



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**DISEÑO DE UNA INTERFAZ DE UN
SISTEMA DE CONTROL DIGITAL
PARA ANTENAS EN ARREGLO DE
FASE**

TESIS

Que para obtener el título de
Ingeniero Eléctrico Electrónico

P R E S E N T A

Jorge Luis Badillo Díaz

DIRECTOR DE TESIS

Dr. Saúl de la Rosa Nieves



Ciudad Universitaria, Cd. Mx., 2024

Agradecimientos

A mi familia:

Les doy las gracias por el apoyo que me han brindado a lo largo de mi vida. Estos últimos años han sido difíciles y agradezco infinitamente tenerlos a mi lado. Cada uno de ustedes me ha dejado una enseñanza diferente con la cual he podido afrontar la vida con optimismo y motivación.

A mi hermano, Iván:

Los caminos de la vida son más turbulentos que un descenso en avión en las costas de Cancún y las batallas que uno debe librar cada día se vuelven más complejas. Son en esos momentos en los que agradezco tener a un guerrero tan fuerte como tú a mi lado. Gracias por los consejos y las enseñanzas.

A mi papá:

Me enseñaste a luchar hasta el final y jamás rendirme ante cualquier adversidad. Gracias por permitirme ver las distintas tonalidades de la vida, y que de esta manera yo pudiera elegir mi camino. Fuiste el mejor papá del mundo, y siempre serás mi “campeón mundial”. Por mucho tiempo estuve enojado con la vida porque no te permitió estar junto a mí en este momento, sin embargo, ahora sé que mientras yo te lleve en el corazón tu siempre estarás conmigo. *Que la fuerza te acompañe.*

A mi mamá:

Gracias por tu apoyo en los momentos más difíciles que he pasado y por siempre estar ahí para escucharme. La resiliencia y humildad son valores que he aprendido de ti todos estos años. No me queda la menor duda que no estaría aquí parado si no hubieras estado para mis todas esas noches de angustia. Espero algún día poder ser tan fuerte como tú.

“No se dice, se hace”.

Índice general

Capítulo 1: INTRODUCCIÓN.....	9
1.1 Presentación del problema	9
1.2 Objetivos	10
1.2.1 Objetivo general	10
1.2.2 Objetivos específicos.....	10
Capítulo 2: MARCO TEORÍCO	11
2.1 Antenas.....	11
2.2 Antenas en arreglos de fase.....	12
Capítulo 3: SISTEMA DE CONTROL DIGITAL ESCALABLE Y RECONFIGURABLE PARA ANTENAS EN ARREGLOS DE FASE.....	16
3.1 Requerimientos del sistema.....	16
3.2 Funcionamiento	16
3.3 Arquitectura.....	18
3.4 Análisis de módulos	21
Capítulo 4: DISEÑO	24
4.1 Diseño conceptual	24
4.1.1 Interfaz de usuario.....	24
4.1.2 Propuestas de diseño para seguimiento de satélites	25
4.1.3 Propuesta de diseño para enviar fases para cada elemento desde la interfaz.	26
4.1.4 Propuesta de diseño para graficar el patrón de radiación de la antena.....	27
4.2 Diseño a nivel de sistema	27
4.2.1 Interfaz de usuario	27
4.2.2 Seguimiento de satélites.....	32
4.2.3 Envío de fases para cada elemento.....	33
4.2.4 Graficar el patrón de radiación.....	35
4.3 Diseño de detalle	36

4.3.1	Interfaz de usuario	36
4.3.2	Seguimiento de satélites	39
4.3.3	Envío de fases para cada elemento.....	48
4.3.4	Graficar patrón de radiación de la antena.....	58
Capítulo 5:	RESULTADOS	68
Capítulo 6:	CONCLUSIONES.....	75
Capítulo 7:	REFERENCIAS	77
Anexo 1	78

Índice de figuras

Figura 2-1. Patrón de radiación de una antena parabólica.	11
Figura 2-2. Propagación de ondas electromagnéticas emitidas por un elemento. [2].....	12
Figura 2-3. Propagación e interferencia de ondas electromagnéticas emitidas por dos elementos. [2]	12
Figura 2-4. Propagación e interferencia de ondas electromagnéticas emitidas por seis elementos. [2]	13
Figura 2-5. Ondas con diferencia de fase.	13
Figura 2-6. Propagación e interferencia de ondas electromagnéticas con desfase, emitidas por dos elementos. [2].....	14
Figura 2-7. Reorientación del haz provocada por la diferencia de fase de los elementos. [3].....	14
Figura 2-8. Estructura interna de un diodo PIN. [2].....	15
Figura 3-1. Diagrama de arreglo de fase y alimentador. [2]	17
Figura 3-2. Posiciones para el desplazador de fase de dos bits con líneas conmutadas. [2].....	18
Figura 3-3. Diagrama de arquitectura del sistema de control. [2].....	19
Figura 3-4. Diagrama de la estructura de un dispositivo esclavo. [2].....	19
Figura 3-5. Diagrama simplificado de la etapa de configuración del sistema. [2].....	20
Figura 3-6. Diagrama simplificado de la etapa de ejecución del sistema. [2].....	21
Figura 3-7. Estructura interna de la FPGA maestra. [2].....	21
Figura 3-8. Interconexión de módulos para la etapa de configuración. [2]	22
Figura 3-9. Interconexión de módulos para la etapa de ejecución. [2]	23
Figura 4-1. Diagrama de radiación de un sintetizador de antenas de arreglo de fase de parche de microcinta acopladas. [4]	25
Figura 4-2. Puertos seriales resaltados en la estructura del FPGA.....	28
Figura 4-3. Diagrama de bloques para el envío de fases desde la interfaz.	28
Figura 4-4. Puerto de recepción resaltado en la FPGA Maestra.	29
Figura 4-5. Diagrama de bloques para el envío de los ángulos desde la interfaz.	29
Figura 4-6. Núcleo de ejecución resaltado en la estructura interna del FPGA.	30
Figura 4-7. Diagrama de bloques para el envío de las fases a la interfaz.	30
Figura 4-8. Diagrama de bloques simplificado de la interfaz con la FPGA maestra.	31
Figura 4-9. Orbita de un satélite calculada por MATLAB.	32
Figura 4-10. Diagrama de bloques del puerto de recepción de ángulos de orientación. [2].....	33
Figura 4-11. Módulos involucrados en el envío de fases.	34
Figura 4-12. Diagrama simplificado de señales para los Puertos seriales.	34
Figura 4-13. Diagrama simplificado de la conmutación para el envío de fases desde la interfaz. ..	35
Figura 4-14. Diagrama simplificado para la obtención de las fases del sistema.....	36
Figura 4-15. Menú principal de la interfaz de usuario.	36

Figura 4-16. Menú inicial de la interfaz de usuario.	37
Figura 4-17. Funcionamiento de la equivalencia de fases. [2].....	38
Figura 4-18. Ejemplo del redondeo y codificación de una fase. [2]	38
Figura 4-19. Mensaje de error en puerto serial.	39
Figura 4-20. Mensaje de error en la detección de satélite.....	39
Figura 4-21. Menú para seguimiento de satélites en modo manual.....	41
Figura 4-22. Radar para detección de satélites.....	41
Figura 4-23. Diagrama de bloques del transceptor UART [2].....	43
Figura 4-24. Diagrama de estados del puerto de ángulos de orientación UART.....	44
Figura 4-25. Diagrama de conexiones del puerto de ángulos de orientación UART.....	47
Figura 4-26. Diagrama de bloques del puerto de ángulos de orientación UART.....	47
Figura 4-27. Diagrama de estados de la máquina de recepción de fases.	49
Figura 4-28. Diagrama de conexiones de la máquina de recepción de fases.....	53
Figura 4-29. Diagrama de bloques de la máquina de recepción de fases.....	53
Figura 4-30. Diagrama de estados de la Máquina de distribución de fases.	54
Figura 4-31. Diagrama de conexiones de la máquina de distribución de fases.	57
Figura 4-32. Diagrama de bloques de la máquina de distribución de fases.....	57
Figura 4-33. Conjunción de las máquinas para recibir las fases.	58
Figura 4-34. Diagrama de estados de la máquina de detección de fases.	59
Figura 4-35. Diagrama de conexiones de la máquina de detección de fases.	63
Figura 4-36. Diagrama de bloques de la máquina de detección de fases.....	63
Figura 4-37. Diagrama de estados de la máquina de envío de memoria.....	64
Figura 4-38. Diagrama de conexiones de la máquina de envío de memoria.	67
Figura 4-39. Diagrama de bloques de la máquina de envío de memoria.....	67
Figura 5-1. Diagrama esquemático del módulo de pruebas.....	68
Figura 5-2. Experimento en modo manual.....	69
Figura 5-3. Simulación de órbita de la estación espacial.	69
Figura 5-4. Tabla de ángulos con su respectivo tiempo, obtenidos de la simulación orbital.....	69
Figura 5-5. Habilitación del botón “Realizar seguimiento sobre el satélite”.....	70
Figura 5-6. Ventana de comandos de Matlab con los ángulos de orientación que se enviaron.....	70
Figura 5-7. Paso del satélite PAINANI-2 sobre el punto de observación.....	71
Figura 5-8. Seguimiento de satélites en modo automático.....	71
Figura 5-9. Elección de archivos en la interfaz.....	72
Figura 5-10. Ventana para el envío de fases.	72
Figura 5-11. Recibimiento de fases en el FPGA.....	73
Figura 5-12. Fases recibidas por la interfaz.	73
Figura 5-13. Patrón de radiación calculado en la interfaz.....	74

Índice de tablas

Tabla 3-1. Matriz de funcionamiento de un arreglo hipotético de dos bits. [2].....	18
Tabla 4-1. Ejemplo de valores generados por la ecuación 4.1.....	37
Tabla 4-2. Ejemplo de obtención de ángulos para satélite.....	42
Tabla 4-3. Descripción de señales del transceptor UART.....	42
Tabla 4-4. Descripción de señales del puerto de ángulos de orientación UART.....	44
Tabla 4-5. Descripción de estados y transiciones del puerto de ángulos de orientación UART.	45
Tabla 4-6. Tabla de estados presentes y futuros del puerto de ángulos de orientación UART.....	46
Tabla 4-7. Funciones de entradas y salidas del puerto de ángulos de orientación UART.....	46
Tabla 4-8. Descripción de señales de la máquina de recepción de fases.....	49
Tabla 4-9. Descripción de estados y transiciones de la máquina de recepción de fases.....	50
Tabla 4-10. Estados presentes y futuros de la máquina de recepción de fases.....	52
Tabla 4-11. Funciones lógicas de la máquina de recepción de fases.....	52
Tabla 4-12. Descripción de señales de la máquina de distribución de fases.....	54
Tabla 4-13. Descripción de los estados de la máquina de distribución de fases.....	55
Tabla 4-14. Estados presentes y futuros de la máquina de distribución de fases.....	56
Tabla 4-15. Funciones lógicas de la máquina de distribución de fases.....	57
Tabla 4-16. Descripción de señales de la máquina de detección de fases.....	59
Tabla 4-17. Descripción de estados de la máquina de detección de fases.....	60
Tabla 4-18. Estados presentes y futuros de la máquina de detección de fases.....	62
Tabla 4-19. Funciones lógicas de la máquina de detección de fases.....	62
Tabla 4-20. Descripción de señales de la máquina de envío de memoria.....	64
Tabla 4-21. Descripción de estados de la máquina de envío de datos.....	65
Tabla 4-22. Estados presentes y futuros de la máquina de envío de memoria.....	66
Tabla 4-23. Funciones lógicas de la máquina de envío de memoria.....	66

Capítulo 1: INTRODUCCIÓN

1.1 Presentación del problema

Las antenas en arreglo de fase son tecnología que proviene principalmente de investigaciones de las tecnologías de radares militares. Estas presentan ventajas sobre las antenas convencionales, tales como el peso y la más destacable, su alta velocidad de redirección en su haz, que es hacia donde la antena dirige las ondas electromagnéticas. Para poder sacarle todo el provecho a esta última característica, es necesario un sistema de control digital que permita manipular electrónicamente la dirección del haz.

Investigadores del Laboratorio de Electrónica de Alta Frecuencia y Microondas de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México (UNAM) desarrollan antenas en arreglos de fase y requieren sistemas de control digital para sus desarrollos.

Un proyecto en específico del laboratorio mencionado para el que se requiere un sistema de control digital es una antena de arreglos de fase cuyos elementos están conmutados por diodos PIN. Los requerimientos del control digital del sistema descrito y su justificación son enlistados a continuación:

- Graficar el patrón de radiación. Es necesario conocer la orientación de las antenas en arreglos de fase, porque de esta manera se puede comprobarla configuración electrónica de los elementos del arreglo. Por lo tanto, el sistema deberá ser capaz de ilustrar el patrón de radiación de las antenas, para caracterizar la forma e intensidad en que las ondas electromagnéticas son reflejadas.
- Se requiere controlar digitalmente la dirección del haz de la antena para realizar un seguimiento sobre la trayectoria de un satélite.
- Envío de fases de orientación. Con el propósito de controlar la dirección del lóbulo principal de las antenas de forma interactiva y personalizable.
- Crear una interfaz gráfica que ofrezca un fácil acceso a todos los requerimientos anteriores.

Para realizar este trabajo de tesis se tomará de antecedente la tesis de maestría, llevada a cabo por el M.I. Ernesto Hernández Gastaldi, titulada “Diseño de un sistema de control digital escalable y reconfigurable para antenas en arreglo de fase” desarrollada por el Laboratorio de Electrónica de Alta Frecuencia y Microondas, y el Laboratorio de Instrumentación Electrónica de Sistemas Espaciales, ambos laboratorios pertenecientes a la Facultad de Ingeniería de la UNAM.

1.2 Objetivos

1.2.1 Objetivo general

Diseñar e implementar una interfaz y un sistema de control digital para el apuntamiento del lóbulo principal de antenas en arreglo de fase.

1.2.2 Objetivos específicos

-Diseñar una interfaz que permita la comunicación con el sistema de control del arreglo de antenas de fase.

-Desarrollar un módulo en FPGA mediante un lenguaje de descripción de hardware que permita la recepción de tramas digitales, desde una interfaz, con el fin de que esta pueda redireccionar su haz.

-Diseño de un módulo en FPGA mediante un lenguaje de descripción de hardware que le permita al sistema realizar el seguimiento de la trayectoria de un satélite.

-Desarrollo de una interfaz que obtenga del sistema de control digital de antenas los datos necesarios para graficar el patrón de radiación.

Capítulo 2: MARCO TEORÍCO

2.1 Antenas

Las comunicaciones por vía satelital se han vuelto muy importantes en los últimos años debido a su alta velocidad y eficiencia, por esto se requieren antenas de alta ganancia que tengan un haz de alta directividad que beneficie la transmisión y recepción. Existen antenas, como las parabólicas, que concentran la mayor parte de su energía en una sola dirección, a esta característica se le conoce como directividad. Esta describe la distribución de la potencia electromagnética radiada en el espacio circundante. Entonces mientras más concentrado esté el haz, más directiva es la antena.

Otra característica importante es el patrón de radiación, esta es una representación gráfica en 3D o 2D que permite caracterizar la forma e intensidad en que las ondas electromagnéticas son transmitidas o recibidas. El patrón o diagrama de radiación de las antenas directivas cuenta con una zona en dónde la radiación es máxima, a esta se le conoce como lóbulo principal o haz principal. A los lóbulos con menor amplitud, que rodean al principal, se les llama lóbulos secundarios. Mientras que al lóbulo que se encuentra a la espalda del principal se le denomina lóbulo trasero [1]. En la Figura 2-1 se muestra la distribución de los lóbulos de radiación de una antena parabólica convencional. En el centro del disco de la antena se concentra la mayor parte del haz, entonces ahí se encuentra el lóbulo principal.

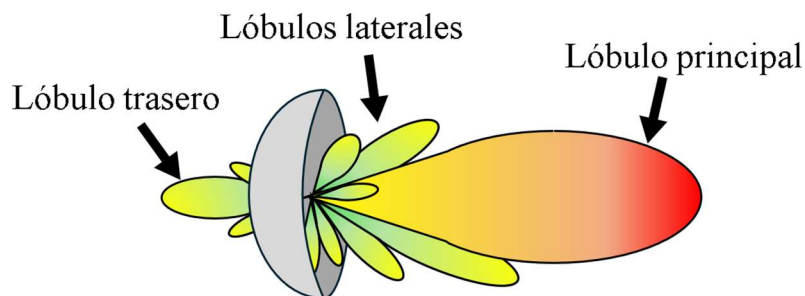


Figura 2-1. Patrón de radiación de una antena parabólica.

En apariencia, las antenas directivas son la respuesta a la necesidad actual de equipos que requieren transmitir una gran cantidad de información. Sin embargo, en el caso de las antenas parabólicas la limitante que tienen es su lenta respuesta ante una reconfiguración en la dirección del haz principal. Cabe recordar que la mayor parte de la energía emitida por estas antenas está concentrada alrededor de la dirección que coincide con el eje del paraboloide, por lo tanto, se debe mover el disco completo para variar la dirección a la que apunta el haz. Este proceso se vuelve dependiente de elementos externos como actuadores o motores que permitan que se mueva la antena de una dirección a otra. Es aquí cuando los arreglos reflectivos reconfigurables toman relevancia. Esta tecnología se basa en un arreglo de antenas que tienen la capacidad de reflejar una onda incidente en un ángulo deseado y se explicarán a detalle en el apartado siguiente.

2.2 Antenas en arreglos de fase

Estas antenas están formadas por grupos de antenas, llamados elementos, que trabajan en conjunto y cuyo propósito es funcionar como una única antena. Como se comentó anteriormente, son capaces de reflejar una onda hacia una dirección deseada. La ventaja principal que ofrecen estos sistemas es que no requieren de un movimiento físico, entonces se puede reorientar el lóbulo principal de una veloz forma, lo que vuelve al sistema más eficaz y rápido.

Es importante abordar como es que se consigue que el haz se reoriente sin que haya un movimiento físico en la antena. Supóngase un elemento que emite ondas electromagnéticas a cierta frecuencia. Como se puede ver en la Figura 2-2 el patrón de radiación es muy amplio.

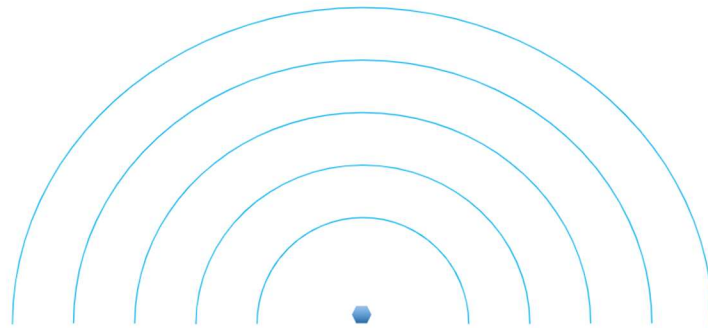


Figura 2-2. Propagación de ondas electromagnéticas emitidas por un elemento. [2]

Si se coloca otro elemento al lado del primero, que transmita a la misma frecuencia y en el mismo tiempo, ocasionará que ambas señales se interfieran mutuamente. Dado el fenómeno de interferencia que se provoca, hay un área en la que se forma una zona de interferencia constructiva, en esta las energías de ambas señales se suma. Esto último se puede ver en el área delimitada por líneas grises en la Figura 2-3.

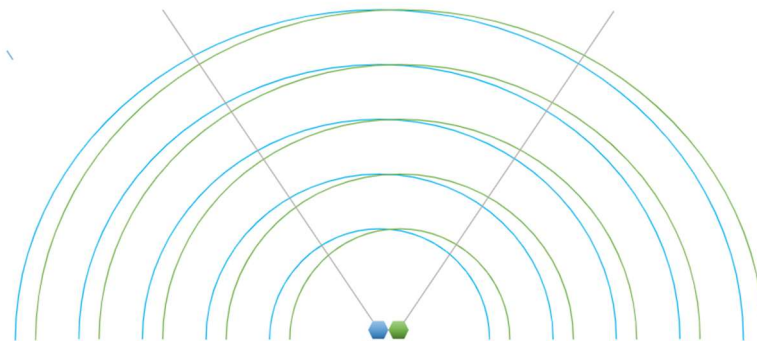


Figura 2-3. Propagación e interferencia de ondas electromagnéticas emitidas por dos elementos. [2]

Si se continúan agregando elementos se formará una zona de interferencia constructiva con mayor energía y amplitud. Por lo tanto, se obtiene un lóbulo principal con una alta directividad formado por los elementos del arreglo.

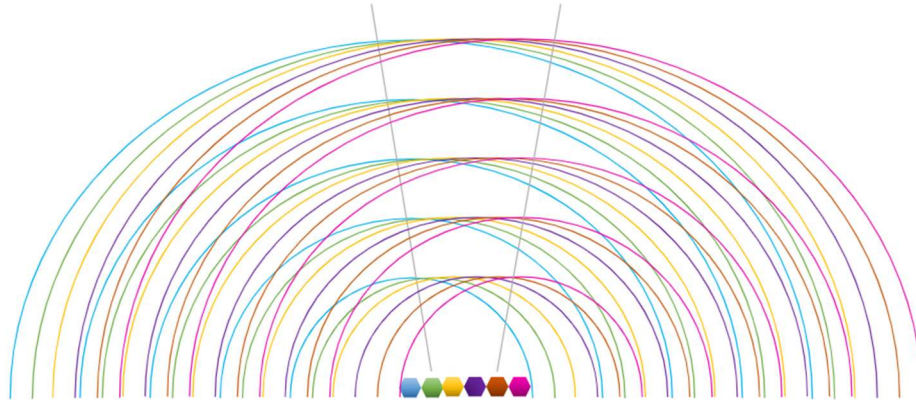


Figura 2-4. Propagación e interferencia de ondas electromagnéticas emitidas por seis elementos.
[2]

Ahora que se sabe cómo se crea un lóbulo principal directivo con base en más de un elemento, se explicará la forma en la que este lóbulo puede redireccionarse sin que exista un movimiento físico en elementos de la antena. Primero, retomando el ejemplo de la Figura 2-3, el haz que se genera por la interferencia de los elementos es perpendicular porque las ondas emitidas por estos elementos tienen la misma fase. Cuando una onda se atrasa o se adelanta respecto a otra, se dice que hay una diferencia de fase, tal y como se puede apreciar en la Figura 2-5 en donde la onda de color azul está atrasada con respecto a la onda roja.

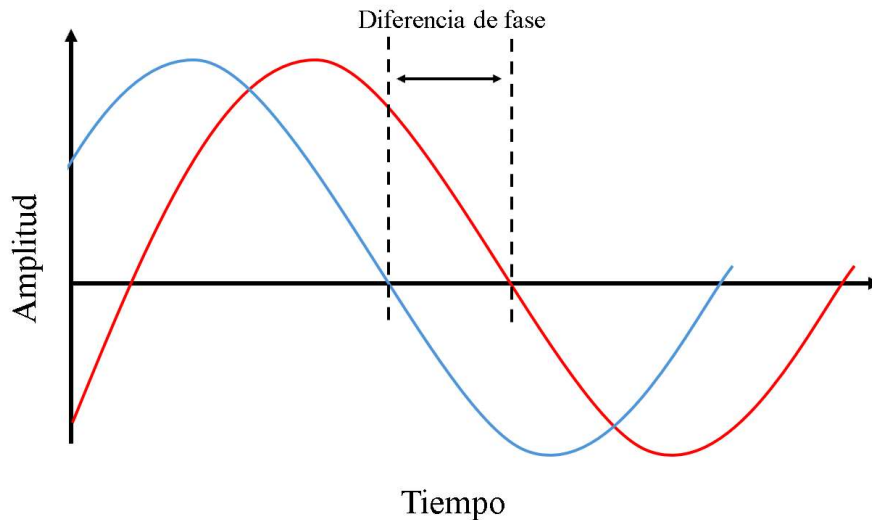


Figura 2-5. Ondas con diferencia de fase.

Si se aplica una diferencia de fase en las ondas propagadas por los elementos de la Figura 2-3, se verá un cambio en la zona de interferencia constructiva lo que ocasiona que haya una inclinación, por lo tanto, un redireccionamiento el lóbulo principal, como se muestra en la Figura 2-6.

