text{ MHz}$ . A parte de esto, el parámetro  $I_{Cmax}$  debe ser suficiente para proporcionar la potencia deseada sin deteriorarse. Se elige un transistor 2N2222, cuya  $f_t > 30\text{ MHz}$  e  $I_{Cmax} = 0,6\text{ A}$ . Además, se fija una  $V_{CC} = 9\text{ V}$  el cual será el mayor valor que pueda tener, pero esto no implica que el transmisor

pueda funcionar con tensiones de alimentación inferiores. Se opta por limitar a  $I_c = 100 \text{ mA}$ , que supondrá una potencia de aproximadamente  $I_C \cdot V_{CC} = 0,72 \text{ W}$  y  $V_{CE} = \frac{V_{CC}}{2} = 4,5 \text{ V}$ , condición necesaria para trabajar en activa directa con el mayor rango de margen de distorsión.

Adicionalmente, se obtienen los parámetros necesarios de la hoja de características del 2N2222<sup>8</sup>. Se conoce  $h_{FEmax} \approx 300$  de la hoja de datos, aunque, medido con un multímetro, se obtiene el valor  $h_{FE} = 280$ , por lo que se utilizará este último. En lugar de repetir el cálculo que se hizo para seleccionar el valor de las resistencias de polarización, se opta por verificar si los valores elegidos satisfacen las imposiciones. Esto es debido a que, una vez se realizaron los cálculos, los valores de las resistencias de polarización fueron aproximados a valores comerciales para la construcción del dispositivo. De esta forma, se obtiene una comprobación doble. Además, en todo el proceso se considera despreciable la caída de tensión en el transistor PNP, la cual es de unos 0,2 V en condición de saturación.

Se utiliza equivalente de Thevenin para las resistencias en paralelo. En la malla que aparece se obtiene:

$$V_{th} - 0,7 - I_c \cdot R_e = I_b \cdot R_{th}$$

Siendo:

$$\begin{aligned} V_{th} &= \frac{V_{CC} \cdot R_2}{R_1 + R_2} & I_b \cdot h_{FE} &= I_c \\ R_{th} &= \frac{R_1 \cdot R_2}{R_1 + R_2} & h_{FE} &= 280 \end{aligned}$$

Se obtienen  $I_c$  y  $V_{CE}$  con las siguientes dos ecuaciones sustituyendo los valores correspondientes:

$$\begin{aligned} R_1 &= 6,8 \text{ k}\Omega & R_2 &= 22 \text{ k}\Omega & R_E &= 47 \Omega & V_{CC} &= 9 \text{ V} \\ I_c \cdot \left( R_E + \frac{R_{th}}{h_{FE}} \right) &= V_{th} - 0,7 \\ V_{CC} &= V_{CE} + I_c \cdot R_E \\ V_{CE} &= 4,57 \text{ V} & I_C &= 94,2 \text{ mA} \end{aligned} \tag{11}$$

Una vez calculado el punto de operación, se obtienen los parámetros híbridos en base común siguiendo la metodología expuesta en el apartado 2.2. En primer lugar, se deben calcular los parámetros híbridos en emisor común a partir de los resultados obtenidos en el punto de operación, utilizando las ecuaciones 5 y 6. En segundo lugar, se deben aplicar las transformaciones indicadas en la ecuación 7. Además se debe calcular el dato  $V_{AF}$  con ayuda de la hoja de datos del transistor, en este caso el dato se obtuvo como una media del rango de valores proporcionado. El resultado del cálculo de los parámetros es el siguiente:

$$V_{AF} = \frac{I_{Cdata}}{h_{OEdata}} = \frac{1 \text{ mA}}{6 \mu\text{s}} = 50 \text{ V} \tag{12}$$

$$\begin{aligned} h_{ib} &= 7,4 \Omega & h_{rb} &= 0,014 \\ h_{fb} &= -0,996 & h_{ob} &= 6,7 \mu\text{s} \end{aligned} \tag{13}$$

## Modelo en pequeña señal

---

<sup>8</sup>ON Semiconductor. (2016). 2N2222A: Small Signal Transistor Datasheet. Recuperado de <https://www.onsemi.com/pdf/datasheet/2n2222a-d.pdf>.

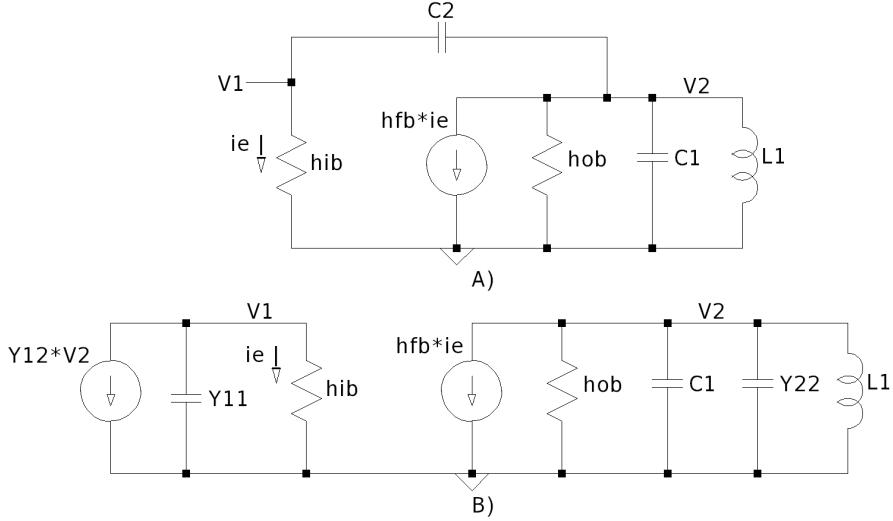


Figura 11: A) *Modelo en pequeña señal del bucle de oscilación para frecuencias medias* B) *Modelo en pequeña señal del oscilador sustituyendo el condensador de realimentación  $C_1$  por su equivalente en parámetros  $Y$*

El objetivo de este modelo es el cálculo de la frecuencia de resonancia del oscilador. En la figura 11 se muestra el modelo en pequeña señal del oscilador para frecuencias intermedias, en torno a la frecuencia de oscilación. El bucle de oscilación se trata de una realimentación paralelo-paralelo, por lo que se representa el condensador de realimentación  $C_1$  como su equivalente en parámetros  $Y$ . El valor de dichos parámetros son:

$$\begin{aligned} Y_{11} &= \frac{i_1}{v_1}|_{v_2=0} = s \cdot C_2 & Y_{21} &= \frac{i_2}{v_1}|_{v_2=0} = -s \cdot C_2 \\ Y_{12} &= \frac{i_1}{v_2}|_{v_1=0} = -s \cdot C_2 & Y_{22} &= \frac{i_2}{v_2}|_{v_1=0} = s \cdot C_2 \end{aligned}$$

Se deben tener en cuenta ciertas consideraciones previas como consecuencia del análisis del esquema B) en la figura 11. La realimentación es positiva en el momento que  $Y_{12} < 0$  y  $h_{fb} < 0$  por lo que  $V_2 > 0$ . El modelo que se muestra corresponde a frecuencias intermedias en torno a la de oscilación. Para frecuencias bajas, la impedancia de  $C_2$  tendrá un valor tan alto que corta la realimentación, siguiendo el esquema A) figura 11. Para frecuencias altas, la impedancia de  $C_2$  tendrá un valor tan bajo que supondrá un cortocircuito a tierra para la corriente de realimentación, por lo que  $i_e = 0A$ .

Se calcula la frecuencia de resonancia, en base al modelo B) de la figura 11. Siguiendo el criterio de Barkhausen, la frecuencia de resonancia se corresponde con la única con desfase  $\angle A_l(f_0) = 180^\circ$  a lo largo del bucle y una magnitud  $|A_l(f_0)| \geq 0$ . Se obtiene la función de transferencia de la ganancia en lazo abierto. Siguiendo el modelo general de la realimentación (ecuación 8 aplicada al esquema B) de la figura 11), se calcula:

$$f = Y_{12} = -s \cdot C_2 \quad (14)$$

Se muestra el desarrollo para el cálculo de  $A = \frac{V_2}{i_E}$ :

$$\begin{aligned} \frac{i_E}{V_1} &= Y_{T1} = s \cdot C_2 + \frac{1}{h_{ib}} & \frac{V_2}{h_{fb} \cdot i_e} &= Y_{T2}^{-1} \\ \frac{i_e}{V_1} &= \frac{1}{h_{ib}} & Y_{T2} &= s \cdot C_2 + s \cdot C_1 + \frac{1}{s \cdot L_1} + h_{ob} \\ i_E &= Y_{T1} \cdot i_e \cdot h_{ib} & V_2 &= \frac{h_{fb} \cdot i_e}{Y_{T2}} \end{aligned} \quad (15)$$

Se calcula la ganancia en lazo abierto como  $A_l = A \cdot f$  y sustituyendo los valores de  $Y_{T1}$  e  $Y_{T2}$ :

$$A_l = \frac{h_{fb} \cdot C_1 \cdot s^2}{(C_2 + C_1)(s \cdot h_{ib} \cdot C_2 + 1) \left( s^2 + s \cdot \frac{h_{ob}}{C_1 + C_2} + \frac{1}{(C_1 + C_2) \cdot L_1} \right)} \quad (16)$$

De la expresión obtenida en la ecuación 28, se deduce el siguiente análisis. En primer lugar, se analiza el desfase, el cual a bajas frecuencias es  $0^\circ$ , debido a la suma de los  $180^\circ$  del cero doble junto a los  $180^\circ$  de  $h_{fb} < 0$ . A su vez, el polo cuadrático introduce un desfase de  $-180^\circ$ , al que se llega de forma rápida debido al bajo valor del coeficiente de amortiguación, el cual se obtiene como:

$$\zeta = \frac{h_{ob} \cdot \sqrt{(C_1 + C_2) \cdot L_1}}{(C_1 + C_2) \cdot 2} = 2,9 \times 10^{-4}$$

Además, se añaden los  $-90^\circ$  del polo simple, el cual, se considera no dominante por su alto valor:  $s = \frac{-1}{h_{ib} \cdot C_2} = 4,5 \times 10^9 \text{ rad s}^{-1}$ . Consecuentemente, se obtiene que la frecuencia de resonancia con desfase  $180^\circ$  es aproximadamente igual que la frecuencia de resonancia del polo cuadrático, es decir:

$$\omega_0 = 1,925 \times 10^8 \text{ rad s}^{-1}$$

Por lo que:

$$f_0 = \frac{\omega_0}{2 \cdot \pi} = 30,63 \text{ MHz} \quad (17)$$

### Fabricación de la inductancia del circuito tanque

La inductancia del circuito tanque  $LC$ , es construida a mano para facilitar la radiación al medio. Esta inductancia será el elemento principal de radiación. En contraparte, las inductancias comerciales en su fabricación se centran en tener un reducido tamaño y evitar radiación al medio, produciendo bajas interferencias. Este hecho es contrario al objetivo que se busca.

La expresión de inductancia  $L$  de la bobina en función de sus parámetros físicos se desarrolla de la siguiente forma<sup>9</sup>:

Por un lado se tiene la definición de inductancia ( $L$ ), es decir, la variación del flujo con respecto a la variación de corriente por un bobinado:

$$L = N \cdot \frac{d\theta}{di} \quad (18)$$

---

<sup>9</sup>Kulkarni, S. V., & Khaparde, S. A. (2004). *Transformer Engineering: Design and Practice* (Capítulo 1: Transformer Fundamentals). Indian Institute of Technology, Bombay, Mumbai, India.

Por otro lado, se aplica la ley de Ampère para un único bobinado:

$$\mathcal{R} = N \cdot \frac{di}{d\theta} \quad \mathcal{R} = \frac{l}{\mu \cdot A_c} \quad (19)$$

Combinando las ecuaciones 19 y 18, se obtiene la expresión para el cálculo de la inductancia:

$$L = \frac{N^2 \cdot \mu_0 \cdot \left(\frac{d}{2}\right)^2 \cdot \pi}{l} \quad (20)$$

En este caso, los parámetros medidos en la inductancia fabricada son:  $N = 8$ ,  $\mu = \mu_0$ ,  $d = 8$  mm y  $l = 9$  mm. El resultado de la aplicación de la ecuación 20 con los parámetros medidos resulta en:

$$L_1 = 450 \text{ nH}$$

### Carril de +5 V

Para conseguir independencia de la fuente de alimentación general con respecto de la alimentación del microcontrolador de la parte digital, se añade un pequeño regulador de tensión formado por una resistencia, un diodo *zener* y un condensador. Este método también actúa a modo de protección contra posibles picos de tensión de la fuente principal, protegiendo el microcontrolador, además de proporcionar estabilidad, mejorando su rendimiento. En contraparte, el regulador, al tener un diseño simple, sufre de un excesivo consumo estático, siendo este, el mayor calculado independientemente del consumo de su carga. El cálculo del valor de la resistencia de protección se realiza según las necesidades de potencia. Se tiene:

$$\begin{aligned} P_{zener_{max}} &= 0,5 \text{ W} \\ V_{CC_{max}} &= 9 \text{ V} \\ I_{AVR_{max}} &\approx 50 \text{ mA} \\ P_{load_{max}} &= V_{zener} \cdot I_{AVR_{max}} < P_{zener_{max}} \\ P_{R_{Lim}} &= (V_{CC_{max}} - V_{zener}) \cdot I_{AVR_{max}} \\ R_{Lim} &= 78 \Omega \end{aligned} \quad (21)$$

El valor de  $R_{Lim}$  obtenido, se trata del mínimo para satisfacer los requisitos impuestos, pues  $I_{AVR_{max}}$  se encuentra sobredimensionado.

