

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



PROYECTO FIN DE CARRERA

Ingeniería de Telecomunicación

**SISTEMA DE CONTROL SERIE PARA APUNTAMIENTO
ELECTRÓNICO DE ANTENAS PLANAS PARA
RECEPCIÓN DE SATÉLITE DBS**

Melani Corral Baltasar

Julio de 2015

**SISTEMA DE CONTROL SERIE PARA APUNTAMIENTO
ELECTRÓNICO DE ANTENAS PLANAS PARA
RECEPCIÓN DE SATÉLITE DBS**

AUTOR: Melani Corral Baltasar

TUTOR: Guillermo González de Rivera



Dpto. Tecnología Electrónica y de las Comunicaciones

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Julio de 2015

Resumen

Este proyecto tiene como objetivo controlar el apuntamiento de antenas planas utilizadas como receptoras de sistemas DBS, mediante un sistema de control electrónico.

En un principio, se diseña, construye y verifica una placa de control manual que permitirá conocer cómo se controla la antena. Se realizan pruebas que posteriormente permitirán comprobar si el control electrónico es eficiente.

En segundo lugar, se diseña, construye y verifica la placa principal de este proyecto, por medio de la cual se realiza el control serie. Esta placa da comodidad y facilidad al usuario a la hora de controlar la antena.

La tercera placa a tener en cuenta en este PFC, es la placa de RF. Ésta contiene los desfases digitales, necesarios para construir la antena plana. Esta placa de RF no se ha construido en este proyecto, pero sí se ha participado en su diseño.

Finalmente, para hacer más accesible el sistema, también se diseña en este PFC un interfaz software (firmware) que permite que el usuario pueda acceder al sistema únicamente modificando ciertos parámetros a través del PC, sin necesidad de conocer los detalles de bajo nivel.

Las pruebas de verificación de las placas de control se realizan en primer lugar con instrumental disponible en el laboratorio de investigación del Grupo HCTLab, y posteriormente, se realizan las pruebas del sistema completo, con un analizador de redes y en una cámara anecoica (sala que elimina la reflexión de cualquier tipo de onda), para ver que el desfase se realiza correctamente y que no existen parámetros que afecten a la radiación de las señales de radiofrecuencia por parte de los sistemas de control.

Palabras clave

Antena, array, microcontroladores, USB, variación de fase, automatización, diseño PCB, array de desfase.

Abstract

This project gets the control the pointing of planar antennas used as receptors of DBS systems by an electronic control.

First, it designs, builds and verifies manual control, which will reveal how the antenna is controlled, and testing, which subsequently possible to check whether the electronic control is efficient.

Second, it is designed, built and verified, the project's main board, through which the serial or electronic control is performed, and allow comfort and ease to the user in controlling the antenna.

Later, the RF board is designs and builds, which contains the digital phase shifters, necessary to construct the planar antenna. The RF board hasn't been built in this project, but if your design has participated.

Finally, to make the system more accessible, it is also designed in this PFC, a software (firmware) interface that allows the user to access the system, only modifying soe parameters throughout the PC, without having to know the details of low-level system.

The inicial testing is make using the instrumental avalible on the HCTLab's group investigation, then, the entire system is testing in a network analyzer and an anechoic chamber (a room which eliminate the waves's reflexion), to see that the gap is successful and there are no parameters that affect the radiation of radio frequency signals by control systems.

Key Words

Antenna, array, microcontrollers, USB, phase shifter, automatization, PCB design, phased array.

Agradecimientos

Me gustaría agradecer mi trabajo, en primer lugar, a toda mi familia tanto a mis padres y a mi hermana, como a mis abuelos, tíos y primos, incluso a aquellos que se encuentran lejos y día a día se han preocupado por cómo iba desarrollando mi proyecto.

A mis amigas y mis compañeras de piso, que han tenido que aguantar el que no quiera salir de mi habitación por dar vueltas y vueltas a algún diseño o programa, pero también cuando algo me salía bien y quería celebrarlo.

A mi tutor Guillermo, por tantas veces que le he interrumpido en su despacho con a veces solo chorradas, y que siempre ha estado ahí, y porque ha tenido que soportar llevarse algún chispazo en la cámara anecoica, por resolver problemas.

A Alberto Sanchez, profesor del HCTLab, que gracias a él, he podido desarrollar la parte software del proyecto, que tan malos ratos me hizo pasar en un principio. Y por último, a Jose Luis Masa, que aunque no a mi directamente, si ha dado mucha ayuda a Adrian Cobos, que muchas veces me beneficiaba a mí.

INDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Motivación.....	1
1.2	Objetivos.....	2
1.3	Organización de la memoria.....	3
2	Estado del arte	5
2.1	Sistemas de control.....	5
2.2	Antenas planas en sistemas DBS.....	8
	2.2.1 <i>Conceptos básicos</i>	8
	2.2.2 <i>Antenas planas o microstrip</i>	9
	2.2.3 <i>Sistema DBS</i>	11
2.3	Algunos sistemas de control para antenas en la actualidad.....	11
	2.3.1 <i>Módulo para el control y posicionamiento de antenas formadas por rotores Yaesu G5400 y G5500</i>	11
	2.3.2 <i>Control automático para la red satelital VSAT-ICE</i>	13
	2.3.3 <i>Módulo para el control de apuntamiento de un conjunto de antenas planas para un sistema WiMAX 3.5 GHz.</i>	13
2.4	Protocolo USB.....	14
	2.4.1 <i>Introducción</i>	14
	2.4.2 <i>Tipos</i>	14
	2.4.3 <i>Aplicaciones USB</i>	15
	2.4.4 <i>Protocolo USB CDC</i>	16
	2.4.4.1 <i>Introducción</i>	16
	2.4.4.2 <i>Características del CDC Clase y limitaciones</i>	17
4	Diseño.....	19
3.1	Introducción.....	19
4.1	Control manual	19
4.2	Control electrónico	20

4.3 Placa de RF.....	20
4.4 Protocolo de comunicación entre microcontrolador y antena microstrip.....	20
4.4.1 Protocolo SPI.....	20
4.4.2 Salidas digitales	21
4.5 Protocolo de comunicación entre PC y la placa de control	21
5 Desarrollo	23
5.1 Desarrollo hardware	23
5.1.1 Placa de control manual.....	23
5.1.2 Placa de control electrónico.....	29
5.1.2.1 Topologías de conexión entre la placa de control serie y la placa de RF....	30
5.1.2.2 Diseño placa control electrónica.....	32
5.1.3 Placa de RF.....	37
5.2 Programación del microcontrolador en Atmel Studio.....	40
5.2.1 <i>Programa de control de placa de RF</i>	41
5.3 Desarrollo software.....	43
5.3.1 <i>Introducción a LUFA</i>	44
5.3.2 Demo VirtualSerial de LUFA en Linux	46
5.3.3 Control del sistema de apuntamiento con una terminal windows	49
5.3.3.1 Control mediante una terminal disponible en la red.....	50
5.3.3.2 Interfaz gráfica personalizada en Java Eclipse	53
5.3.3.3 Modificaciones necesarias para utilizar un array de 16 desfasadores	60
6 Integración, pruebas y resultados	61
6.1 Calibración	61
6.2 Pruebas	63
6.2.1 <i>Pruebas control manual en analizador de redes</i>	63
6.2.1.1 Resultados obtenidos con control manual	64
6.2.2 Pruebas control serie en analizador de redes	66
6.2.2.1 Control serie para placa con un desfasador digital	66

6.2.2.2 Control serie para placa phased array	69
7 Conclusiones y trabajo futuro.....	75
7.1 Conclusiones.....	75
7.2 Trabajo futuro	75
Referencias	77
Anexos.....	79
A Datasheet	79
B Esquemáticos	84
C Archivos GERBERS	87

INDICE DE FIGURAS

FIGURA 1. DISEÑO PHASED ARRAY.....	3
FIGURA 2. ESQUEMA BÁSICO DE UN SISTEMA DE CONTROL.....	5
FIGURA 3. FORMATO BÁSICO DE UNA ANTENA MICROSTRIP	10
FIGURA 4. CIRCUITO CONTROLADOR CON MICROCONTROLADOR PIC 16F688	12
FIGURA 5. SÍMBOLO USB	14
FIGURA 6. TIPOS DE CONECTORES USB	15
FIGURA 7. RELACIÓN ENTRE UN DISPOSITIVO Y UN SISTEMA HOST	16
FIGURA 8. IDEA DE DISEÑO DEL SISTEMA COMPLETO	19
FIGURA 9. PROTOCOLO DE COMUNICACIÓN SPI	21
FIGURA 10 SOLUCIÓN ESQUEMA FINAL PROPUESTO DEL SISTEMA COMPLETO.....	22
FIGURA 11. FLUJO DE DESARROLLO DEL SISTEMA.....	23
FIGURA 12. DISEÑO COMPONENTE (I)	24
FIGURA 13. DISEÑO COMPONENTE (II).....	24
FIGURA 14. DISEÑO COMPONENTE (II).....	25
FIGURA 15. DISEÑO COMPONENTE (IV)	25
FIGURA 16. DISEÑO COMPONENTE (V).....	26
FIGURA 17. SWITCH PLACA DE CONTROL MANUAL.....	26
FIGURA 18. REGULADOR MC7805 PLACA DE CONTROL MANUAL	27
FIGURA 19. REGULADOR MC7905 PLACA DE CONTROL MANUAL	27
FIGURA 20. CONECTOR DE CONEXIÓN A LA PLACA RF.....	27
FIGURA 21. CLEMA DE ALIMENTACIÓN.....	28
FIGURA 22. PLACA CONTROL MANUAL O PARALELO	28
FIGURA 23. ESQUEMA DE CONEXIÓN ENTRE LA PLACA CONTROL PARALELO Y PLACA RF	28