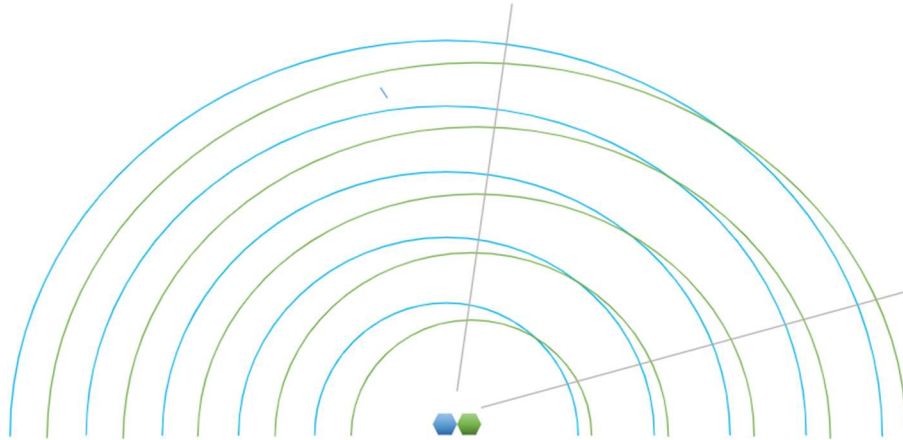


Figura 2-6. Propagación e interferencia de ondas electromagnéticas con desfase, emitidas por dos elementos. [2]

Lo anterior es aplicable para arreglos con un número mayor de elementos, si cada elemento tiene una diferencia de fase, se logrará reorientar el lóbulo principal hacia una dirección deseada. Por lo tanto, si se controla la fase de cada uno de los elementos, se controlará el redireccionamiento del lóbulo principal, ver Figura 2-7.

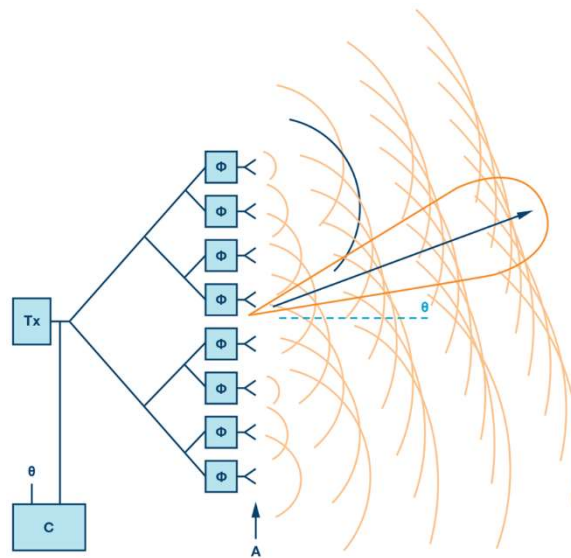


Figura 2-7. Reorientación del haz provocada por la diferencia de fase de los elementos. [3]

Los desplazadores de fase son causantes que la orientación del haz de una antena sea controlada por la diferencia de fase de sus elementos. Los desplazadores con los que trabajan las antenas a las que está dirigido este sistema de control son basados en los diodos PIN. El control de la fase se obtiene mediante la polarización de estos diodos.

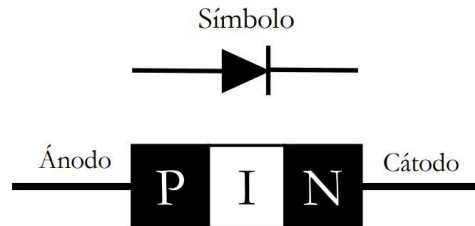


Figura 2-8. Estructura interna de un diodo PIN. [2]

Los diodos PIN se encuentran formados por una difusión tipo P y N, con una capa de silicio en medio llamada región intrínseca (I), ver Figura 2-8. Una característica de estos diodos es que tienen una resistencia interna que varía inversamente a la polarización aplicada. Entonces, si a los diodos PIN se les polariza de forma directa se comportan como una resistencia que disminuirá su valor mientras el mayor sea el voltaje de polarización. Cuando se encuentran en polarización nula prácticamente no hay cargas libres en la región I, esto hace que el diodo no conduzca y se comporte como un capacitor. Sin embargo, usualmente se le polariza de forma inversa para asegurar que las cargas libres sean las menos posibles. [2]

Capítulo 3: SISTEMA DE CONTROL DIGITAL ESCALABLE Y RECONFIGURABLE PARA ANTENAS EN ARREGLOS DE FASE

Este sistema de control digital corresponde al trabajo de tesis de maestría del M.I. Ernesto Gastaldi Hernández realizada bajo los requerimientos del laboratorio de electrónica de alta frecuencia y microondas de la Facultad de Ingeniería de la UNAM. Este sistema se tomará como base para realizar la interfaz de control requerida en el presente trabajo de tesis, entonces, se analizará a detalle su arquitectura, diseño de módulos y funcionamiento con el fin de tener el conocimiento necesario para poder añadir módulos para realizar las nuevas tareas sin afectar el funcionamiento previo del sistema de control digital.

3.1 Requerimientos del sistema

El sistema fue creado con el fin de calcular y controlar digitalmente la fase de cada elemento de una antena en arreglos de fase para que esta reoriente su haz principal con base en los ángulos ingresados θ_0 y φ_0 . Los requerimientos de este sistema fueron:

Escalabilidad y reconfigurabilidad. Como los arreglos son distintos en cuanto a tamaño, número de elementos y disposición de ellos, fue necesario implementar un controlador que funcione en distintas antenas.

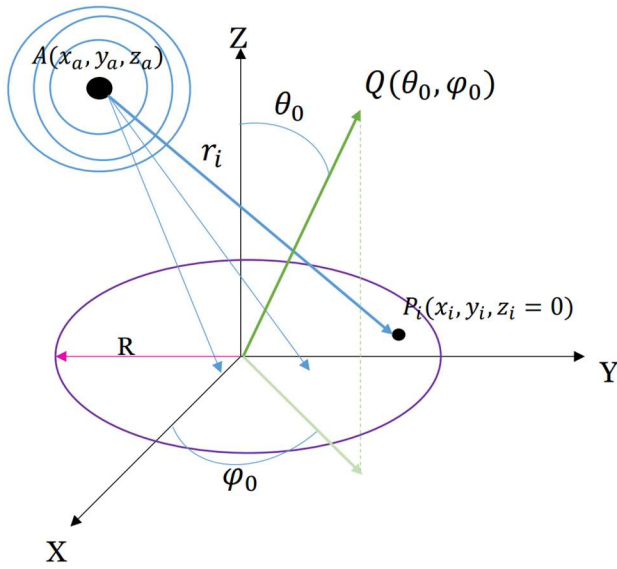
Rapidez de respuesta y cantidad de señales: Dado que las antenas en arreglo de fase ofrecen la ventaja de controlar la dirección del haz reflejado sin un movimiento físico, se implementó un sistema de control que es capaz de aplicar las polarizaciones adecuadas a los diodos PIN y así desplazar la onda a una velocidad aproximada a un microsegundo. También se implementó que sistema de control fuera capaz de proveer todas las señales necesarias sin descuidar la rapidez de respuesta del sistema.

Operaciones matemáticas: Cada uno de los elementos del arreglo debe proporcionar una fase distinta para que la antena pueda reflejar la onda hacia la dirección deseada. Estas fases son calculadas con operaciones matemáticas realizadas el mismo sistema de control.

3.2 Funcionamiento

La tarea principal de este sistema de control es calcular las fases necesarias para orientar el haz reflejado de las antenas en arreglo de fase. A continuación, se muestran los cálculos que realiza el sistema para obtener las fases.

En la Figura 3-1. se muestra un diagrama simplificado de un arreglo de fase circular, en esta se muestra un elemento desplazador, el alimentador (A), el haz incidente y el haz reflejado por el conjunto de elementos (Q), así como nomenclaturas de parámetros y ángulos asociados.



R: radio del arreglo
A: centro de fase del alimentador
P_i: posición del elemento *i*
Q: haz reflejado con ángulos θ_0 y φ_0
r_i: distancia entre el centro del alimentador y el elemento *i*.

$$r_i = \sqrt{(x_a - x_i)^2 + (y_a - y_i)^2 + (z_a - z_i)^2}$$

Figura 3-1. Diagrama de arreglo de fase y alimentador. [2]

Se desea reflejar la onda incidente hacia un vector cuya orientación está dada por los ángulos θ_0 y φ_0 . El ángulo de la onda reflejada con respecto al eje Z es θ_0 . Por otro lado, el ángulo φ_0 es con respecto al eje X. La distribución de fase en cada elemento *i*, siendo x_i e y_i sus coordenadas, para que la onda reflejada se dirija a la dirección deseada, dada por los ángulos θ_0 y φ_0 se define en la (3.1 en donde $k = \frac{2\pi}{\lambda}$, k es el número de onda y λ la longitud de onda.

$$P_{desp_i} = k[-x_i \cos(\varphi_0) \text{sen}(\theta_0) - y_i \text{sen}(\varphi_0) \text{sen}(\theta_0) + r_i] \quad (3.1)$$

Como los desplazadores son dispositivos finitos solo se permiten una cantidad limitada de posiciones (fases). Entonces el resultado de la (3.1) se redondea al valor discreto más cercano de $P_{cod_i}(2\pi/N)$, en donde P_{cod_i} es el código de la posición discreta y N es el número de fases en función del número de bits del elemento (b_a).

Por ejemplo, se tiene un arreglo de $b_a = 2$, las 2^2 posiciones de cada desplazador de fase son 0, 90, 180, 270 [°].

Como se mencionó en la sección 2.2, los desplazadores de fase de las antenas con las que se trabaja este sistema de control están basados en los diodos PIN. Las polarizaciones de estos para las posiciones discretas están definidas por el diseño y la estructura física del arreglo, por lo tanto, para el sistema de control son constantes y se definen en la matriz de funcionamiento de la Tabla 3-1. Como ejemplo en la Figura 3-2 se muestran las polarizaciones y el recorrido para las cuatro fases discretas.

Tabla 3-1. Matriz de funcionamiento de un arreglo hipotético de dos bits. [2]

Fase [°]	P_{cod}	DPIN1	DPIN2	DPIN3	DPIN4	DPIN5	DPIN6	DPIN7	DPIN8
0	0	OFF (0)	OFF (0)	OFF (0)	OFF(0)	ON(1)	ON(1)	ON(1)	ON(1)
90	1	OFF(0)	OFF(0)	ON(1)	ON(1)	ON(1)	ON(1)	OFF(0)	OFF(0)
180	2	ON(1)	ON(1)	OFF (0)	OFF (0)	OFF (0)	OFF(0)	ON(1)	ON(1)
270	3	ON(1)	ON(1)	ON(1)	ON(1)	OFF (0)	OFF (0)	OFF (0)	OFF(0)

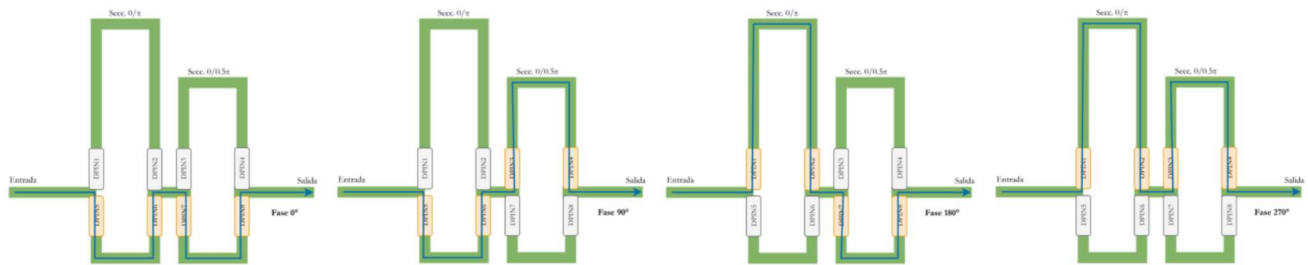


Figura 3-2. Posiciones para el desplazador de fase de dos bits con líneas conmutadas. [2]

3.3 Arquitectura

La arquitectura con la que se realizó este sistema consta de una FPGA maestra y de dispositivos esclavos. La FPGA maestra es la encargada de realizar todos los cálculos matemáticos para cada elemento con los parámetros que recibe desde una interfaz de usuario, después los envía codificados hacia los dispositivos esclavos. Estos cuentan con lógica combinatorial de decodificación interna para cada elemento. Entonces, la tarjeta maestra envía las fases codificadas de forma serial a los dispositivos esclavos y estos las decodifican. En la Figura 3-3 se puede apreciar un diagrama general de la arquitectura.

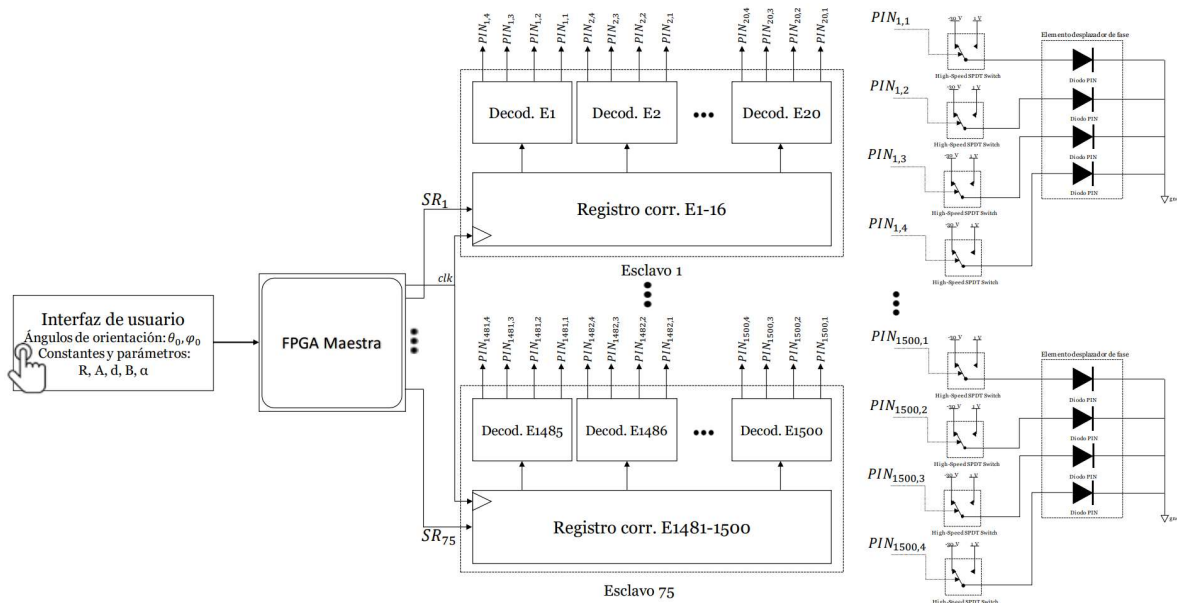


Figura 3-3. Diagrama de arquitectura del sistema de control. [2]

A continuación, se dará una explicación más detallada a cerca de los componentes de la arquitectura del sistema de control:

- Dispositivos esclavos

Están conformados por un registro de corrimiento y lógica que decodifica el valor de la fase correspondiente a cada elemento de la antena. La lógica combinacional de decodificación consta de 20 módulos que convierten las señales provenientes de la FPGA Maestra a señales de control para los diodos de cada desplazador de fase [3]. La Figura 3-4 muestra la estructura interna de uno de los dispositivos que funciona como esclavo.

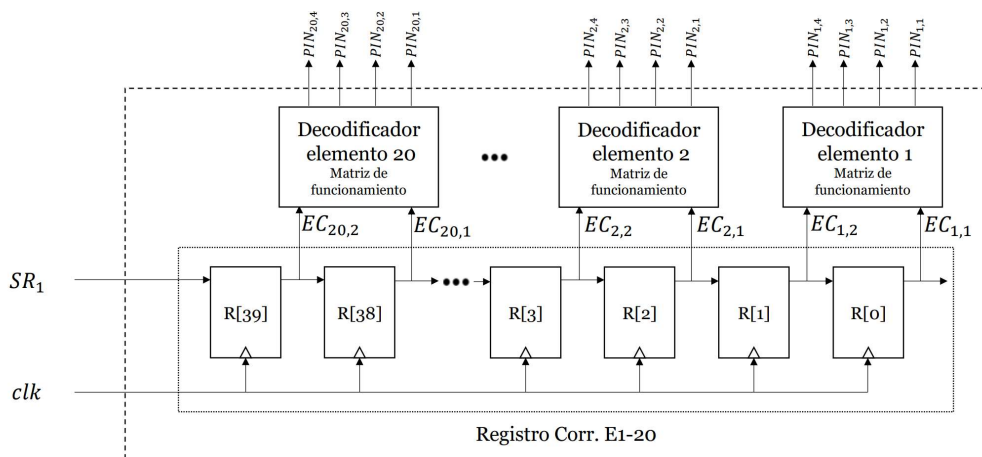


Figura 3-4. Diagrama de la estructura de un dispositivo esclavo. [2]

- FPGA Maestra

Este es el módulo principal, es el encargado de realizar todos los cálculos, recibir y almacenar toda la información correspondiente al arreglo. El funcionamiento de este módulo se divide en dos etapas nombradas, por el autor, como *etapa de configuración* y *etapa de ejecución*.

La *etapa de configuración* consiste en obtener los parámetros del arreglo y con esos datos calcular la posición de cada elemento, así como la distancia que hay entre ellos y el alimentador (r_i). Estas tareas son llevadas a cabo gracias a sistemas de cálculos y almacenamiento, además de una interfaz de usuario por donde se reciben los parámetros del arreglo. Esta etapa se toma como previa al funcionamiento del sistema, por lo que el tiempo de respuesta no es crucial. Los parámetros recibidos de la interfaz son: radio del arreglo (R), distancia vertical (d) y horizontal (B) entre elementos, inclinación del eje Y' (α), posición del alimentador (A) y longitud de onda (λ).

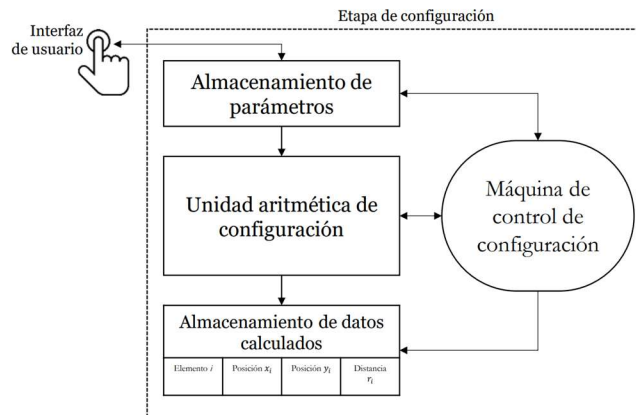


Figura 3-5. Diagrama simplificado de la etapa de configuración del sistema. [2]

La *etapa de ejecución* recibe los ángulos de orientación para orientar el haz, posteriormente se hacen los cálculos de las fases para cada elemento, después se redondean y se codifican para enviarlas por los puertos seriales hacia el dispositivo esclavo correspondiente. En la Figura 3-7 se muestra la estructura completa de la FPGA maestra.

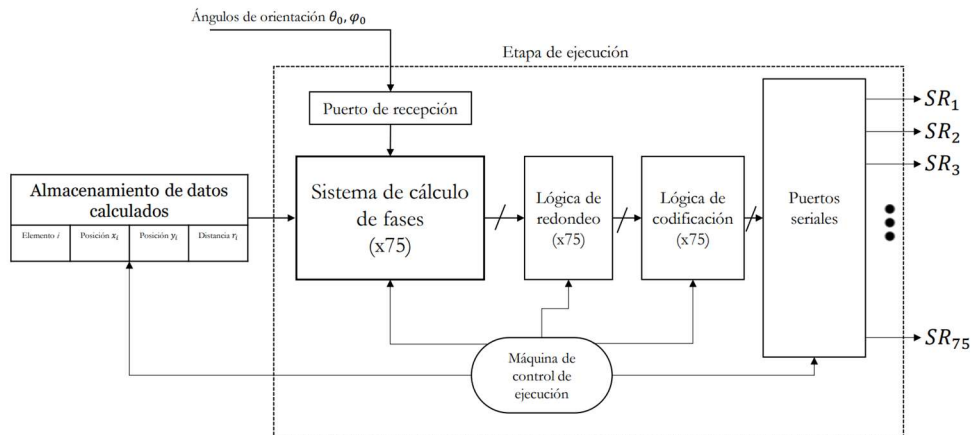


Figura 3-6. Diagrama simplificado de la etapa de ejecución del sistema. [2]

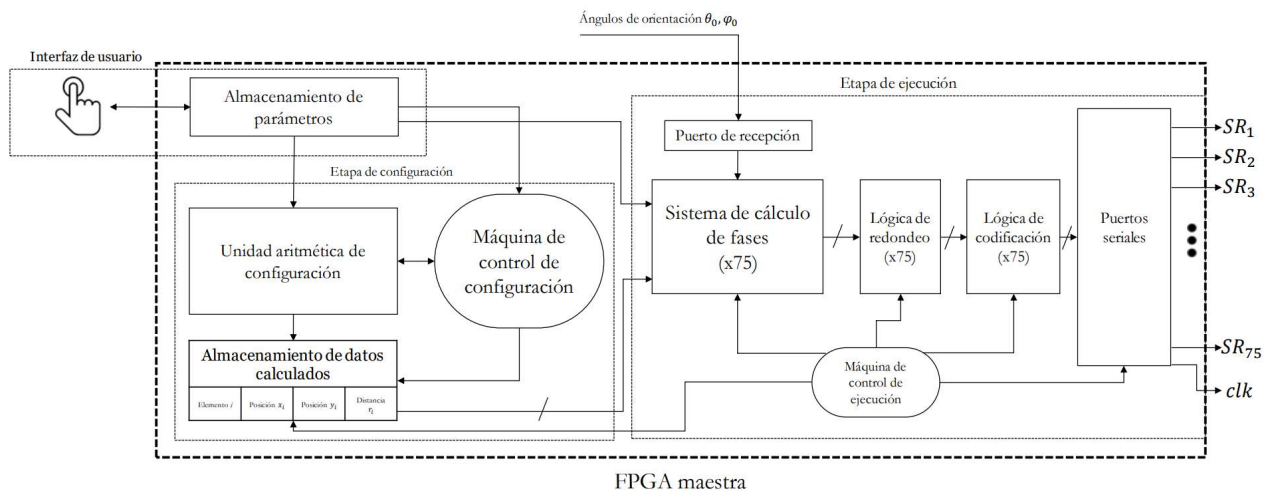


Figura 3-7. Estructura interna de la FPGA maestra. [2]

Ahora que se ha descrito de forma general la interfaz de usuario, FPGA maestra y dispositivos esclavos, se profundizará a detalle en cada módulo creado.

3.4 Análisis de módulos

Con el propósito de entender el funcionamiento del sistema de control digital se revisará los módulos más relevantes que componen a la FPGA Maestra. También, se analizarán sus diagramas de bloques en donde se muestren las entradas y salidas para conocer cada una y poder diseñar módulos propios que sean compatibles con todo el sistema de control.

- Etapa de configuración

Está conformada por máquinas que le permiten recibir los parámetros del arreglo, calcular las posiciones de los elementos y guardar datos que servirán para calcular la fase requerida para orientar el haz de antena. Se enlistarán los bloques principales y la función que realizan:

-Memoria de elementos: Se encarga de guardar los índices para cada elemento, así como su distancia entre ellos y el alimentador dividida entre la longitud de onda.

-Máquina de control de configuración: Su función es coordinar y sincronizar todos los módulos de esta etapa con el fin de almacenar los datos referentes al arreglo dentro de la memoria de elementos.

La Figura 3-8 es un diagrama de bloques de la etapa de configuración en donde se muestran todos sus módulos interconectados, con sus respectivas entradas y salidas.

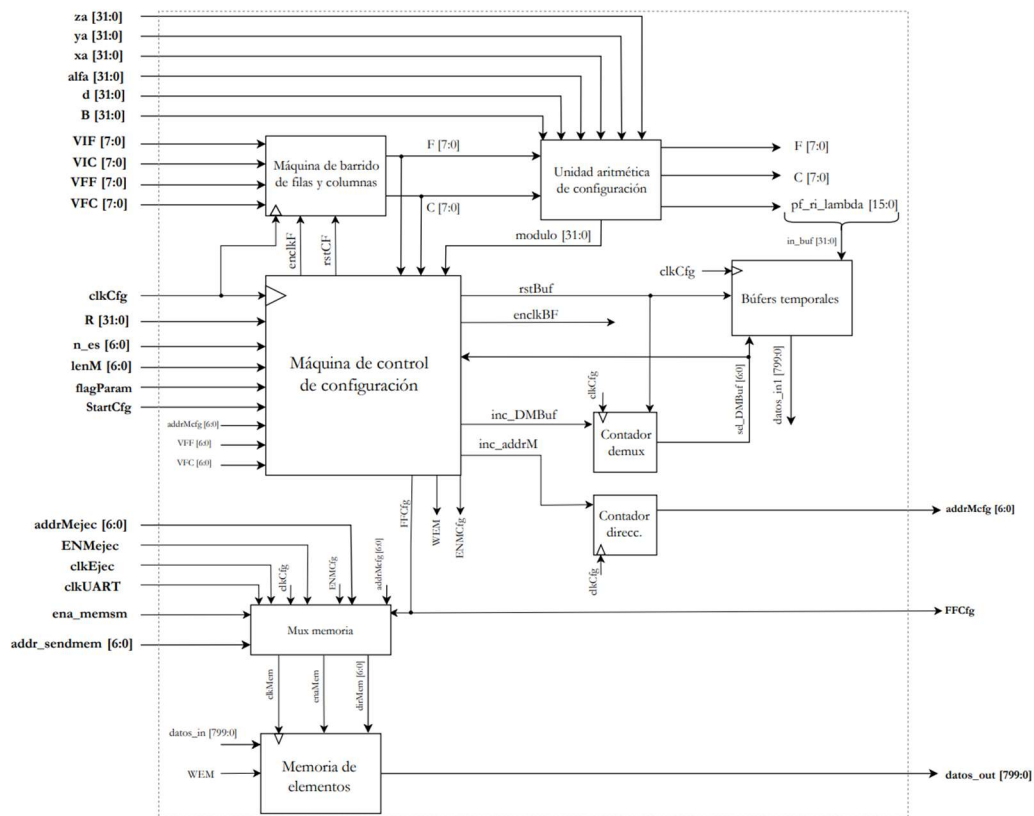


Figura 3-8. Interconexión de módulos para la etapa de configuración. [2]

- Etapa de ejecución

Con los datos que recibe de la etapa anterior, se calcula la fase para cada elemento y posteriormente se envía hacia los dispositivos esclavos. Los principales módulos son:

-Unidad de cálculo de fases: Recibe los datos del elemento actual y los ángulos de orientación, calcula la fase a la que cada elemento se desplazará. Su salida es el coeficiente de la fase.