Empíricamente, en la figura 10, se usa  $R_{Lim} = R_6 = 100 \Omega$ , aunque este valor puede ser aún mayor. En cuanto a  $C_5$ , se usa un valor de  $C_5 = 4,7 \mu\text{F}$ . Este valor se considera suficientemente alto como para estabilizar la tensión y además suficientemente bajo como para que, al arranque del circuito, cuando  $C_5$  se encuentre descargado, no suponga un pico de corriente muy alto y prolongado para la fuente de alimentación. La resistencia  $R_5 = 50 \Omega$  actúa de separador entre la parte de RF y digital. Además, se comporta como un filtro paso bajo, evitando que señales de RF se filtren hacia el microcontrolador.

Por último, se hace referencia al transistor PNP. Este transistor es utilizado como conmutador, de tal forma que, una señal digital activa o corta el paso de corriente por el circuito principal. La señal digital de inicio o corte es producida por el microcontrolador Atmega328p, y se desarrollará en el correspondiente apartado (apartado 3.2.2).

## Antena

Para mejorar la eficiencia de radiación del transmisor, se debe garantizar la máxima transferencia de potencia de señal a la antena. Para poder adaptar la impedancia de salida del circuito a la impedancia de la antena, se diseña un transformador como adaptador de impedancias. Se considera esta opción como la alternativa más sencilla de implementar, debido a que el circuito necesita una impedancia de salida bastante alta para poder producir la oscilación. El método del transformador adapta las impedancias considerablemente bien, mediante un acople magnético, es decir, sin cargar el circuito.

El objetivo del diseño es calcular la relación del número de vueltas óptimo entre el primario y el secundario para mejorar la transferencia de potencia entre el transmisor y la antena. Para facilitar los cálculos, pues existen demasiados parámetros reales que se deben aproximar, se seguirá un modelo ideal sencillo del transformador. Esto es debido a que solo se busca mejorar el aspecto de la transmisión de potencia que se tiene de base. Se sigue el siguiente modelo de relación de impedancias en un transformador<sup>10</sup>.

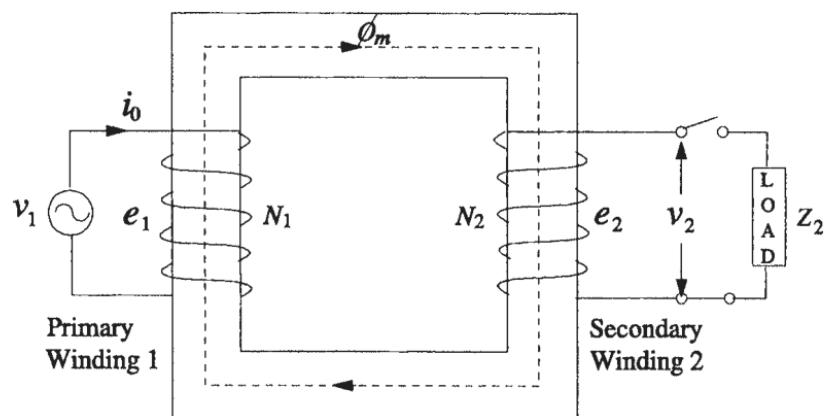


Figura 12: *Modelo de transformador ideal*

---

<sup>10</sup>Kulkarni, S. V., & Khaparde, S. A. (2004). *Transformer Engineering: Design and Practice* (Capítulo 1: Transformer Fundamentals). Indian Institute of Technology, Bombay, India.

Siguiendo el modelo de la figura 12, se obtienen las siguientes relaciones:

$$\begin{aligned}\frac{N_1}{N_2} &= \frac{E_1}{E_2} = \frac{V_1}{V_2} = \frac{I_2}{I_1} = n \\ Z_1 &= V_1 \cdot I_1 \\ Z_2 &= V_2 \cdot I_2\end{aligned}$$

$$\frac{Z_1}{Z_2} = n^2 \quad (22)$$

En particular para el diseño propio se debe calcular tanto la impedancia de salida del transmisor como la resistencia de radiación de la antena utilizada para la frecuencia de trabajo. Se tiene que  $Z_1 = h_{ob}^{-1}$  y que la impedancia de la antena es aproximadamente  $Z_2 = R_{rad} = 1 \text{ k}\Omega$ . Por lo tanto, se usa la ecuación 22 para obtener el ratio de vueltas óptimo siendo:

$$n = \sqrt{\frac{Z_1}{Z_2}} = 10$$

Esta relación de vueltas óptima no es realizable, pues el bobinado primario se construye con ocho espiras. La relación de vueltas aproximada será de  $n = 8 : 1$

### Resultado de la simulación

En este apartado se muestra una simulación del circuito en función del tiempo de los puntos de interés del circuito. En la figura 13 se observa el  $V_C$ , que es la tensión que se aplicará en el transformador de impedancias a la antena, y  $V_{digital}$ , la señal moduladora que produce la modulación ASK.

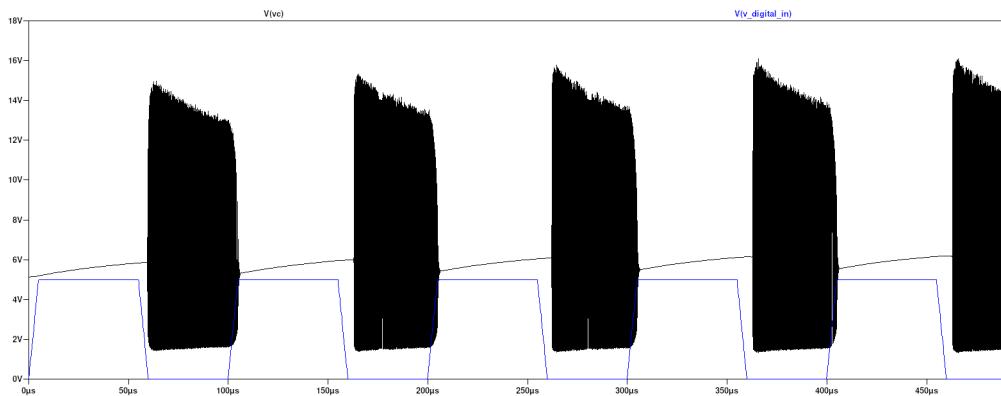


Figura 13: Simulación de  $V_C$  modulada por  $V_{digital}$

En la figura 14 se observa la FFT de la señal  $V_C$  de forma general, con un span de frecuencias alto. Por otro lado, en la figura ?? se muestra ampliada la frecuencia de trabajo, en la que se pueden observar los detalles de la modulación AM.

En la figura general de la FFT, al tratarse de una modulación ASK, se observa con acentuada potencia la señal moduladora en banda base. Además, esta señal posee una forma de onda cuadrada, por lo que su espectro se extiende ampliamente en el dominio de la frecuencia, aportando numerosos armónicos.

En la figura ??, se observa el espectro de la modulación ampliado a la frecuencia de trabajo. Se sitúan cursores a la frecuencia de trabajo y los armónicos fundamentales a 10 kHz. Además, se pueden observar multitud de armónicos secundarios a distancias múltiplos de la frecuencia fundamental 10 kHz.

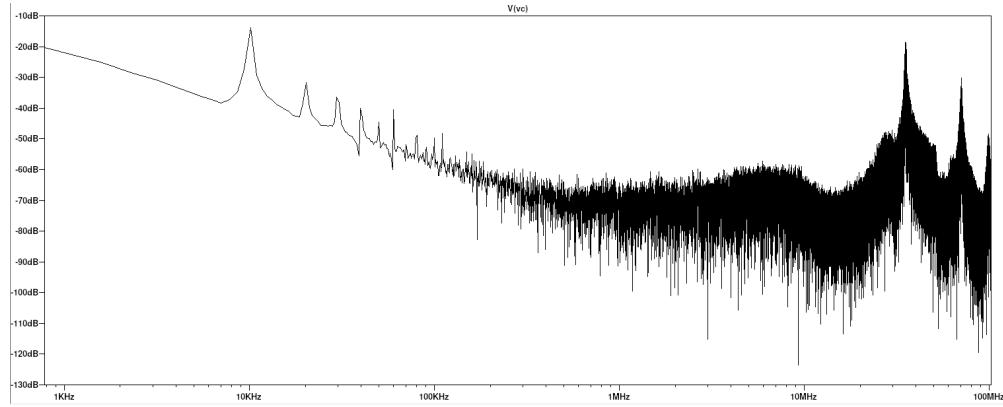


Figura 14: Simulación de la FFT de  $V_C$  de forma general

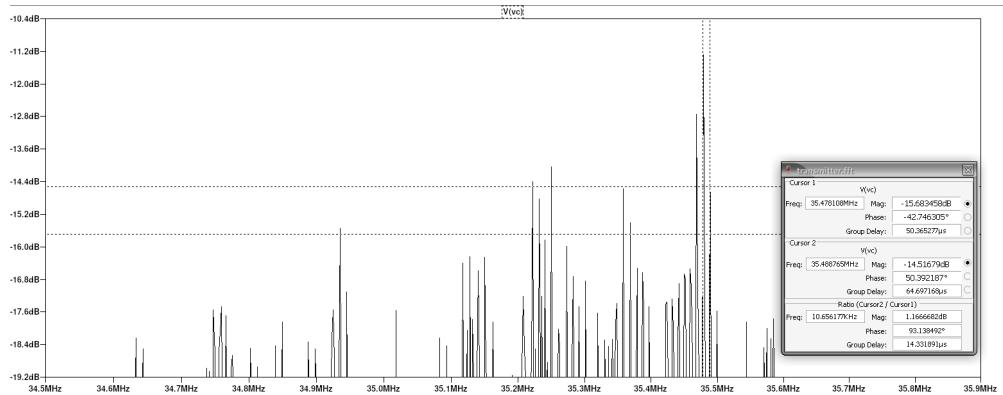


Figura 15: Simulación de la FFT de  $V_C$  ampliada a la frecuencia de trabajo

## Resultado de la práctica

En la parte práctica se comparan los resultados de la simulación con los resultados obtenidos en el circuito real.

El circuito está fabricado en placa soldada de agujeros, la realización de estas placas se muestra en la figura 17. Los resultados se miden con un osciloscopio en los mismos puntos de interés que en el apartado de simulación. Las figuras corresponden a capturas realizadas por el osciloscopio al tomar las medidas pertinentes.

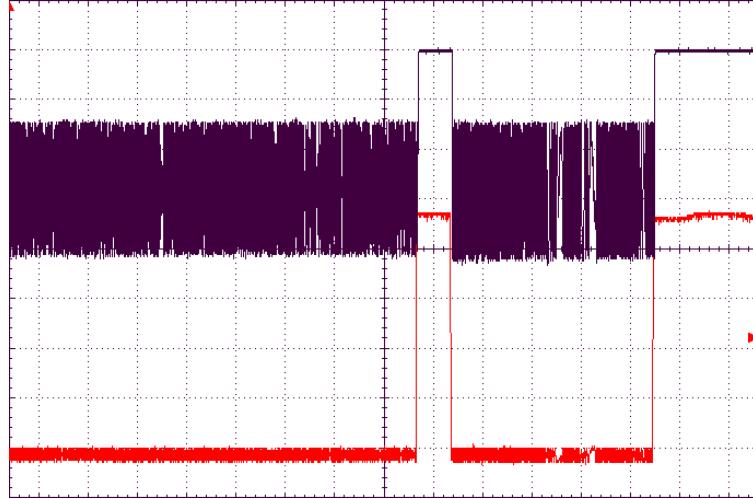


Figura 16: *Experimental: captura de osciloscopio de  $V_c$  modulada por  $V_{digital}$*

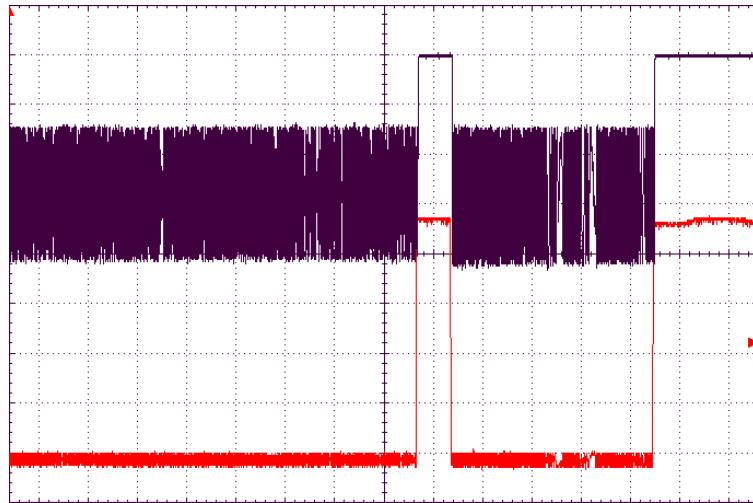


Figura 17: *Placa de transmisor soldada*

### 3.1.2. Diseño de la parte analógica de RF del receptor

El diseño del receptor se ha de separar en tres partes diferenciadas.