FIGURA 24. ALTERNATIVA 1 PROTOCOLO DE COMUNICACIÓN ARRAY (I).....	30
FIGURA 25. ALTERNATIVA 1 PROTOCOLO DE COMUNICACIÓN ARRAY (II)	31
FIGURA 26. ALTERNATIVA 2 PROTOCOLO DE COMUNICACIÓN ARRAY	31
FIGURA 27. ESQUEMÁTICO ATMEGA32U4	32
FIGURA 28. ESQUEMÁTICO CONECTOR AVRSPi MKII DE LA PLACA PRINCIPAL	33
FIGURA 29. ESQUEMÁTICO CONECTOR MINI USB DE LA PLACA PRINCIPAL	33
FIGURA 30. SELECTOR DE ALIMENTACIÓN DE LA PLACA PRINCIPAL	33
FIGURA 31. ESQUEMÁTICO LEDS Y PULSADORES PLACA PRINCIPAL	34
FIGURA 32. CONECTORES GENÉRICOS PLACA PRINCIPAL	34
FIGURA 33. PLACA DE CONTROL ELECTRÓNICO PRINCIPAL.....	35
FIGURA 34. PLACA DE INTERFAZ DE CONTROL.....	35
FIGURA 35. ESQUEMA DE CONEXIÓN ENTRE LA PLACA DE CONTROL SERIE Y LA PLACA RF.....	36
FIGURA 36. PLACA RF EN SOPORTE CON UN DESFASADOR DIGITAL.....	38
FIGURA 37. PLACA RF CON 3 PHASED ARRAY (PARTE DE ADELANTE)	38
FIGURA 38. PLACA RF CON 3 PHASED ARRAY (PARTE DE ATRÁS)	39
FIGURA 39. FRESADORA CIRCUITOS IMPRESOS.....	39
FIGURA 40. PLACA MICROCONTROLADOR ATMEGA32U4 (PLACA PRINCIPAL).....	40
FIGURA 41. PROGRAMADOR AVRISP MKII	40
FIGURA 42. LOGOTIPO ATMEL STUDIO	40
FIGURA 43. OSCILOSCOPIO AGILENT MSO-X-3014A	42
FIGURA 44. RESULTADO OSCILOSCOPIO DE ENVÍO DE DATOS A LA PLACA RF	43
FIGURA 45. DIAGRAMA DE FLUJO SOFTWARE.....	44
FIGURA 46. ENDPOINT PUERTO SERIE	45
FIGURA 47. VID Y PID	46
FIGURA 48. ICONO ARCHIVO .INF	49

FIGURA 49. ACTUALIZACIÓN CONTROLADOR	50
FIGURA 50. TERMINAL REALTERM (I)	51
FIGURA 51. TERMINAL REALTERM (II).....	52
FIGURA 52. SEÑALES OSCILOSCOPIO ENVÍO DE DATOS CON REALTERM.....	52
FIGURA 53. DESCARGA JAVA	53
FIGURA 54. LOGOTIPO JAVA ECLIPSE	54
FIGURA 55. INSTALACIÓN WINDOWS BUILDER (I).....	54
FIGURA 56. INSTALACIÓN WINDOWS BUILDER (II)	54
FIGURA 57. EXTERNAL JARs ECLIPSE(I)	55
FIGURA 58. EXTERNAL JARs ECLIPSE(II).....	56
FIGURA 59. EXTERNAL JARs ECLIPSE(III)	56
FIGURA 60. INTERFAZ USUARIO, MENÚ PRINCIPAL	57
FIGURA 61. INTERFAZ USUARIO, 3 PHASED ARRAY	57
FIGURA 62. INTERFAZ USUARIO, 16 PHASED ARRAY	58
FIGURA 63. INTERFAZ USUARIO ENVÍO A LA PLACA RF 3 PHASED ARRAY	58
FIGURA 64. SEÑALES OSCILOSCOPIO ENVÍO DE DATOS CON INTERFAZ PERSONAL.....	59
FIGURA 65. MODIFICACIÓN ARCHIVO VIRTUALSERIAL.C PARA ANTENA DE 16 PHASED ARRAY (I) 60	
FIGURA 66. MODIFICACIÓN ARCHIVO VIRTUALSERIAL.C PARA ANTENA DE 16 PHASED ARRAY (II)	60
FIGURA 67. CAPTURA ARRAY DE ENVÍO, CÓDIGO JAVA.....	60
FIGURA 68. ANALIZADOR DE REDES AGILENT E5071C.....	61
FIGURA 69. CÁMARA ANECOICA.....	62
FIGURA 70. ESQUEMA DE MONTAJE CONTROL MANUAL EN ANALIZADOR DE REDES	63
FIGURA 71. MONTAJE ANALIZADOR DE REDES CON CONTROL PARALELO Y PLACA RF CON UN DESFAZADOR DIGITAL.....	63
FIGURA 72. TRANSMISIÓN (S21) CON PLACA DE CONTROL PARALELO.....	64

FIGURA 73. REFLEXIÓN (S11) CON PLACA DE CONTROL PARALELO	64
FIGURA 74. DESFASE CON REFERENCIA AL ESTADO 0 DE LA PLACA DE CONTROL PARALELO.....	65
FIGURA 75. ESQUEMA DE CONEXIÓN CONTROL SERIE Y ANALIZADOR DE REDES.....	66
FIGURA 76. MONTAJE ANALIZADOR DE REDES CON CONTROL SERIE Y PLACA RF CON UN DESFAZADOR DIGITAL.....	66
FIGURA 77. TRANSMISIÓN (S21) CON PLACA DE CONTROL SERIE Y UN DESFAZADOR DIGITAL.....	67
FIGURA 78. REFLEXIÓN (S11) CON PLACA DE CONTROL SERIE Y UN DESFAZADOR DIGITAL.....	67
FIGURA 79. DESFASE CON REFERENCIA AL ESTADO 0 DE LA PLACA DE CONTROL SERIE Y UN DESFAZADOR DIGITAL.....	68
FIGURA 80. ESQUEMA DE CONEXIÓN DE CONTROL SERIE CON ANALIZADOR DE REDES Y 3 PHASED ARRAY.....	69
FIGURA 81. MONTAJE ANALIZADOR DE REDES CON CONTROL SERIE Y PLACA RF CON UN 3 PHASE ARRAY.....	69
FIGURA 82. TRANSMISIÓN (S21) CON PLACA DE CONTROL SERIE EN ARRAY (DESFAZADOR 1).....	70
FIGURA 83. REFLEXIÓN (S11) CON PLACA DE CONTROL SERIE EN ARRAY(DESFAZADOR 1).....	70
FIGURA 84. TRANSMISIÓN (S21) CON PLACA DE CONTROL SERIE EN ARRAY (DESFAZADOR 2).....	71
FIGURA 85. REFLEXIÓN (S11) CON PLACA DE CONTROL SERIE EN ARRAY (DESFAZADOR 2).....	71
FIGURA 86. TRANSMISIÓN (S21) CON PLACA DE CONTROL SERIE EN ARRAY (DESFAZADOR 3).....	72
FIGURA 87. REFLEXIÓN (S11) CON PLACA DE CONTROL SERIE EN ARRAY (DESFAZADOR 2).....	72
FIGURA 88. DESFASE PARA DESFAZADOR 1 DEL ARRAY	73
FIGURA 89. DESFASE PARA DESFAZADOR 2 DEL ARRAY	73
FIGURA 90. DESFASE PARA DESFAZADOR 3 DEL ARRAY	74

INDICE DE TABLAS

TABLA 1. RANGO DE TENSIÓN	29
TABLA 2. RANGO DE FRECUENCIA.....	29
TABLA 3. ESTADOS DIGITALES RESPECTO A LA FASE.....	37

1 Introducción

En la actualidad existen multitud de equipos que necesitan de un sistema de control que permita manejarlos. Un sistema de control consiste en un conjunto de componentes mecánicos, hidráulicos, eléctricos y/o electrónicos que, interconectados, permiten modificar un proceso para alcanzar el resultado deseado.

En un principio la mayor parte de los sistemas de control existentes eran manuales, es decir, un sistema manual necesita la intervención de una persona en el sistema, que es quién tendría que realizar las modificaciones necesarias, para en este caso modificar el desfase producido en la antena en cada momento. Esto se haría, por ejemplo, por medio de pulsadores, y resulta un sistema poco eficiente, sobre todo en el caso del control de antenas, puesto que sería necesario, que las personas tuvieran incluso que subirse al tejado para modificar su fase.

En la actualidad, se han ido haciendo cada vez más imprescindibles y necesarios los sistemas de control automático, los cuales “operan por sí solos”, esto es, efectuando los cambios necesarios por parte del ser humano pero no directamente en los equipos finales, sino en los dispositivos tecnológicos. Y esto es lo que se consigue en este proyecto.