-Núcleo de ejecución: Consta de tres etapas, en las que se encarga de obtener la equivalencia de la fase en caso de ser negativa, dado el coeficiente obtenido en la *Unidad de cálculo de fases*. También realiza la lógica de redondeo necesaria por la cantidad finita de fases alcanzables por los elementos y codifica la fase a un valor discreto, dada la matriz de funcionamiento. Se tiene un *Núcleo de ejecución* por cada tres *Puertos seriales*.

-Puertos seriales: Encargados del envío de las fases codificadas hacia los dispositivos esclavos.

-Puerto de recepción de ángulos de orientación: Sirve para recibir por medio del protocolo SPI los ángulos de orientación y estos vienen de un sistema externo.

-Máquina de control de ejecución: Es la encargada de coordinar los cálculos, su redondeo y codificación, y el envío de las fases por los puertos seriales. También controla la lectura de la memoria de elementos perteneciente a la etapa de configuración.

En la Figura 3-9 se pueden observar los módulos de la etapa de ejecución interconectados y resaltados en colores las distintas tareas que realiza esta etapa.

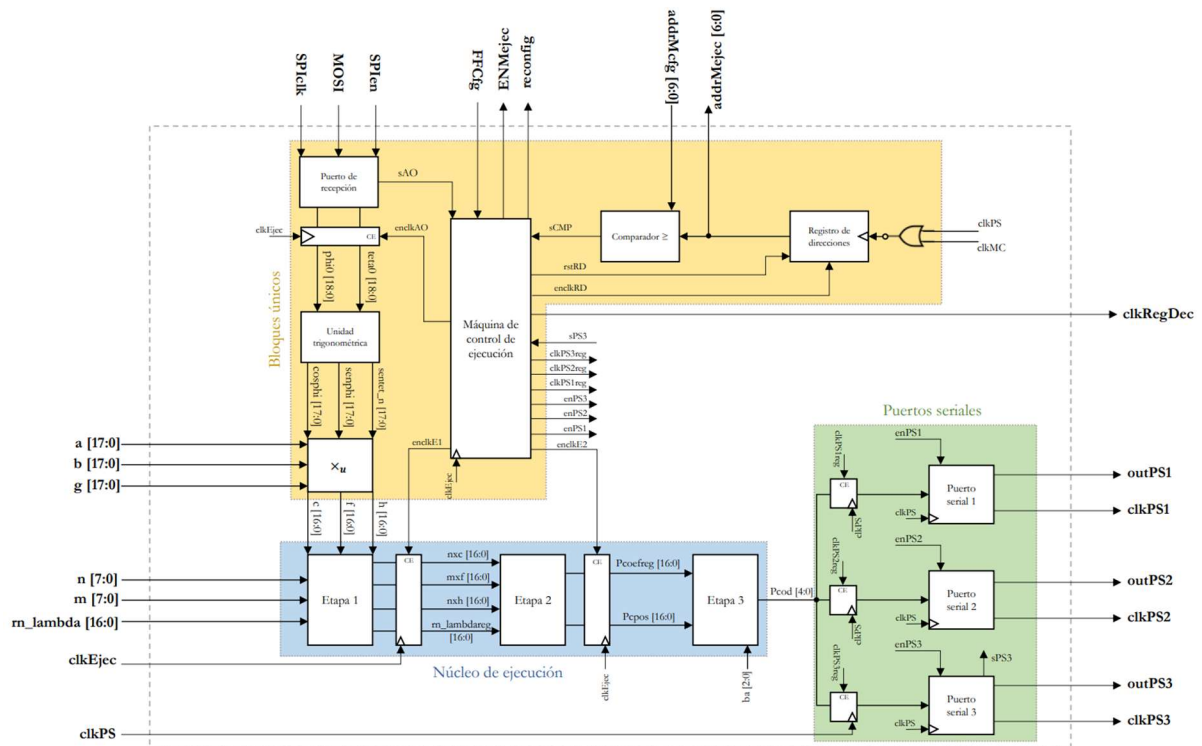


Figura 3-9. Interconexión de módulos para la etapa de ejecución. [2]

Capítulo 4: DISEÑO

4.1 Diseño conceptual

Tomando como base el sistema de control digital descrito en el Capítulo 3 de este trabajo de tesis y resaltando la importancia de su capacidad de orientar antenas en arreglos de fase, se necesitan aplicaciones que aprovechen esta característica del sistema. El laboratorio de electrónica de alta frecuencia y microondas de la Facultad de Ingeniería de la UNAM requiere tareas concretas que sean ejecutadas por el sistema de control con el propósito de verificar su correcto funcionamiento y hagan uso de la reorientación del haz principal de las antenas. Las tareas requeridas para este sistema de control por el laboratorio de electrónica de alta frecuencia y microondas son las siguientes:

- Desarrollar una interfaz de usuario en una computadora que permita tener un fácil acceso a las demás tareas y que esté conectada al sistema de control para enviar y recibir datos.
- Controlar la reorientación del haz principal de las antenas en arreglos de fase para realizar un seguimiento sobre un satélite de órbita baja.
- Enviar, desde la interfaz, fases de orientación para cada elemento de la antena.
- Graficar el patrón de radiación de la antena con las fases calculadas por el sistema.

Por lo tanto, se debe crear una interfaz gráfica que tenga la capacidad de enviar, procesar y recibir datos, así como tener un elemento gráfico para poder visualizar el patrón de radiación que se dibuje con las fases obtenidas del sistema de control. A esto hay que sumarle el diseño y desarrollo de los módulos en descripción de hardware que permitan llevar a cabo las tareas requeridas, estos módulos deberán ser compatibles con todo el sistema de control ya creado anteriormente, por lo cual, se deberá ser cuidadoso en la arquitectura de estos.

A continuación, se describirán los requerimientos de diseño del sistema con más detalle.

4.1.1 Interfaz de usuario

El propósito principal de esta interfaz es que al usuario u operador se le facilite el uso del sistema de control digital, entonces deberá ser intuitiva y sencilla de usar para quién desee realizar alguna tarea solicitada para este trabajo.

Además, la interfaz podrá enviar y recibir datos del sistema de control digital, entonces se tiene que crear un apartado en donde se carguen los datos y se procesen para posteriormente enviarlos.

Otra de las capacidades solicitadas es que se realice una gráfica 3D del patrón de radiación asociado a la antena que se esté controlando. Entonces, la interfaz deberá contar con un elemento gráfico que permita dibujar de forma precisa este diagrama. En la Figura 4-1 se puede ver un ejemplo de lo que es el patrón de radiación de una antena en arreglos de fase.

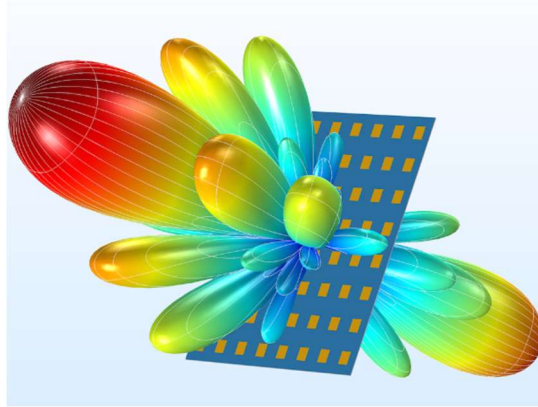


Figura 4-1. Diagrama de radiación de un sintetizador de antenas de arreglo de fase de parche de microcinta acopladas. [4]

4.1.2 Propuestas de diseño para seguimiento de satélites

Se requiere que la antena reoriente su patrón de radiación de manera continua para realizar el seguimiento sobre la trayectoria de un satélite, para esto, el primer paso será obtener su posición actual. Existen diferentes tipos de software y páginas en internet que otorgan datos y posición, se podría utilizar uno de estos medios para determinar la posición de algún satélite. Se ha pensado en dos propuestas para realizar esta tarea:

A) Generar ángulos continuos

Consiste en construir un módulo que genere los valores de manera continua para reorientar la antena durante un tiempo y área definidos por el usuario en la interfaz. Estos valores se generan con base en datos recibidos desde la interfaz a cerca de la posición y tiempo del siguiente paso del satélite sobre la estación terrena. Este módulo debe tener la capacidad de recibir la posición inicial y final, dada en ángulos de orientación, así como hacer el procedimiento matemático necesario para generar el recorrido que deberá hacer el haz principal de la antena para reorientarse continuamente y, por último, enviar esos valores de ángulos hacia los módulos existentes en el sistema de control.

Posibles ventajas:

- Uso del FPGA para hacer cálculos, lo que volvería al sistema más rápido.
- Se recibirían solo los ángulos iniciales y finales, entonces el puerto por donde entran los datos no se usaría demasiado.

Posibles desventajas:

- Como se trata de una predicción del movimiento del satélite, se podría tener un error respecto a la posición real del mismo.

B) Recibir ángulos en tiempo real

Es posible conectarse a un sitio web dedicado al seguimiento y predicción de satélites. Dada la API (interfaz de programación de aplicaciones) de algunas páginas en internet se puede acceder a estos datos. Al lograr una comunicación se podría obtener la ubicación en tiempo real de cualquier satélite, y que esta sea enviada hacia el sistema de control digital para que la antena reoriente su haz principal hacia la dirección deseada. Esta ubicación debe consultarse de manera periódica, a modo que, se asegure un correcto seguimiento del satélite por la antena.

Posibles ventajas:

- Se obtendrá la posición exacta del satélite, lo que resulta en un seguimiento muy preciso.
- Al contar con un acceso a la API de la página de internet, se podrían realizar más aplicaciones en un trabajo a futuro.

Posibles desventajas:

- Se debe esperar la respuesta del servidor y posteriormente hacer el envío de los datos hacia el FPGA, esto generará un retardo.
- El sistema se vuelve dependiente de una conexión a internet

Haciendo un análisis detallado junto con el titular del grupo de electrónica de alta frecuencia y microondas de la Facultad de Ingeniería de la UNAM, el Dr. Oleksandr Martynyuk, y determinar que tratándose de un sistema para realizar pruebas la velocidad de respuesta es indiferente y, al contrario, la precisión es indispensable para un correcto seguimiento de satélites. Se decidió que la alternativa de recibir los ángulos en tiempo real es la indicada para realizar este trabajo.

4.1.3 Propuesta de diseño para enviar fases para cada elemento desde la interfaz.

Una de las tareas del sistema de control digital es calcular fases y después enviarlas a su elemento correspondiente, logrando así que el haz principal de la antena se reoriente. Esto puede ser visto como dos etapas, la primera *calcula* las fases y la segunda las *envía*. Lo que se desea hacer en este trabajo es instrumentar una alternativa a la primera etapa y que las fases enviadas hacia los elementos sean las dadas por el usuario desde la interfaz.

Entonces, se piensa construir un apartado en la interfaz en donde se puedan cargar las fases y después se envíen al sistema de control digital. Este no tiene la capacidad de recibir las fases de forma externa, por lo tanto, se deberá diseñar un módulo que las reciba.

4.1.4 Propuesta de diseño para graficar el patrón de radiación de la antena

Con el propósito de verificar la orientación de las antenas, se requiere graficar el diagrama de radiación asociado. Para realizar este gráfico, se necesitará recibir del sistema de control la fase que se calculó para cada elemento de la antena, su posición y distancia del alimentador. Posteriormente se empleará la ecuación (4.1), dada por el titular del grupo de electrónica de alta frecuencia y microondas, con la que se graficará el patrón de radiación. Esta ecuación se puede ver como un algoritmo que servirá para corroborar la energía emitida por las fases discretas de todos los elementos el arreglo, considerando su posición y distancia del alimentador.

$$F(\theta, \varphi) = \sum_{i=1}^N e^{j\varphi\rho_{ei}} e^{jkx_i \sin(\theta)\cos(\varphi)} e^{jky_i \sin(\theta)\sin(\varphi)} \quad (4.1)$$

Entonces será necesario construir un módulo en descripción de hardware que envíe las fases calculadas por el sistema de control digital, la posición y distancia del alimentador de cada elemento hacia la interfaz. También se deberá desarrollar un algoritmo que le permita a la interfaz de usuario recibir estos datos y operar con cada uno de ellos para obtener el patrón de radiación. Es necesario aclarar en este punto que la (4.1) no se ejecutará dentro del sistema de control por límites de tiempo y su implementación en hardware se llevará a cabo en un trabajo futuro.

4.2 Diseño a nivel de sistema

En esta sección se definirá la arquitectura del sistema y se le hará una descomposición en subsistemas y componentes. Los requerimientos planteados anteriormente serán analizados y definidos en términos de electrónica digital y programación. Se dará una especificación funcional de cada uno de los subsistemas generados y se presentarán diseños a bloques del sistema. El diseño tomará como prioridad los módulos ya existentes, entonces se evitará alterarlos en mayor medida.

4.2.1 Interfaz de usuario

Esta es el enlace del operador con el sistema de control digital y sus tareas. Lo que sale de la interfaz entra al sistema y lo que sale del sistema entra a la interfaz. De esta se identificarán cuáles serán las señales de entrada y salida, para esto será necesario revisar el envío de las fases desde la interfaz, envío de los ángulos de orientación para seguimiento de satélites y graficar el patrón de radiación.

En esta sección únicamente se analizarán las señales relacionadas a la interfaz de usuario y su conexión con el sistema de control. Se hará una descripción más detallada de cada módulo en su sección correspondiente. Además, se dará una explicación funcional de cada etapa de la interfaz.

Analizando el apartado 4.1 se obtiene que la comunicación se hará directamente con la FPGA Maestra, se necesitará acceder a sus dos etapas (configuración y ejecución). De la sección 3.4, se obtuvieron las señales que se usarán del sistema de control para realizar las tareas requeridas.

Para enviar las fases desde la interfaz al sistema, se necesita acceder a la etapa de ejecución de la FPGA Maestra. En esta etapa es donde se envían las fases codificadas que calculó el sistema a los dispositivos esclavos, el módulo encargado de realizar esta tarea se muestra resaltado en el diagrama de bloques de la FPGA Maestra de la Figura 4-2. Se planea acceder a este módulo para transmitir a los dispositivos esclavos las fases enviadas desde la interfaz.

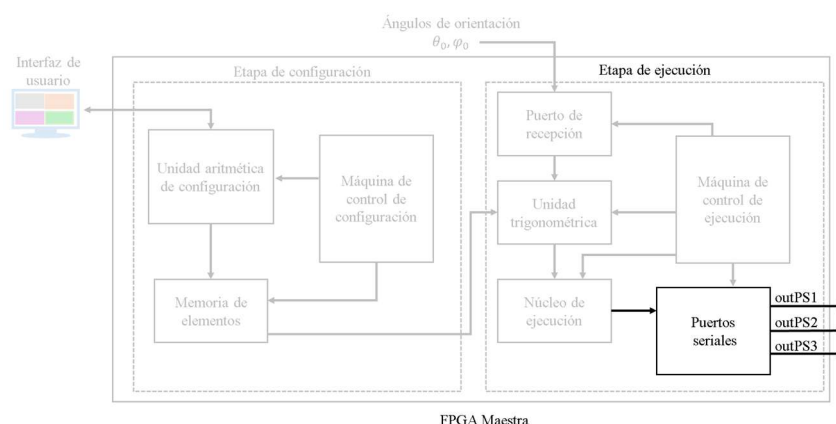


Figura 4-2. Puertos seriales resaltados en la estructura del FPGA.

Ignorando por el momento los demás módulos y señales, se tendría una conexión para el envío de fases desde la interfaz a la etapa de ejecución como se muestra en la Figura 4-3.

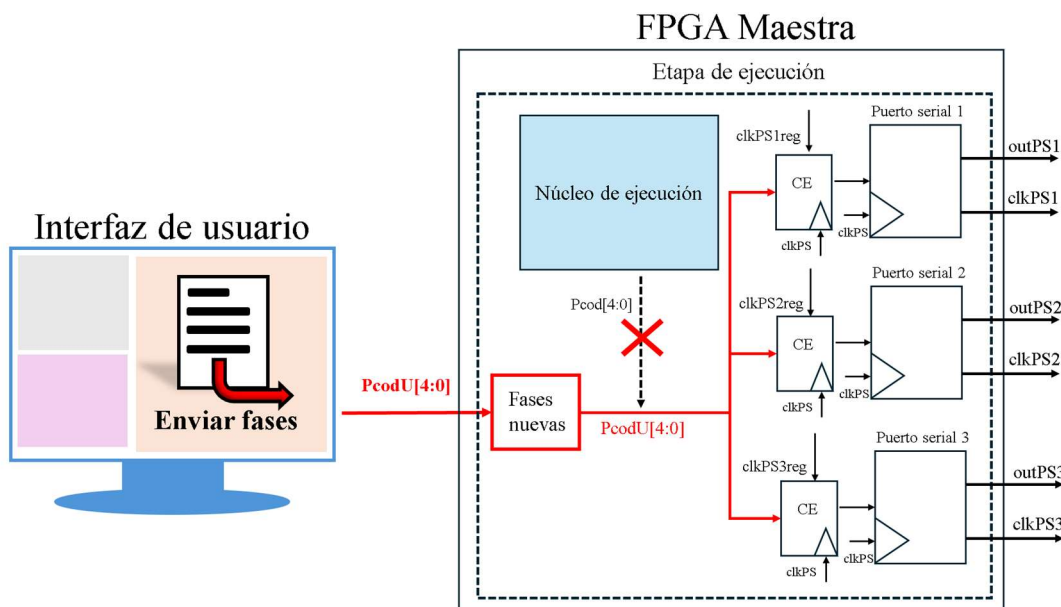


Figura 4-3. Diagrama de bloques para el envío de fases desde la interfaz.

En la Figura 4-3 se puede notar que la señal de 5 bits que sale del *Núcleo de ejecución* llamada *Pcod* ha sido sustituida por una señal del mismo tamaño pero que viene de la interfaz, esta contiene las fases que el usuario ha ingresado. Hay que hacer énfasis en que esta modificación del sistema de

control digital en esa última etapa únicamente es para el envío de fases desde la interfaz. Se hablará a profundidad de esta modificación más adelante.

Para el seguimiento de satélites se planean enviar los ángulos de orientación θ_0 y φ_0 desde la interfaz hacia el sistema de control. Este ya cuenta con un puerto de recepción de ángulos, ver Figura 4-4, y cada vez que un nuevo par es recibido, el sistema calcula las fases para reorientar el haz. Entonces, se necesita controlar el envío de nuevos ángulos cada cierto tiempo.

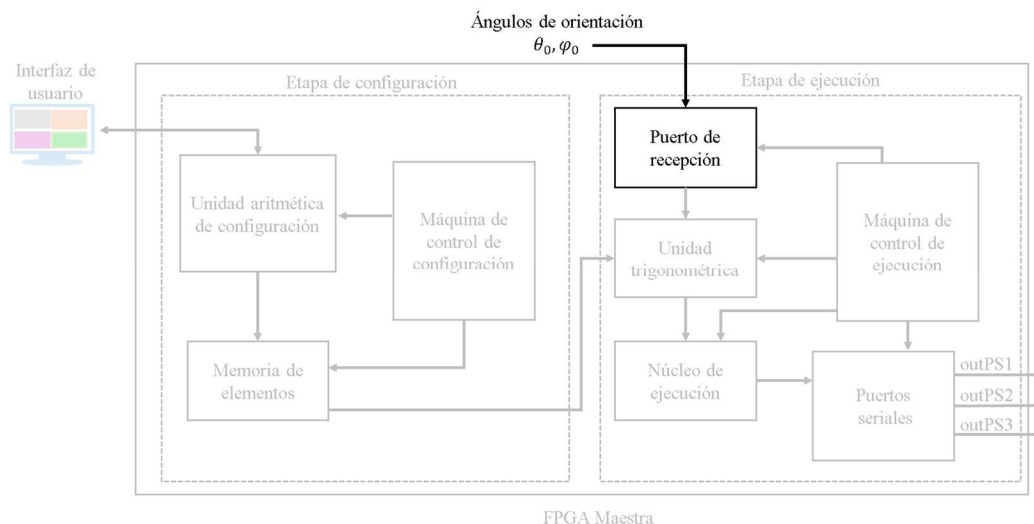


Figura 4-4. Puerto de recepción resaltado en la FPGA Maestra.

La Figura 4-5 ignora los demás módulos de la FPGA y muestra la conexión entre el puerto de recepción y la interfaz de usuario. En la sección 4.2.2 dedicada al diseño a nivel sistema del seguimiento de satélites se profundizará más en los cambios que se le harán al puerto de recepción de ángulos para hacerlo compatible con la interfaz de usuario.

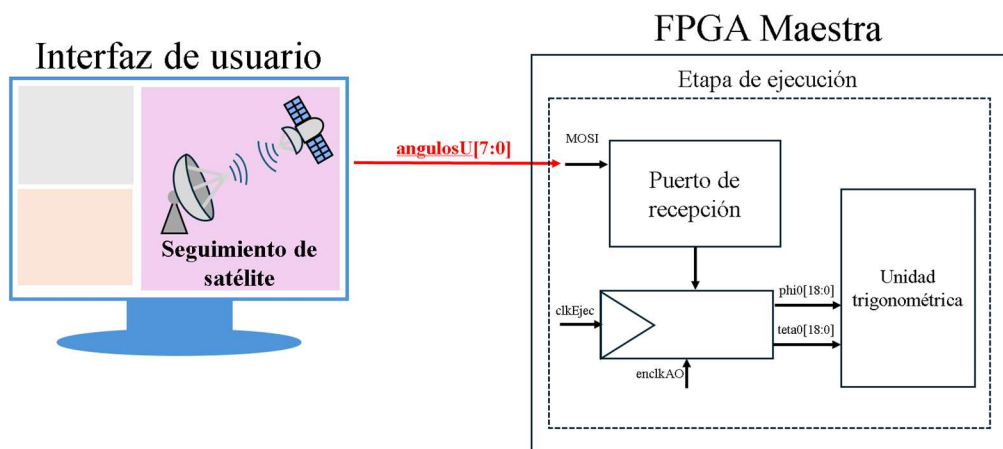


Figura 4-5. Diagrama de bloques para el envío de los ángulos desde la interfaz.

Por último, se analizará la conexión entre la interfaz y la FPGA Maestra para graficar el patrón de radiación. Para poder realizar esta tarea se necesitarán las fases para cada elemento que el sistema de control calcula, estas se obtienen en la salida del *Núcleo de ejecución* resaltado en la Figura 4-6. Entonces, las fases se deben enviar de la FPPA a la interfaz.

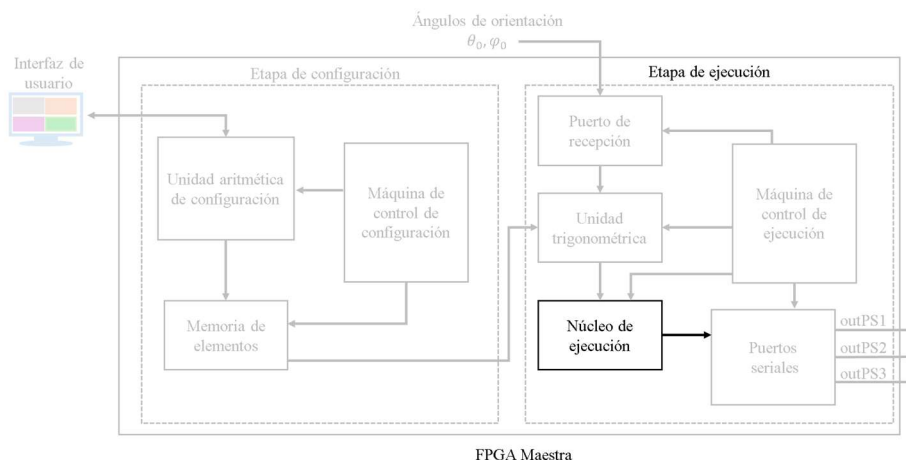


Figura 4-6. Núcleo de ejecución resaltado en la estructura interna del FPGA.