En primer lugar, la parte correspondiente al punto de operación de los transistores, tanto del oscilador de RF como del amplificador de la señal de *quench*. En este apartado se trabaja con la componente DC.

En segundo lugar, se desarrolla la parte de RF correspondiente al oscilador, el cual define la frecuencia de trabajo.

Por último, la parte correspondiente a la *quench-signal*, encargada de gestionar el paro y arranque de la oscilación. Esta parte trabaja a una frecuencia intermedia que puede diferenciarse de la parte de RF y de la componente DC.

El esquema completo del receptor se expone en la figura 18.

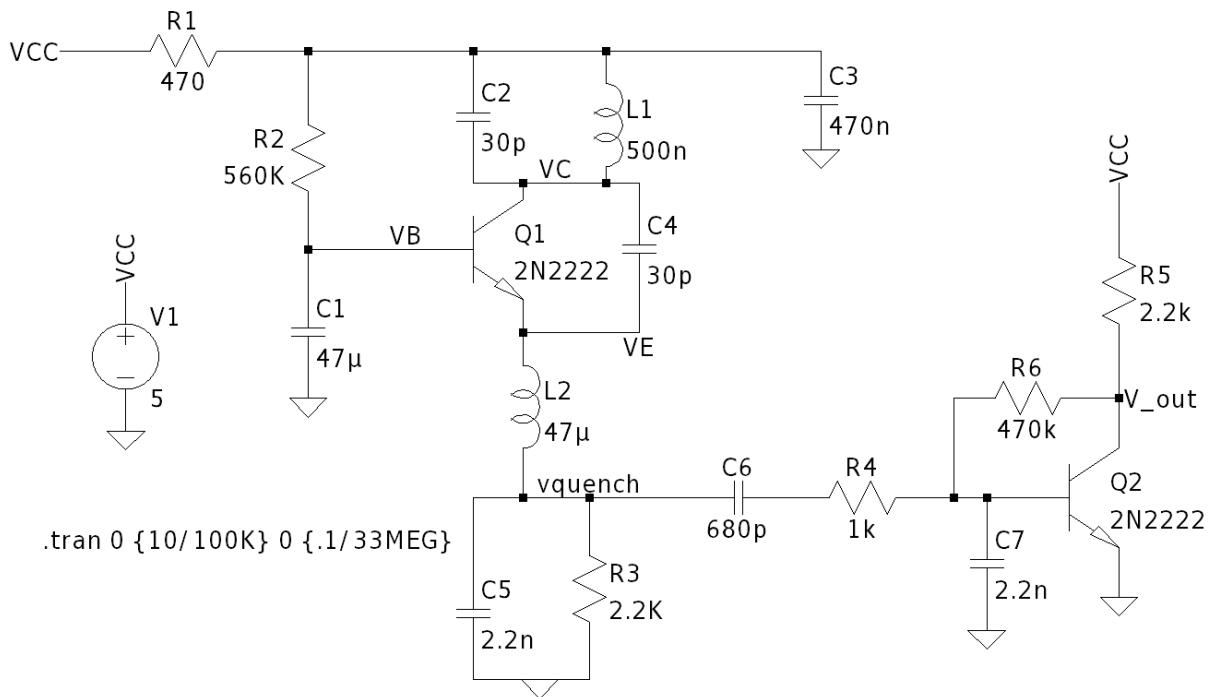


Figura 18: Esquema eléctrico del receptor

## Polarización

En este caso, la estrategia para fijar el punto de operación es ligeramente distinta a como se diseña en el transmisor, pues se utiliza una estructura conocida como realimentación de colector.

El transistor debe trabajar en zona activa directa, por lo que se debe cumplir  $V_{CE} > V_{sat} \approx 0,2\text{ V}$ . Se fija tanto  $V_{CC} = 5\text{ V}$ , como una corriente de colector aproximada de  $I_C = 600\text{ }\mu\text{A}$ , en este caso para que el transistor trabaje introduciendo el mínimo ruido posible. Este hecho es importante, pues cuando el receptor se encuentra en la etapa de inicio de oscilación, un bajo nivel de ruido ayuda a aumentar la sensibilidad del receptor. Esto es debido a que la suma mínima de todos los ruidos generados por un transistor se encuentra en este rango de corriente de colector<sup>11</sup>. Cabe recalcar, que la estructura de realimentación de colector provoca una realimentación negativa que fija el punto de operación de manera más independiente a los parámetros característicos del transistor. Esta realimentación negativa debe eliminarse en corriente alterna para provocar la oscilación. La estrategia para eliminarla se verá en el apartado de pequeña señal.

Se realizan los cálculos para estimar los valores de las resistencias de polarización en función de los valores anteriormente fijados. Las expresiones son válidas para ambos transistores, pues poseen

<sup>11</sup>Horowitz, P., & Hill, W. (2015). *The Art of Electronics* (Capítulo 8: Low Noise Techniques). Cambridge University Press.

la misma configuración con diferentes valores.

$$\begin{aligned}
 I_i &= I_C + I_B = I_E \\
 I_i &= I_E = I_B \cdot (h_{FE} + 1) \\
 I_B \cdot (h_{FE} + 1) &= I_E = (h_{FE} + 1) \cdot \frac{V_{CE} - 0,7}{R_B} \\
 V_{CC} - V_{CE} - I_E \cdot R_E - I_E \cdot R_C &= 0 \\
 V_{CE} \cdot \left(1 + \frac{(h_{FE} + 1) \cdot (R_E + R_C)}{R_B}\right) &= V_{CC} - \frac{0,7 \cdot (R_E + R_C) \cdot (h_{FE} + 1)}{R_B}
 \end{aligned}$$

De las anteriores expresiones se obtiene, para el transistor oscilador:

$$V_{CE} = 1,74 \text{ V} \quad I_C = 518 \mu\text{A} \quad (23)$$

Y para el transistor amplificador de la señal de *quench*:

$$V_{CE} = 1,76 \text{ V} \quad I_C = 633 \mu\text{A} \quad (24)$$

### Modelo en pequeña señal

La estructura del oscilador en el receptor es idéntica al transmisor. En primer lugar, para evitar la realimentación negativa provocada por la parte de polarización, se coloca el condensador  $C_3 = 470 \text{ nF}$ . Este valor es suficiente para que su impedancia para la frecuencia de RF suponga un cortocircuito a tierra. La inclusión de este condensador es imprescindible para que el circuito funcione.

Debido a que el diseño es estructuralmente igual que en el transmisor, los cálculos serán idénticos sustituyendo los valores correspondientes. Se incluyen los valores característicos junto a las ecuaciones de interés. En función de los valores del punto de operación obtenido, se calculan los parámetros híbridos para el receptor.

$$V_{AF} = \frac{I_{Cdata}}{h_{OEdata}} = \frac{1 \text{ mA}}{6 \mu\text{S}} = 50 \text{ V} \quad (25)$$

$$\begin{aligned}
 h_{ib} &= 1,35 \text{ k}\Omega & h_{rb} &= 0,014 \\
 h_{fb} &= -0,996 & h_{ob} &= 36,9 \text{ nS}
 \end{aligned} \quad (26)$$

El modelo en pequeña señal para las frecuencias de RF es sustancialmente igual a la parte del transmisor (apartado ??). En la figura 19, se muestra el modelo del receptor en pequeña señal para las frecuencias de RF.  $L_2$  tiene una impedancia suficientemente grande como para considerarla circuito abierto. El objetivo del modelo es la obtención de una expresión para la frecuencia de resonancia.

Debido a la dualidad con respecto al tx, la expresión de la función de transferencia de lazo cerrado es idéntica al transmisor. La expresión de la función de transferencia se muestra a continuación:

$$A_l = \frac{h_{fb} \cdot C_1 \cdot s^2}{(C_2 + C_1) (s \cdot h_{ib} \cdot C_2 + 1) \left( s^2 + s \cdot \frac{h_{ob}}{C_1 + C_2} + \frac{1}{(C_1 + C_2) \cdot L_1} \right)} \quad (27)$$

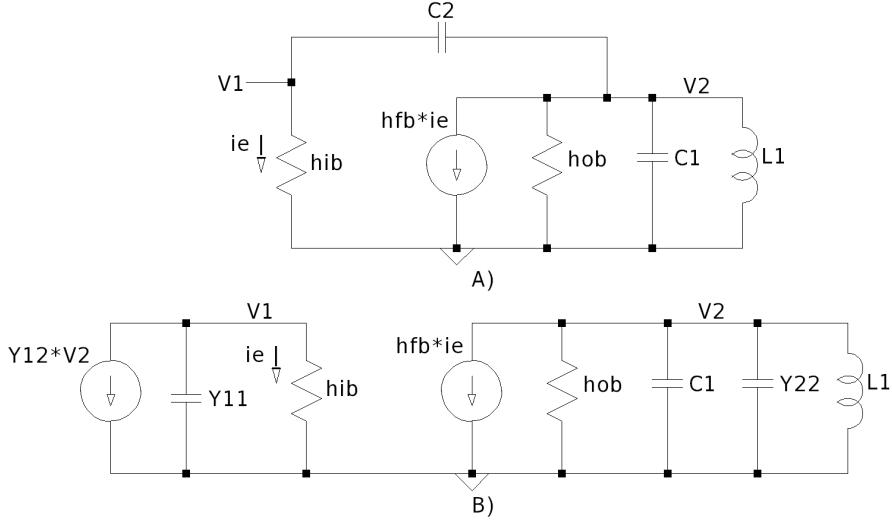


Figura 19: A) *Modelo en pequeña señal del bucle de oscilación para frecuencias de RF* B) *Modelo en pequeña señal del oscilador sustituyendo el condensador de realimentación  $C_2$  por su equivalente en parámetros  $Y$*

Se obtiene la frecuencia de resonancia como:

$$\omega_0^2 = \frac{1}{L_1 \cdot (C_1 + C_2)}$$

$$f_0 = \frac{1}{2 \cdot \pi \cdot \sqrt{L_1 \cdot (C_1 + C_2)}} = 30,63 \text{ MHz}$$

El valor de la inductancia  $L_1$  es el mismo que el calculado en el apartado 3.1.1, pues esta se construye con las mismas características. El valor resultante es calculado mediante la ecuación 20:

$$L_1 = 450 \text{ nH}$$

### **Quench-signal**

En este caso, al contrario que en el receptor, la bobina de RFC no podrá ser arbitrariamente grande, pues debe permitir el paso de la frecuencia de *quench* pero no de la señal de RF.

En primer lugar, es necesario una explicación analítica del fenómeno para facilitar el entendimiento antes de realizar los cálculos de interés. El transistor trabaja en configuración de base común. Esto implica que la tensión de base  $V_B$  es fija, mientras que  $V_E$  varía en el tiempo cortando y activando el transistor. Para un mejor entendimiento, se habla de  $V_E$  como  $V_{quench}$  indistintamente, pues  $V_{quench}$  es  $V_E$  tras el filtro paso bajo, eliminando visualmente la componente RF. La componente de baja frecuencia es quien corta el transistor de forma general. En la figura 20 se observa de forma gráfica la explicación dada a continuación.

Partiendo de una tensión  $V_B - V_{quench} \approx 0,7 \text{ V}$ , la oscilación comienza a generarse. A medida que la oscilación, al encontrarse dentro de un bucle de realimentación positiva, va incrementando su amplitud, la tensión media  $V_{quench}$  también aumenta. En el momento que  $V_{quench}$  aumenta de forma que  $V_B - V_{quench} < 0,7 \text{ V}$ , el transistor se corta, matando la oscilación y provocando que la tensión  $V_{quench}$  descienda, volviendo de esta forma a completar el ciclo.