Además, conociendo que existe una gran necesidad de que el ser humano pueda interactuar de forma sencilla y a distancia con los sistemas tecnológicos sin tener que conocer su funcionamiento, también es necesario desarrollar aplicaciones software que nos permitan dar accesibilidad al sistema. Esta actuación se podrá realizar modificando unas opciones, por ejemplo a través de un PC."Para esto sería necesario:

- Un componente hardware.
- Un medio de comunicación, por ejemplo, USB.
- Una arquitectura (software y firmware).

1.1 Motivación

El grupo de investigación RFCAs (RadioFrecuencia, Circuitos, Antenas y Sistemas) de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid, ha desarrollado una antena plana de tipo phased array con apuntamiento variable diseñada y construida en el proyecto *“Diseño de un phased array lineal de parches sobre guía de onda para comunicaciones por satélite DBS”* realizado por Jesús Vázquez [1], el control de esta antena se realizaba de forma manual por medio de selectores. Actualmente, Adrián Cobos en su proyecto *“Integración de desfases digitales en un array lineal de parches sobre guía de onda para comunicaciones satélite DBS”*, busca intentar mejorar las características de radiación de una antena de las mismas características, así el grupo RFCAs ve necesario modificar el sistema de control manual de la antena por un control automático, por comodidad a la hora de hacer pruebas y para su uso final. Ya que el grupo de radiofrecuencia no dispone de los equipos y conocimientos necesarios de electrónica, es el grupo de investigación HCTLab de la Escuela Politécnica Superior de la Universidad

Autónoma de Madrid, quien se encarga de conseguir el control electrónico de la antena, concretamente en este PFC.

Por lo tanto, en este proyecto se busca conseguir un sistema de control electrónico, que permita el control de antenas planas con un diseño phased array lineal de parches sobre guía de onda para satélites DBS, ya que como se ha mencionado estos sistemas de control son más eficientes en la actualidad, y permiten un mejor control por parte de cualquier usuario, sin que este tenga la necesidad de tener conocimientos electrónicos. Para ello, además se implementa un firmware, que completará el sistema de control electrónico diseñado.

Para conseguir el control electrónico se tienen en cuenta los sistemas de control desarrollados en el PFC *“Sistemas de Control de Apuntamiento electrónico de una agrupación de antenas planas reconfigurables operando en el sistema WIMAX a 3.5 GHz”* diseñado por Alfredo Manuel de Castro [2], dicho sistema fue construido en el proyecto *“Integración de una antena plana TX/RX de apuntamiento variable con desfasadores digitales y matrices de Butler para comunicaciones WiMAX a 3.5 GHz”* por Colino RJL [3], en el que se conseguía automatizar el apuntamiento de una antena y sirve como referencia a la hora de tomar decisiones.

Además para dar utilidad al sistema, se implementa un firmware que permite configurar la antena a través de un PC, logrando:

- Aprovechar las capacidades de procesamiento, y almacenamiento del PC.
- Mayor interacción con el mundo físico.
- Hacer más accesibles los dispositivos por parte del usuario.
- Aumentar la potencialidad de los dispositivos.

1.2 Objetivos

Desarrollo de la parte hardware y software necesarias para implementar un sistema de control que permita facilitar la comunicación con dispositivos electrónicos, por medio de USB, eliminando así la necesidad de controlar el sistema manualmente. Y por tanto, aumentando la potencialidad de los dispositivos electrónicos disponibles.

La parte hardware final, consiste en un sistema de control electrónico, donde se reutiliza la placa diseñada en el proyecto [2], y en la que se implementarán algunas modificaciones, para controlar una antena plana formada por un phased array lineal de desfasadores digitales como se aprecia en la Figura 1.

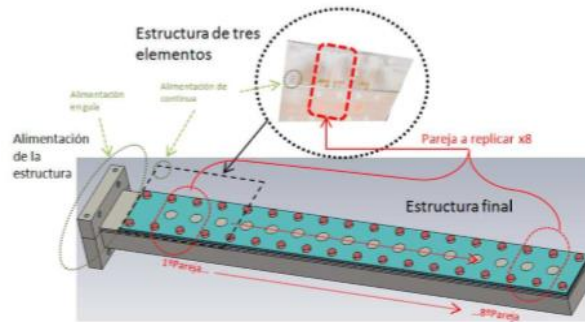


Figura 1. Diseño phased array

El phased array, que se controla en este PFC, es un prototipo intermedio, formado por 3 desfasadores digitales, puesto que el grupo RFCAs implementará posteriormente la antena conformada por 16 desfasadores, se deja implementada la interfaz y ciertas modificaciones que serán necesarias para poder controlar dicho array futuro.

1.3 Organización de la memoria

La memoria está dividida en 6 partes:

- La primera parte se centra en introducir el trabajo que se realiza, y lo que se busca conseguir con su desarrollo.
- La segunda parte, es el estado del arte, en el que se habla de aspectos que son importantes en este PFC, utilizado para la parte de control, antenas planas phased array, que es el elemento a controlar, y el protocolo de control utilizado para dar accesibilidad al sistema.
- La tercera parte, explica el diseño tanto de las placas de control como de la de radiofrecuencia, así como de la firmware necesaria, para crear la interfaz de usuario.
- La cuarta parte, es el desarrollo del diseño propuesto.
- La quinta parte, son las pruebas con los distintos sistemas de control y la respuesta de la parte RF.
- La sexta, se da una pequeña conclusión e ideas sobre trabajos futuros.

2 Estado del arte

En la Figura 2 se muestra un esquema básico de un sistema de control para antena plana receptora de sistema DBS:

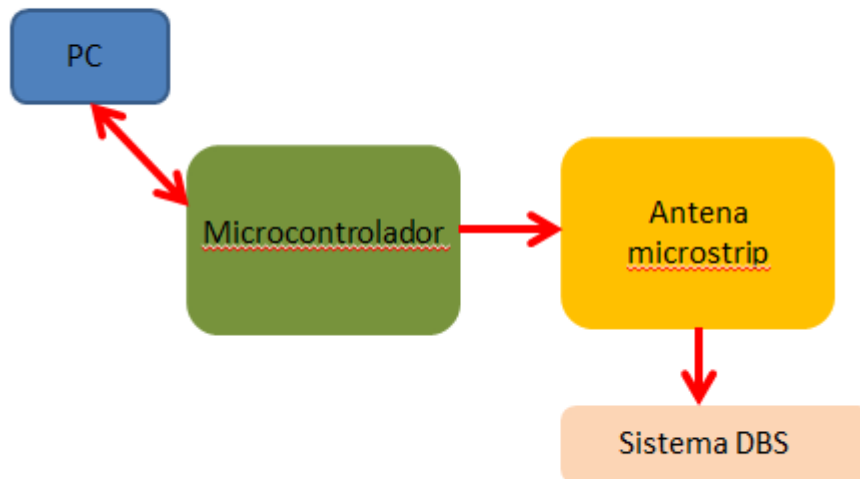


Figura 2. Esquema básico de un sistema de control

En este capítulo se mencionan características sobre las partes principales del sistema en su totalidad, el sistema de control y el sistema RF, además del protocolo de comunicación USB, utilizado.

También se destacan algunos proyectos desarrollados, que tenían como meta el control de antenas de distintos tipos, y con distintos sistemas de control.

2.1 Sistemas de control

En la actualidad existen una gran variedad de sistemas de control, estos pueden ser manuales o electrónicos. El interés en este proyecto son los sistemas electrónicos. Para conseguir el control electrónico de un sistema se utilizan microcontroladores. A continuación, se procede al análisis de algunos microcontroladores distribuidos comercialmente que existen en la actualidad:

- **Atmel Corporation:** Dispone de tres arquitecturas de procesadores ARM, AVR y 8051. Las unidades ARM son muy potentes y disponen de una gran variedad de módulos de comunicaciones y capacidad de multitarea. Están principalmente orientados a dispositivos multimedia de bajo consumo, como por ejemplo móviles

y reproductores de contenidos multimedia, muy por encima de los requisitos planteados.

Los chips de la arquitectura 8051, disponen de interfaz de comunicación I2C, temporizadores, módulos USB y suficientes pines para manejar una pantalla LCD. No obstante, es una tecnología madura y por tanto no está recomendada para nuevos diseños.