Si se contemplan enviar las fases ya codificadas el número de bits máximo para cada fase es de cinco, esto involucra que en la interfaz se diseñe una lógica de decodificación. En la Figura 4-7 se muestra como la salida del *Núcleo de ejecución* es la señal que debe enviarse a la interfaz, esto sin modificar el destino final de las fases, que son los puertos seriales. En la sección 4.2.3 se hará un análisis detallado de cómo se hará este envío.

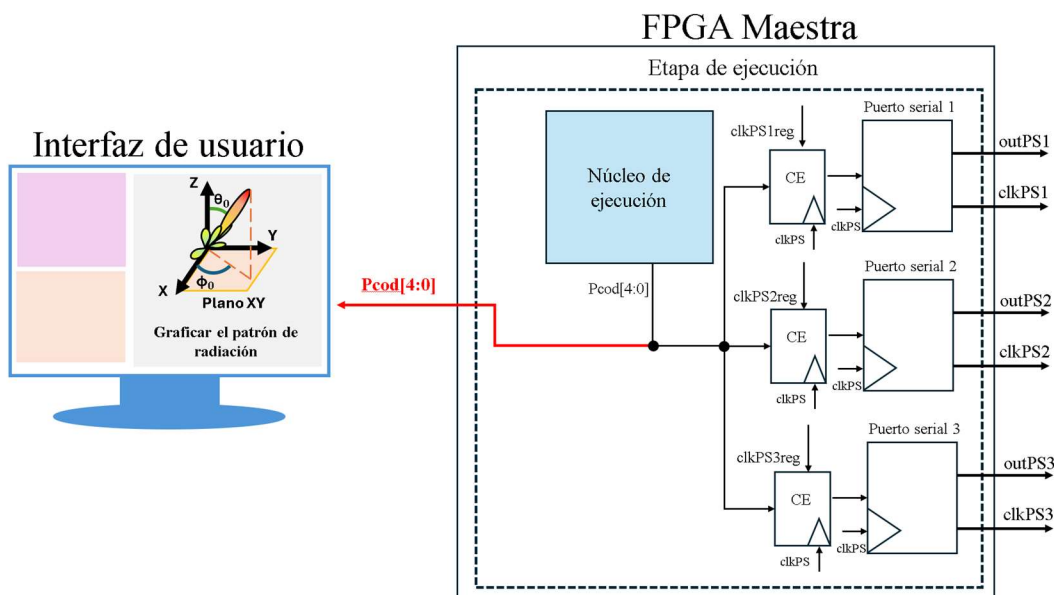


Figura 4-7. Diagrama de bloques para el envío de las fases a la interfaz.

En resumen, se estarán enviando y recibiendo datos de la FPGA Maestra continuamente, entonces la interfaz se deberá construir de forma que respete los datos y tiempos de ejecución para cada tarea. Esto se conseguirá monitoreando las banderas de cada etapa, con el fin de verificar su inicio y final.

En la Figura 4-8 se muestra un diagrama con las señales que la interfaz tiene que enviar y recibir, en este no se han contemplado las banderas de cada ejecución. En las secciones posteriores se analizará a detalle cada señal y bandera que se debe tomar en cuenta para realizar todas las tareas requeridas.

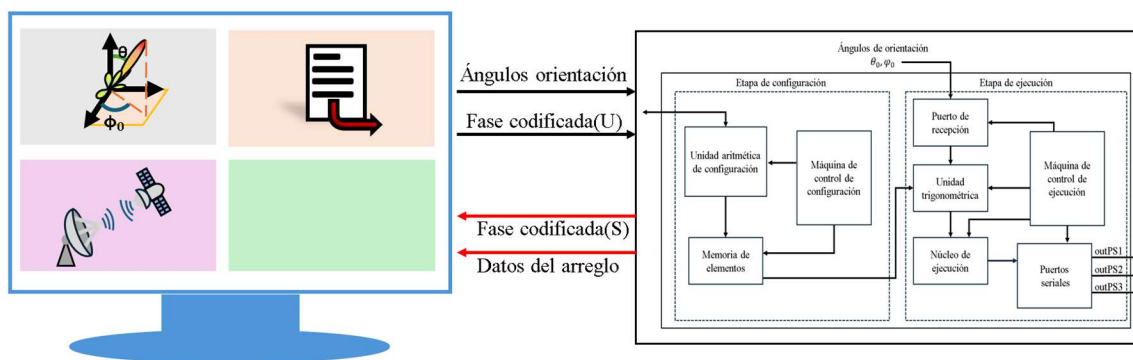


Figura 4-8. Diagrama de bloques simplificado de la interfaz con la FPGA maestra.

Ahora se describirá el funcionamiento de la interfaz para poder llevar a cabo todas las tareas ya mencionadas. Comenzando con el seguimiento de satélites, se tiene que obtener la hora y los ángulos de azimut y elevación del siguiente paso del satélite sobre nuestra ubicación. De esta manera se sabrá en cuanto tiempo y en que posición se hará el seguimiento. De entre muchas opciones se han elegido usar herramientas de MATLAB que contienen algoritmos de diseño, simulación y pruebas con las que se puede determinar datos de un satélite, como la trayectoria vista en la Figura 4-9, usando elementos de dos líneas (por sus siglas en inglés, TLE). Estos son un conjunto de datos que contienen parámetros orbitales como la posición del satélite y su velocidad. Esto abre la puerta para que se pueda realizar el seguimiento de tantos satélites como elementos TLE se proporcionen en la programación.

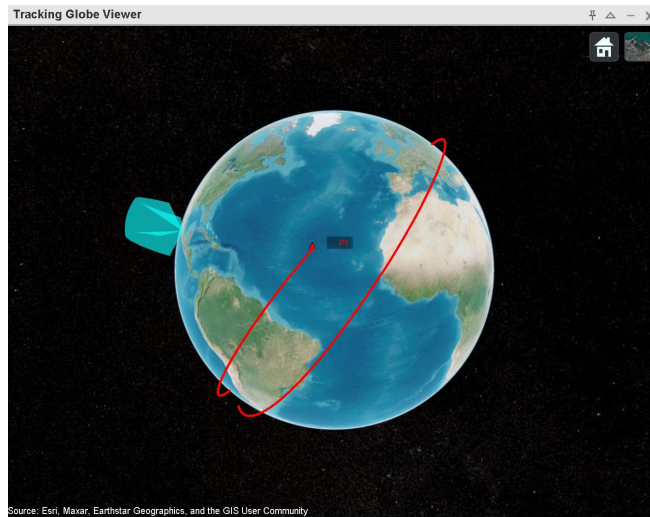


Figura 4-9. Órbita de un satélite calculada por MATLAB.

Para el envío de fases desde la interfaz se considera práctico que el usuario tenga un archivo de Excel o de texto que contenga los datos. Por lo tanto, se desarrollará un botón a través del cual se acceda a los archivos de la PC y el operador deberá seleccionar el que contenga las fases. La interfaz se encargará de redondear y codificar todos los datos para posteriormente enviarlos al sistema de control. Así mismo se mostrará una vista previa del archivo elegido con el fin de corroborar que es el correcto.

Por último, para graficar el patrón de radiación emitido por la antena, la interfaz debe almacenar todos los datos recibidos desde el sistema de control. Posteriormente se calculará, por medio de la (4.1), la energía emitida por cada elemento de la antena con respecto a cada ángulo de orientación posible. Existen muchos lenguajes de programación en los que se puede programar la (4.1), pero dados los elementos gráficos que ofrece y la facilidad de usar el puerto serial de la PC, MATLAB es la mejor opción para realizar esta tarea.

Dado que para dos de las tres tareas que requiere hacer la interfaz de usuario es la mejor opción, se ha decidido ocupar MATLAB para construir esta interfaz.

4.2.2 Seguimiento de satélites

Como se observó anteriormente, ver Figura 4-4, el sistema de control digital cuenta con un puerto de recepción, por el cual recibe los ángulos de orientación. Este puerto está compuesto por dos máquinas, una que sirve como receptor por medio del protocolo SPI y otra que controla el llenado de los registros en donde se almacenan los datos, estas máquinas se pueden observar en la Figura 4-10.

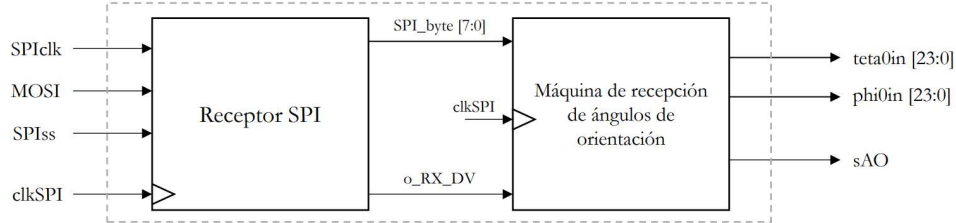


Figura 4-10. Diagrama de bloques del puerto de recepción de ángulos de orientación. [2]

Al módulo que controla el llenado de los registros en dónde se almacenan los datos se le denominó *Máquina de recepción de ángulos de orientación*. Para cada uno de los dos ángulos (θ_0, φ_0) se cuenta con un registro de 24 bits (8 bits para la parte entera y 16 para la fraccionaria), estos ángulos se reciben en radianes. Así mismo, cuenta con una señal de salida que le indica a la *Máquina de control de ejecución* que un nuevo par de ángulos ha sido recibido y esta máquina ejecuta un nuevo cálculo de fases.

Por lo tanto, se requiere que cada nuevo par de ángulos de orientación que se reciban de la interfaz mantenga el formato numérico con el que trabaja el sistema. Para esto primero se enviarán los 8 bits correspondientes a la parte entera del ángulo de orientación y posteriormente los 16 de la parte fraccionaria, tal y como se muestra a continuación.

$$207^\circ = 3.6128 \text{ rad}$$

P. Fracc. Baja
3.6128
P. entera P. Fracc. Alta

Basándose en esto, se diseñará un nuevo puerto de recepción de ángulos que funcione con UART. De modo que se puedan recibir desde la interfaz de usuario.

4.2.3 Envío de fases para cada elemento

El envío de las fases que se generan en el sistema es controlado por la *Máquina de control de ejecución*, esta máquina, entre otras cosas, monitorea la bandera que indica cuando una nueva fase ha sido calculada en el *Núcleo de ejecución*, posteriormente habilita los *Puertos seriales* para permitir el envío de dicha fase, estos módulos se observan resaltados en la Figura 4-11.

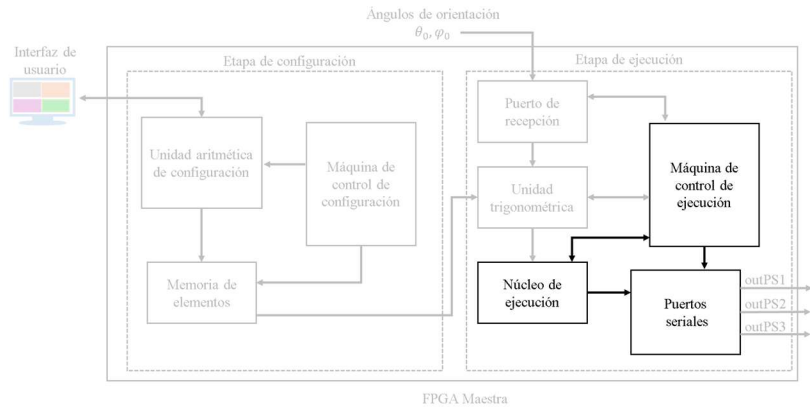


Figura 4-11. Módulos involucrados en el envío de fases.

Haciendo un análisis más a fondo de los módulos de la Figura 4-11 y de las señales involucradas únicamente en enviar las fases por los puertos seriales se obtendría algo similar al diagrama de la Figura 4-12, en este se observa la *bandera* que le indica a la *Máquina de control de ejecución* que la fase está lista, la *fase* calculada en el *Núcleo de ejecución* y las señales que habilitan a los *Puertos seriales*.

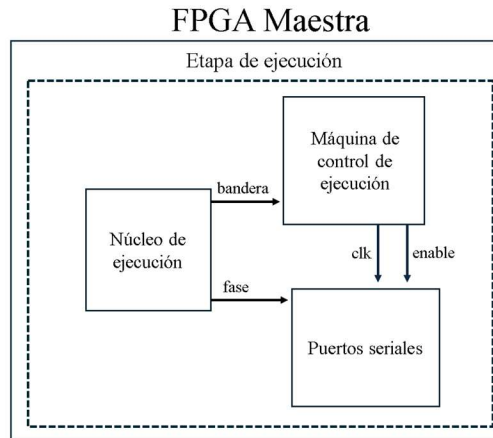


Figura 4-12. Diagrama simplificado de señales para los Puertos seriales.

Considerando que para esta tarea en concreto las fases no se calculan, se reciben, entonces la bandera proveniente del *Núcleo de ejecución* que le indica a la *Máquina de control de ejecución* que la fase esta lista no se levantará, anulando la activación de los *Puertos seriales*. Teniendo en cuenta todo esto se debe crear una señal que reemplace la bandera original y le permita al sistema realizar la función de envío de fases. Si se hace esto, se tendrían dos señales similares para una sola entrada. Para resolver este problema se concluyó en usar un multiplexor, de esta manera por medio de un selector se elegirá cual será la bandera que le indique a la *Máquina de control de ejecución* que la fase está lista para enviarse. Así mismo, con un multiplexor se puede conmutar entre las fases que se calculan en el sistema y las fases recibidas. En la Figura 4-13 se muestra la adición del módulo

que recibe las fases de la interfaz y los multiplexores necesarios para conmutar entre el funcionamiento original del sistema y el nuevo.

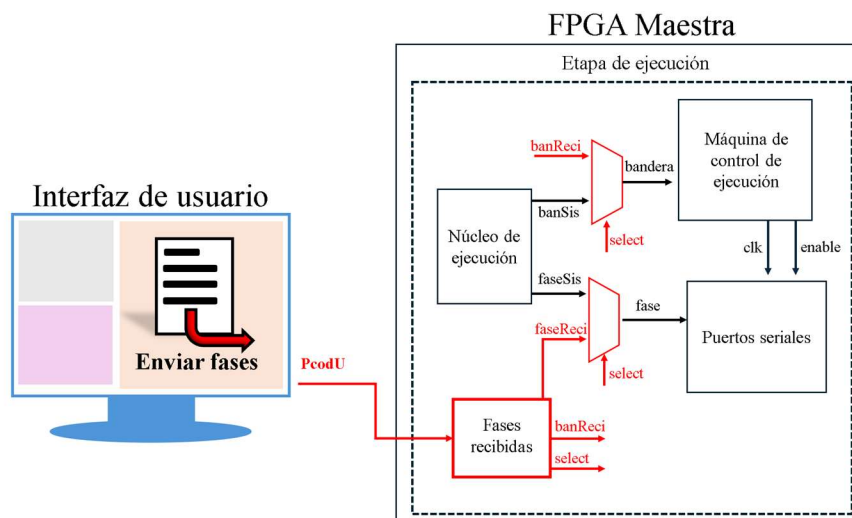


Figura 4-13. Diagrama simplificado de la conmutación para el envío de fases desde la interfaz.

4.2.4 Graficar el patrón de radiación

La gráfica de la forma en que las ondas electromagnéticas son reflejadas por las antenas en arreglos de fase se puede obtener por medio de la ecuación 4.1. Esta sirve para corroborar el patrón emitido por las fases discretas de varios elementos de la antena, considerando sus coordenadas de posición y distancias al alimentador. Se realiza un barrido en todo el espacio dado por los ángulos de orientación, comprobando mediante la suma de la magnitud del campo generado por los elementos para estos ángulos, valor que se almacena en una matriz. Si se grafica cada punto se obtendrá una malla en la que se resaltará el valor máximo almacenado en la matriz, que corresponderá al haz principal del patrón de radiación.

La *Máquina de control de ejecución* levanta una bandera cuando se obtiene una nueva fase lista para enviar por los puertos seriales, entonces se utilizará un módulo de memoria que se habilite cada que esta señal se ponga en alto, de esta manera se tomará el valor de entrada, que es la fase actual, y se guardará para posteriormente enviar todos los datos de la memoria a la interfaz. Estas señales se pueden ver en el diagrama de la Figura 4-14.

En un inicio se consideró enviar las fases a la interfaz cada que el *Núcleo de ejecución* las calculara, pero debido a la alta frecuencia a la que este trabaja, aproximadamente 142.856 MHz [2], se puede complicar la sincronización del envío y habría fases que no serían tomadas en cuenta para enviarse.

Las fases recibidas se almacenarán dentro de la interfaz para posteriormente operarlas con respecto a la (4.1) y obtener el patrón de radiación.

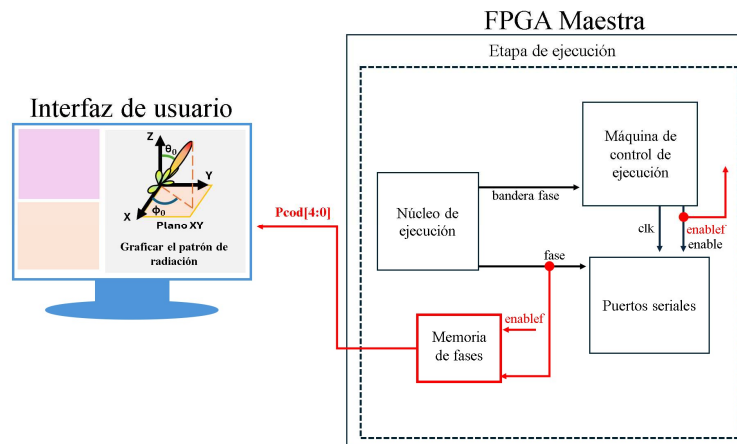


Figura 4-14. Diagrama simplificado para la obtención de las fases del sistema.

4.3 Diseño de detalle

Ya se han definido los requerimientos, capacidades y especificaciones técnicas de entradas y salidas de cada módulo, en esta sección se detallará como se hizo la construcción de cada uno.

4.3.1 Interfaz de usuario

Con ayuda de App Designer de MATLAB, se puede construir una interfaz gráfica que cuente con botones que ejecuten las tareas requeridas, estos botones se encuentran juntos en un menú principal. Al presionar cualquiera de los botones, el menú se cierra y se abre otra pestaña destinada a la tarea que se requiere.

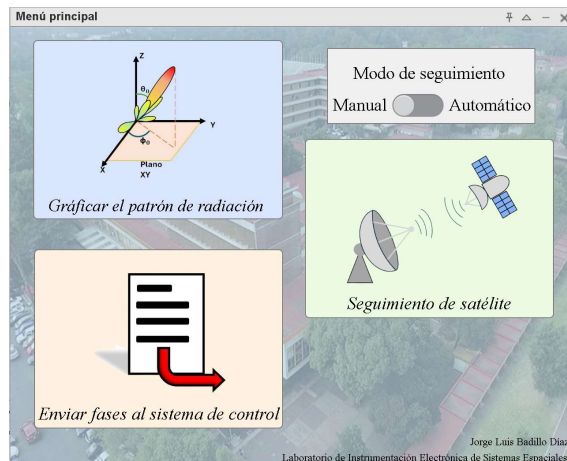


Figura 4-15. Menú principal de la interfaz de usuario.

Previamente a la pestaña mostrada en la Figura 4-15 se cuenta con un menú inicial, mostrado en la Figura 4-16, en dónde se preselecciona el satélite que se desea seguir, así como el número de bits con los que está trabajando el sistema de control y el puerto serial en dónde se encuentra conectado.

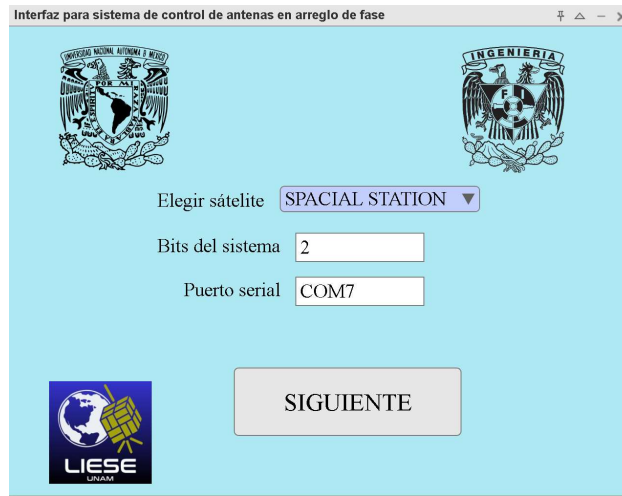


Figura 4-16. Menú inicial de la interfaz de usuario.

Si se elige el botón *Graficar el patrón de radiación* se cierra la ventana de la Figura 4-15 y se abre otra que contiene un elemento gráfico en dónde se dibuja el patrón de radiación dado por puntos calculados con la (4.1), estos puntos están asociados a los ángulos de orientación. Entonces, cada valor corresponde a dos ángulos diferentes, un ejemplo de esto se observa en la Tabla 4-1, en dónde el valor más alto le corresponde al par de ángulos 45° y 180° , hacia esa dirección estaría dirigido el lóbulo principal del patrón de radiación.

Tabla 4-1. Ejemplo de valores generados por la ecuación 4.1.

	0°	90°	180°	360°
0°	3.24	4.9	5.7	4.2
30°	3.6	5.1	6.2	4.9
45°	3.62	5.4	6.8	5.2
60°	3.4	4.9	6.1	4.7

Si se presiona el botón denominado *Enviar fases al sistema de control* se despliega una ventana que le permite al usuario elegir un archivo, desde su PC, que contenga las fases, si estas no se encuentran codificadas, la interfaz se encarga de encontrar su equivalencia, redondear y codificar cada una. La equivalencia se programó basándose en el diagrama de flujo de la Figura 4-17, en dónde se muestra un ejemplo para una mejor comprensión.

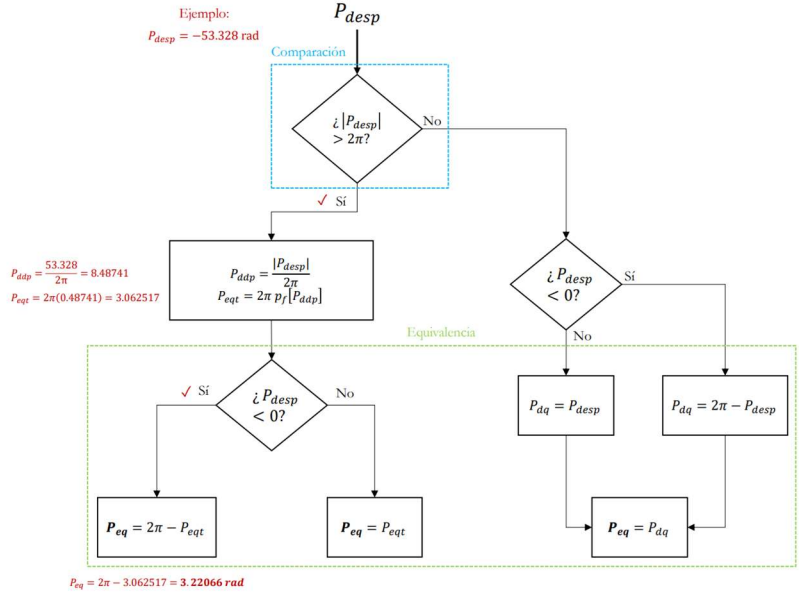


Figura 4-17. Funcionamiento de la equivalencia de fases. [2]

Para realizar el redondeo y la codificación se tomó como base lo hecho dentro del sistema de control, en donde se realiza una multiplicación de la fase equivalente y P_m . Este último es el valor inverso de $2\pi/N$, en donde N es el número de fases del sistema, $N = 2^{b_a}$. Al resultado de esta operación se le suma el primer dígito de la parte fraccionaria, como si fuera un entero. En la Figura 4-18, se muestra un ejemplo con $P_{eq} = 3.61283 \text{ rad}$ (207°) y $b_a = 3$.

$$\begin{array}{l}
 P_{eq} = 3.61283 \text{ y } P_m = 1/\frac{2\pi}{2^3} = 1.27323 \\
 P_{eq} \times P_m = 4.5999 = \underline{00100}.\underline{10011}_b \dots \\
 P_{cod} = \underline{00100}_b + 0000\underline{1}_b = 00101_b
 \end{array}
 \left. \begin{array}{l} \\ \\ \\ \end{array} \right\} \begin{array}{l} \text{Redondeo} \\ \\ \text{Codificación} \end{array}$$

Figura 4-18. Ejemplo del redondeo y codificación de una fase. [2]

Por último, el botón denominado *Seguimiento de satélite* tiene dos modos, el manual y el automático. Si se elige el modo manual el usuario escoge el número de horas, a partir de la actual, sobre las que se hará la predicción de la trayectoria del satélite. Con ayuda de MATLAB se obtienen los ángulos de azimut y elevación (φ_0 y θ_0 , respectivamente), junto con la hora en donde se podrá observar el satélite basados en un punto de observación definido en Ciudad Universitaria. Estos ángulos de orientación se encuentran en grados, entonces se debe obtener su equivalencia en radianes para posteriormente enviarlos al sistema de control digital, este se logra con la Ecuación 4-2.

$$[ang]rad = \frac{[ang]^\circ}{180^\circ} * \pi \tag{Ecuación 4-2}$$

Con el propósito de evitar ciertos errores se han implementado mensajes de aviso dentro de la interfaz. Por ejemplo, si el puerto serial que se ingresa en la ventana observada en la Figura 4-16 no es válido se muestra un mensaje como el de la Figura 4-19.