Para calcular la frecuencia de *quench*, se debe tener en cuenta el filtro paso bajo formado por  $L_2$ ,  $C_5$  y  $R_3$ . Se calcula la función de transferencia del conjunto de estos tres elementos, al cual se le llamará circuito de *quench*. Se pretende obtener la expresión para la función de transferencia de  $H(s) = \frac{i_{L_2}}{v_i}$ .

$$\begin{aligned}
v_i &= i_{L_2} \cdot (Z_1 + Z_2) \\
Z_1 &= s \cdot L_2 + r_s \\
Z_2 &= \frac{R_3}{s \cdot R_3 \cdot C_5} \\
\frac{i_{L_2}}{v_i} &= \frac{1}{(Z_1 + Z_2)} = \frac{(s \cdot R_3 \cdot C_5 + 1)}{rs \cdot (s \cdot R_3 \cdot C_5 + 1) + s \cdot L_2 \cdot (s \cdot R_3 \cdot C_5 + 1) + R_3} \\
H(s) &= \frac{i_{L_2}}{v_i} = \frac{(s \cdot R_3 \cdot C_5 + 1)}{R_3 \cdot (s \cdot r_s \cdot C_5 + s^2 \cdot L_2 \cdot C_5 + 1)} \tag{28}
\end{aligned}$$

Se calcula la frecuencia de resonancia como

$$\begin{aligned}
\omega_0^2 &= \frac{1}{L_2 \cdot C_5} \\
f_0 &= \frac{1}{2 \cdot \pi \cdot \sqrt{L_2 \cdot C_5}} = 495 \text{ kHz}
\end{aligned}$$

En la figura 20, se muestra por un lado, a la izquierda, la respuesta en frecuencia del circuito de *quench*, compuesto por:  $L_2$ ,  $R_3$  y  $C_5$  en el esquemático del transmisor (figura 18). La respuesta en frecuencia es simulada para varios valores de  $L_2$ , y se indica con cursorres. En el caso el caso de  $L_2 = 47 \mu\text{H}$ , el valor de su frecuencia de corte es  $f_c = 483 \text{ kHz}$ .

Por otro lado, a la derecha, se muestra la simulación en función del tiempo del receptor. Se hace hincapié en la corriente de  $L_2$ . Esta está cursorizada de forma aproximada al valor que debería tener si a través de la bobina pudiera conducir corriente negativa, siendo este valor igual que el de la frecuencia de corte en la simulación en frecuencia. La corriente de la inductancia  $L_2$  no puede ser negativa, pues en este sentido de circulación la unión PN base emisor se encuentra inversamente polarizada y el condensador de realimentación tiene alta impedancia para frecuencias bajas.

En cualquier caso, la frecuencia real de *quench* no es la calculada en la ecuación 28, sino que la señal de *quench* se genera como se muestra en la figura 20. En el momento que  $i_{L_2} = 0$ , la tensión en  $V_{quench}$  se descarga a través de  $R_3$ , siguiendo  $\tau$  de  $C_3$  y  $R_3$  hasta que  $V_B - V_{quench} < 0,7$ , reiniciando el ciclo. El tiempo de subida depende tanto del tiempo que tarda en reconstruirse la oscilación como de la constante de elongación del sistema de la ecuación REF. Esta constante viene dada por  $r_s$  y  $C_5$  EXPRESION y define  $v_{oi}$ . Se calcula aproximadamente la frecuencia de *quench*

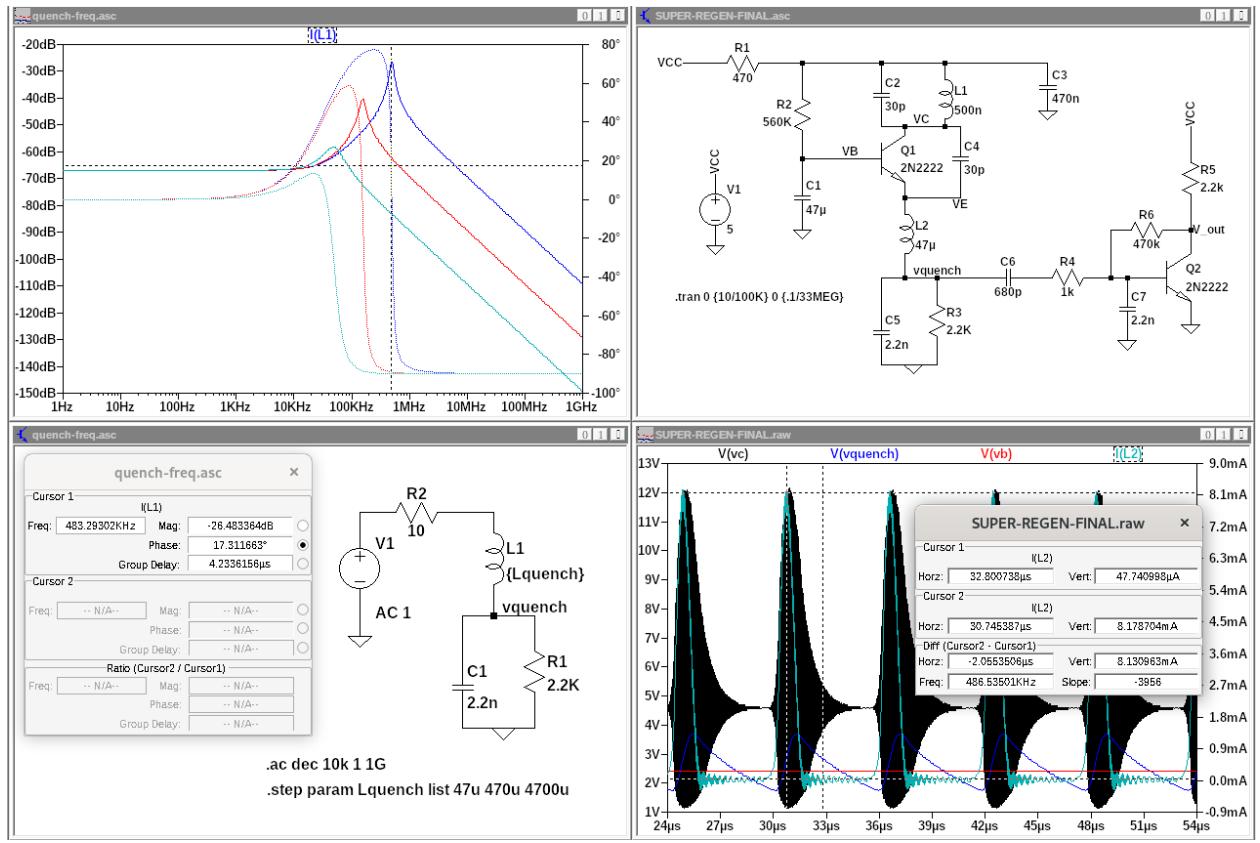


Figura 20: Explicación del comportamiento de la señal de quench

siguiendo los valores de la figura 20.

$$\begin{aligned}
 i_{C_5} + i_{R_3} &= 0 \\
 C_5 \cdot \frac{dv_0(t)}{dt} + \frac{v_0(t)}{R_3} &= 0 \\
 \int_{v_{0i}}^{v_0(t)} \frac{1}{v_0(t)} dv_0(t) &= \int_0^t \frac{-1}{R_3 \cdot C_5} dt
 \end{aligned}$$

$$\ln\left(\frac{v_0(t)}{v_{oi}}\right) = \frac{-t}{R_3 \cdot C_5} \quad (29)$$

$$\frac{v_0(t)}{v_{oi}} = e^{\left(\frac{-t}{R_3 \cdot C_5}\right)} \quad (30)$$

Se usa 29 para calcular el tiempo de descarga con los valores de la figura 20.

$$t_{disch} = 3,62 \mu s$$

A este tiempo, se le debe sumar el tiempo de subida que viene dado por el tiempo de construcción de la oscilación junto con el tiempo de subida del sistema del circuito *quench*. Este tiempo, se aproximarán como  $t_{rise} \approx t_{disch}$ . Por lo que

$$t_{quench} \approx 2 \cdot t_{disch}$$

y por tanto

$$f_{quench} = \frac{1}{t_{quench}} = 138 \text{ kHz}$$

Empíricamente, esta frecuencia resulta en  $f_{quench} = 67 \text{ kHz}$ . Esto es debido a que los valores en la simulación difieren bastante de la realidad, pero no las formas de onda. El ajuste de la frecuencia de  $quench$  empírica se realiza probando distintos valores de  $C_5$  y  $L_2$ , pues  $R_3$  se fija al polarizar el transistor.

## Resultado de la simulación

En el apartado de simulación, se trata de obtener una representación gráfica de lo desarrollado anteriormente sobre el receptor. Por ello, en la figura REF se observan los puntos de interés del circuito como son  $V_C$ ,  $V_{quench}$  y  $V_B$ , además de añadir la diferencia  $V_B - V_{quench}$  como  $V(B, quench)$ . Estas cuatro medidas son suficientes para entender el ciclo de paro y marcha del transistor.

Como se puede observar en la figura 33, a medida que se construye la oscilación, el valor medio de la tensión de  $V_C$  (es decir  $V_{quench}$ ) aumenta hasta que, finalmente, la diferencia  $V_B - V_{quench} < 0,7 \text{ V}$  hace desaparecer la oscilación. Este corte provoca que el valor medio de  $V_C$  descienda, y por tanto  $V_{quench}$ , provocando finalmente que la diferencia  $V_B - V_{quench} > 0,7 \text{ V}$ , reactivando al transistor y reiniciando el ciclo de oscilación.

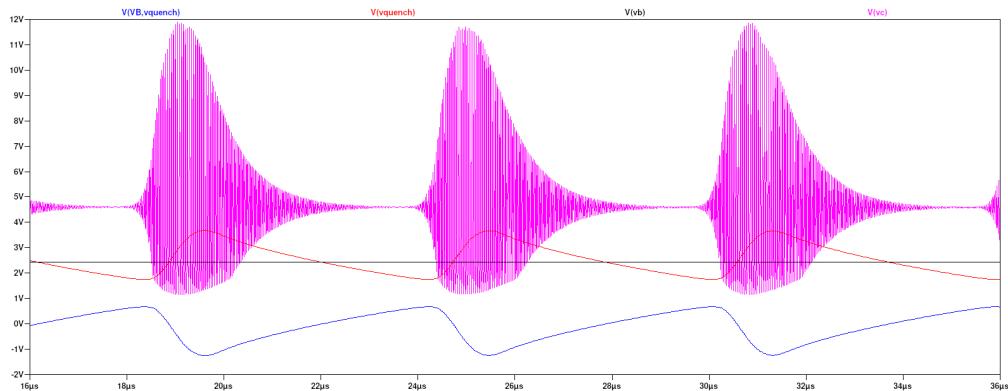


Figura 21: Simulación de los puntos de interés en varios ciclos ampliados

Se añade también, en la figura 22, la forma de onda de la tensión de salida  $V_{out}$ , que es la señal de entrada al microcontrolador atmega328p, el cual se encargará de demodular la señal. Esta señal debe ser una señal digital entre 0 V y 5 V.

## Resultado práctico

En este caso se muestran las medidas tomadas en el apartado de simulación, esta vez tomadas en el circuito real. Las medidas se toman usando un osciloscopio digital. La configuración es la misma que en el apartado 3.1.1 del transmisor. En este caso, al disponer de dos canales, se muestran en la figura 24 los puntos  $V_{quench}$  (rojo) y  $V_C$  (negro) y en la figura 25 los puntos  $V_{out}$  (negro) y  $V_{quench}$  (rojo).