La serie AVR se presenta como los sustitutos de los procesadores basados en la arquitectura 8051. Tienen un set de instrucciones renovado, permiten la programación en ensamblador, ANSI C y C++. Se agrupan en 6 familias en función de sus características y prestaciones:

- 32-bit AVR UC3: Procesadores de alto rendimiento de 32 bits y bajo consumo. DSP integrado para operar decimales eficientemente. Disponen de controladores DMA (Direct Access memory) que junto con el bus de datos multinivel permite trabajar eficientemente con memorias de almacenamiento. Cuentan con gran variedad de módulos de comunicaciones entre los que destacan Ethernet/Mac USB High Speed y OTG (On the go). Además, cuentan con un módulo criptográfico AES/DES para cifrado eficiente de datos. La mayoría requieren una alimentación entre 3V y 3.6V aunque dispone de una serie con voltajes de alimentación de 3V a 5.5V. Algunas versiones disponen de hasta 109 líneas de E/S. Pueden trabajar hasta 66 MHz y tienen una capacidad de almacenamiento de hasta 250 Kbytes.
- AVR XMega: Esta familia se sitúa entre la familia UC3 y la familia megaAVR. Son procesadores de 8 y 16 bits, con las características de la familia megaAVR: ADC/DAC módulos de comunicación I2C, SPI, USB (OTG), múltiples contadores Watchdog. Pero que además incluyen algunas características importantes de la serie UC3 como son el módulo criptográfico AES/DES y el controlador DMA.
- megaAVR: Esta línea de procesadores de propósito general de 8 bits es la más popular de la familia AVR. Se compone de una amplia variedad de procesadores que permiten ajustarse a necesidades muy específicas sin desaprovechar funcionalidades. Muchos de ellos disponen de interfaz USB integrada, módulos de comunicación I2C, USART, SPI, CAN y LIN. También dispone de convertidores ADC/DAC para aplicaciones con señales analógicas, por ejemplo, para trabajar con sensores. Disponen de varios temporizadores y algunas versiones hasta controladores de pantallas líquidas LCD integrados dentro del propio encapsulado. Los más grandes tienen hasta 69 pines de entrada/salida de propósito general. La capacidad para el programa oscila entre 16kb y 64 kb y muchos de ellos disponen de interfaces JTAG y DebugWire para el depurado de código en tiempo real. Pueden alimentarse entre 1.8V y 5.5V.
- 8-bit tinyAVR: Familia de procesadores de propósito general de 8 bits de pequeñas dimensiones y muy versátiles. Ideados para el control de pequeños dispositivos que no requieran una capacidad de proceso elevada (frecuencia máxima 20MHz). Cuentan con módulos de comunicaciones serie, USART, I2C, SPI, convertidores ADC, varias líneas de entrada/salida de propósito general y PWM. Disponen de 1kb a 8kb de espacio para el programa y también disponen de una memoria

EEPROM interna para almacenar datos de forma permanente. Requieren una tensión de alimentación entre 2.7V y 5.5V, aunque existen versiones con convertidores Boost incorporados que permiten trabajar hasta con 0.7V.

- Battery Management: Procesadores de 8 y 32 bits de muy bajo consumo orientados a la supervisión y gestión de baterías.
- Automotive AVR: Esta familia está pensada para aplicaciones donde el entorno es hostil y requiere una elevada resistencia a impactos y grandes variaciones de temperatura. La mayoría de procesadores incorporan interfaces de conexión pensadas para el sector automovilístico (LIN, CAN...).

- **Fabricante Freescale:**

Freescale (antiguamente Motorola Semiconductors) es una de las compañías más antiguas y grandes en el mercado de semiconductores.

Dispone de una inmensa variedad de microcontroladores y microprocesadores en su catálogo y por tanto se analizan sólo las familias de microcontroladores más relevantes para este proyecto. Dispone de tres arquitecturas de microcontroladores: ARM, Arquitectura Freescale y PowerPC.

- Arm: •
 - Kinetis: Procesadores de 32bits basados en el núcleo ARM Cortex-M0 y ARM Cortex-M4. Se dividen en 8 series. Serie K (alto rendimiento), serie L (Bajo consumo), serie E (Resistentes a ESD/EMC), serie EA (Automoción), serie V (Motor y Potencia), serie M (Meteorología y sensores), serie W (Comunicaciones inalámbricas) y serie MINI (Encapsulado pequeño). Las series K, V y W utilizan el núcleo Cortex-M4. Son por tanto los más potentes. Trabajan a frecuencias entre 50Mhz y 200Mhz, con capacidades de almacenamiento de 32kB a 2 MB. Se pueden alimentar hasta 5V. Disponen de DSPs para realizar cálculos decimales eficientemente, RTC (Real time clock) y convertidores DAC/ADC de 12 y 16 bits, En cuanto a las comunicaciones disponen de múltiples puertos UART, I2C SPI, SAI (Serial audio interface) y soporte para Ethernet/Mac. Tiene módulos criptográficos y CRC (sumas de verificación). El patillaje oscila entre 32 pines (QFP) hasta encapsulados con hasta 256 patillas (BGA) El resto de series trabajan con el núcleo Cortex-M0. Trabajan a frecuencias más bajas, hasta 48 MHz y algunas series se pueden alimentar con 5.5V. Integran módulos de conexión USB High speed y capacidad USB-OTG, I2C, SPI y CAN. DMA y capacidad de almacenamiento de hasta 256Kb.

Arquitectura Freescale:

- Coldfire/Coldfire+: Procesadores de 32 bits orientados a aplicaciones de bajo nivel con múltiples opciones de conectividad, I2C, I2S, CAN, SPI.

Dispone de versiones con módulo USB integrado o Ethernet. Pueden trabajar a 80 MHz y la mayoría permiten una tensión de alimentación de 3.6V aunque tiene varias series con un rango mayor, hasta 5.5V. Tienen una gran capacidad de almacenamiento (32kB hasta 512 kB). Como complementos tienen múltiples convertidores ADC/DAC, módulos de criptografía y generación de números aleatorios e interfaces de depuración JTAG y On-chip Emulation.

- S08/RS08/08: Procesadores de 8 Bits para aplicaciones pequeñas y muy específicas. Tienen un coste muy competitivo y están pensados para ser ubicados en los actuadores dentro de un sistema de control distribuido. Disponen de interfaces SPI y I2C pueden trabajar hasta velocidades de 50 MHz. Las capacidades de almacenamiento alcanzan los 128 KB y se pueden alimentar hasta con 5.5V.
- **Microchip: PIC**
 - PIC: Familia de procesadores de 8, 16 y 32bits muy populares enfocados a aplicaciones de control. Dispone de un amplio abanico de modelos con capacidades de almacenamiento hasta 128KB y memoria EEPROM hasta 4 KB para los procesadores de 8 bits llegando hasta los 2MB en el caso de los procesadores de 32 bits. Compatibles en patillaje y código entre las distintas series permiten ampliar el procesador sin complicaciones. Disponen de módulos de comunicación serie I2C, SPI, UART, USB y Ethernet. Permiten realizar el código en C y ensamblador. Las versiones de 32bits permiten realizar multitarea y pueden ser utilizados para aplicaciones multimedia. Las tensiones de alimentación según la serie alcanzan los 5.5V.

2.2 Antenas planas en sistemas DBS

2.2.1 Conceptos básicos

Una antena es un dispositivo diseñado con el objetivo de emitir y/o recibir ondas electromagnéticas hacia/desde el espacio libre. Una antena transmisora, transforma la corriente eléctrica en ondas electromagnéticas, y una receptora lo contrario. [4][5]

Una antena se comporta de igual forma en transmisión que en recepción, y se caracteriza por ciertos parámetros, entre los más habituales:

- **Ancho de banda:** Margen de frecuencias en el cual los parámetros de la antena cumplen unas determinadas características. Se puede definir: ancho de banda de potencia, polarización, ganancia u otros parámetros.
- **Directividad:** es la relación entre la densidad de potencia radiada en la dirección de máxima radiación a una cierta distancia, y la potencia total radiada dividida por el área de la esfera de radio R.
- **Ganancia:** Es la relación entre la densidad de potencia radiada en la dirección del máximo a una distancia R, y la potencia total entregada a la antena dividida por el área de una esfera de radio R.

- **Rendimiento en la antena:** es la relación entre la potencia de radiación y la potencia total aplicada a la antena, en la cual se tiene en cuenta, además de la potencia de radiación, la potencia de pérdida.
- **Impedancia:** Una antena, se tendrá que conectar a un transmisor o receptor y deberá radiar o recibir el máximo de potencia posible, con un mínimo de pérdidas. Se deberá adaptar el transmisor o receptor a la antena para la máxima transferencia de potencia, que se suele hacer a través de una línea de transmisión. Esta línea también influirá en la adaptación, debiéndose considerar entre otros, su impedancia característica y atenuación.
- **Anchura de haz:** coincide con el ancho de haz a -3dB, que es el intervalo angular en el que la densidad de potencia radiada es igual a la mitad de la máxima.
- **Polarización:** es una determinada dirección, es la figura geométrica que traza el extremo del vector campo eléctrico a una cierta distancia de la antena, al variar el tiempo. La polarización puede ser lineal, circular y elíptica. La polarización lineal, puede tomar distintas orientaciones(horizontal, vertical, +45°, -45°). Las polarizaciones circular o elíptica pueden ser a derechas o izquierdas.
- **Diagrama copolar:** es el diagrama de radiación con la polarización deseada, y diagrama contrapolar al diagrama con polarización contraria.

2.2.2 Antenas planas o microstrip

Existen distintos tipos de antenas, pero este proyecto busca modificar el apuntamiento o polarización de una antena plana o microstrip de tipo parche. Su configuración:

- Un parche(dimensiones comparadas a λ).
- Un sustrato dieléctrico:
 - grosor de $0.003\lambda \leq h \leq 0.05\lambda$
 - constante dieléctrica de $2.2 \leq \epsilon_r \leq 12$
- - Un plano de masa en la cara opuesta.

Los parches pueden tener cualquier forma, en este proyecto se usan redondos.