Figura 4-19. Mensaje de error en puerto serial.

Otro mensaje implementado se encuentra en el modo manual del *Seguimiento de satélites*, si para el tiempo elegido por el usuario no se detecta ningún paso por la ubicación actual, se muestra el aviso de la Figura 4-20.



Figura 4-20. Mensaje de error en la detección de satélite.

4.3.2 Seguimiento de satélites

Como se describió en la sección 4.2.1 los ángulos de Elevación y Azimut se obtendrán por medio de MATLAB, este cuenta con cajas de herramientas (ToolBox, por su nombre en inglés,) que son

capaces de realizar una predicción sobre la órbita de un satélite usando los elementos de dos líneas (TLE). Para este trabajo de tesis se han añadido los TLE de los siguientes satélites:

- TERRA. Satélite lanzado en diciembre de 1999 en una misión del programa Empresa de Ciencia de la Tierra (ESE, por sus siglas en inglés) de la NASA, con la colaboración de Japón y Canadá. Su objetivo es obtener información sobre las propiedades físicas y radiactivas de las nubes, intercambios de aire-tierra y aire-mar, mediciones de gases y vulcanología. [5]
- PAINANI-2. Diseñado y construido por científicos del Instituto Politécnico Nacional y de la UNAM. Este nanosatélite es propiedad de la Secretaría de la Defensa Nacional (SEDENA) y fue puesto en órbita a bordo de un cohete Falcon 9 de la empresa Space X en julio del 2021. Cuenta con subsistemas de comunicaciones para establecer enlaces Tierra- satélite, así mismo, cuenta con sistema de potencia que convierte energía solar a eléctrica. Se le estima un tiempo de vida de 2.5 a 3 años a partir de su lanzamiento. [6]
- Estación Espacial Internacional (por sus siglas en inglés, ISS). Diseñada en colaboración con varias agencias, pero liderada por Estados Unidos y Rusia. Se lanzaron sus primeros elementos en 1998. Actualmente la operan diversas agencias, entre las que se encuentran la Agencia Espacial Europea y la Agencia de Exploración Aeroespacial de Japón. Se estima que seguirá en funcionamiento como laboratorio y puesto de avanzada en órbita hasta el menos 2030. [7]

Para obtener los datos de la órbita satelital, primero hay que definir un rango de tiempo para que se realice la predicción, se debe declarar uno de inicio y uno de fin. Los rangos pueden variar desde un segundo hasta aproximadamente 2 días, esto dependerá de la memoria del propio MATLAB, si el rango es muy amplio no se podrá obtener la predicción. Por lo tanto, es importante definir correctamente el intervalo de tiempo para evitar errores en el programa, a este se le denominará **tiempo de simulación orbital**.

Para realizar esta tarea se ha preestablecido como tiempo de inicio la hora correspondiente a cuando se ejecute el *Seguimiento de satélite* dentro de la interfaz de usuario, por otro lado, el tiempo final varía dependiendo de las selecciones del operador. Si se ha seleccionado el modo manual, se podrá elegir cualquier rango de horas dentro de una lista dada, así como cambiar el satélite elegido en el menú inicial, ver Figura 4-21.

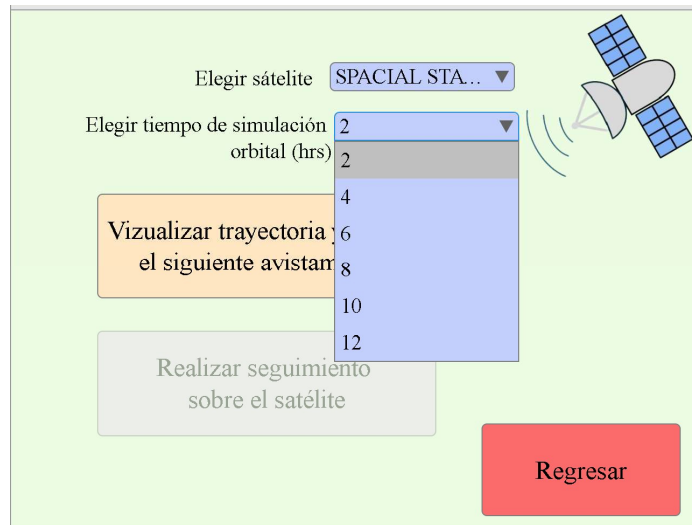


Figura 4-21. Menú para seguimiento de satélites en modo manual.

En el caso en el que se haya elegido el modo automático, el tiempo final está establecido en 18 horas. Si en cualquiera de los rangos de tiempo elegidos no se detecta que el satélite pase sobre la ubicación establecida la interfaz mandará un mensaje de error.

Esto último nos lleva a la pregunta ¿Cómo se detecta un paso sobre nuestra ubicación? Después de seleccionar el tiempo de simulación orbital, se determina el área en la que se realizarán las detecciones, dada por un radar monoestático. Para construir uno, primero se establecen sus coordenadas en latitud y longitud, y luego el rango de ángulos para formar el campo de observación. La ubicación de este radar se encuentra preestablecida en la interfaz de usuario y se eligió poner en Ciudad Universitaria con un rango de 2000 km y un campo de observación de 360° en Azimut y 180° en Elevación para que simule la forma de un domo y poder detectar los satélites en cualquier dirección.

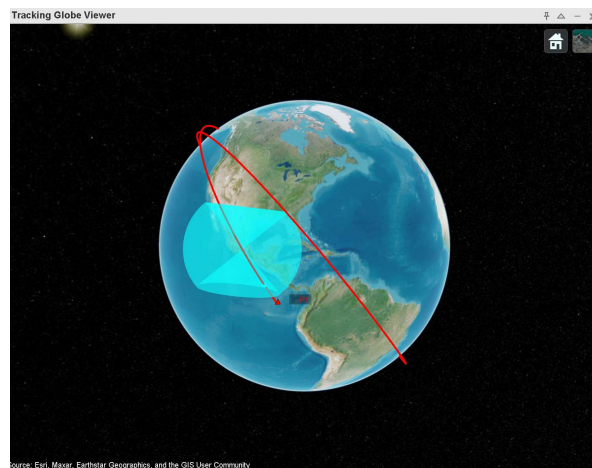


Figura 4-22. Radar para detección de satélites.

Con todo esto dado y una vez iniciada la simulación de la órbita, la interfaz se encarga de recopilar los datos de posición y velocidad, asociados a cada punto generado de la trayectoria simulada. Estos puntos son analizados para comprobar si alguno de ellos cruzó por el radar, de ser así se obtiene el tiempo exacto dado por el día, hora, minutos y segundos para esa detección. Con este valor del tiempo es posible obtener la ubicación del satélite dada en los ángulos de orientación θ_0 y φ_0 , recordando que equivalen a Azimut y Elevación, respectivamente. Estos tres valores son resultado de una detección, al almacenar los datos asociados a cada nueva detección se obtiene una tabla que describe la trayectoria del satélite en el campo de observación generado por el radar. La Tabla 4-2 muestra un ejemplo de cómo se almacenan los ángulos de orientación, con una diferencia de 10 segundos entre cada uno.

Tabla 4-2. Ejemplo de obtención de ángulos para satélite.

Hora	Elevación [°]	Azimut [°]
24/03/2024 23:58:02	41.6907	339.3381
24/03/2024 23:58:12	46.6819	342.7923
24/03/2024 23:58:22	52.2949	347.7716
24/03/2024 23:58:32	58.3855	355.1397
24/03/2024 23:58:42	64.4718	7.0755
24/03/2024 23:58:52	69.3728	26.6871
24/03/2024 23:59:02	71.0565	54.4229

Los ángulos que se obtengan por la interfaz deben ser enviados al sistema de control. Como se mencionó en la sección 4.2.2 es necesario un módulo que reciba los ángulos de orientación por medio de UART, para usar implementar este protocolo se utilizó una máquina secuencial proporcionada en el trabajo de tesis del M. I. Ernesto Hernández Gastaldi, la cual recibe el nombre de Transceptor UART [2]. Este utiliza dos máquinas, una de recepción y otra de transmisión. Las señales que se utilizan para la creación de los módulos que se comunican con la interfaz se describen en la Tabla 4-3. Y en la Figura 4-23 se observan las dos máquinas secuenciales que conforman este puerto.

Tabla 4-3. Descripción de señales del transceptor UART.

Tipo	Nombre	Descripción
Entradas	transmit	Señal que sirva para iniciar la transmisión
	tx_byte[7:0]	Byte que se desea transmitir
Salidas	rx_byte[7:0]	Byte recibido
	is_receiving	Bandera que indica cuando se está recibiendo un dato

	is_transmitting	Bandera que permanece en alto mientras se está transmitiendo un dato
	received	Indica cuando se ha recibido el dato

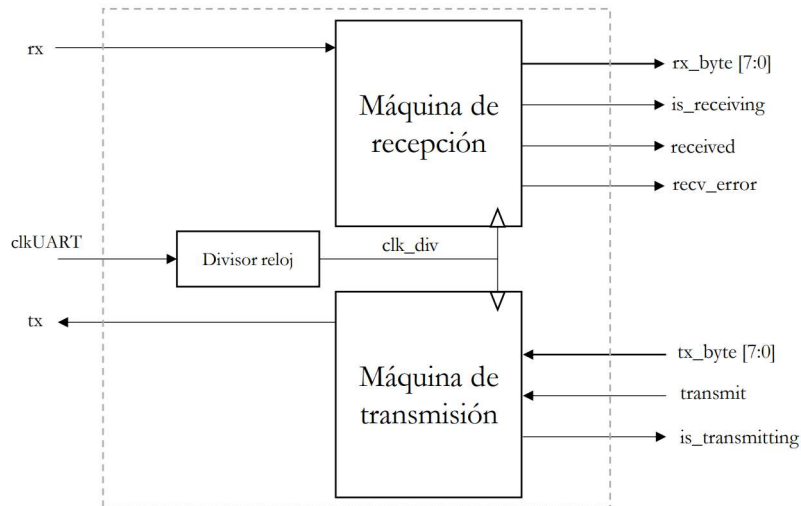


Figura 4-23. Diagrama de bloques del transceptor UART [2].

Con ayuda del transceptor y bajo los requerimientos de recepción de datos de esta etapa de diseño se construyó el *Puerto de ángulos de orientación UART*. El diagrama de estados correspondiente se encuentra en la Figura 4-24. Seguido de las tablas que describen las señales y los estados, Tabla 4-4 y Tabla 4-5, respectivamente. En la Tabla 4-6, se muestran los estados presentes y futuros, con los que se obtuvo el diagrama de conexiones de la Figura 4-25. Por último, el diagrama de bloques del puerto se encuentra en la Figura 4-26.

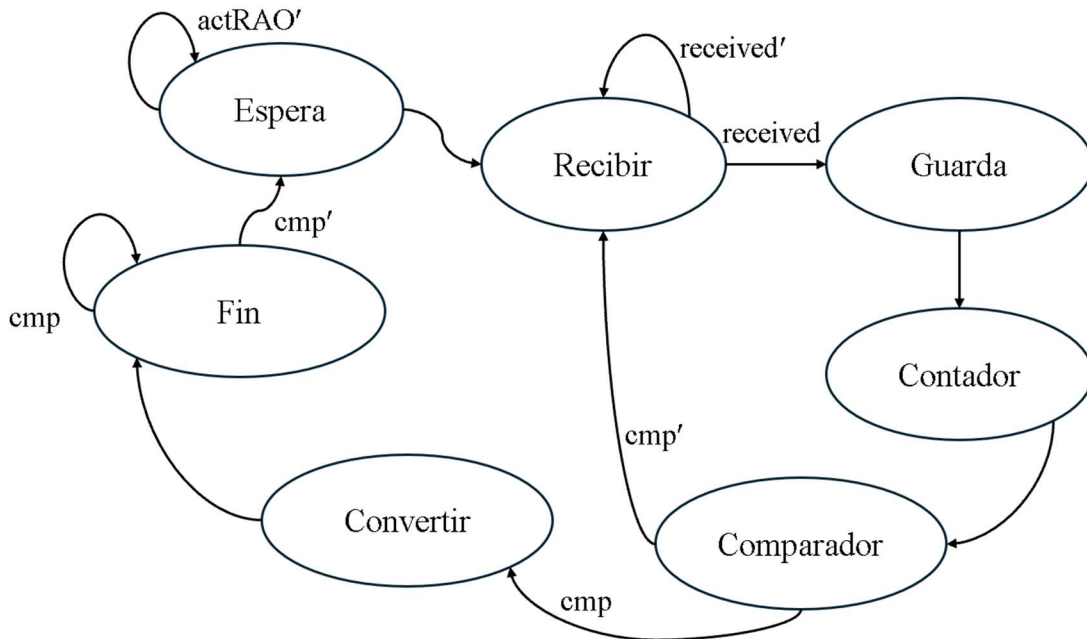


Figura 4-24. Diagrama de estados del puerto de ángulos de orientación UART.

Tabla 4-4. Descripción de señales del puerto de ángulos de orientación UART.

Señales		
Tipo	Nombre	Descripción
Entrada	actRAO	Señal de activación para la recepción de ángulos de orientación.
	cmp	Comparación realizada para identificar ángulo recibido.
	received	Señal que indica que se está recibiendo un dato por el puerto UART.
	AOReci [7:0]	Valor de la fase que se recibe.
Salida	sAOn	Indica que el par de ángulos de orientación se ha recibido
	Cont	Contador
	Theta0[23:0]	Valor de ángulo theta recibido.
	Phi0[23:0]	Valor de ángulo phi recibido.

Tabla 4-5. Descripción de estados y transiciones del puerto de ángulos de orientación UART.

Estados		
Nombre	Descripción y transiciones	Valores de señales de salida (las que no se muestran explícitas tienen un valor de 0).
Espera	<p>Inicia la máquina de estados, estado de reposo, en espera comando de activación.</p> <p>actRAO=0, el sistema no ha recibido comando de activación, permanece en <i>Espera</i>.</p> <p>actRAO=1, el sistema ha recibido activación. Pasa a <i>Recibir</i>.</p>	Todas las señales de salida son cero.
Recibir	<p>Se espera a que se reciba un dato por el puerto UART.</p> <p>received=0, no se ha recibido ningún dato. Permanece en <i>recibir</i>.</p> <p>received=1, significa que un dato ha sido recibido. Pasa a <i>Guarda</i>.</p>	Todas las señales de salida son cero.
Guarda	<p>Estado en el que guarda en un registro el dato recibido.</p> <p>Transición incondicional a <i>Contador</i>.</p>	Todas las señales de salida son cero.
Contador	<p>Estado que pone en alto la señal del reloj del contador para que se aumente la cuenta.</p> <p>Transición incondicional a <i>Comparador</i>.</p>	Cont=1
Comparador	<p>Compara si se ha llegado al final de la cuenta.</p> <p>Cmp=0, pasa a <i>Recibir</i>.</p> <p>Cmp=1, pasa a <i>Convertir</i>.</p>	Todas las señales de salida son cero.
Convertir	<p>Se obtiene en un solo registro el equivalente del par de ángulos de orientación recibidos.</p>	Phi0[23:0] Theta0[23:0]
Fin	<p>Estado final. Se levanta la bandera que indica que se han recibido los ángulos.</p>	sAOn=1

Tabla 4-6. Tabla de estados presentes y futuros del puerto de ángulos de orientación UART.

Estado	Q_2 Q_1 Q_0	Entradas						Salidas	
		actRAO		received		cmp		sAOn	Cont
		0	1	0	1	0	1		
Espera	000	000	001	000	000	000	000	0	0
Recibir	001	001	001	001	010	001	001	0	0
Guarda	010	011	011	011	011	011	011	0	0
Contador	011	100	100	100	100	100	100	0	1
Comparador	100	100	100	100	100	001	101	0	0
Convertir	101	110	110	110	110	110	110	0	0
Fin	110	110	110	110	110	000	110	1	0

De la Tabla 4-6 se obtuvieron las funciones lógicas mostradas en la Tabla 4-7. Para mejor visualización de estas, solo se usa la letra inicial de cada señal de entrada.

Tabla 4-7. Funciones de entradas y salidas del puerto de ángulos de orientación UART.

$T_2 = Q_1Q_0 + a'r'c'Q_2Q_0' + Q_1Q_0$
$T_1 = a'r'c'Q_2'Q_1 + a'rc'Q_0 + Q_1Q_0 + Q_2Q_0$
$T_0 = Q_2'Q_1 + a'r'Q_2Q_1' + a'rc'Q_0 + ar'c'Q_2'Q_0' + Q_2Q_0$
$sAOn = Q_2Q_1Q_0'$
$Cont = Q_2'Q_1Q_0$

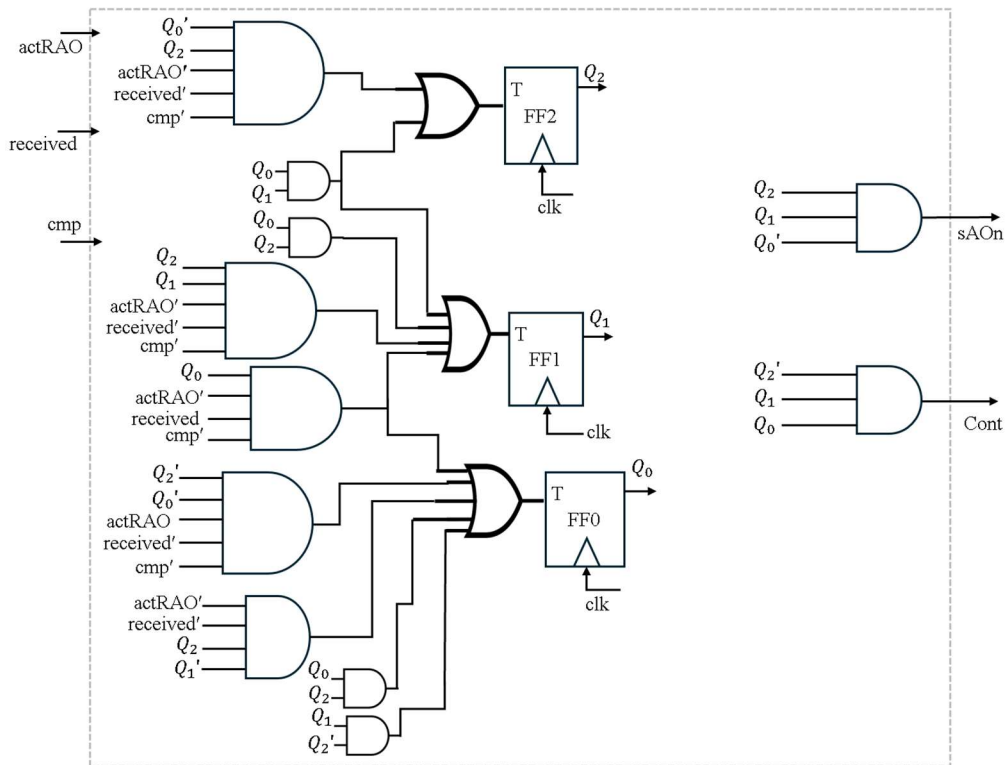


Figura 4-25. Diagrama de conexiones del puerto de ángulos de orientación UART.

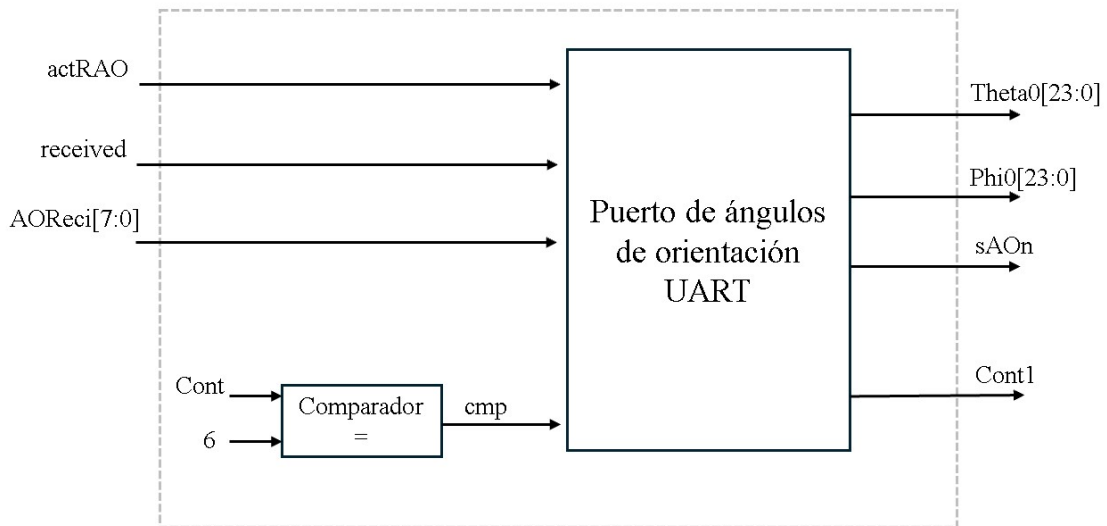


Figura 4-26. Diagrama de bloques del puerto de ángulos de orientación UART.

4.3.3 Envío de fases para cada elemento

En la sección 4.2.3 se planteó que cuando el sistema de control recibiera una fase enviada desde la interfaz se transfiriera directamente hacia los *Puertos seriales*, sin embargo, dada la construcción del sistema de control se ve que una mejor solución es guardar las fases que provienen de la interfaz. A continuación, se detallará la explicación de este análisis.

Recordando, la *Máquina de control de ejecución* es la encargada de dirigir todas las señales involucradas con el cálculo de las fases, realizado en el *Núcleo de ejecución*, y de enviarlas por los *Puertos seriales*. Este es un proceso complejo que se ejecuta automáticamente cuando se recibe un nuevo par de ángulos de orientación a través del *Puerto de recepción*. Cada que se calcula una fase, la *Máquina de control de ejecución* envía una señal para activar los *Puertos seriales* y realizar la distribución.

Con el propósito de asegurar que las fases que se envíen por los *puertos seriales* sean las que se recibieron de la interfaz, estas serán guardadas en memoria antes que el sistema de control comience la tarea de calcular y enviar. Una vez que se hayan guardado todos los datos provenientes de la interfaz se cambia la entrada de los puertos seriales, que normalmente es la fase calculada por el sistema de control, pero para este diseño se necesita la que está guardada en la memoria; este cambio es llevado a cabo por un multiplexor.

Posteriormente, se realiza un envío interno de unos ángulos de orientación al *Puerto de recepción* con la intención de activar la *Máquina de control de ejecución* y hacer uso de sus capacidades para distribuir las fases guardadas en memoria a los *Puertos seriales*.

En resumen, se necesitan hacer las siguientes tareas:

1. Enviar fases de la interfaz.
2. Guardarlas en el sistema de control.
3. Enviar ángulos de orientación.
4. El sistema arranca su funcionamiento de forma normal.
5. En lugar de las fases generadas por el *Núcleo de ejecución*, se enviarán las fases guardadas en el punto 2.

Teniendo en cuenta las acciones requeridas en esta etapa de diseño se construyeron dos máquinas secuenciales con el propósito de facilitar su implementación, estas reciben los nombres de: *Máquina de recepción de fases* y *Máquina de distribución de fases*. La primera, recibe y guarda las fases de forma ordenada. Para finalizar la recepción de fases es necesario que la interfaz envíe el comando 0xEE, para detectarlo se utiliza un comparador a la entrada de la máquina, En la Figura 4-27 se muestra el diagrama de estados.

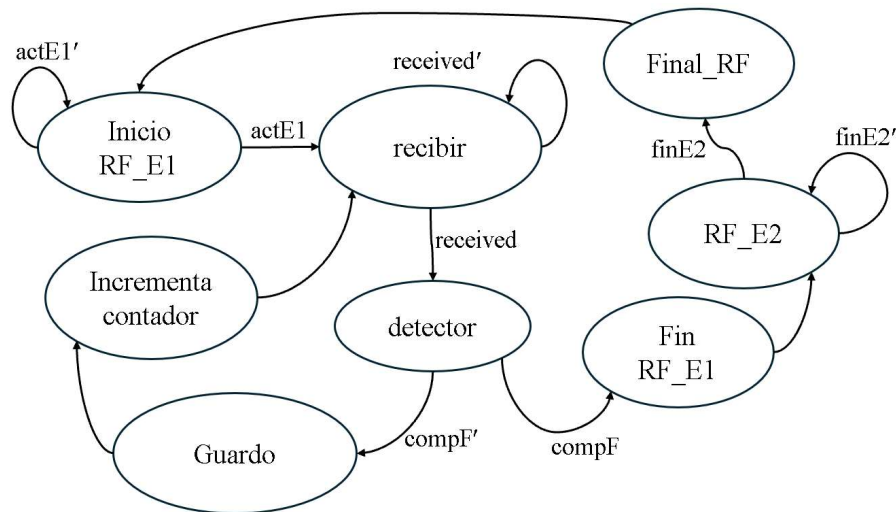


Figura 4-27. Diagrama de estados de la máquina de recepción de fases.