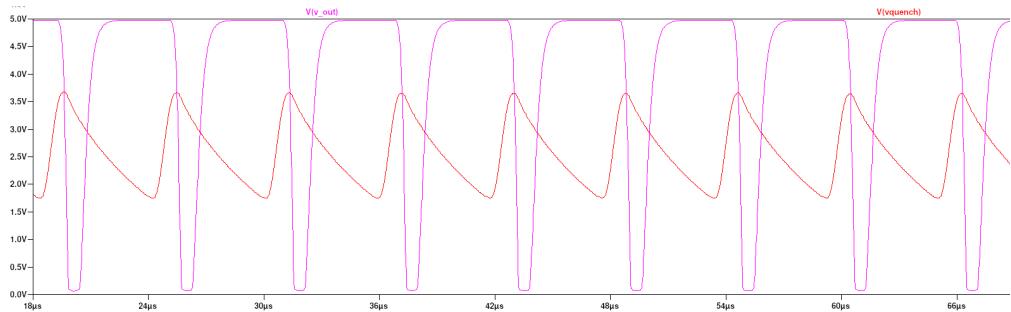


Figura 22: Simulación de  $V_{out}$  junto a  $V_{quench}$

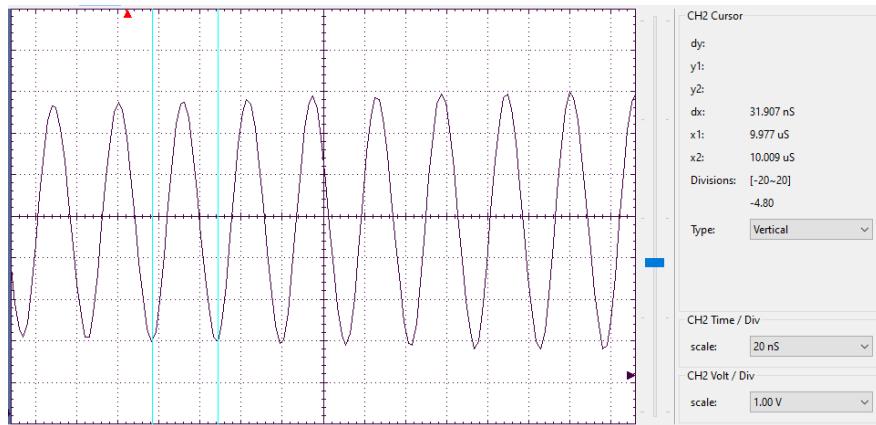


Figura 23: Experimental: captura de osciloscopio de la frecuencia de trabajo

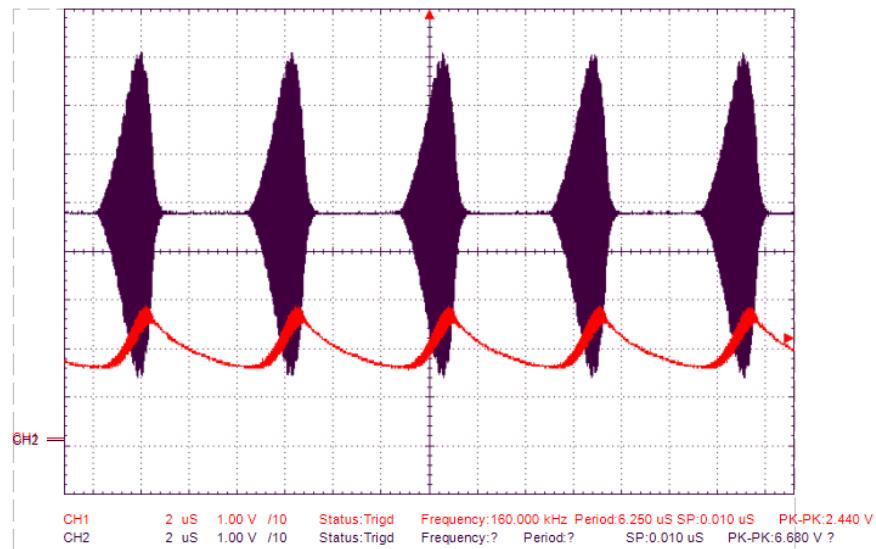


Figura 24: Experimental: captura de osciloscopio de  $V_C$  (negro) y  $V_{quench}$  (rojo)

### 3.2. Parte digital

Para establecer un canal de comunicación de datos digital, se utilizan dos microcontroladores para la codificación y la demodulación de los mismos. En este caso, el modelo de microcontrolador utilizado es el mismo en ambos dispositivos, el Atmega328p, pero con distintos programas



Figura 25: Experimental: captura de osciloscopio de  $V_{out}$  (negro) y  $V_{quench}$  (rojo)

dependiendo de si se utiliza en el transmisor o receptor.

### Configuración del entorno de trabajo

El microcontrolador se programa por medio de un proyecto escrito en C. Para ello, se trabaja con las herramientas que permiten la compilación de este lenguaje a un archivo ejecutable entendible para la plataforma de AVR.

En primer lugar, es necesario compilar el programa en código fuente a un archivo binario para la plataforma objetivo, para ello se usará el compilador *avr-gcc*. Este archivo binario generado no puede ser grabado directamente a la flash del microcontrolador, sino que se necesita la traducción a código hexadecimal del mismo. Para ello, se utiliza el programa *avr-objcopy*. Finalmente, el programa es grabado en la flash. Este proceso se realiza de la siguiente forma: el archivo hexadecimal debe ser grabado en el microcontrolador configurando el microcontrolador en modo programación de la flash y transfiriendo el programa por medio del protocolo ISP. Para ello se hará uso de un programador software, *avrdude*, y un programador hardware que traduzca el protocolo USB del ordenador de trabajo a ISP para ser grabado en la memoria del microcontrolador objetivo. En este proyecto se utiliza como programador un microcontrolador Atmega2560 en una placa Mega2560 R3. La placa Mega2560 R3 se programa con un software que permite el proceso de traducción anteriormente descrito. Este software se ofrece oficialmente desde la página web de Arduino<sup>12</sup>. Para automatizar todo el proceso de compilación y programación del microcontrolador objetivo, se hace uso de la herramienta *make*. A continuación, se muestra un archivo *Makefile* utilizado para clarificar el proceso anteriormente descrito:

```
all: main.hex

main.hex: main.elf
        avr-objcopy -O ihex main.elf main.hex

main.elf: main.c
        avr-gcc -mmcu=atmega328p main.c -o main.elf

upload: main.hex
```

---

<sup>12</sup><https://docs.arduino.cc/built-in-examples/arduino-isp/ArduinoISP/>

```
avrdude -p m328p -c arduino -P /dev/ttyACM0 -b 19200 -U flash:w:  
main.hex
```

Es importante destacar que los microcontroladores AVR poseen unos registros especiales denominados *fuses*. Estos registros poseen configuraciones críticas para el microcontrolador como la fuente que se utiliza como reloj de CPU. Esta puede ser configurada como externa o interna, a diferentes frecuencias. Por defecto de fábrica, viene configurado usando el oscilador interno de 8 MHz, con un prescaler de 8, resultando en  $F_{CPU} = 1 \text{ MHz}$ .

En la práctica se tuvo que modificar uno de los *fuses*, el *lfuse* en concreto. El propósito final era modificar la frecuencia de reloj de la CPU modificando el bit de prescaler. Para lidiar con los aspectos referentes a los *fuses*, se crearon dos guiones *bash*: *fuses.sh* y *write\_fuses.sh*, para leer y escribir respectivamente. A continuación se muestra *fuses.sh*:

```
#!/bin/bash  
set -x  
  
avrdude -p m328p -c arduino -P /dev/ttyACM0 -b 19200 -U lfuse:r:-:b -U  
hfuse:r:-:b -U efuse:r:-:b  
  
# to write fuses  
# avrdude -p m328p -c arduino -P /dev/ttyACM0 -b 19200 -U lfuse:w:0xFF:m -  
# U hfuse:w:0xDA:m -U efuse:w:0xFD:m  
# replace hex values with hex values of choice
```

Por último, cabe mencionar, que los archivos de código fuente junto a todo el proyecto han sido desarrollados mediante el programa de control de versiones *git*. Los repositorios tanto del proyecto como del código fuente se encuentran en GitHub<sup>13</sup>.

### 3.2.1. Diseño del demodulador digital para la recepción

El objetivo del microcontrolador en la parte de recepción tiene dos funciones. Implementar un contador de frecuencia que identifique las variaciones recibidas por el módulo de RF correspondientes a los diferentes símbolos digitales, de tal forma que sirva como demodulador. y además decodificar la señal digital recibida, identificando la orden concreta transmitida por el transmisor.

#### Contador de frecuencia

Esta parte se implementa por medio de dos timers/counters incorporados en el SOC del Atmega328p. La configuración y uso de estos dispositivos se encuentra en la hoja de datos del microcontrolador (REF). La estrategia de implementación es la siguiente: mientras uno de los timers genera interrupciones periódicas en un intervalo de tiempo conocido. Durante el mismo espacio de tiempo, el segundo timer/counter, se encarga de detectar el número de flancos de subida o bajada producidos por la señal de salida codificada en FM del módulo de RF. Este proceso provoca un número de interrupciones variable en función de la frecuencia de la señal de entrada en un intervalo de tiempo conocido.

---

<sup>13</sup><https://github.com/josegu05/tfg2>

Cada vez que el timer produzca su interrupción periódica, la rutina de tratamiento de interrupción (IRQ) se encargará de examinar el número de interrupciones producidas por el counter en ese lapso de tiempo y decidir si se ha recibido señal, en función del número de interrupciones del counter.

Se hacen uso tanto del TIMER0 como del TIMER2, esto es debido a que poseen las mismas características necesarias las cuales se encuentran expuestas en la hoja de datos. Existen a su vez, más timers/counters con características más complejas, pero no serán necesarias en este proyecto.

Se configura TIMER0 como temporizador, generando la interrupción periódica necesaria conocida como *gate*. mediante el registro de configuración propio del timer, se configura la frecuencia a la cual se genera esta interrupción periódica. La rutina de tratamiento de interrupción ISR(TIMER0), incrementa una variable que permite conocer el número de veces que se genera la interrupción.

Por otro lado, TIMER2, se configura como contador, identificando los flancos de bajada de una señal externa introducida por el pin OSC2. La interrupción del TIMER2, se puede producir cada cierto número de flancos detectados. la rutina de tratamiento de interrupción ISR(TIMER2), incrementa la variable global de cuenta.

Finalmente, el objetivo del programa main(), es identificar la recepción de señal y actuar en consecuencia. Para ello, el programa actúa de la siguiente forma. En primer lugar, espera a que la interrupción del temporizador *gate* se haya activado el número de veces necesaria. Ambos contadores durante el periodo de espera se encuentran continuamente actualizando. Después se realiza una media aritmética dividiendo el número de flancos detectados por el contador entre el número de veces que *gate* se activó. En caso de que la media calculada supere la media calculada anteriormente, quiere decir que se ha detectado señal, por lo que el programa procede a actuar en consecuencia. El programa implementa una máquina de estados en función del número de señales recibida.

## Limitaciones

Algunos problemas que surgieron a la hora de llevar a cabo esta realización fueron. La velocidad de procesamiento de instrucciones debe ser decenas de veces más rápida que la frecuencia de entrada, *quench-signal*. Siendo la rutina IRQ(TIMER2), del contador la función crítica. En las primeras versiones del código recargaban mucho las rutinas de tratamiento de interrupción, escalando muy rápido este problema.

También, ha sido necesario introducir una función de startup(), ya que, en el momento que se conecta el circuito a la fuente de alimentación, la frecuencia de *quench* tarda un tiempo en estabilizarse. La función startup(), asegura que la señal de *quench* es estable.

## Presentación del código

Todas las versiones de los códigos, tanto del emisor como del receptor, se encuentran en el repositorio de GitHub<sup>14</sup>. La versión final del código del transmisor se muestra a continuación. El código se presenta en dos columnas por hoja, el orden de lectura es el siguiente: primera columna, segunda columna, hoja siguiente.

---

<sup>14</sup><https://github.com/josegu05/tfg2>

```

#include <avr/io.h>
// definiciones de registros
    especificos del controlador de io
#include <avr/interrupt.h>
// gestion de IRQs
#include <stdint.h>
// definiciones de tipos estandar

#ifndef define WAIT_TIME 1000
#define GATE_COUNT_LIMIT 8
#define TIMES_STABLE_BOOT 500
#define THRESHOLD 1.0
uint8_t volatile decoding = 0;
uint32_t volatile count = 0;
float mean = 0;
float last_mean = 0;
uint16_t volatile gate_times = 0;

void setup_timer_0_counter(void)
{
    // set configuration registers
    // set CTC mode and COM on toggle
    // with compare match
    //TCCR0A = (1 << COM0A0) | (1 <<
    //WGM01);
    //
    TCCR0A = (0 << COM0A0) | (1 <<
    WGM01);
    // 4 count limit frecuency for
    // F_in = 60 KHz and F_CPU = 1
    // MHz
    OCROA = 3;

    // select clock source as external
    // source falling edge no prescaler
    // PD4 is the Alternate function
    // to External T0 clk
    TCCR0B = (1 << FOC2A) | (1 <<
    FOC2B) | (1 << CS12) | (1 <<
    CS11);

    // set PD6 as output wich is
    // TIMERO COMP_A OUTPUT
    //DDRD |= (1 << DDD6);
    // set TIMERO interrupt COMP_A mask
    // enable
    TIMSK0 = (1 << OCIE0A);
}

ISR(TIMERO_COMPA_vect)
{
    //counter
    count++;
}

void setup_timer_2_gate(void)
{
    // set CTC mode and COM on toggle
    // with compare match
    TCCR2A = (1 << COM2A0) | (1 <<
    WGM21);
    //TCCR2A = (0 << COM2A0) | (1 <<
    WGM21);
    OCR2A = 220;

    // select clock source as internal
    // prescaler 32
    // TCCR2B = (1 << CS21) | (1 <<
    CS20);
    // select clock source as internal
    // prescaler 32*8 at 8MHz
    TCCR2B = (1 << CS22) | (1 << CS21
    ) | (0 << CS20);

    // PB3 as output compare match A
    // timer 2
    DDRB |= (1 << DDB3);

    // set TIMER2 interrupt COMP_A mask
    // enable
    TIMSK2 = (1 << OCIE2A);
}

ISR(TIMER2_COMPA_vect)
{
    // gate
    gate_times++;
}

void setup_gpios(void)
{
    // as output gpios pb0=general
    DDRB |= (1 << DDB0);
    // as output decoded outputs and
    // set to 0
    DDRD |= ((1<< DDD5) | (1<< DDD6)
    | (1<< DDD7));
    PORTD &= ~(1 << PORTD5);
    PORTD &= ~(1 << PORTD6);
    PORTD &= ~(1 << PORTD7);
}

//void wait_init(void)
//{
//    for(int i=0; i<WAIT_TIME; i
//    ++);
//}

void startup (uint16_t* stable_boot)
{
}