Una antena microstrip se compone de 4 elementos: el parche, el sustrato dieléctrico, el plano de masa y el tipo de alimentación. El formato de antena parche más común:

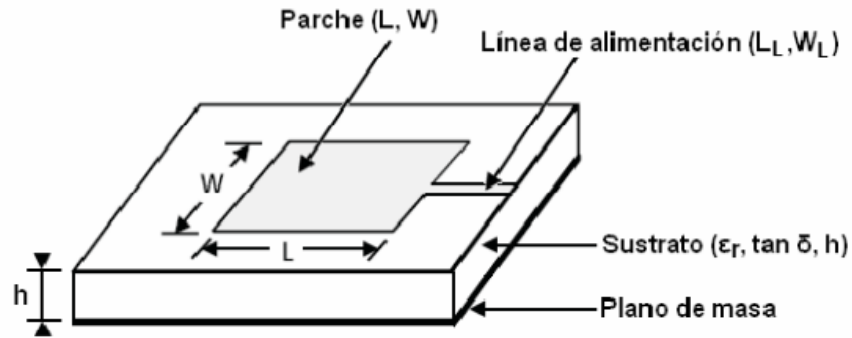


Figura 3. Formato básico de una antena Microstrip

Las antenas microstrip tienen la ventaja de que son livianas y ocupan poco volumen, facilidad de adaptación a distintas superficies, bajos costos de fabricación, facilidad para fabricación serie, soporte de polarización lineal y circular, facilidad de integración en sistemas integrados microondas, posibilidad de diseños a diferentes frecuencias y son mecánicamente robustas al ser montadas sobre superficies rígidas.

Pero también tienen algunas desventajas, como el pequeño ancho de banda, baja potencia, baja ganancia, limitación de potencia, baja fuerza de polarización y la radiación de los bordes puede afectar a los parámetros de la antena.

Destacando el parámetro de polarización de la antena, se define “coeficiente de polarización” a la diferencia entre la polarización de la antena y la onda incidente. Este coeficiente debería ser:

$$C_p = |\widehat{e}_a \times \widehat{e}_i|^2$$

Donde e_a es la polarización de la antena receptora, y e_i es la polarización de la onda incidente. Una onda incidente con la misma polarización que la antena tendrá un vector despolarización $C_p = 1$.

Algunas de sus aplicaciones típicas:

- Antenas embarcadas en misiles.
- Altímetros radar en aviones.
- Antenas de exploración radar en satélites.
- Sistemas GPS.
- Telefonía móvil.
- Comunicaciones móviles por satélite.
- Aplicadores de calor en medicina (hipertermia).

La antena que se controla en este PFC, se implementa en un sistema DBS.

2.2.3 Sistema DBS

Un sistema DBS(Direct Broadcast System/Satellite) se define como el servicio que distribuye una señal de audio, video o datos sobre una extensa zona determinada, haciendo uso de sistemas dedicados a ello, especialmente permitiendo la recepción en terminales de pequeño tamaño.

Los sistemas DBS son muy empleados en sistemas de televisión de pago que no necesita de conexión por cable para su visualización, y por tanto se utiliza en zonas rurales donde los cables no están disponibles. Además, es un servicio fiable puesto que no depende de las caídas en la red o de la rotura de cables, únicamente puede verse afectado por un tiempo meteorológico muy severo o por interferencias solares durante los equinoccios, lo cual es raro.

2.3 Algunos sistemas de control para antenas en la actualidad

En la actualidad se conoce la necesidad de utilizar sistemas de control para modificar las características de una antena con mayor comodidad. A continuación, se van a mencionar algunos proyectos cuyo objetivo es conseguir el control de distintos tipos de antenas, al igual que el de este PFC.

2.3.1 Módulo para el control y posicionamiento de antenas formadas por rotores Yaesu G5400 y G5500

Existe una versión diseñada por WA8SME [6], Mark Spencer, que utiliza un microcontrolador PIC 16F688, para conseguir el control de posicionamiento de rotores Yaesu G5400 Y G5500. Este microcontrolador programable está compuesto por un puerto serie (UART) que permite la comunicación con el PC, un conversor A/D de 10-bit para leerla posición de las antenas (en voltios) y por último un puerto I/O con 12 señales para controles externos.

El microcontrolador PIC 16F688 programable básicamente está compuesto por un puerto serie (USART) para comunicarse con el PC, un conversor A/D de 10-bit para leer la posición de las antenas (en voltios) y por ultimo un puerto I/O con 12 señales para controles externos. En este caso específico y personalizado, la fuente de voltaje proviene del propio rotor y es reducido a valores de 5 voltios por el CI 78L05. El transistor Q1 convierte los valores de voltaje del puerto serie RS232 del ordenador a niveles TTL necesarios para la comunicación con el PIC 16F688. Los transistores Q2 a Q5 controlan los diferentes relés de movimientos azimutal y elevación que se encuentran en la unidad de potencia y control del rotor. Los diodos LED que figuran en el esquema no son parte necesaria para el correcto funcionamiento del circuito, pero ofrecen una ayuda visual del funcionamiento.



Figura 4. Circuito controlador con microcontrolador PIC 16F688

Los pasos requeridos para el correcto funcionamiento serán:

1. Convertir el valor de la posición en grados, enviado por el programa a través del puerto serie, en el valor digital correspondiente a un voltaje en función de esa posición.
2. Leer a través de los puertos A/D la posición real de la antena que nos envía el rotor. Un voltaje en función de la posición.
3. Comparar la posición deseada que envía el programa, con la posición real leída en las patillas 9 y 10 del PIC-16F688.
4. Determinar la dirección de giro del rotor en función del error obtenido.
5. Girar los rotores y continuar leyendo la posición de las antenas hasta que alcancen la posición deseada.
6. Comenzar el proceso nuevamente leyendo la nueva posición enviada por el programa de seguimiento NOVA, SatPc32.

La configuración y operación del circuito es totalmente transparente. El PIC está programado para comunicarse con un estándar de 9600 Baud con los programas NOVA o SatPc32, que deberán tener sus puertos COM configurados correctamente y coincidentes con el PIC. Configurar la opción "Tracking" del NOVA para que utilice el protocolo EASYCOMM1 en la comunicación con el controlador del rotor o configurar el SatPc32 para que use el protocolo SAEBRTrackBox en su comunicación con el controlador de posición (nuestra PIC-16F688) Fijar la posición de aparcamiento del rotor azimutal (norte o sur) en función de las propias características del rotor usado. Fijar el modo de parada del rotor de elevación de 0° y 180°. En definitiva, adaptar al software tanto del NOVA como del SatPc32 las condiciones particulares de funcionamiento de nuestro sistema. En este punto, escogemos un satélite para su seguimiento y podremos observar cómo en el momento de inicio del seguimiento, el diodo LED de datos (D4) comienza a parpadear indicándonos que los datos están siendo recibidos por la PIC16F688. Seguidamente y dependiendo de la dirección de giro calculada por el programa, los rotores comienzan a moverse en la dirección demandada. Pueden verse las órdenes de dirección de marcha con los diodos LED que figuran en el esquema general, como decíamos al principio, son una ayuda para observar el funcionamiento del sistema. (D1-D2, bajar-subir, y D3-D4,

izquierda - derecha). El proceso continuará activo mientras el satélite esté dentro de la ventana de uso y dentro de los parámetros configurados en el programa de seguimiento.

2.3.2 Control automático para la red satelital VSAT-ICE

Desarrollo de un sistema de control automático que permite el control de la red satelital VSAT-ICE [7], cuya antena se encuentra en Tarbaca, este sistema se encarga de proporcionar servicios de banda ancha y teléfono para zonas remotas a lo largo y ancho del país. Puesto que en un primer momento, el sistema de control del que se disponía era muy antiguo de la década de los 80, y por tanto era poco eficiente, ya que necesitaba de un control manual por parte del usuario.

Para conseguir dicho control, el sistema hardware consiste en un kit de desarrollo disponible en el mercado SBC65EC. Este kit, posee un micro controlador PIC18F6627.

Como protocolo de comunicación utiliza el protocolo UDP.

2.3.3 Módulo para el control de apuntamiento de un conjunto de antenas planas para un sistema WiMAX 3.5 GHz.

En el proyecto desarrollado por Alfredo Manuel de Castro, desarrollado en el laboratorio HCTLab de la Universidad Autónoma de Madrid, se logra el control de antenas planas para un sistema WiMAX.

Para conseguir dicho control, utiliza como elemento principal de la parte hardware, el microcontrolador ATmega32u4, diseñando y construyendo, una placa, donde extrae las funcionalidades necesarias para conseguir el control de la antena a partir del mismo. Utilizando el protocolo de comunicación serie, SPI.

La placa de control diseñada, es una de las placas que forman parte de este PFC, y que también realizando ligeras modificaciones permitirán controlar las antenas planas de un sistema satelital DBS.

En el proyecto de control de un sistema WiMAX, la parte de radiofrecuencia a controlar está formada, por desfasadores de tipo MAPS 008343-PK6003.

2.4 Protocolo USB

2.4.1 Introducción

El Bus Universal en Serie (USB), es un bus estándar industrial que define los cables, conectores y protocolos usados en un bus para conectar, comunicar y proveer de alimentación eléctrica entre computadoras, periféricos y dispositivos electrónicos.



Figura 5. Símbolo USB

En la actualidad, las aplicaciones USB están muy extendidas, ya que están implementadas en cualquier dispositivo electrónico o componentes. Incluso Europa, ha creado una norma, que obliga que todos los móviles deberán venir con un cargador microUSB, pero también están disponibles en aplicaciones como: periféricos de ordenadores, reproductores, juguetes modernos...