El tamaño de la memoria es igual a 1500, que es el número máximo de elementos que se pueden controlar por el sistema. Para poder cambiar la dirección de almacenamiento de cada fase se utiliza un contador que aumenta con cada ciclo de reloj. Dado que este es el que controla la dirección de la memoria se necesita que alcance la cuenta de 1500, entonces debe ser de 11 bits. Este valor de cuenta se utilizará en las maquinas que se diseñen más adelante.

$$1500_{10} = 10111011100_2$$

En la Tabla 4-8 se describen las señales de entrada y salida de la máquina de recepción de fases. Posteriormente, en la Tabla 4-9 se detallan los estados y sus transiciones, seguida de la Tabla 4-10 en dónde se presentan los estados presentes y futuros. Por último, se muestran los diagramas de conexiones y de bloques, en la Figura 4-29 y Figura 4-29, respectivamente.

Tabla 4-8. Descripción de señales de la máquina de recepción de fases.

Señales		
Tipo	Nombre	Descripción
Entrada	actE1	Señal de activación para la recepción de fases.
	finE2	Indica que la máquina de distribución de fase ha concluido.
	received	Señal que indica que se está recibiendo un dato por el puerto UART.

	compF	Se activa cuando se recibe el comando de termino de recepción.
	faseReci [7:0]	Valor de la fase que se recibe.
Salida	finRF	Bandera que indica que ambas máquinas han terminado su proceso.
	faseSal [7:0]	Valor de fase guardado en la memoria.
	clkContaRF	Señal de reloj para incrementar el contador.
	vConta[10:0]	Valor del contador, sirve como dirección para la memoria.
	actWriteRF	Señal que habilita la escritura de la memoria
	clkMemRF	Señal de reloj para la memoria.
	actE2	Se utiliza para activar la máquina de distribución de fases.

Tabla 4-9. Descripción de estados y transiciones de la máquina de recepción de fases.

Estados		
Nombre	Descripción y transiciones	Valores de señales de salida (las que no se muestran explícitas tienen un valor de 0).
Inicio RF_E1	Inicia la máquina de estados, estado de reposo, en espera comando de activación. actF=0 , el sistema no ha recibido comando de activación, permanece en <i>Espera</i> . actF=1 , el sistema ha recibido activación. Pasa a <i>Recibir</i> .	Todas las señales de salida son cero.
Recibir	Estado de espera. Se recibe un dato por el puerto UART. Se prepara la memoria para cargarle datos.	Todas las señales de salida son cero.

	<p>received=0, no se ha recibido ningún dato. Permanece en <i>recibir</i>.</p> <p>received=1, significa que un dato ha sido recibido. Pasa a <i>Detector</i>.</p>	
Detector	<p>Estado en el que se compara el dato recibido con el comando de finalización del recibimiento de fases.</p> <p>compF=0, no es el comando, se pasa a <i>Guardo</i>.</p> <p>compF=1, es el comando que finaliza la recepción. Pasa a <i>FinRF E1</i>.</p>	Todas las señales de salida son cero.
Guardo	<p>Estado en el que se guarda la fase que se recibió.</p> <p>Transición incondicional a <i>resConta</i>.</p>	<p>clkMemRF=1</p> <p>actWriteRF=1</p>
Incrementa Contador	<p>Se pone un flanco de subida en el reloj del contador, lo que provoca que se sume una unidad. Se avanza en dirección de memoria.</p> <p>Transición incondicional a <i>recibir</i>.</p>	clkContaRF=1
finRF_E1	<p>Se ha detectado el comando de finalización del proceso actual. El valor del contador se guarda para determinar cuál es el valor máximo que se alcanzará al enviar la memoria. Se resetea el contador y se deshabilita la escritura en memoria.</p> <p>Transición incondicional a RE E2.</p>	resConta=1
RF_E2	<p>Se da inicio y se cede el control de la memoria a la máquina de distribución de fases.</p> <p>finE2=0, la máquina de distribución aún no termina su proceso. Permanece en <i>RF_E2</i>.</p> <p>finE2=1, la máquina de distribución ha finalizado su proceso.</p> <p>Pasa a <i>Final RF</i>.</p>	actE2=1
Final_RF	<p>Estado final de la recepción de fases, se ha concluido el envío por los <i>Puertos seriales</i>.</p>	finRF=1

Tabla 4-10. Estados presentes y futuros de la máquina de recepción de fases.

		Entradas							
Estado	$Q_2Q_1Q_0$	actE1		received		compF		Fin_E2	
		0	1	0	1	0	1	0	1
Inicio RE_E1	000	000	001	000	000	000	000	000	000
Recibir	001	001	001	001	010	001	001	001	001
Detector	010	010	010	010	010	011	101	010	010
Guardo	011	100	100	100	100	100	100	100	100
I. Contador	100	001	001	001	001	001	001	001	001
finRF_E1	101	110	110	110	110	110	110	110	110
RFE2	110	110	110	110	110	110	110	110	111
Final_RF	111	000	000	000	000	000	000	000	000

		Salidas				
$Q_2Q_1Q_0$	FinRF	resConta	clkContaRF	actE2	FinRF	clkMemRF actWriteRF
000	0	0	0	0	0	0
001	0	0	0	0	0	0
010	0	0	0	0	0	0
011	0	0	0	0	0	0
100	0	0	1	0	0	1
101	0	1	0	0	0	0
110	0	0	0	1	0	0
111	1	0	0	0	1	0

Con los datos de la Tabla 4-10, se obtuvieron las funciones lógicas de la Tabla 4-11. Se usan solo la primera letra de cada señal de entrada para apreciar cada función de buena forma.

Tabla 4-11. Funciones lógicas de la máquina de recepción de fases.

$T_2 = Q_1Q_0 + Q_2Q_1'Q_0' + a'r'c'F'Q_2'Q_1$	
$T_1 = Q_2Q_0 + a'r'c'F'Q_2'Q_1 + Q_1Q_0 + a'rc'F'Q_0'$	
$T_0 = Q_1Q_0 + Q_2Q_1' + a'r'F'Q_2'Q_1 + a'r'c'F'Q_2 + a'rc'F'Q_0 + ar'c'F'Q_1'Q_0'$	
$clkMemRF = actWriteRF = Q_2'Q_1Q_0$	
$clkContaRF = Q_2Q_1'Q_0'$	$resConta = Q_2Q_1'Q_0$
$actE2 = Q_2Q_1Q_0'$	$FinRF = Q_2Q_1Q_0$

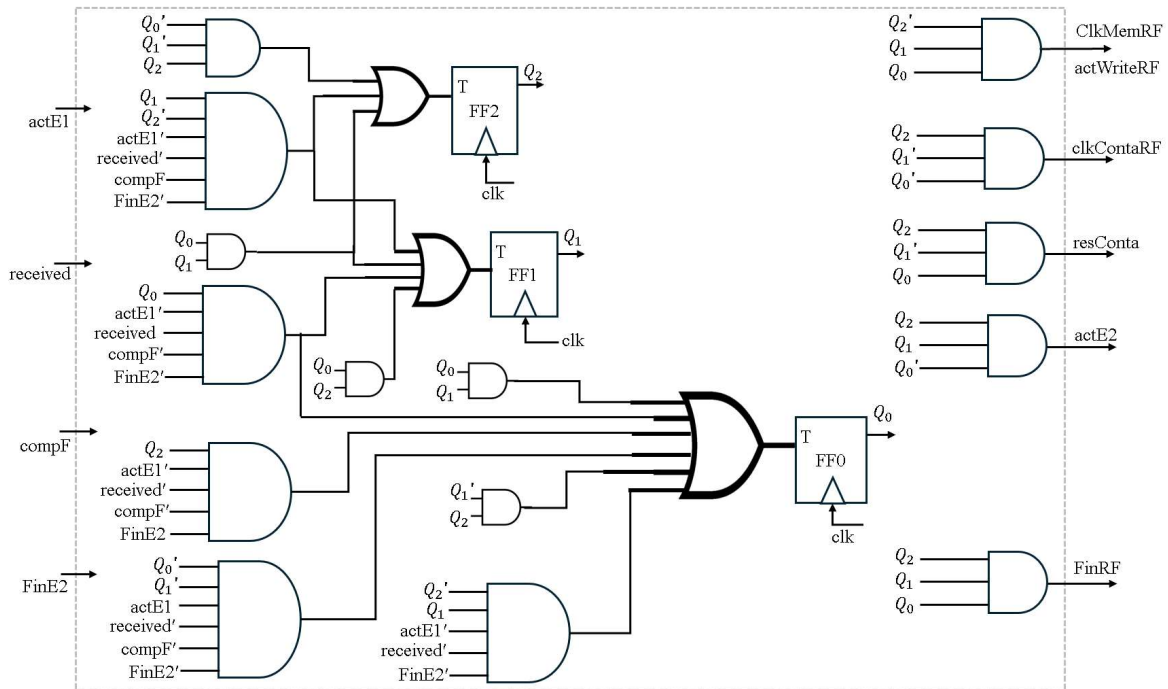


Figura 4-28. Diagrama de conexiones de la máquina de recepción de fases.

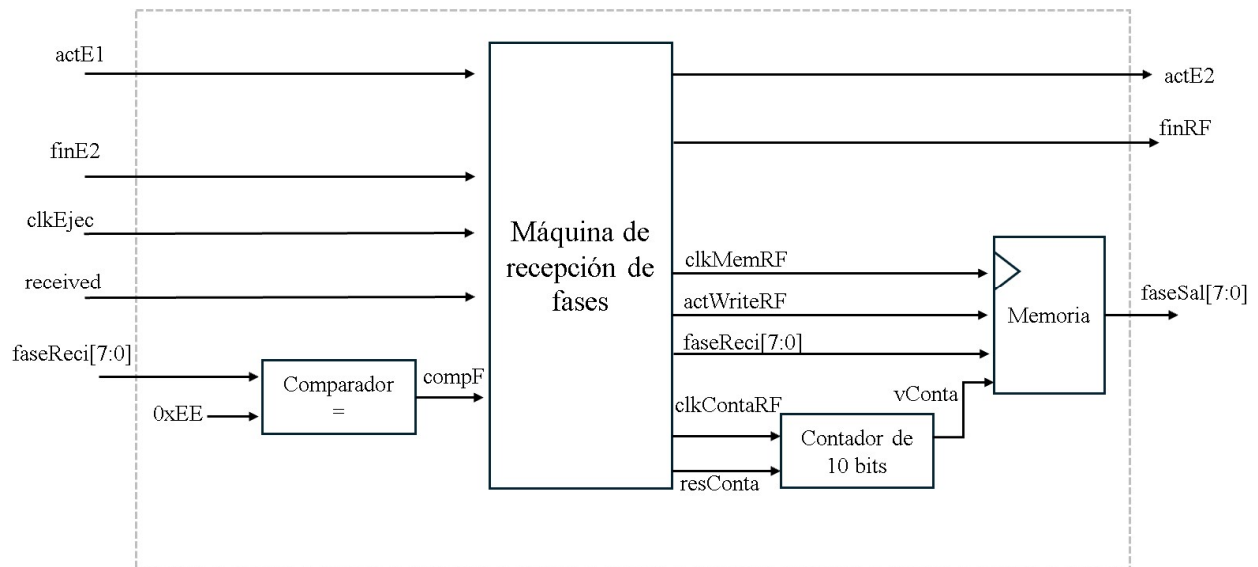


Figura 4-29. Diagrama de bloques de la máquina de recepción de fases.

Por otro lado, la *Máquina de distribución de fases* se encarga de monitorear las señales del sistema de control para determinar en qué momento ingresar los datos que se recibieron de la interfaz. Cuando esta máquina es activada le envía unos ángulos de orientación al sistema de control, con el propósito que este comience a calcular las fases y a generar las señales de activación para los *Puertos*

seriales. Para sincronizar las fases recibidas con las fases generadas se utilizará la señal llamada **clkEjec**, que se pone un flanco de subida cada que el *Núcleo de ejecución* calcula una nueva fase, esta será ingresada al contador de manera que en cada ciclo de reloj se apunte a la siguiente fase guardada en la memoria. Por otra parte, **orPSn** es una señal que funcionará para conocer si el sistema sigue enviando fases, cuando esta bandera ya no se encuentre activa se finaliza la lectura de la memoria. El diagrama de estados de la *Máquina de distribución de fases* se observa en la Figura 4-30. Mas adelante, en la Tabla 4-12 se encuentran las señales de entrada y salida, seguida de la descripción de los estados y transiciones de la máquina, en la Tabla 4-13. Finalmente se muestran los estados presentes y futuros en la Tabla 4-14, con estos se diseñaron los diagramas de conexiones y de bloques de la Figura 4-31 y Figura 4-32.

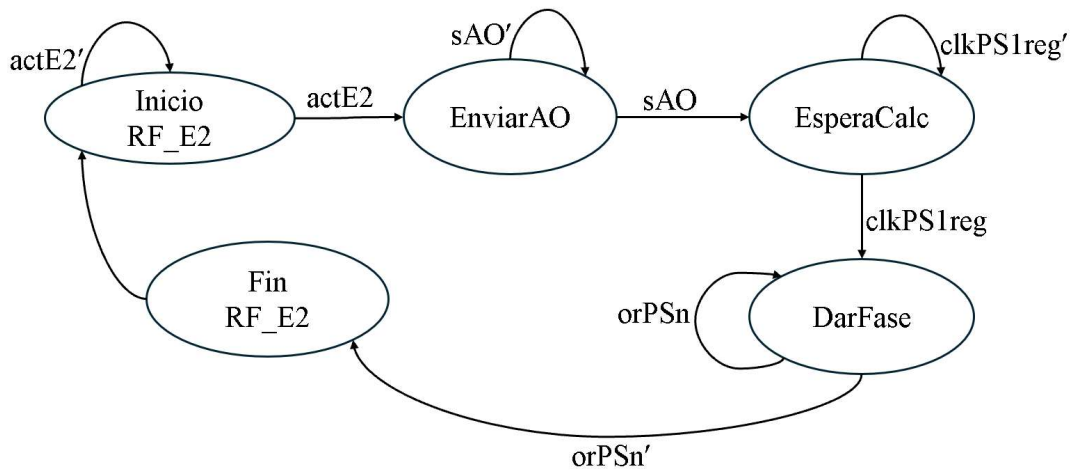


Figura 4-30. Diagrama de estados de la Máquina de distribución de fases.

Tabla 4-12. Descripción de señales de la máquina de distribución de fases.

Señales		
Tipo	Nombre	Descripción
Entrada	actE2	Señal de activación de máquina de distribución de fases.
	sAO	Señal de salida del puerto de recepción. Indica que se han terminado de recibir los ángulos de orientación.
	orPSn	Bandera que mientras esté activa, se siguen mandando fases.

	clkEjec	Señal de reloj dada por el sistema, indica que se ha calculado una nueva fase.
	clkPS1Reg	Señal de reloj que habilita al primer puerto serial.
	faseReci [7:0]	Valor de la fase en memoria.
Salida	finE2	Indica que la máquina de distribución de fases ha concluido
	thetaE[7:0]	Ángulo theta que se enviará al puerto de recepción de ángulos.
	phiE[7:0]	Ángulo phi que se enviará al puerto de recepción de ángulos
	faseEnvi [7:0]	Valor de fase que se enviará a los puertos.
	vCont[10:0]	Valor que determina la dirección de la memoria.
	resetConta	Resetea el valor del contador.
	finE2	Indica el fin de la máquina de distribución de fases.

Tabla 4-13. Descripción de los estados de la máquina de distribución de fases.

Estados		
Nombre	Descripción y transiciones	Valores de señales de salida (las que no se muestran explícitas tienen un valor de 0).
Inicio RF_E2	Inicia la máquina de estados, estado de reposo, en espera de activación. actE2=0 , el sistema no ha recibido comando de activación, permanece en <i>Espera</i> . actE2=1 , el sistema ha recibido activación. Pasa a <i>EnviarAO</i> .	Todas las señales de salida son cero.
EnviarAO	Estado de espera. Espera a que se reciban los ángulos de orientación. sAO=0 , no se han recibido ángulos. Permanece en <i>EnviarAO</i> .	phi[23:0]= 90° theta[23:0]= 30°

	sAO=1 , se han recibido los ángulos correctamente. Pasa a <i>EsperaCalc</i>	
EsperaCalc	Estado que espera a que el sistema de control haga el cálculo de las fases. clkPS1reg=0 , no ha concluido, se mantiene en <i>EsperaCalc</i> . clkPS1reg=1 , ha terminado de calcular fases. Pasa a <i>DarFase</i> .	Todas las señales de salida son cero.
DarFase	Estado en dónde la fase guardada en memoria se vuelve la entrada de los Puertos seriales para enviarla. orPSn=1 , el envío de fases sigue activo. Se mantiene en <i>DarFase</i> . orPSn=0 , ha terminado el envío de fases de parte de los puertos seriales. Pasa a <i>Fin RF E2</i> .	faseEnvi [7:0]= faseReci [7:0]
Fin RF_E2	Estado final de la máquina de distribución de fases. Se resetea el contador y se levanta la bandera que indica el fin de esta máquina. Transición incondicional a inicio RE E2.	resConta=1 finE2=1

Tabla 4-14. Estados presentes y futuros de la máquina de distribución de fases.

		Entradas								Salidas	
Estado	$Q_2 Q_1 Q_0$	actE2		sAO		clkPS1reg		orPSn		resConta	finE2
		0	1	0	1	0	1	0	1		
Inicio RF_E2	000	000	001	000	000	000	000	000	000	0	0
EnviarAO	001	001	001	001	010	001	001	001	001	0	0
EsperaCalc	010	010	010	010	010	010	011	010	010	0	0
DarFase	011	011	011	011	011	011	011	100	011	0	0
Fin RF_E2	100	000	000	000	000	000	000	000	000	1	1

Con los datos de la Tabla 4-14 se obtuvieron las funciones lógicas de la Tabla 4-15.

Tabla 4-15. Funciones lógicas de la máquina de distribución de fases.

$T_2 = Q_2 + a's'c'o'Q_1Q_0$
$T_1 = a's'c'o'Q_1Q_0 + a's'c'o'Q_1'Q_0$
$T_0 = a's'c'o'Q_1Q_0 + a's'co'Q_1Q_0' + a'sc'o'Q_1'Q_0 + as'c'o'Q_2'Q_1'Q_0'$
$FinE2 = resConta = Q_2 Q_1'Q_0'$

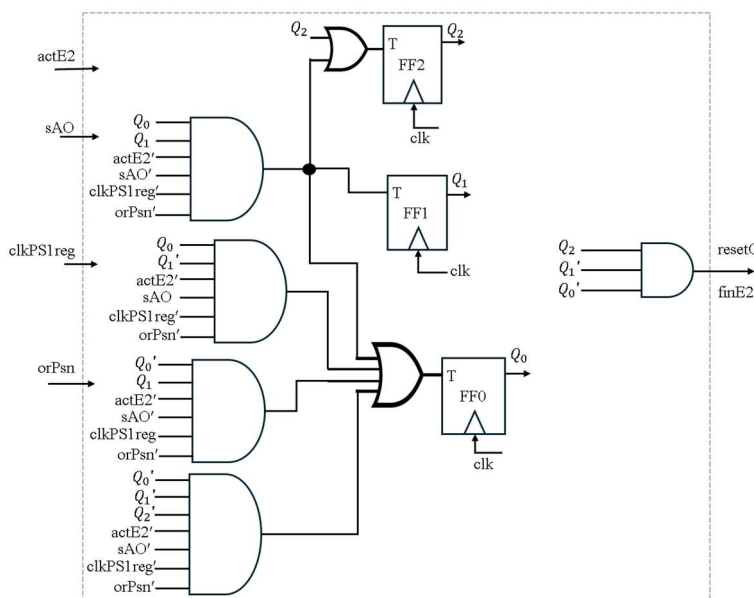


Figura 4-31. Diagrama de conexiones de la máquina de distribución de fases.

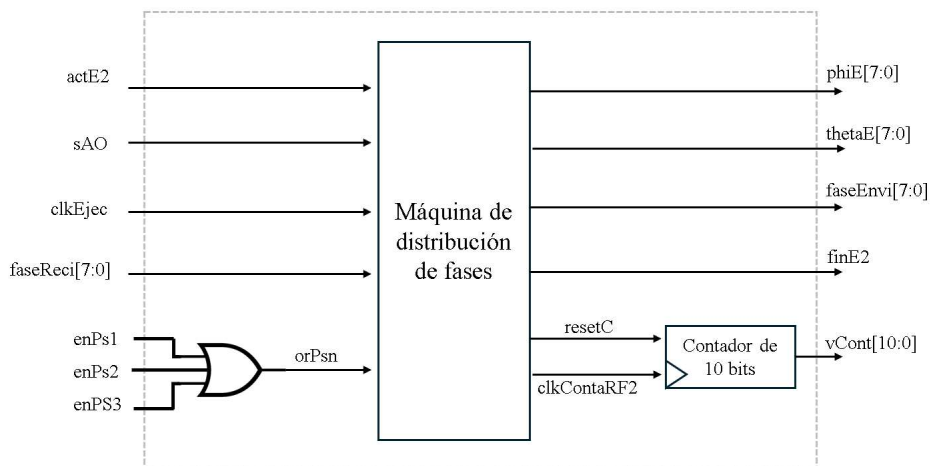


Figura 4-32. Diagrama de bloques de la máquina de distribución de fases.

Uniendo las máquinas diseñadas se obtiene el diagrama de la Figura 4-33, en donde se muestran las señales de entrada y salida correspondientes a cada máquina. De esta unión se construye el módulo que recibe las fases desde la interfaz y las distribuye por los puertos seriales.

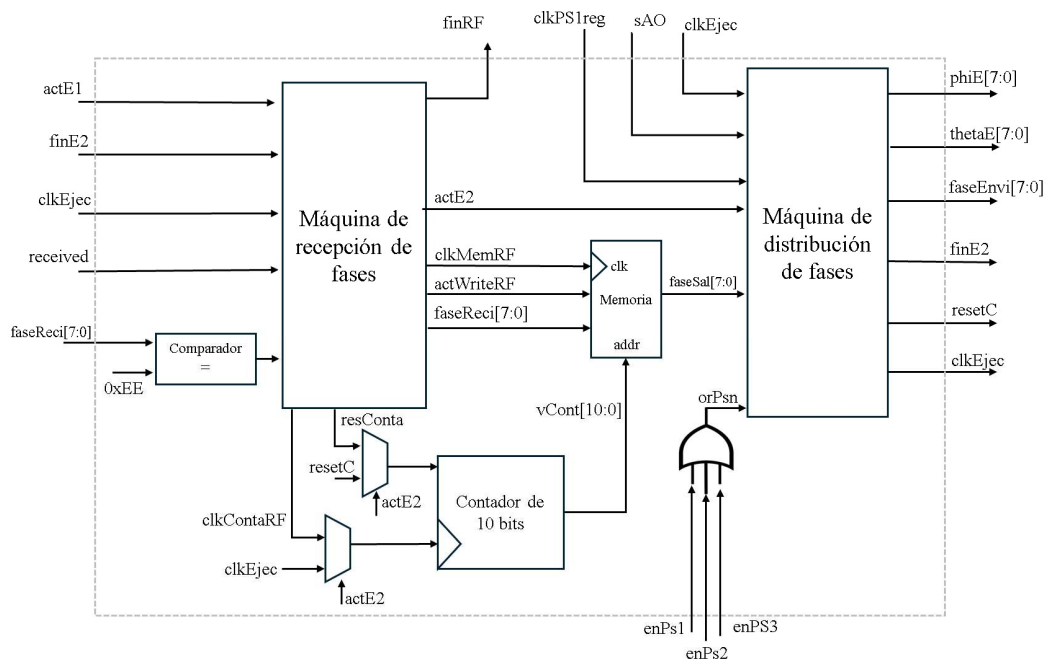


Figura 4-33. Conjunción de las máquinas para recibir las fases.

4.3.4 Graficar patrón de radiación de la antena

Siguiendo con lo detallado en la sección 4.2.4, se necesitan las fases que el sistema de control calcula para poder realizar la gráfica del patrón de radiación, entonces estas deben ser enviadas a la interfaz. Dado que esta representación gráfica solamente es para corroborar la orientación de la antena no se requiere una alta velocidad en la transferencia de las fases. Por lo tanto, la mejor alternativa es almacenar estos datos tal y como los vaya generando el Núcleo de ejecución, para posteriormente enviarlos hacia la interfaz.