```

```

float diff;

if (gate_times > GATE_COUNT_LIMIT)
{
    // disable global interrupts
    cli();

    // calculate mean
    mean = (float) count /
        gate_times;

    // evaluate if signal is
    // stable
    diff = mean - last_mean;
    if ( (diff < THRESHOLD) )
    {
        // one time stable
        *stable_boot = *stable_boot
            + 1;
    }
    else
    {
        // continue booting
        *stable_boot = 0;
    }

    last_mean = mean;
    gate_times = 0;
    count = 0;
    // clear past interrupts
    //TIFR2 |= (1 << OCF2B) |(1 <<
    //OCF2A) |(1 << TOV2);
    TIFR2 |= (1 << OCF2A);
    TIFR0 |= (1 << OCF0A);
    //reset timers
    TCNT0 = 0;
    TCNT2 = 0;
    // enable interrupts
    sei();
}

int main(void)
{
    uint16_t stable_boot = 0;

    setup_timer_0_counter();
    setup_timer_2_gate();
    setup_gpios();

    //enable global interrupts
    sei();

    // while booting turn on led
    PORTB |= (1 << PORTB0);

    //startup function
    while (stable_boot <
        TIMES_STABLE_BOOT)
    {
        startup(&stable_boot);
    }
    // turn off led
    PORTB &= ~(1 << PORTB0);

    while(1)
    {
        if (gate_times >
            GATE_COUNT_LIMIT)
        {
            // disable global
            // interrupts
            cli();

            // calculate mean
            mean = (float) count /
                gate_times;

            // evaluate if signal has
            // been detected
            if ( (mean - last_mean) >
                THRESHOLD)
            {
                // signal detected
                switch (decoding)
                {
                    case 0:
                        // pd5 = 1;
                        PORTD ^= (1 <<
                            PORTD5);
                        break;
                    case 1:
                        // pd6 = 1;
                        PORTD ^= (1 <<
                            PORTD6);
                        break;
                    case 2:
                        // pd7 = 1;
                        PORTD ^= (1 <<
                            PORTD7);
                        break;
                    default:
                        break;
                }
                decoding = (decoding +
                    1) % 3;
            }
            else
            {
                // not detected
            }
        }
    }
}

```

```

last_mean = mean;
gate_times = 0;
count = 0;
// clear past interrupts
// clear past interrupts
//TIFR2 |= (1 << OCF2B) |(1
    << OCF2A) |(1 << TOV2); }
TIFR2 |= (1 << OCF2A); }

TIFR0 |= (1 << OCF0A);
//reset timers
TCNT0 = 0;
TCNT2 = 0;
// enable interrupts
sei();

```

### 3.2.2. Diseño del codificador digital para la transmisión

#### Introducción

El objetivo del microcontrolador en la parte de transmisión, codifica mensajes según los botones pulsados. Cada pulsador codifica un símbolo diferente para que el receptor actúe de manera distinta según el botón pulsado. El algoritmo de comunicación entre transmisor y receptor se realiza de manera asíncrona. Para diferenciar los símbolos digitales se ha de tener en cuenta el tipo de modulación ASK, donde el '1' implica recibir señal y el '0' no haberla recibido. /paragraph Los relojes o timers encargados de la codificación y decodificación tanto en transmisión como en recepción, deben trabajar a la misma tasa de baudios para identificar correctamente los mensajes.

#### Codificación de los mensajes

La codificación de los diferentes símbolos se desarrolla de forma que, los algoritmos de codificación y decodificación se realicen de la forma más sencilla y robusta posible, teniendo en cuenta el tipo de modulación ASK.

Cada símbolo se diferencia en función del número de ciclos de TIMER2, *gate*, que mantengan activos la señal de RF. Tanto emisor como receptor poseen un *gate timer* configurado a la misma frecuencia. El receptor decodifica los símbolos en función del número de ciclos de timers detectados.

Previamente al envío del símbolo, se envía el *header*, el cual es utilizado para estabilizar al transmisor y sincronizar al receptor. Este *header* concluye con un '0', transmitiendo, seguidamente el símbolo correspondiente.

Un ejemplo real del algoritmo se puede observar en la figura 16, donde se aprecia el final del header y posteriormente el envío del código correspondiente.

#### Limitaciones

En anteriores versiones del código, la lógica de transmisión pretendía ser esencialmente la misma, pero fue implementada de una forma más limitada. Esta limitación fue provocada por errores de programación. Los errores vinieron dados principalmente porque las rutinas de tratamiento de interrupción (IRQ), pretendían hacer funciones que no les correspondían.

En la versión final, anteriormente mostrada, las funciones principales son implementadas en main(), mejorando esta flexibilidad de implementación. Cabe recalcar, que el código de la versión anterior, también cumplía su función.

## Presentación del código

Todas las versiones de los códigos, tanto del emisor como del receptor, se encuentran en el repositorio de GitHub<sup>15</sup>. La versión final del código del transmisor se muestra a continuación. El código se presenta en dos columnas por hoja, el orden de lectura es el siguiente: primera columna, segunda columna, hoja siguiente.

```
#include <avr/io.h>
// definiciones de registros
    especificos del controlador de io
#include <avr/interrupt.h>
// gestion de IRQs
#include <stdint.h>
// definiciones de tipos estandar

// while testing PB0, in board PB7
#define PIN_OUT PINB7
#define PIN_TEST PIND7
#define WAIT_TIME 100000UL
#define DELAY 10000UL

uint8_t volatile lock_button = 0;
uint8_t volatile timer_complete = 0;
uint16_t volatile code = 0;
#define TEST 1
#define CODE0 TEST
#define CODE1 4
#define CODE2 8
#define CODE3 12

#ifndef DEBUG
void delay(void)
{
    for(uint16_t i=0; i<DELAY; i++)
}
void wait_time(void)
{
    for(int i=0; i<WAIT_TIME; i++)
}
#endif DEBUG
#define COUNT 100000UL
void debug_toggle_led(void)
{
    //toggle pb0
    PORTB ^= (1 << PORTB0);
    for(int i=0; i<COUNT; i++);
    PORTB ^= (1 << PORTB0);
    for(int i=0; i<COUNT; i++);
}
void debug_stop_exec(void)
{
    //toggle pb0
    PORTB ^= (1 << PORTB0);
    for(int i=0; i<COUNT; i++);
    PORTB ^= (1 << PORTB0);
    for(int i=0; i<COUNT; i++);
}

//toggle pb0
PORTB ^= (1 << PORTB0);
for(int i=0; i<COUNT; i++);
PORTB ^= (1 << PORTB0);
for(int i=0; i<COUNT; i++);
while(1);
}

#endif

void enable_timer2(void)
{
    // TIMER2 interrupt COMP_A
        interrupt clear by
        writing a logic 1 if a 1
        is written
    TIFR2 |= (TIFR2 & (1 <<
        OCIE2A));
    //restart the count
    TCNT2 = 0;

    //enable timer2 interrupt
    mask
    TIMSK2 = (1 << OCIE2A);
}

void transmit_sync(void)
{
    // first transmit sync
    sequence
    PORTB &= ~(1 << PIN_OUT);
    wait_time();
    PORTB |= (1 << PIN_OUT);

    void setup_timer_2_gate(void)
    {
        // set CTC mode and COM on toggle
        with compare match
        TCCR2A = (1 << COM2A0) | (1 <<
            WGM21);
        //TCCR2A = (0 << COM2A0) | (1 <<
        //    WGM21);
        OCR2A = 220;
        // select clock source as internal
        prescaler 32
    }
}
```

<sup>15</sup><https://github.com/josegu05/tfg2>

```

TCCR2B = (1 << CS21) | (1 << CS20           //debug_toggle_led();
);                                                 }

// PB3 as output compare match A
// timer 2
DDRB |= (1 << DDB3);

// set TIMER2 interrupt COMP_A mask
// enable
// TIMSK2 = (1 << OCIE2A);

// enable timer2 in power
// reduction
// PRR &= ~(1 << PRTIM2);

}

// IRQ of gate timer
ISR(TIMER2_COMPA_vect)
{
    //gate
    timer_complete = 1;
}

// IRQ if any button is pressed
ISR(PCINT2_vect)
{
    // wait for button oscillations
    delay();
    // auto clear the interrupt
    // read which pin invoked the
    // interrupt line
    uint8_t irq = 0;
    irq = PIND;

    if (lock_button == 0)
    {
        lock_button = 1;

        // set code value depending on
        // which button has been
        // pressed
        // PIND reg has state value of
        // the PINx -> 0 is button
        // pressed
        if ((irq & (1 << PIND5)) == 0
            )
            code = CODE1;
        if ((irq & (1 << PIND6)) == 0
            )
            code = CODE2;
        if ((irq & (1 << PIN_TEST))
            == 0
            )
            code = TEST;
    }
}
}

}

void setup_gpio_pins(void)
{
    //configurar gpios pd5, pd6, pd7
    // as input
    DDRD &= ~(1 << DDD5);
    DDRD &= ~(1 << DDD6);
    DDRD &= ~(1 << DDD7);

    // activate pull-ups
    PORTD |= (1 << PORTD5);
    PORTD |= (1 << PORTD6);
    PORTD |= (1 << PORTD7);

    // activate the pin change
    // interrupts
    // first activate PC2vector
    PCICR |= (1 << PCIE2);
    // second activate pd5 -> PC21,
    // pd6 -> PC22, pd7 -> PC23 to
    // trigger interrupt
    PCMSK2 |= ((1 << PCINT21) |(1 <<
        PCINT22) |(1 << PCINT23));

    //config pb7 as output
    DDRB |= (1 << PIN_OUT);
    //set init high level pnp
    //transistor
    PORTB |= (1 << PIN_OUT);
    //PORTB &= ~(1 << PIN_OUT);
}

int main(void)
{
    uint16_t i = 0;
    uint8_t irq = 0;

    setup_timer_2_gate();
    setup_gpio_pins();

#ifndef DEBUG
    DDRB |= (1 << PORTB0);
#endif

    //enable global interrupts
    sei();

    while(1)
    {
        // code transmit algorithm
    }
}

```

```

// if code > CODE0, means code
// set with CODEn value
if (code > CODE0 &&
    lock_button == 1)
{
    //wait for realised
    //while (button-pressed)
    irq = PIND;
    irq = ~irq & ((1 << PIND5)
                   | (1 << PIND6));
    while ( irq != 0 )
    {
        irq = PIND;
        irq = ~irq & ((1 <<
                        PIND5) | (1 << PIND6)
                      );
    }
    //disable PC_INT
    PCICR &= ~(1 << PCIE2);

    transmit_sync();
    enable_timer2();
    //transmit code sequence
    for (i=0; i<code; i++)
    {
        // set low level pnp
        // transistor (activate
        // RF)
        PORTB &= ~(1 << PIN_OUT)
        ;
        // wait for timer
        // interrupt
    }
    while(!timer_complete);      }
}

timer_complete = 0;
}
PORTB |= (1 << PIN_OUT);

// disable timer2
interrupts
TMSK2 &= ~(1 << OCIE2A);

code = 0;
lock_button = 0;
timer_complete = 0;
//enable PC_INT
PCICR |= (1 << PCIE2);
}

// TEST mode
if (code == TEST &&
    lock_button == 1)
{
    //wait for button released
    irq = PIND;
    while ( (irq & (1 <<
                      PIN_TEST)) == 0 )
        irq = PIND;

    PORTB ^= (1 << PIN_OUT);

    code = 0;
    lock_button = 0;
    timer_complete = 0;
}
}

```

## 4. Cronología del proyecto y diagrama de Grant

En este apartado se expone a modo de resumen, un historial de desarrollos del proyecto los cuales no se terminaron llevando a cabo por diferentes motivos. Se trata de dar a entender las diferentes alternativas que han surgido, el aprendizaje que se ha obtenido de las mismas y el por qué fue abandonada su línea de desarrollo.

### 4.1. transmisor FM a varactor

#### Desarrollo técnico

la idea original del proyecto fue el realizar un transmisor y receptor de FM digital. El sistema se pensó de forma que su frecuencia de trabajo fuera aproximadamente 1MHz. El transmisor modulaba la frecuencia portadora de forma que una tensión inversa de baja frecuencia (moduladora), se aplicaba a unos diodos capacitivos o varactores. Estos diodos varactores se encontraban de circuito tanque en el bucle de oscilación. Este modelo de transmisor funcionaba correctamente.

Por otro lado, el receptor de FM era bastante complejo. Un filtro de entrada exigente, seguido de un amplificador, y a continuación una etapa de filtrado muy agudo a la frecuencia de la portadora. Esta estrategia permite la conversión de una señal modulada en FM a una señal modulada en AM. Posteriormente se realizó la etapa del demodulador AM el cual requería de amplificadores y acondicionamiento de señal tedioso.