USB, es un bus punto a punto puesto que parte de un host(PC o hub), y el destino es un periférico u otro hub. Un hub permite que en un puerto USB permita más de dos periféricos simultáneamente.

2.4.2 Tipos

Los dispositivos USB, pueden clasificarse por su velocidad de transmisión:

- Velocidad baja (1.0): hasta 1.5 Mbits/s, es utilizado mayoritariamente por dispositivos HID(Human Interface Device), son los teclados, los ratones(mouse), las cámaras web,...
- Velocidad completa (1.1): hasta 12Mbits/s, son dispositivos que dividen el ancho de banda de conexión USB entre ellos.
- Alta velocidad (2.0): hasta 480Mbits/s, pero la tasa real es de 280Mbits/s. Dispone de 4 líneas, dos para datos y otros dos para alimentación.
- Velocidad super alta (3.0) hasta 4.8 Gbits/s, será compatible con los estándares anteriores.

A nivel eléctrico, se transmite sobre 4 hilos, dos para la alimentación(5V) y otros dos para la señal(D+ y D-).

Los conectores USB pueden tener formas y tamaños distintos:

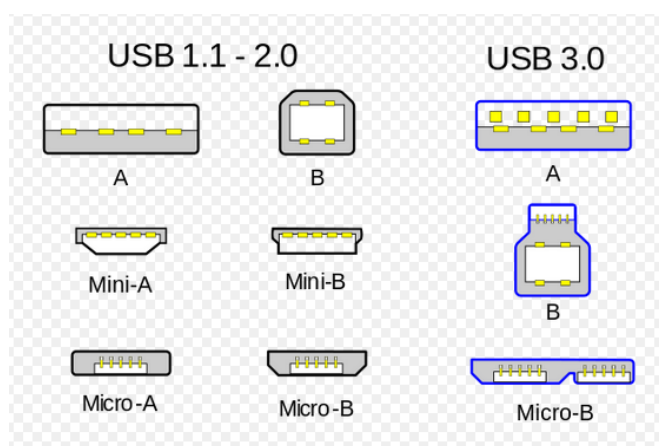


Figura 6. Tipos de conectores USB

2.4.3 Aplicaciones USB

A la hora de desarrollar una aplicación USB, es necesario:

- Un microcontrolador o CI que soporte la interfaz USB.
- Un programa sobre el periférico para transmitir la información en cualquier lenguaje de programación.
- Una computadora con puerto USB.
- Herramientas para programación de microcontroladores.

En una aplicación con protocolo de comunicación USB, los periféricos comparten la banda de paso del USB, esto se conoce como token. Los ordenadores proporcionan el token al periférico y este los devuelve una respuesta. Se permite la conexión y desconexión en todo momento sin necesidad de apagar el equipo.

En una comunicación USB, existen varias partes diferenciadas:

- **Host:** Dispositivo maestro que inicia la comunicación, el PC en este caso.
- **Hub:** Dispositivo que contiene uno o más conectores o conexiones internas hacia otros dispositivos USB. Habilita la comunicación del host con distintos dispositivos.
- **Puerto USB:** Cada host soporta solo un bus, y cada conector en el bus representa un puerto USB por tanto sobre el bus puede haber distintos conectores, pero con una única ruta y solo un dispositivo puede transmitir información a un tiempo.
- **Driver:** Programa que habilita aplicaciones para poderse comunicar con el dispositivo. Cada dispositivo sobre el bus debe tener un driver, algunos periféricos utilizan los drivers que trae Windows.

- **Puntos terminales (Endpoints):** Es una localidad específica dentro del dispositivo. El Endpoint es un buffer que almacena múltiples bytes, típicamente es un bloque de la memoria de datos o un registro dentro del microcontrolador. El Endpoint 0 siempre es el buffer de control.
- **Tuberías(Pipes):** Es un enlace virtual entre el host (PC) y el dispositivo USB, este enlace configura los parámetros asociados con el ancho de banda que tipo de transferencia (Control, Bulk, Isocrona o Interrupt) dirección del flujo de datos y el máximo y/o mínimo tamaño de los paquetes/buffers.

Cada enlace se caracteriza por su banda de paso(Token), su tipo de servicio, el número de endpoint y el tamaño de los paquetes.

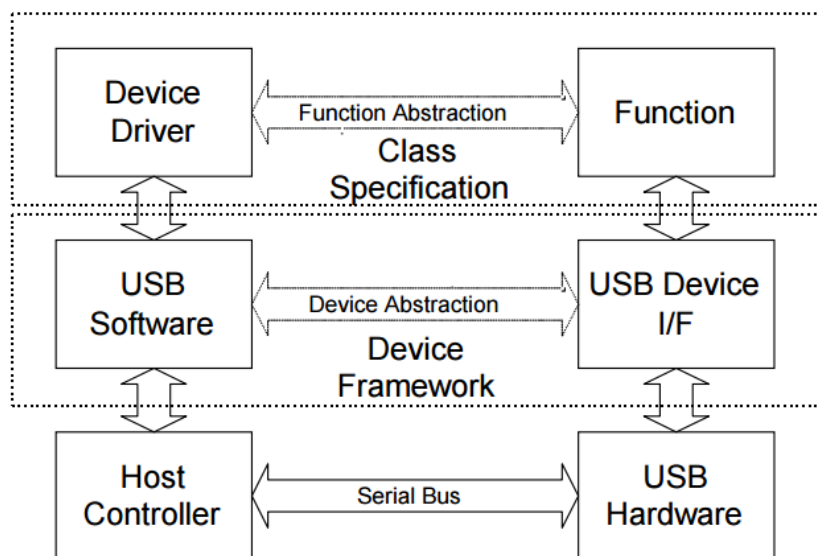


Figura 7. Relación entre un dispositivo y un sistema Host

2.4.4 Protocolo USB CDC

2.4.4.1 Introducción

La clase de dispositivo de comunicación (CDC), que simplifica al usuario la comunicación USB proporcionando una funcionalidad de puerto COM. Es como la comunicación interfaz de dispositivos a USB. Es compatible con una amplia gama de dispositivos que pueden realizar funciones de telecomunicaciones y de redes. Ejemplos de equipos de comunicaciones son:

- Equipos de telecomunicaciones, tales como teléfonos analógicos y módems, adaptadores de terminal ISDN, teléfonos digitales, así como los dispositivos COM-puerto

- Dispositivos de red, tales como ADSL y módems de cable, así como adaptadores Ethernet y concentradores

El CDC proporciona una sencilla UART para transmitir y recibir datos a y desde el host, ocultando la mayor parte de los datos del USB a distancia desde la aplicación.

2.4.4.2 Características del CDC Clase y limitaciones

Un dispositivo de comunicaciones tiene tres tareas básicas:

- Gestión de dispositivos (controlar un configuración de un dispositivo específico y notificar al host USB de ciertos eventos)
- Gestión de llamadas (establecimiento y terminación de las llamadas telefónicas u otras conexiones)
- La transmisión de datos (enviar y recibir datos de aplicación)

Nota: Se debe tener en cuenta que el máximo tamaño de transmisión es de 64 bytes.

La implementación de los CDC en el componente USB tiene las siguientes características y limitaciones:

- Emulación de un puerto COM virtual utilizando la subclase ACM (Extracto Modelo de Control) de los CDC.
- Una interrupción EN punto final para las notificaciones a la USB Host
- Un punto final a granel IN y OUT uno mayor para la transferencia de datos
- El componente USB soporta la clase CDC (ACM) para aplicaciones de dispositivos USB y USB Host.

3 Diseño

3.1 Introducción

La Figura 8 muestra una idea de esquema de control del proyecto, donde se puede ver que la antena es controlada por un sistema manual y un sistema electrónico de control.

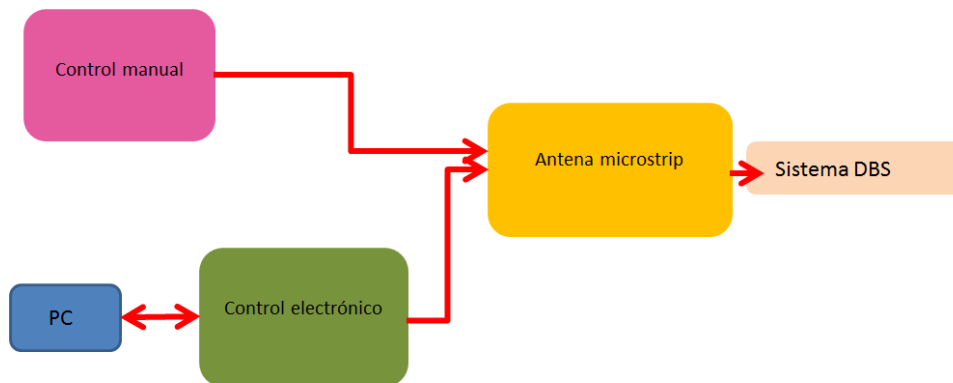


Figura 8. Idea de diseño del sistema completo

Para ello es necesario, elegir el componente utilizado en cada uno de los módulos de este esquema.

3.2 Control manual

Todos los componentes utilizados en esta placa, han sido seleccionados por disponibilidad inmediata en el laboratorio, puesto que como es una placa temporal y utilizada en pruebas iniciales, se necesitaba construir en el menor tiempo posible.