El sistema de control genera unas nuevas fases cuando recibe un par de ángulos de orientación, después comenzará con el cálculo y finalmente con la distribución de estas por los *Puertos seriales*. Cada uno de estos puertos recibe una señal de activación llamada **enPsi** ($i=1,2,3$) que indica que las fases se están enviando, el monitoreo de estas señales da como resultado un habilitador para la memoria. Por otro lado, la señal **clkEjec** indica cuando se ha calculado una nueva fase, igualando esta señal a la de reloj del contador y de la memoria, y el dato recién obtenido por el sistema de control se guarda. De esta manera se evitan pérdidas de fases o que algunas fases se almacenen dos veces.

Para poder controlar la escritura en memoria y posteriormente el envío de datos hacia la interfaz, se han diseñado dos máquinas secuenciales, la *Máquina de detección de fases* y la *Máquina de envío*

de memoria. La primera está encargada de monitorear la recepción de un nuevo par de ángulos de orientación y de guardar las fases que se calculen en el sistema de control. El diagrama de estados de esta máquina se presenta en la Figura 4-34, después en la Tabla 4-16 se indican las señales involucradas. Posteriormente en la Tabla 4-17 se explican los estados, sus transiciones y salidas correspondientes. Y en la Tabla 4-18 se muestran los estados presentes y futuros.

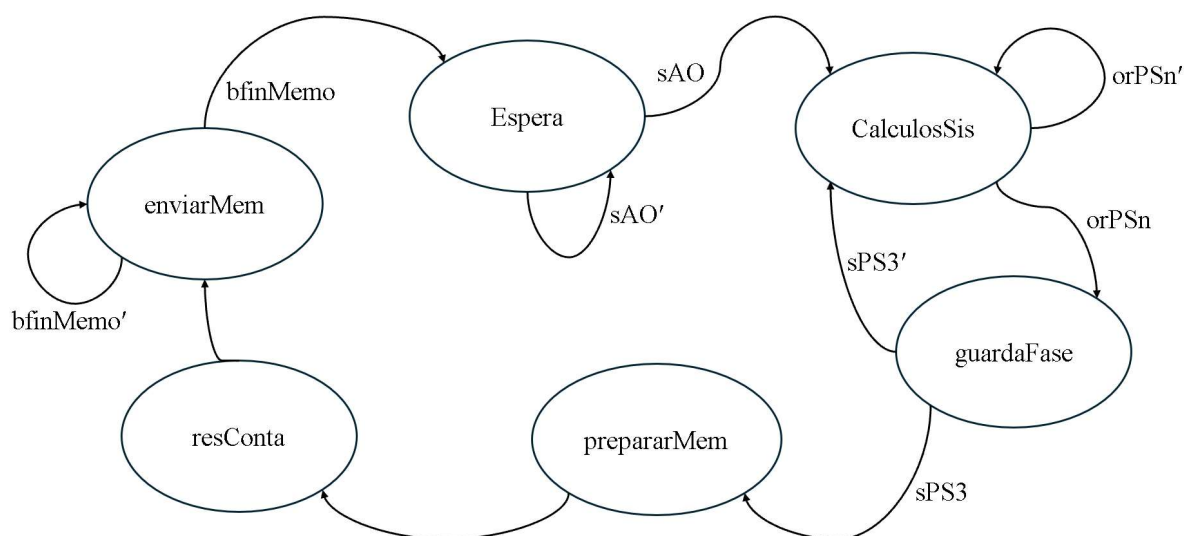


Figura 4-34. Diagrama de estados de la máquina de detección de fases.

Tabla 4-16. Descripción de señales de la máquina de detección de fases.

Señales		
Tipo	Nombre	Descripción
Entrada	sAO	Señal de salida del puerto de recepción. Indica que se han terminado de recibir los ángulos de orientación.
	clkEjec	Señal que indica que una fase nueva se ha calculado.
	sPS3	Proveniente de los puertos seriales. Se usa para saber que se han terminado de calcular las fases
	bfinMemo	Bandera de finalización del envío de la memoria.
Salida	clkConta	Señal de reloj del contador
	Vconta [10:0]	Valor del contador, indica la dirección de la memoria.

	resetC	Señal que reinicia al contador.
	limiteMem[10:0]	Se guarda el valor máximo al que llegue la memoria, para que al enviar los datos ese sea su límite.
	actWrite	Señal que habilita la escritura de la memoria
	clkMem	Señal de reloj para la memoria.
	actEnv	Se utiliza para activar la máquina de envío de memoria.

Tabla 4-17. Descripción de estados de la máquina de detección de fases.

Estados		
Nombre	Descripción y transiciones	Valores de señales de salida (las que no se muestran explícitas tienen un valor de 0).
Espera	Inicia la máquina de estados, en espera a recibir ángulos de orientación. sAO=0 , el sistema no ha recibido un nuevo par de ángulos de orientación, permanece en <i>Espera</i> . sAO=1 , el sistema ha recibido ángulos de orientación. Pasa a <i>CalculosSis</i> .	Todas las señales de salida son cero.
CalculosSis	Estado de espera. Se permanece en este estado mientras el sistema de control realiza el cálculo de las fases. Se prepara la memoria para cargarle datos. clkEjec =0 , no se ha calculado ninguna fase. Permanece en <i>CalculosSis</i> . clkEjec =1 , significa que una fase ha sido obtenida. Pasa a <i>guardaFase</i> .	actWrite=1
guardaFase	La fase que ha sido calculada por el sistema de control es guardada y se incrementa el contador. sPS3=0 , aún hay fases por calcular, regresa a <i>CalculosSis</i> .	clkMem=1 clkConta=1

	sPS3=1 , el sistema ha terminado de calcular todas las fases. Pasa a <i>prepararMem</i> .	
prepararMem	Estado en el que el valor del contador se guarda para determinar cuál es el valor máximo que alcanzará al enviar la memoria. Transición incondicional a <i>resConta</i> .	limiteMem=Vconta
resConta	Se regresa el contador a 0. Y se deshabilita la escritura de la memoria, Transición incondicional a <i>enviarMem</i> .	resetC=1 actWrite=0
enviarMem	Se da inicio y se le cede el control a la máquina de envío de memoria. bfinMemo=0 , la máquina de envío de memoria aún no termina. Permanece en <i>enviarMem</i> . bfinMemo=1 , se ha concluido el envío de fases a la interfaz. Pasa al estado inicial <i>Espera</i> .	actEnv=1 resetC=0

Tabla 4-18. Estados presentes y futuros de la máquina de detección de fases.

		Entradas							
Estado	$Q_2Q_1Q_0$	sAO		clkEjec		sPS3		bfinMemo	
		0	1	0	1	0	1	0	1
Espera	000	000	001	000	000	000	000	000	000
CalculosSis	001	001	001	001	010	001	001	001	001
GuardaFase	010	010	010	010	010	001	011	010	010
prepararMem	011	100	100	100	100	100	100	100	100
resConta	100	101	101	101	101	101	101	101	101
enviarMem	101	110	110	110	110	110	110	110	000

Salidas				
$Q_2Q_1Q_0$	actEnv	resC	clkConta clkMemRF	actWrite
000	0	0	0	0
001	0	0	0	1
010	0	0	1	0
011	0	0	0	0
100	0	1	1	0
101	1	0	0	0

Con los datos obtenidos en la Tabla 4-18 se obtuvieron las funciones lógicas de la Tabla 4-19.

Tabla 4-19. Funciones lógicas de la máquina de detección de fases.

$T_2 = Q_1Q_0 + s'c'p'bQ_2Q_0$
$T_1 = Q_1Q_0 + s'c'p'b'Q_1 + s'cp'b'Q_2'Q_0$
$T_0 = s'c'b'Q_1 + Q_2Q_0 + Q_1Q_0 + s'c'p'bQ_2 + s'cp'b'Q_2'Q_0 + sc'p'b'Q_1'Q_0$
$actEnv = Q_2Q_1'Q_0'$
$resC = Q_2'Q_1Q_0$
$actWrite = Q_2'Q_1'Q_0$
$clkMem = clkConta = Q_2'Q_1Q_0$

Para finalizar el diseño de la Máquina de detección de fases se presentan los diagramas de conexiones y de bloques, mostrados en la Figura 4-36 y Figura 4-36 respectivamente.

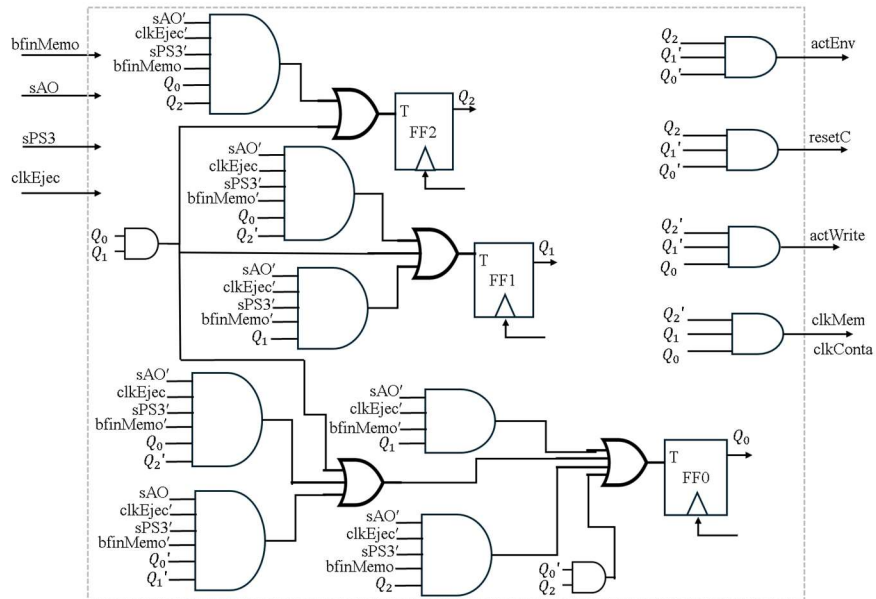


Figura 4-35. Diagrama de conexiones de la máquina de detección de fases.

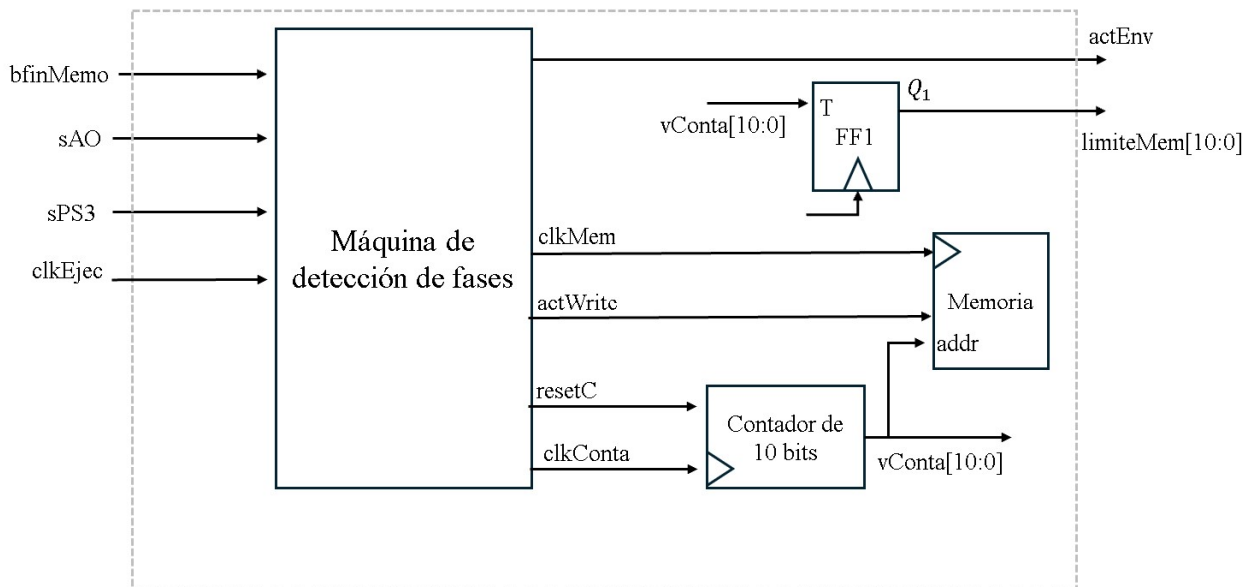


Figura 4-36. Diagrama de bloques de la máquina de detección de fases.

Por otra parte, la *Máquina de envío de memoria* se encarga de enviar las fases a la interfaz. Se diseñó esta máquina secuencial tal y como se ve en el diagrama de estados mostrado en la Figura 4-37. Al comparar el valor del contador y el almacenado en **limiteMem**, si son iguales significa que ya se han enviado todas las fases almacenadas en la memoria y se levanta la bandera **compara** que finaliza el envío de datos. En la Tabla 4-20 se muestran las señales de entrada y salida de la máquina y en la Tabla 4-6 se describen los estados y sus transiciones. Por último, se muestran los diagramas de conexiones y de bloques de la *Máquina de envío de memoria*, en la Figura 4-39 y Figura 4-38.

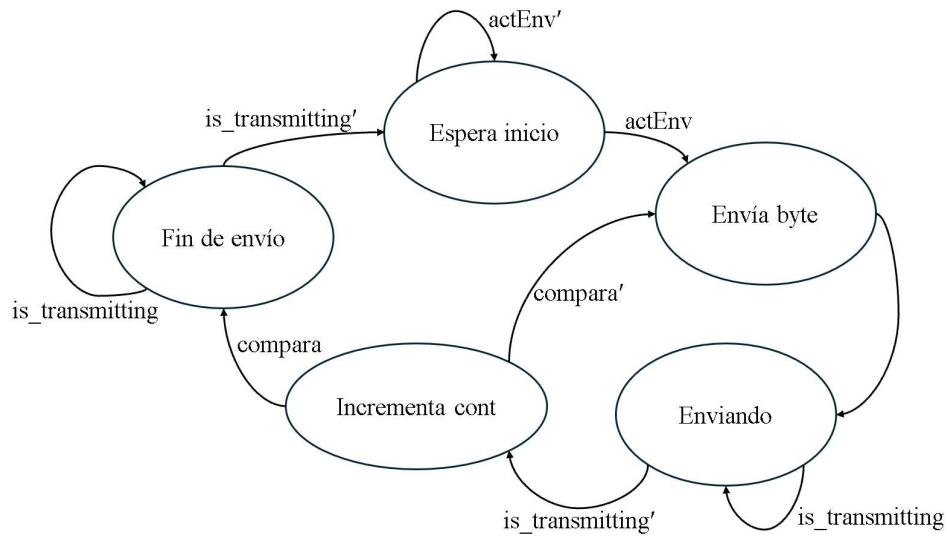


Figura 4-37. Diagrama de estados de la máquina de envío de memoria.

Tabla 4-20. Descripción de señales de la máquina de envío de memoria,

Señales		
Tipo	Nombre	Descripción
Entrada	actEnv	Señal que le indica a la máquina que comience con el envío de datos.
	is_transmitting	Se activa cuando el puerto UART está transmitiendo un dato.
	compara	Se activa cuando el contador ha llegado al valor máximo de direcciones de memoria.
	clkMem	Señal de reloj de la memoria.
	faseEnv[7:0]	Dato de salida de la memoria, es la fase que se enviará.
	actWrite	Señal que habilita la escritura de la memoria.
	limiteMem[10:0]	Contiene el dato máximo al que llegará el contador.
Salida	transmit	Inicio de transmisión del puerto UART.
	tx_byte[7:0]	Byte que se transmitirá por el UART.

	clkConta	Señal de reloj del contador. Cuando se levanta suma uno a la cuenta
	bfinMemo	Señal que indica que la máquina de envío memoria ha finalizado su tarea.

Tabla 4-21. Descripción de estados de la máquina de envío de datos.

Estados		
Nombre	Descripción y transiciones	Valores de señales de salida (las que no se muestran explícitas tienen un valor de 0).
Espera inicio	Inicia la máquina de estados, estado de reposo, en espera de señal de activación. actEnv=0 , permanece en <i>Espera inicio</i> . actEnv =1 , se comienza el envío. Pasa a <i>Envía byte</i> .	Todas las señales de salida son cero.
Envía byte	Se selecciona la fase de la memoria que corresponde a la dirección actual dada por el contador.	transmit=1 tx_byte[7:0]= faseEnv[7:0]
Enviando	Se espera a que se finalice la transmisión de la fase. is_transmitting=1 , se está enviando, permanece en <i>Enviando</i> . is_transmitting =0 , se ha terminado de enviar la fase. Pasa a <i>Incrementa cont</i> .	tx_byte[7:0]=fase[7:0] transmit=1
Incrementa cont	Incrementa el valor del contador para apuntar hacia la siguiente dirección de la memoria y se compara con limiteMem. compara=0 , aún no se llega a la cuenta máxima. Regresa a <i>Envía byte</i> . compara=1 , se ha llegado a la cuenta máxima, se han enviado todas las fases. Pasa a <i>Fin de envío</i> .	clkConta=1
Fin de envío	Estado final que indica que el envío de datos terminó. Se reinicia el contador. Y espera a que se envíe la última fase. is_transmitting=1 , aún se está enviando, se mantiene en <i>Fin de envío</i> .	resC=1 bfinMemo=1

	is_transmitting=0 , finalizó el envío, pasa al estado inicial <i>Espera inicio</i> .	
--	--	--

Tabla 4-22. Estados presentes y futuros de la máquina de envío de memoria.

		Entradas						Salidas		
Estado	$Q_2Q_1Q_0$	act_Env		is_transmitting		compara		transmit	clkConta	bfinMemo resetC
		0	1	0	1	0	1			
Espera Inicio	000	000	001	000	000	000	000	0	0	0
Enviar Byte	001	010	010	010	010	010	010	1	0	0
Enviado	010	010	010	011	010	010	010	1	0	0
I. cont	011	011	011	011	011	001	100	0	1	0
Fin de envío	100	100	100	000	100	100	100	0	0	1

De los datos de la Tabla 4-22 se obtuvieron las funciones lógicas de la Tabla 4-23. En donde se utiliza la primera letra de las señales de entrada para facilitar su lectura.

Tabla 4-23. Funciones lógicas de la máquina de envío de memoria.

$T_2 = a'i'c'Q_2 + a'i'cQ_1Q_0$
$T_1 = Q_1'Q_0 + a'i'Q_0$
$T_0 = ai'c'Q_2'Q_1' + Q_1'Q_0 + a'i'Q_0 + a'i'c'Q_1$
$transmit = Q_2'Q_1'Q_0' + Q_2'Q_1Q_0'$
$clkConta = Q_2'Q_1Q_0$
$bfinMemo = resetC = Q_2Q_1'Q_0'$

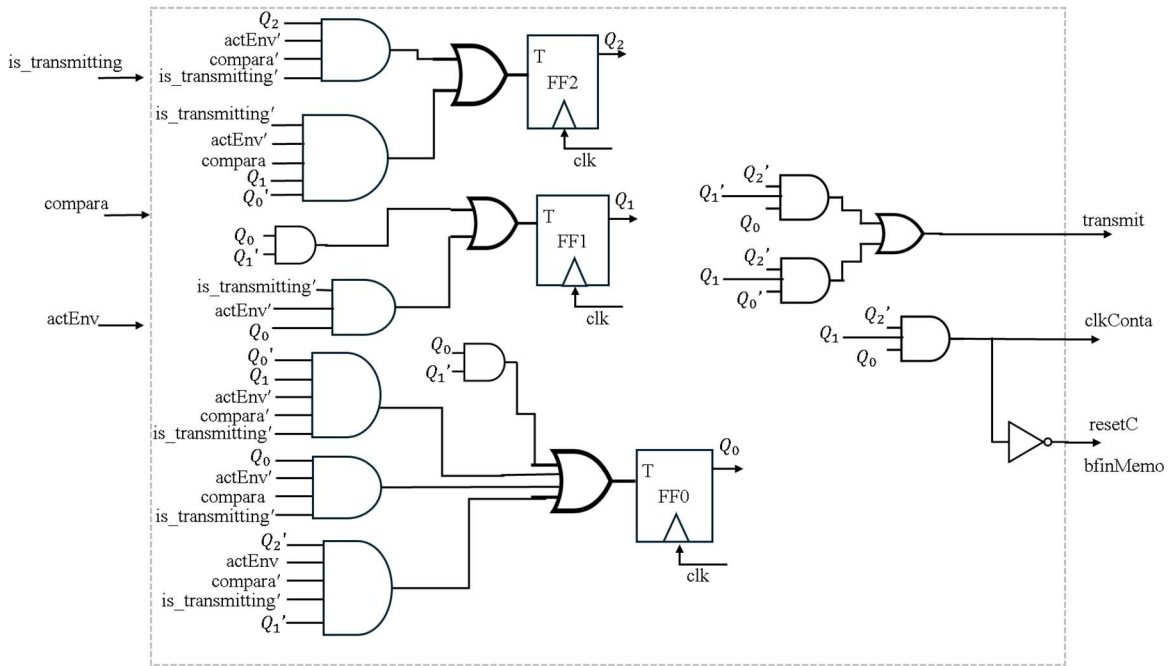


Figura 4-38. Diagrama de conexiones de la máquina de envío de memoria.

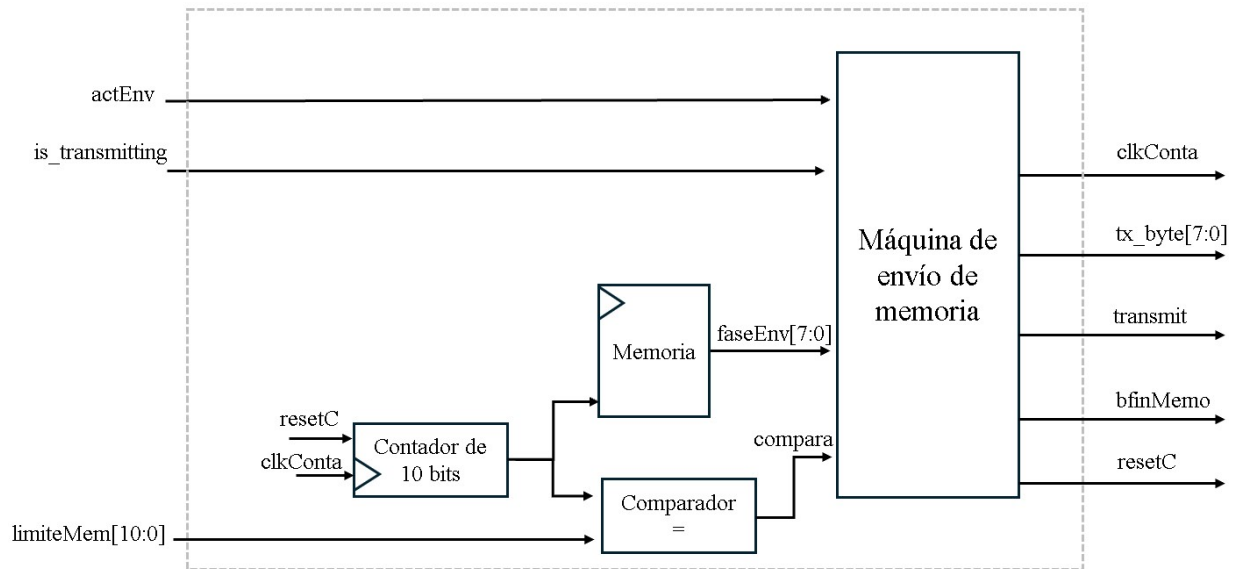


Figura 4-39. Diagrama de bloques de la máquina de envío de memoria.

Capítulo 5: RESULTADOS

Una vez terminada la etapa de diseño, se hacen pruebas del correcto funcionamiento de la interfaz y del sistema creado. En este capítulo se describen los resultados obtenidos de dichos experimentos, así como una profundización en el funcionamiento de la interfaz, ya que, se considera parte importante de este trabajo de tesis. Dado que no se cuenta físicamente con una antena de arreglo de fase, la forma de verificar el funcionamiento del sistema es comprobando que los datos enviados desde la interfaz al sistema de control manipulen las señales digitales y permitan hacer las tareas que se plantearon al comienzo de este trabajo de tesis.

Las pruebas se realizaron en una tarjeta Nexys Video de Digilent con un FPGA XC7A200T de Xilinx [8]. Esta tarjeta cuenta con 8 leds y switches que serán usados para comprobar los estados de las máquinas secuenciales. Para realizar la evaluación del funcionamiento de la máquina diseñada se utilizó un módulo de pruebas que sirve para simular el sistema de control digital sobre el que se realizó este trabajo de tesis. En la Figura 5-1 se muestra el diagrama de conexiones obtenido en el software para síntesis y análisis de diseños de lenguajes de descripción de hardware, Vivado. Este también se encuentra disponible en el Anexo 1, en donde se puede apreciar con más detalle.