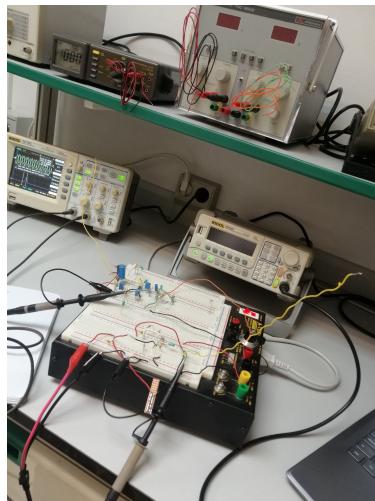


Figura 26: Primer emisor y receptor montados en placa de entrenamiento

#### Motivos de reemplazo

Los motivos de remplazo de este modelo fueron varios. En primer lugar, el sistema de transmisor y receptor no funcionaba a distancias mayores de pocos centímetros. Esto era debido principalmente a la baja potencia radiada. La bobina del circuito tanque de oscilación debía tener numerosas espiras para que a esa frecuencia tan baja consiguiera inducir, por acople magnético, tensión en el receptor. A parte de el problema mencionado, el transmisor tenía un excesivo consumo estático. En cuanto al receptor, los problemas también fueron varios. Los filtros, tan agudos eran complicados de fabricar, más si los filtros debían ser diseñados a la frecuencia de la portadora como conversión

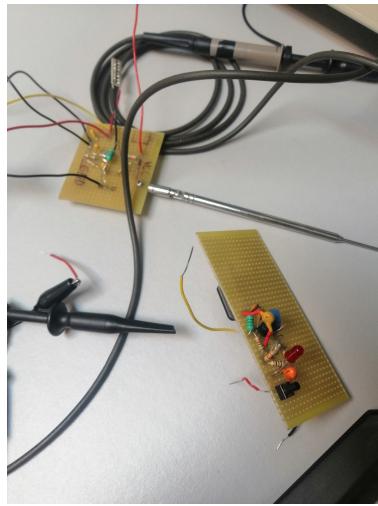


Figura 27: Modelos de transmisores soldados placa

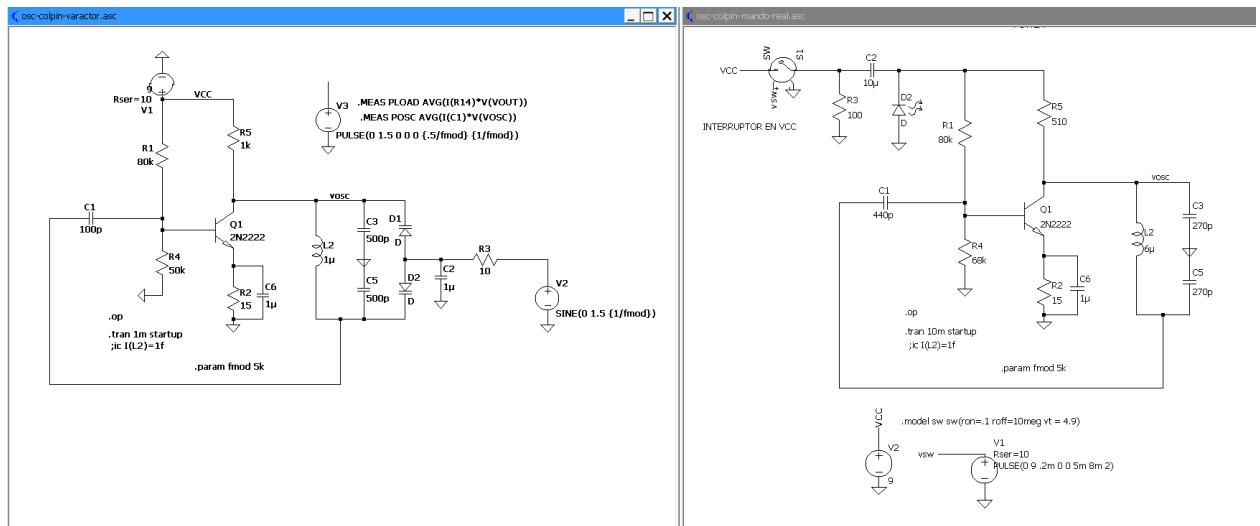


Figura 28: Modelos esquemáticos de transmisores

directa. Además, la necesidad de implementar también el demodulador de AM tomando como entrada la señal filtrada se hacía complicado. Finalmente se opta por mejorar el diseño del sistema proponiendo una segunda versión.

#### 4.2. receptor superheterodino FM

##### Desarrollo técnico

Como intento de mejora a la anterior versión, la cual era de conversión directa, se trató de diseñar un receptor superheterodino. A su vez, se mantuvo el diseño del transmisor a varactores, pero se cambió la frecuencia de trabajo a una mayor, unos 6 MHz.

El diseño consistía en un filtro de entrada que era mezclado en un mezclador con un oscilador. Después se realizaba el tratamiento con la señal de frecuencia intermedia. Un amplificador de dos etapas y posteriormente a un rectificador con filtro paso bajo para demodular la señal. Se muestra

un esquemático en la figura ?? del receptor sin la parte digital, la cual se explica en el apartado ?? La figura ??, consta de la parte de RF explicada a la derecha, y de la interfaz con la parte digital, un amplificador con realimentación positiva denominado *Smith-trigger*.

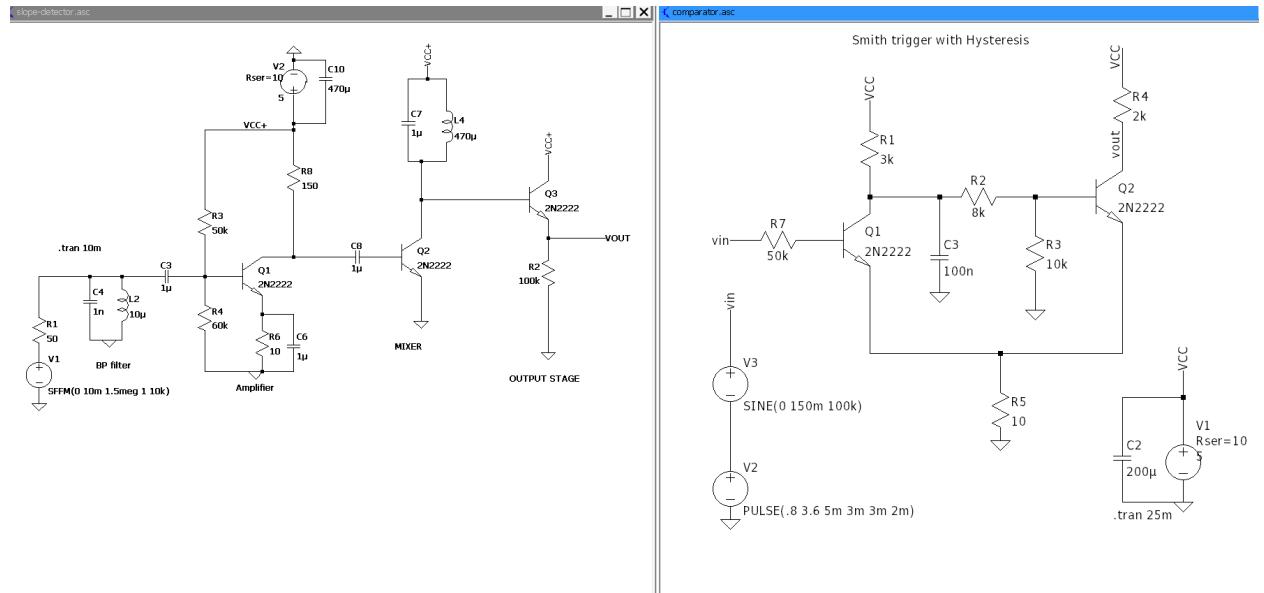


Figura 29: Modelo esquemático del receptor superheterodino de FM

### Motivos de reemplazo

Este diseño funcionaba bien cuando se conectaba a la entrada un generador de frecuencias a la frecuencia de trabajo de muy baja potencia.

El problema surgía cuando se trataba de probar con el transmisor. El receptor no tenía buena selectividad y los amplificadores de frecuencia intermedia, los cuales no estaban correctamente diseñados, producían oscilaciones.

Gracias al mezclador, el circuito era capaz de detectar señales de muy baja potencia con muy buena selectividad. A pesar de todo, el mezclador era bastante sensible al ruido, ya que producía bastantes armónicos, producidos por efectos de segundo orden. Esto explica el por qué al conectarlo con el generador, lo más cercano a un tono puro, funcionaba correctamente, sin embargo cuando se trataba de enlazar con la señal del transmisor la cosa cambiaba.

También, el diseño general, me parecía que se utilizaban demasiados componentes para unas prestaciones tan bajas. El diseño debía ser sencillo y funcional.

Debido a la complejidad a cambio de ninguna clase de beneficio, se opta por transicionar a un esquema de modulación AM.

### 4.3. Máquina de estados digital

#### Desarrollo técnico

El objetivo de este circuito estaba pensado para dar una aplicación a la señal digital recibida por el receptor. El circuito consistía en una máquina de estados cuya salida tenía una aplicación concreta. En este caso, por cada señal recibida, poner en alto una de sus salidas, manteniendo el estado de las anteriores hasta completar el ciclo, donde todas volverían a estado bajo. El esquemático del circuito digital de la máquina de estados se muestra en la figura ???. Además, se añade en la figura ???, la integración del receptor superheterodino FM anterior, junto al circuito digital montados en una placa de entrenamiento.

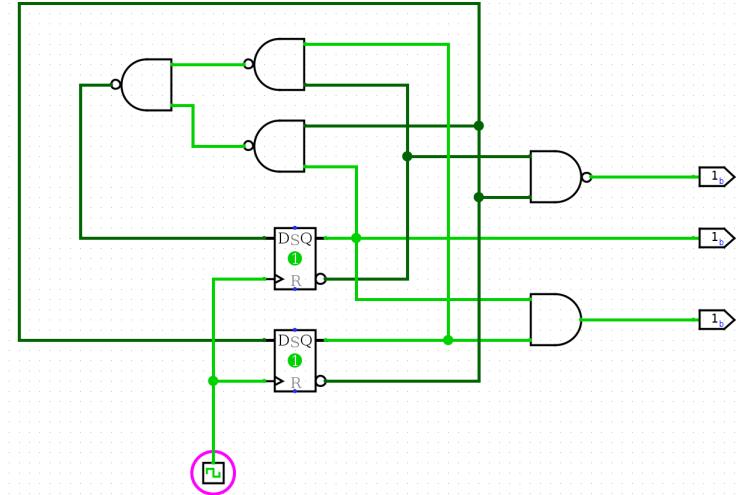


Figura 30: Esquemático de la máquina estados

### Motivos de reemplazo

El diseño del circuito digital era correcto. Sin embargo, al realizar la integración con el receptor daba muchos problemas debido a que la señal de output del receptor no era fiable. Esta señal, al no estar bien filtrada poseía componentes residuales de alta frecuencia que provocaban que el smith trigger metiera señales falsas. esto provocaba un comportamiento no deseado del circuito.

Para solucionar este hecho, se hizo uso de un microcontrolador. El micro abre una inmensidad de posibilidades como la recepción de múltiples canales, reprogramable en función del uso específico, todo contenido en un menor espacio e incluso con un precio más económico.

#### 4.4. Alternativa viable: conversión directa

**Desarrollo técnico** Este diseño, más simplificado, se trataba de un sistema ASK, con un receptor de conversión directa. La parte principal del circuito era la etapa de entrada, compuesta por un filtro de entrada sintonizable, y una etapa de amplificación de corriente formada por un par darlington. Parte de la salida se realimentaba positivamente al filtro de entrada, aumentando la sensibilidad del receptor. La salida del par darlington, era tratada y amplificada generando la señal de salida.

Se muestra un esquema del receptor en la figura ?? junto con la implementación real del diseño con algunas modificaciones en la figura ??.