Los componentes que contiene:

- Un Switch: que es el interruptor de selección de 8 salidas a la placa RF, aunque en realidad únicamente se utilizan 5 de ellas, 4 para el envío de los bits de fase y 1 para selector del Modo Paralelo. La razón de utilizar un interruptor de 8 salidas es porque era del que se disponía en el laboratorio en ese momento.
- 2 Reguladores Lineales de Tensión, uno de ellos tipo mc7805 y otro mc7905.
- Conector macho de 12 pines para la distribución de la alimentación y el bus de datos.
- Clema para introducir la alimentación de la fuente de alimentación VCC y VEE.

3.3 Control electrónico

El elemento principal del control electrónico es un microcontrolador, como se ha visto en el estado del arte de esta memoria existe una variedad de posibilidades de uso. El microcontrolador utilizado en este PFC, es de la arquitectura AVR proporcionada por Atmel. En concreto el microcontrolador ATmega32U4 de la familia AVR Mega. Esta elección se hace porque el grupo de investigación HCTLab dispone de una placa que contenga dicho microcontrolador diseñada en un proyecto previo [2], donde se conseguía el control electrónico de una antena. En dicha placa ya están implementadas una gran cantidad de funcionalidades, que facilitan el desarrollo de este PFC, pues solo sería necesaria una placa interfaz que nos permitiera extraer las salidas o entradas necesarias para lograr el control que se busca en este PFC.

3.4 Placa de RF

En la placa de RF, los desfasadores son del tipo digital phase shifter de 4-bits, esta placa es diseñada por el grupo de investigación RFCAs, por lo que la elección de este componente viene dado como especificación del grupo.

3.5 Protocolo de comunicación entre microcontrolador y antena microstrip

Conociendo que el microcontrolador utilizado en la placa principal, puede configurarse de distintas formas para controlar el sistema. En un primer momento se piensa que la mejor alternativa es el protocolo de comunicación SPI (Serial Peripheral Interface), que es un protocolo maestro/esclavo dirigido por un reloj y una señal de inicialización del sistema. Posteriormente, se ve necesario el estudio de otro método de envío de datos como se explica a lo largo de este apartado.

3.5.1 Protocolo SPI

El microcontrolador ATmega32u4 tiene distintos modos de transmisión, en un principio se pensó utilizar el modo de transmisión SPI (Serial Peripheral Interface). Este es un método de transmisión serie que es considerado el más rápido y síncrono, con una interfaz de transferencia de datos en serie dúplex completo. En una comunicación SPI solo son necesarios cuatro pines (/CS, SCK, MOSI, MISO), y se basa en la transmisión serie de 8 bits entre un módulo maestro y un módulo esclavo.

- /CS: señal de selección. Cuando está en bajo el correspondiente Slave estará seleccionado. Y será el esclavo que recibirá la información serie transmitida por el Master.

- **SCK:** es la señal de reloj, obtenida del reloj MASTER, que controla los tiempos de transmisión de los datos.
- **MOSI:** es la señal de datos a través de la cual se envía los datos del registro interno del Master al Slave.
- **MISO:** es la señal de datos de entrada al Master, por la cual recibe los datos del registro de estado del Slave.

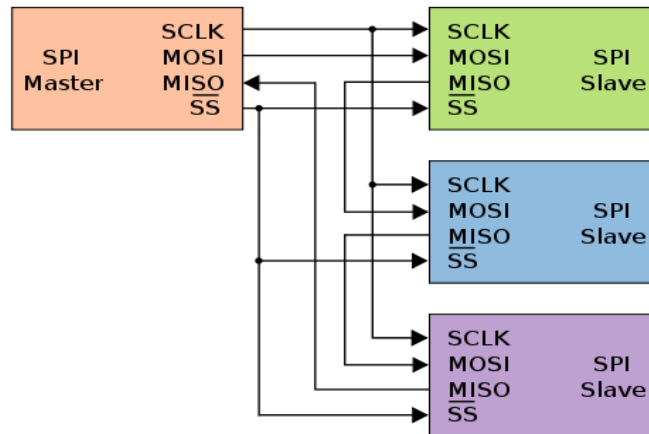


Figura 9. Protocolo de comunicación SPI

Al estudiar las características del Desfasador Digital de 4 bits se descartó la posibilidad de utilizar este protocolo de comunicación. Puesto que en este protocolo, siempre se envía 1 byte de datos y el desfasador solo recibe 6 bits para cada dato de desfase.

3.5.2 Salidas digitales

Ya que la alternativa inicial de comunicación (SPI) no es compatible, es necesario diseñar un protocolo particularizado para este PFC, creando las señales de Enable, Reloj y Datos a través de salidas digitales, utilizando los pines del PORT D del micro.

A la hora de controlar un array de desfasadores lineales, en este caso serán 3, pero en un futuro este número puede ser reconfigurable. Se tienen en cuenta dos topologías de envío de datos, y por tanto de conexionado.

3.6 Protocolo de comunicación entre PC y la placa de control

Ya que se dispone de un conector de tipo Mini USB en la placa facilitada [2], desde el primer momento el protocolo de comunicación que se cree más conveniente es el protocolo USB.

Además de por disponibilidad inmediata, este protocolo de comunicación tiene muy

buenas características. Puesto que al ser por cable no realiza ningún tipo de interferencia con las señales de radiofrecuencia, como podría suceder con el protocolo de comunicación Bluetooth.

Por otro lado, la conexión de los dispositivos USB está disponible en la mayoría de los equipos como se ha dicho anteriormente, y por tanto aumenta la compatibilidad.

También es un punto a favor, que existan librerías facilitadas en internet que nos permita conseguir el control USB de forma más rápida y sencilla, en este PFC se utiliza en concreto la librería LUFA, que permite implementar este protocolo USB en el microcontrolador.

Por lo tanto la solución propuesta finalmente del sistema completo:

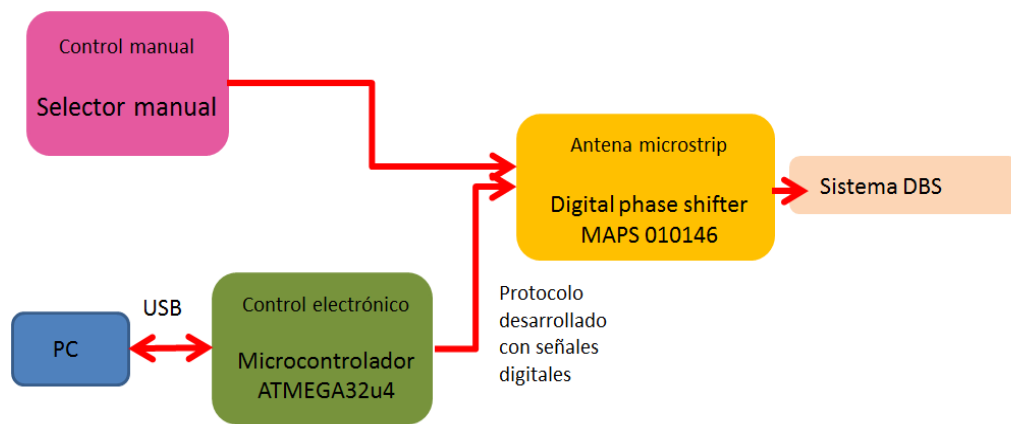


Figura 10 Solución esquema final propuesto del sistema completo.

4 Desarrollo

En esta sección se tendrá en cuenta el diseño de las placas necesarias en el desarrollo de este PFC:

- Placa de Control Manual.
- Placa de Control Electrónico.
- Placa de Radiofrecuencia.

Además, se va a estudiar y probar distintos protocolos para el envío de datos entre el sistema de control electrónico y el array de desfasadores, o en un primero momento, con un único desfasador.

Finalmente, se explican los pasos seguidos para la creación del firmware, además de especificar las modificaciones futuras necesarias, para aumentar el tamaño del phased array.

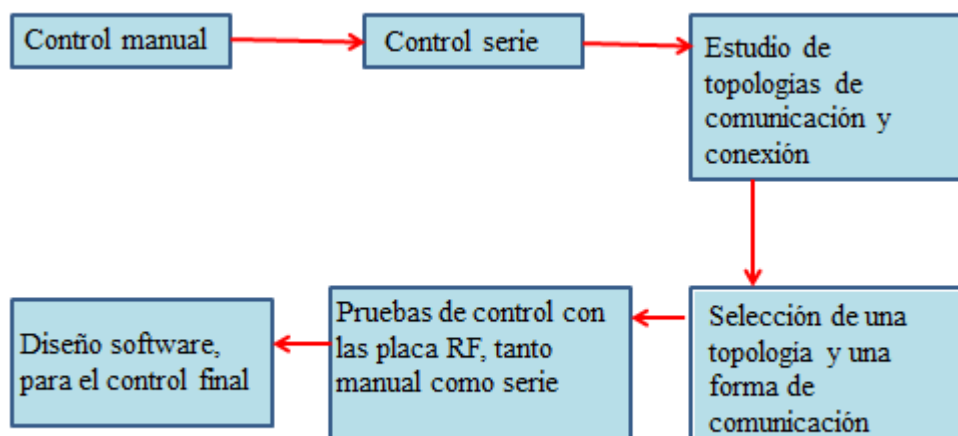


Figura 11. Flujo de desarrollo del sistema

4.1 Desarrollo hardware

4.1.1 Placa de control manual

En un principio se estudia el Control Manual (Paralelo) de la antena. Este control se realiza a partir de una placa independiente a la placa de Radiofrecuencia, y únicamente se diseñará para las pruebas iniciales con un único desfasado digital de RF, ya que solo servirá para comparar los resultados de atenuación y variación de fase con los que se obtiene posteriormente con la placa de control electrónico, y por tanto verificar si se

consiguen los mismos resultados.