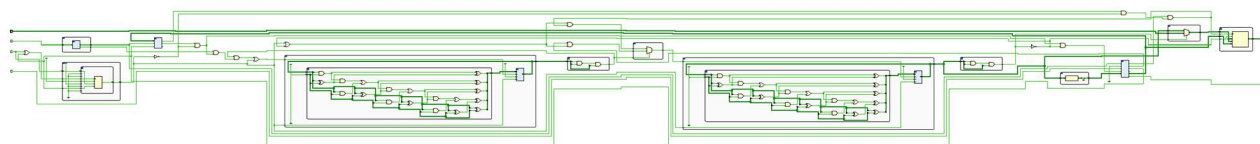


Figura 5-1. Diagrama esquemático del módulo de pruebas.

Comenzando con el primer requerimiento que se atendió en el Capítulo 4: el seguimiento sobre la trayectoria de un satélite. Para probar la capacidad de seguimiento satelital de la interfaz se realizó un experimento en donde se elige seguir el satélite mexicano PAINANI-2 en el modo de seguimiento “manual”, ver Figura 5-2. Por lo tanto, se escoge el tiempo de simulación de órbita y se procede a calcular su trayectoria, misma que se puede apreciar en la Figura 5-3. Una vez que se finaliza la simulación, se obtiene la hora y posición en donde el satélite se ubicará. También se despliega un texto que indica el tiempo que falta para que sea captado por el radar, observe Figura 5-4.

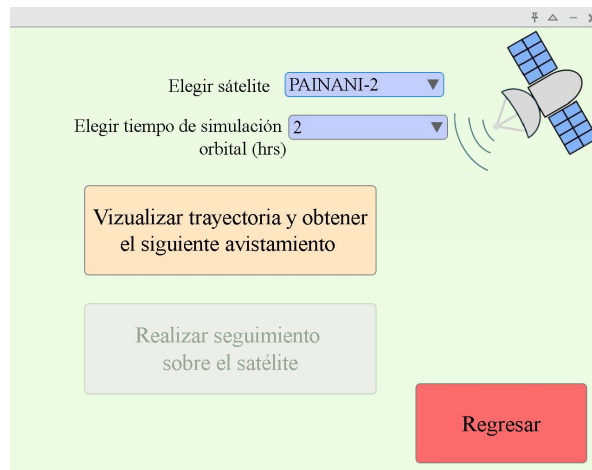


Figura 5-2. Experimento en modo manual.

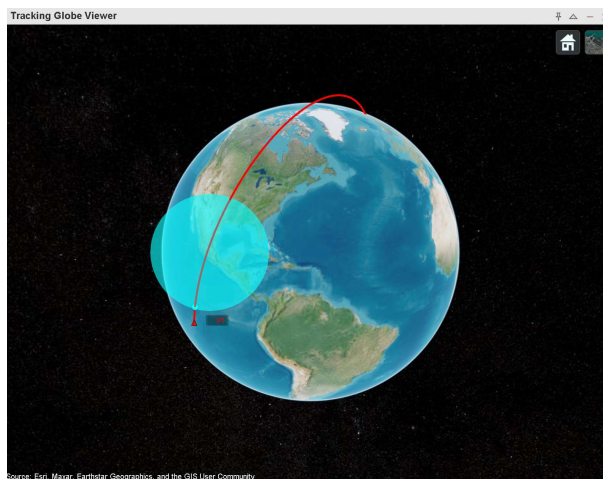


Figura 5-3. Simulación de orbita de la estación espacial.

Tiempo y posición de observación de: PAINANI-2

Fecha y hora	Elevación	Azimuth
28/03/2024 16:02:53	4.4981	16.7392
28/03/2024 16:03:03	5.3186	17.0256
28/03/2024 16:03:13	6.1771	17.3328
28/03/2024 16:03:23	7.0778	17.6636
28/03/2024 16:03:33	8.0259	18.0212
28/03/2024 16:03:43	9.0273	18.4093
28/03/2024 16:03:53	10.0888	18.8325
28/03/2024 16:04:03	11.2183	19.2962
28/03/2024 16:04:13	12.4254	19.8070
28/03/2024 16:04:23	13.7211	20.3728
28/03/2024 16:04:33	15.1185	21.0038
28/03/2024 16:04:43	16.6333	21.7124
28/03/2024 16:04:53	18.2839	22.5143
28/03/2024 16:05:03	20.0925	23.4300

Faltan 00:51 hrs para el siguiente avistamiento

Regresar

Figura 5-4. Tabla de ángulos con su respectivo tiempo, obtenidos de la simulación orbital.

Cuando faltan menos de 10 minutos para que el satélite pase cerca de nuestra posición el botón *Realizar seguimiento sobre el satélite* se habilita para que pueda ser presionado por el usuario, Figura 5-5. Nótese que antes de realizar la simulación orbital, Figura 5-2, este botón estaba deshabilitado.

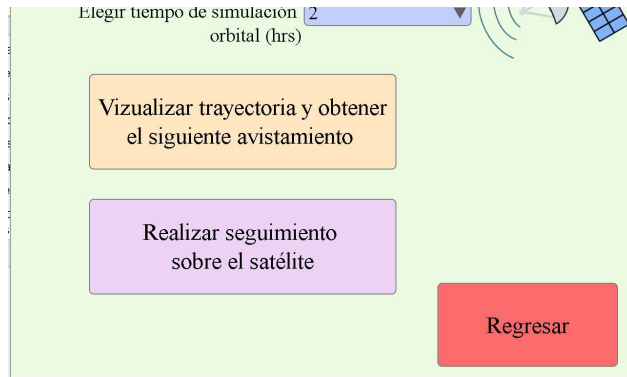


Figura 5-5. Habilitación del botón “*Realizar seguimiento sobre el satélite*”.

Al presionar el botón de *Realizar seguimiento sobre el satélite*, el programa comienza una cuenta regresiva hasta alcanzar la primera hora guardada en la tabla de la Figura 5-4, esta corresponde al momento en el que se comienza con el seguimiento del satélite. Para esto, se calcula el equivalente de los ángulos de Elevación y Azimut en radianes, posteriormente se comienzan a enviar por el puerto serial hacia el sistema de control. En la Figura 5-6 se muestra la ventana de comandos de MATLAB en donde hay dos ángulos dados en radianes que han sido enviados al sistema de control por el puerto serial de la computadora.

```
Command Window
d1 =
    0.0802

d2 =
    0.2927
```

Figura 5-6. Ventana de comandos de Matlab con los ángulos de orientación que se enviaron.

Una característica por resaltar es que mientras se realiza el envío de los ángulos de orientación al sistema de control, en la interfaz se muestra un gráfico en 2D que corresponde al paso del satélite sobre nuestra ubicación en tiempo real.



Figura 5-7. Paso del satélite PAINANI-2 sobre el punto de observación.

Por otro lado, cuando se escoge el modo de seguimiento “automático” se lleva un proceso similar al descrito anteriormente, con la diferencia que en este modo no se cambia de ventana, la interfaz obtiene una tabla como la vista en la Figura 5-4 de forma interna y deshabilita las demás funciones dejando inactivos los botones correspondientes.

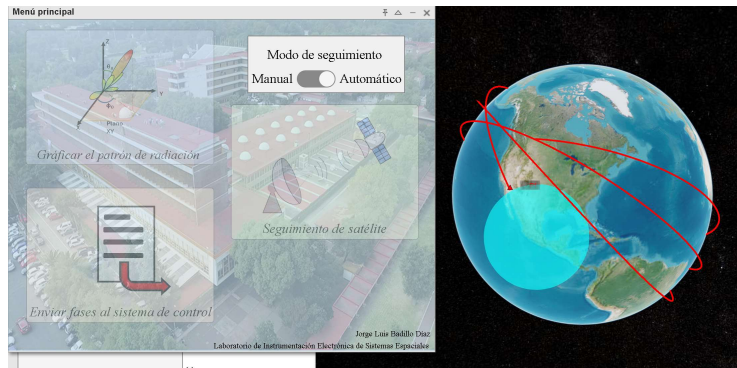


Figura 5-8. Seguimiento de satélites en modo automático.

Continuando con la atención del siguiente requerimiento, la interfaz permite el envío de fases al sistema de control. Para esto solo hay que acceder a la ventana correspondiente a través del botón “*Envío de fases al sistema de control*”. Una vez adentro, se selecciona el documento que contenga los datos que se desean enviar con ayuda del explorador de archivos que se le ha integrado a la interfaz, ver Figura 5-9.

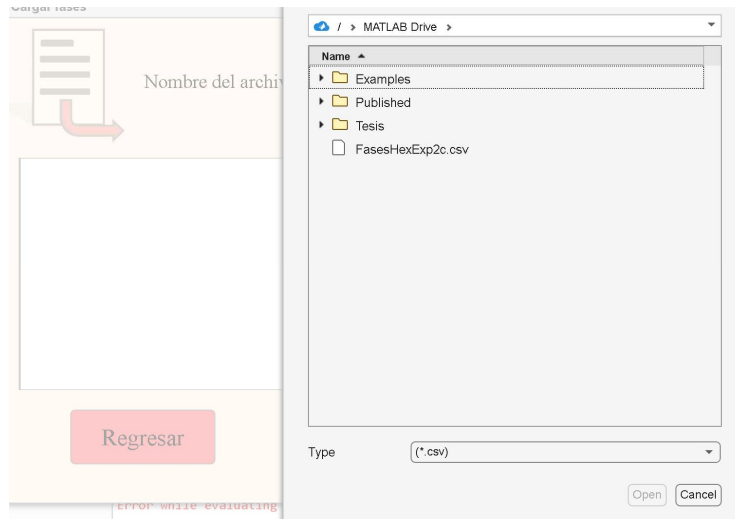


Figura 5-9. Elección de archivos en la interfaz.

Después de seleccionar el archivo deseado, su contenido se copia y se muestra en una vista previa, además, es posible codificar las fases antes de enviarlas por el puerto serial. En la Figura 5-10 se aprecia un ejemplo de lo descrito.

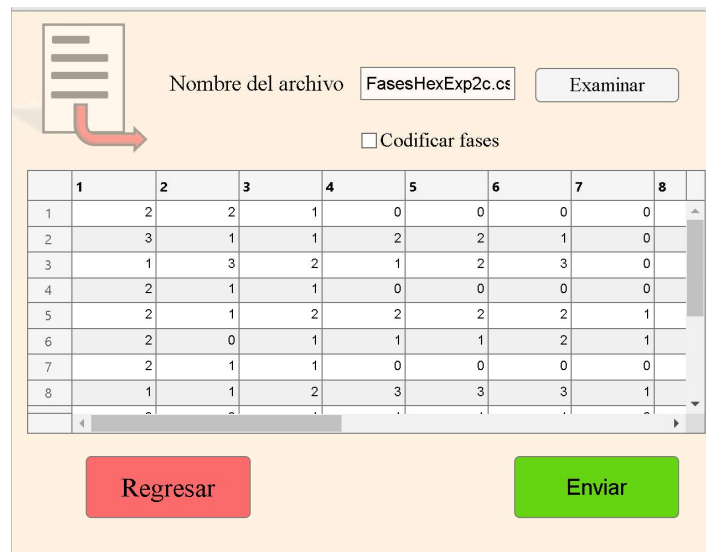


Figura 5-10. Ventana para el envío de fases.

Una vez que se comienza con el proceso de envío de fases, estas se mandan al sistema de control fila por fila. Cuando todos los datos se han transferido, se manda el código de finalización para la máquina secuencial del sistema de control.

Se realizaron pruebas de este envío con un caso de 17 fases, mismas que fueron recibidas y almacenadas con efectividad en el sistema. Para comprobar esto se analizaron las señales digitales de los datos guardados en memoria. En la Figura 5-11, se muestra la alternancia entre un 1 y un 3 para una diferente dirección en la memoria.

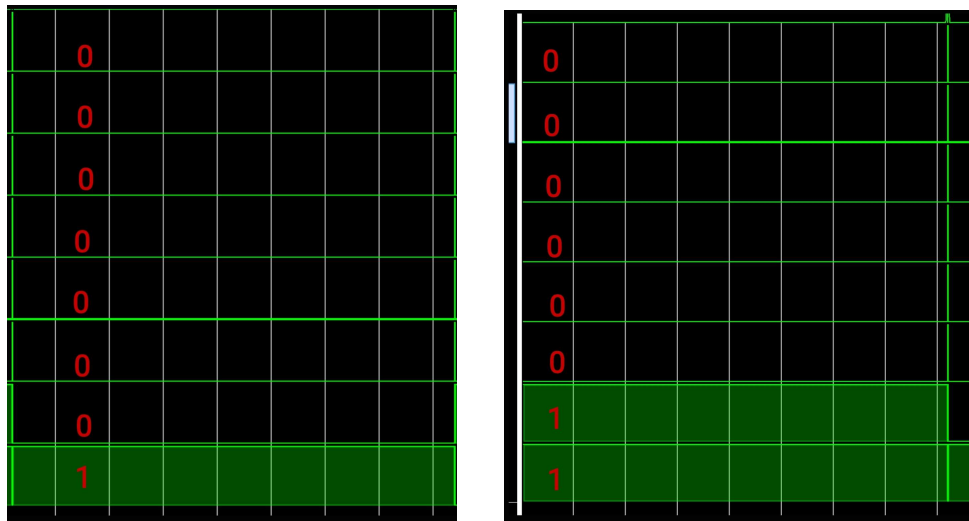


Figura 5-11. Recibimiento de fases en el FPGA.

El último requerimiento de esta interfaz es graficar el patrón de radiación de la antena, dado por las fases que el sistema de control ha calculado. Entonces, al acceder a la pestaña de la gráfica se transmite el comando que activa el envío de estos datos hacia la interfaz. Ya recibidos se calcula el patrón de radiación con la (4.1), con la que se obtiene la energía de cada punto dentro del espacio de 360° en φ y 60° en θ .

Para este experimento también se ocupó un arreglo de 17 fases que fueron enviadas desde el FPGA. Estas se reciben por el puerto serial y se comprueba el valor de cada una en la ventana de comandos de MATLAB, ver Figura 5-12.

```
fases =
    3     1     0     0     0     0     1     2     1     0     3     2     2     3     1     1     0
>>
```

Figura 5-12. Fases recibidas por la interfaz.

Para finalizar se muestra el patrón de radiación calculado gracias a las fases que se recibieron desde el sistema de control. Este se puede observar en la Figura 5-13, en dónde se aprecia su lóbulo principal con los puntos de máxima energía resaltados de color rojo.

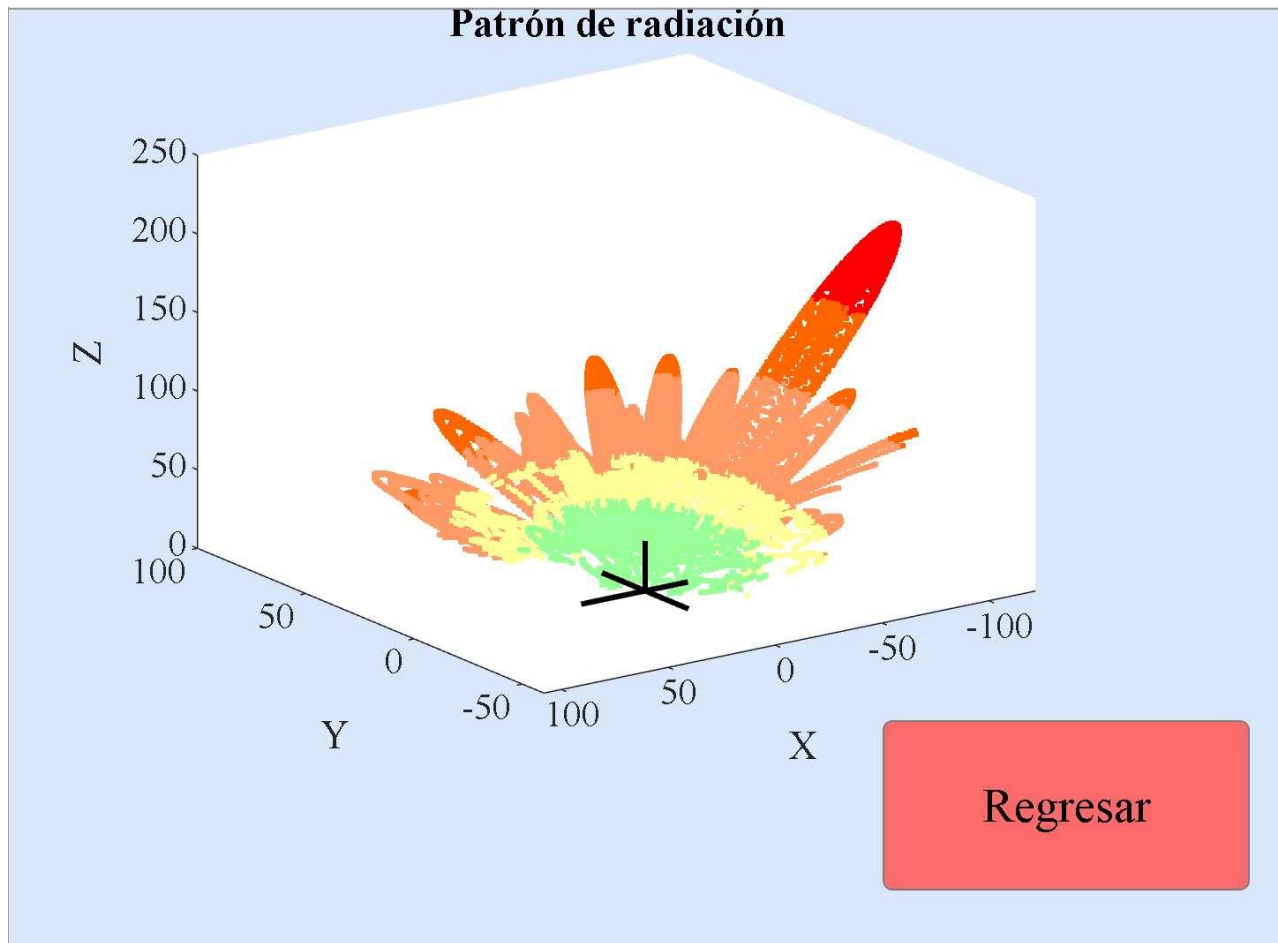


Figura 5-13. Patrón de radiación calculado en la interfaz.

Con esto se finaliza la comprobación del funcionamiento de la interfaz y de los módulos en HDL que se diseñaron. Como ya se mencionó en varias ocasiones, esta interfaz fue desarrollada en MATLAB, por lo tanto, queda el código abierto por si se desea expandir este sistema a futuro o si algún académico o estudiante del nivel superior desea usarlo.

En el capítulo siguiente se realizarán las conclusiones de este trabajo de tesis. Comparando los objetivos planteados al comienzo y los resultados obtenidos.

Capítulo 6: CONCLUSIONES

Se diseñó y construyó una interfaz capaz de enviar y recibir datos de un sistema de control digital para antenas en arreglos de fase implementado en un FPGA. Fue necesario hacer una revisión detallada del sistema de control digital, diseñado y construido en el trabajo de tesis de maestría del M.I. Hernández Gastaldi, para desarrollar los módulos en descripción de hardware que permitan una comunicación entre el sistema y la interfaz.

Al diseñar la interfaz se basó en que el usuario tuviera facilidad de acceder a las funciones dadas por esta, haciendo uso de botones y seleccionar elementos de listas. A pesar de esto, al acceder al código es posible cambiar características visuales y de funcionamiento para mejorar y/o expandir esta interfaz.

El sistema de control calcula las fases necesarias para hacer que el haz de una antena de arreglos de fase se oriente hacia una dirección dada, en un tiempo de respuesta de aproximadamente un microsegundo. Uno de los requisitos para este trabajo de tesis fue añadirle al sistema un módulo descrito en VHDL que le permitiera recibir estos datos desde un medio externo, en este caso la interfaz. Para capturar estas tramas digitales se diseñó e implementó la *máquina de recepción de fase*. Además, se construyó la *máquina de distribución de fases*, encargada de programar al sistema de control para que no envíe los datos calculados, solamente los que se proporcionen en la interfaz. Dentro de esta es posible seleccionar un archivo que contenga los datos que se desean enviar al sistema de control para que este reoriente el haz de la antena. A pesar de que la implementación de todo esto hace que el tiempo de respuesta sea más lento, resulta útil obtener una orientación con la mayor exactitud posible.

Respecto al movimiento del haz de la antena sobre la trayectoria de un satélite, se logró implementar en MATLAB un algoritmo de predicción de la órbita satelital. Con los datos obtenidos de la posición del satélite se determina la hora y posición en la que será observable por una estación terrena ubicada en Ciudad Universitaria, teniendo en cuenta que ahí se encuentra la antena en arreglo de fase. La interfaz de usuario se encarga de transmitir continuamente los datos de la posición del satélite, dada por los ángulos azimut y elevación, al sistema de control. Dentro de este se diseñó e implementó un módulo descrito en VHDL que permite la recepción de estos datos en una trama serial. Una vez recibidos, el sistema calcula las fases necesarias para orientar el haz de la antena hacia donde se encuentra ubicado el satélite en ese momento. En el diseño de este requerimiento tomó mayor relevancia el desarrollo del algoritmo de predicción de la órbita, ya que, era necesario obtener la posición correcta del satélite, así como determinar el rango de tiempo en el que fuera visible por la estación terrena y de esta manera lograr un seguimiento más realista.

Otro de los requerimientos fue la creación de un sistema que pudiera graficar el patrón de radiación de la antena con datos dados por el sistema de control. Se añadió un apartado gráfico en la interfaz para que se pudiera observar esta caracterización de la forma en que las ondas electromagnéticas

son reflejadas por la antena, controlada por las fases que se calculan en el sistema. Para lograr el envío de estos datos se diseñaron y se describieron en VHDL dos máquinas secuenciales. La primera se encarga de guardar todas las fases que calcula el sistema de control, y la segunda transmite estas fases de forma serial hacia la interfaz. Dentro de esta se realizan los cálculos necesarios para obtener el patrón de radiación. Fue importante considerar todas las señales que el sistema calcula para obtener una correcta representación del lóbulo principal. A pesar de que se muestra energía dispersa en otras direcciones, lóbulos secundarios y traseros, el haz de mayor magnitud se concentra en la dirección a la que la antena está apuntando. Es importante conocer la orientación de esta, ya que de esa manera se comprueba el correcto funcionamiento del sistema de control.

En conjunto con la interfaz y los módulos en descripción de hardware se lograron alcanzar todos los objetivos planteados al comienzo de este escrito de tesis. Cabe resaltar que con el desarrollo de un entorno en dónde estos componentes se comuniquen entre sí, se forma una red de transferencia de datos que puede iniciar o concluir en una interfaz, de modo que se facilite la operación de diseños en hardware o mostrar el comportamiento de sistemas en forma gráfica y en tiempo real. Por lo tanto, se considera que la implementación de interfaces graficas que controlen o monitoreen sistemas de bajo nivel es un tema escalable y con mucha posibilidad de aplicación.

Capítulo 7: REFERENCIAS

- [1] A. L. Castellanos, «Antenas,» Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla.
- [2] E. Hernández, Diseño de un sistema de control digital escalable y reconfigurable para antenas en arreglo de fase, Ciudad de México: UNAM, 2023.
- [3] K. Benson, «ANALOG DEVICES,» 1 Enero 2019. [En línea]. Available: <https://www.analog.com/en/resources/analog-dialogue/articles/phased-array-beamforming-ics-simplify-antenna-design.html>. [Último acceso: 31 Enero 2014].
- [4] J. Munn, «Introduction to Efficiently Modeling Antennas in COMSOL Multiphysics,» COMSOL, 31 Octube 2016. [En línea]. Available: <https://www.comsol.com/blogs/introduction-to-efficiently-modeling-antennas-in-comsol-multiphysics/>. [Último acceso: 31 Enero 2014].
- [5] «Terra (EOS/AM-1),» eoPortal, 18 07 2012. [En línea]. Available: <https://www.eoportal.org/satellite-missions/terra#terra-mission-eosam-1>. [Último acceso: 25 03 204].
- [6] A. Nava, «Ponen en órbita satélite mexicano,» Excelsior, 02 07 2021. [En línea]. Available: <https://www.excelsior.com.mx/nacional/ponen-en-orbita-nanosatelite-mexicano/1457996>. [Último acceso: 25 03 2024].
- [7] «International Space Station,» NASA, [En línea]. Available: “International Space Station - NASA”. NASA. Accedido el 25 de marzo de 2024. [En línea]. Disponible: <https://www.nasa.gov/reference/international-space-station/>. [Último acceso: 24 03 2024].
- [8] Xilinx, «Nexys Video FPGA: Trainer Board for Multimedia Applications,» Xilinx, 2024. [En línea]. Available: <https://www.xilinx.com/products/boards-and-kits/1-cfdwic.html>. [Último acceso: 28 03 2024].
- [9] «Learn About the AC Phase Difference,» EEWeb, 28 Febrero 2012. [En línea]. Available: <https://www.eeweb.com/learn-about-the-ac-phase-difference/>. [Último acceso: 31 Enero 2024].
- [10] Pardo, Fernando; Boluda, José, VHDL Lenguaje para síntesis y modelado de circuitos, Madrid: RA-MA, 1999.

Anexo 1