### Motivos de reemplazo

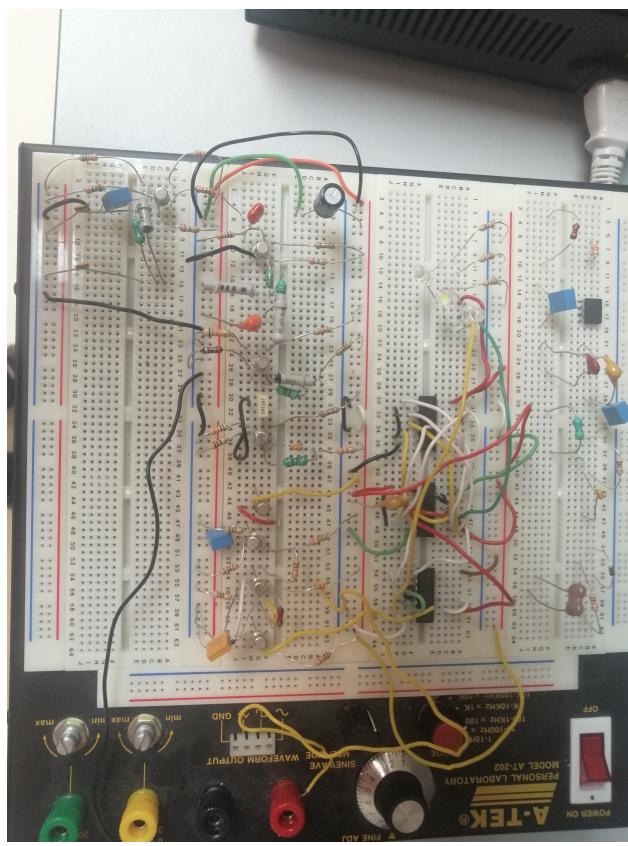


Figura 31: Implementación del receptor superheterodino de FM completo

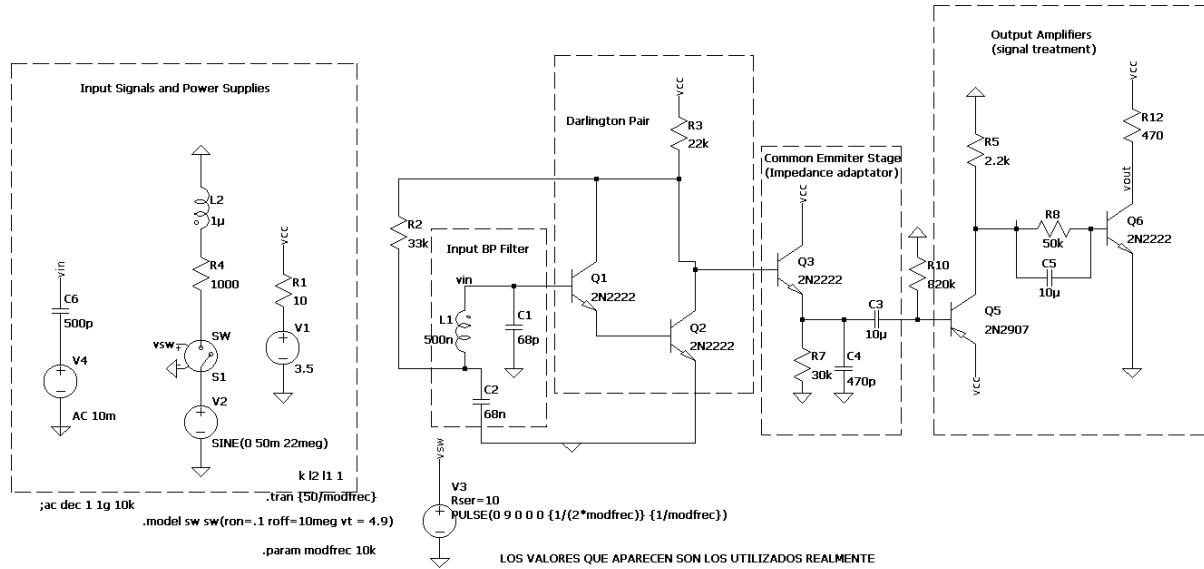


Figura 32: sch radio2

Este diseño, tenía un objetivo principal, ser sencillo y funcional. A pesar de conseguir una sintonización y comunicación adecuada, no estaba suficientemente contento con la distancia. Se trató

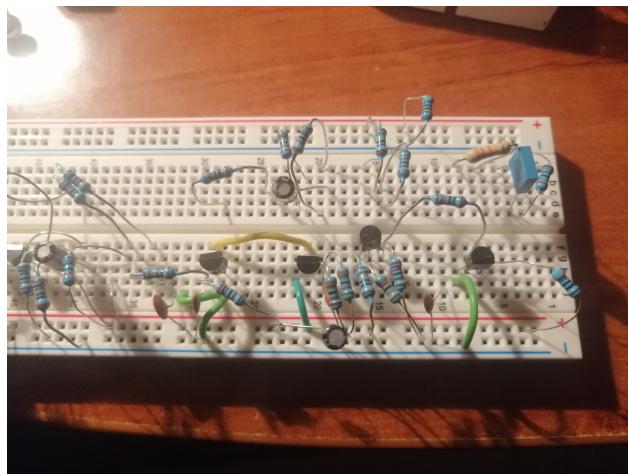


Figura 33: *radio2 real*

de diseñar amplificadores de RF a la entrada junto a un filtro inicial. En ese momento descubrí la dificultad de diseñar amplificadores de RF y buenos filtros.

Otra estrategia, fue la de incorporar un mezclador a la entrada. Tras varias pruebas sin obtener mejoras claras, se cambió al diseño presentado en el proyecto, un receptor super-regenerativo, el cual consigue ser funcional con muy pocos componentes.

#### 4.5. Diagrama de Gantt

En la figura 34 se muestra el diagrama de Gantt del desarrollo general del proyecto.

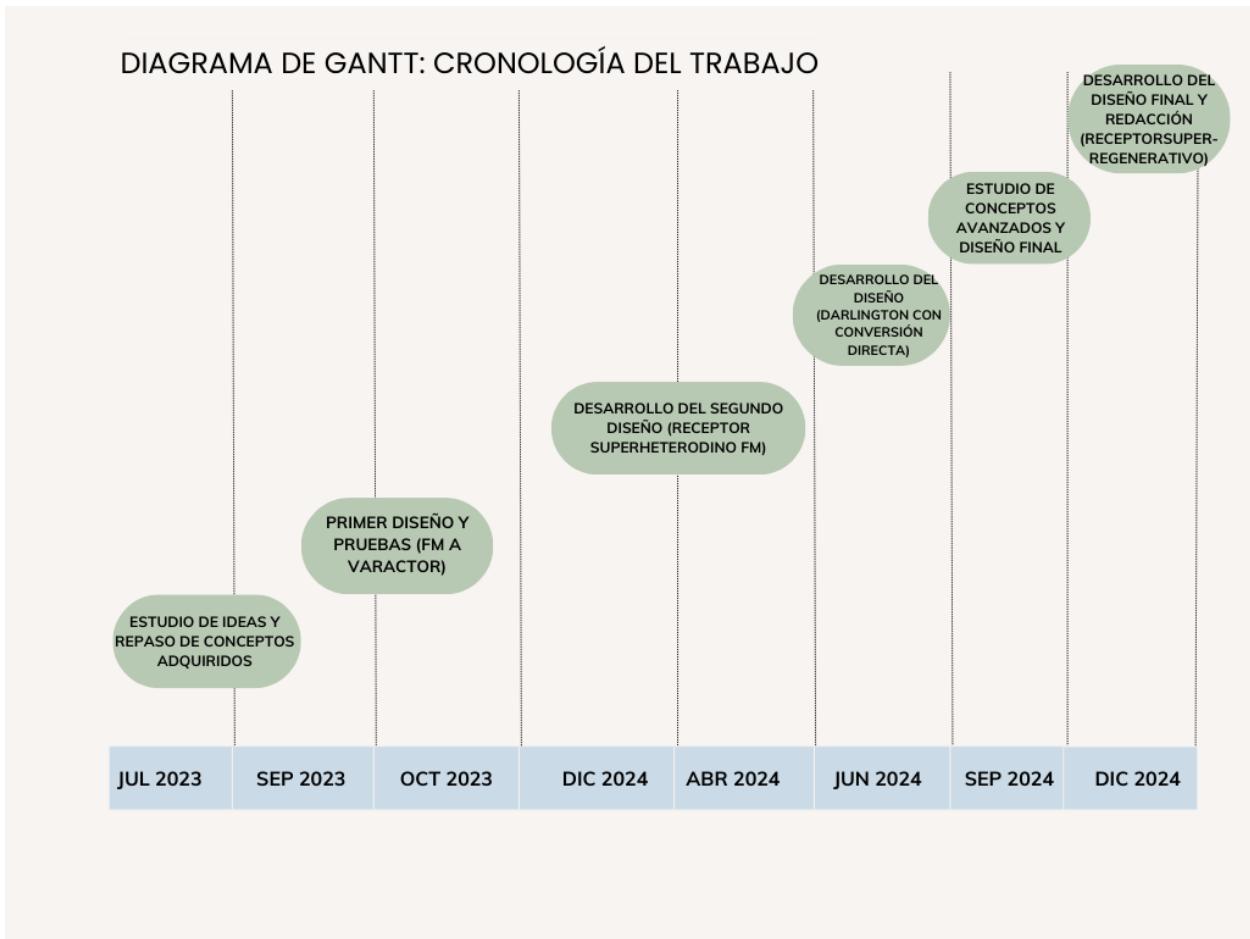


Figura 34: *Diagrama de Gantt del desarrollo del proyecto*

## 5. Resultados y conclusiones

Se concluye con un análisis acerca de si se completaron los objetivos propuestos en el apartado REF. En primer lugar, el objetivo de conocimiento de diseño de un sistema de comunicaciones lo considero satisfactorio. Al haber comprendido e interiorizado las bases de los circuitos de radiofrecuencia y puesto en práctica diversas técnicas de recepción de señales de baja potencia como mezcladores, amplificadores o filtros cada uno de ellos diseñado al nivel de componente. Al tratarse de un diseño realizado con componentes discretos, fueron múltiples los problemas encontrados hasta lograr un diseño final. A su vez fueron múltiples también, las alternativas propuestas de diferentes desarrollos del sistema. Esto me ha permitido conocer diferentes técnicas de diseño de circuitos de radio. En general, considero satisfactorio este objetivo. En cuanto a la puesta en práctica de los diferentes campos considero que es satisfactorio, consiguiendo relacionar los conceptos de los lazos de control en conjunto con los modelos de los componentes electrónicos y contrastados con la realidad. También conocimientos de la rama informática a la hora de programar los microcontroladores y establecer todo el entorno de desarrollo. Con la correspondiente conexión entre los sistemas analógicos. Por último el diseño del transformador de la antena englobando toda la parte de radiofrecuencia, antenas y electromagnetismo. Considero satisfactorio este punto. Finalmente, tras tantas posibles alternativas y problemas, se converge hacia un diseño sencillo y funcional con los requisitos tan exigentes que se habían propuesto del diseño desde la raíz.

En resumen, considero este proyecto como una experiencia, aunque dura y complicada, que me ha llevado mas tiempo del esperado, ha sido satisfactoria, y que ademas me ha permitido llevar con facilidad mi desarrollo a nivel profesional, debido a la asimilacion de numerosos conceptos estudiadas en las materias y llevados a la practica.

## 6. Bibliografía

## 7. Índice de figuras

### Índice de figuras

1. Esquema de una posible modulación ASK . . . . .	5
2. Representación de un transistor como cuadripolo lineal. . . . .	7
3. derecha: Amplificador sin realimentación, izquierda: Sistema realimentado . . . . .	8
4. Amplificador de tensión con realimentación serie paralelo ideal. . . . .	9
5. Amplificador de transresistencia con realimentación paralelo paralelo real. . . . .	10
6. Esquema general de un SOC. . . . .	10
7. Diagrama de bloques del Atmega328p, en detalle la CPU . . . . .	12
8. Mapa de memoria de datos direccionada por bytes . . . . .	12
9. Diagrama de bloques general del proyecto . . . . .	13
10. Esquema eléctrico del transmisor . . . . .	15
11. A) Modelo en pequeña señal del bucle de oscilación para frecuencias medias B) Modelo en pequeña señal del oscilador sustituyendo el condensador de realimentación $C_1$ por su equivalente en parámetros $Y$ . . . . .	17
12. Modelo de transformador ideal . . . . .	20
13. Simulación de $V_C$ modulada por $V_{digital}$ . . . . .	21
14. Simulación de la FFT de $V_C$ de forma general . . . . .	22
15. Simulación de la FFT de $V_C$ ampliada a la frecuencia de trabajo . . . . .	22
16. Experimental: captura de oscilloscopio de $V_C$ modulada por $V_{digital}$ . . . . .	23
17. Placa de transmisor soldada . . . . .	23
18. Esquema eléctrico del receptor . . . . .	24
19. A) Modelo en pequeña señal del bucle de oscilación para frecuencias de RF B) Modelo en pequeña señal del oscilador sustituyendo el condensador de realimentación $C_2$ por su equivalente en parámetros $Y$ . . . . .	26
20. Explicación del comportamiento de la señal de quench . . . . .	28
21. Simulación de los puntos de interés en varios ciclos ampliados . . . . .	29
22. Simulación de $V_{out}$ junto a $V_{quench}$ . . . . .	30
23. Experimental: captura de oscilloscopio de la frecuencia de trabajo . . . . .	30
24. Experimental: captura de oscilloscopio de $V_C$ (negro) y $V_{quench}$ (rojo) . . . . .	30
25. Experimental: captura de oscilloscopio de $V_{out}$ (negro) y $V_{quench}$ (rojo) . . . . .	31
26. primer emisor receptor real . . . . .	40
27. transmisores soldados placa . . . . .	41
28. transmisores schematicos . . . . .	41
29. maquina estados esquematico . . . . .	42
30. receptor general schematic . . . . .	43
31. receptor general real . . . . .	43
32. sch radio2 . . . . .	44
33. radio2 real . . . . .	44

34. Diagrama de Gantt del desarrollo del proyecto . . . . .	45
---	----