El diseño de esta placa se hace en el software de diseño electrónico *Altium Designer 13.3*, para poder utilizar todas sus herramientas es necesario disponer de una licencia. A la hora de diseñar las placas con este software, se tiene en cuenta un diseño esquemático donde se introducen los componentes a utilizar y las conexiones entre ellos, y un diseño de PCB que es la huella del circuito impreso y lo que finalmente imprime la fresadora.

En la página de Altium existe la posibilidad de descargar librerías con multitud de componentes [Download all Libraries, in single ZIP file \(305MB\)](#). Si el componente que se necesita no está disponible en dichas librerías se debe crear su esquemático y su footprint, extrayendo la información de medidas de su datasheet. Por ejemplo, a continuación se detalla la creación del componente switch utilizado en la placa de control manual:

1. Se crea una librería de esquemáticos. Botón izquierdo sobre el proyecto y Add New to Project → Schematic Library

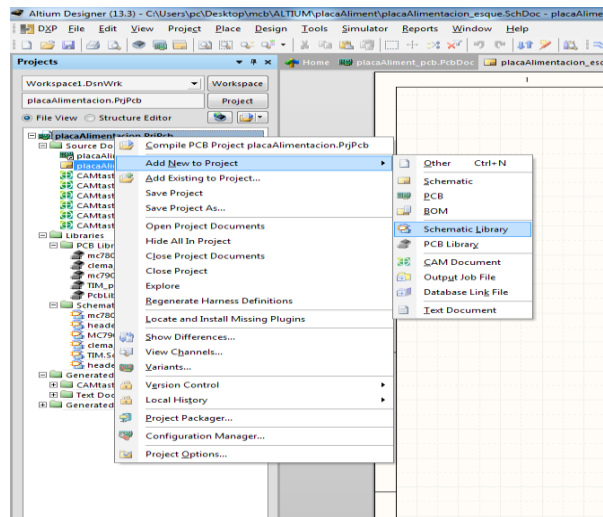


Figura 12. Diseño componente (I)

2. Se añade al proyecto el esquemático del componente.

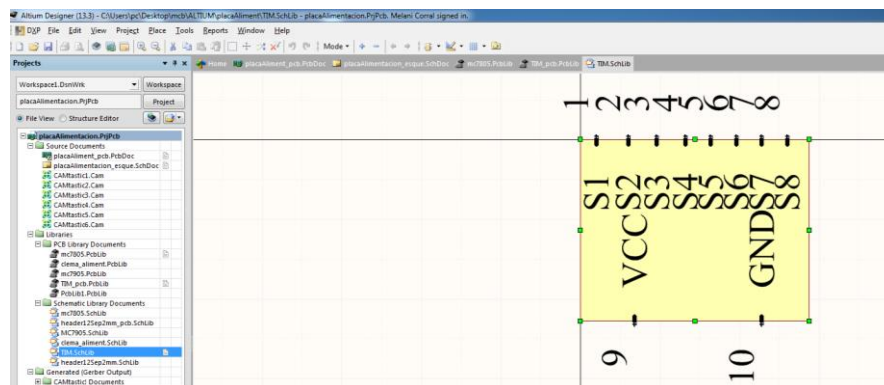


Figura 13. Diseño componente (II)

3. Se añade la librería de footprints. Botón izquierdo sobre el proyecto y Add New to Project → PCB Library

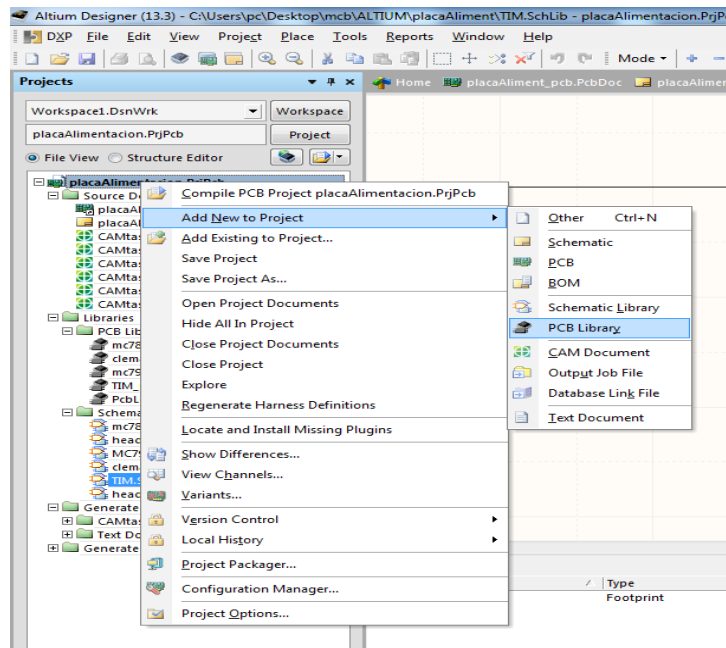


Figura 14. Diseño componente (II)

4. Se diseña el footprint, con las medidas adecuadas del datasheet.



Figura 15. Diseño componente (IV)

5. Se asocia el esquemático y el footprint del componente.

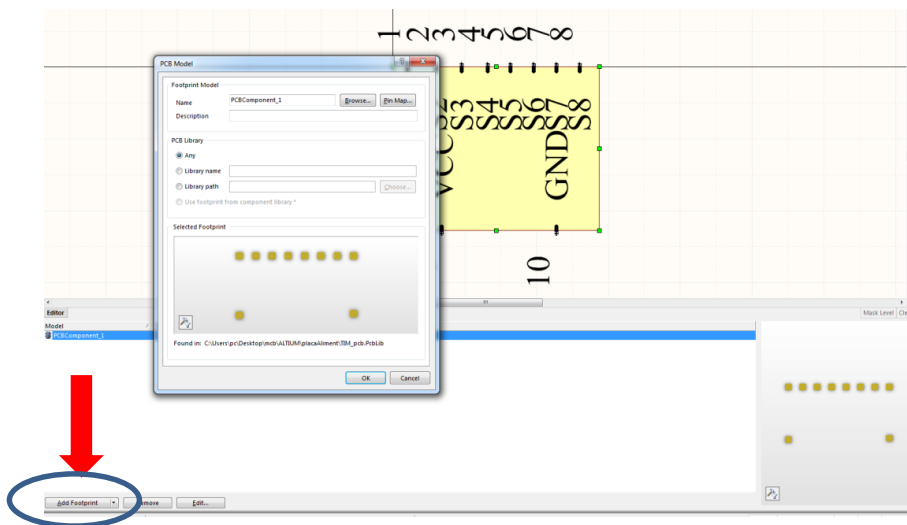


Figura 16. Diseño componente (V)

La placa de Control Manual contiene:

- Un **Switch**: que es el interruptor de selección de 8 salidas a la placa RF, aunque en realidad únicamente se utilizan 5 de ellas, 4 para el envío de los bits de fase y 1 para selector del Modo Paralelo. La razón de utilizar un interruptor de 8 salidas es porque era del que se disponía en el laboratorio en ese momento.

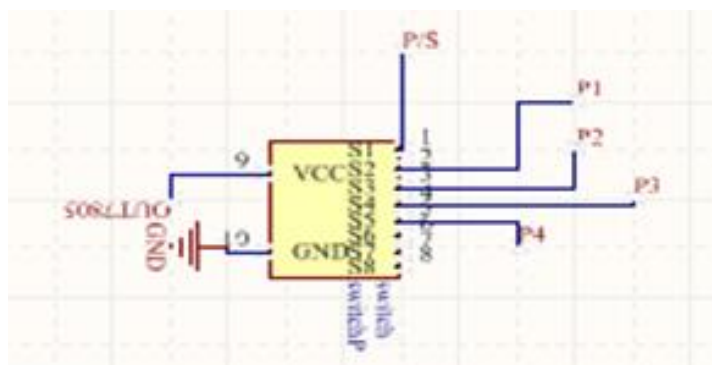


Figura 17. Switch placa de control manual

- 2 Reguladores Lineales de Tensión, uno de ellos tipo *mc7805* y otro *mc7905*.

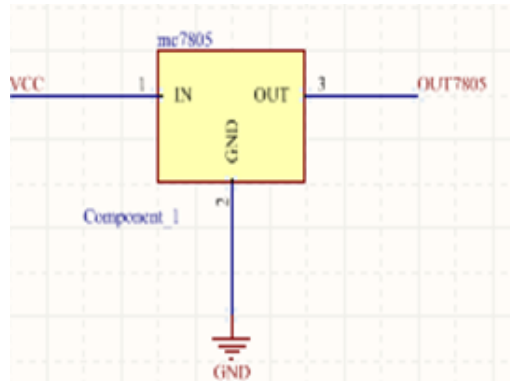


Figura 18. Regulador mc7805 placa de control manual

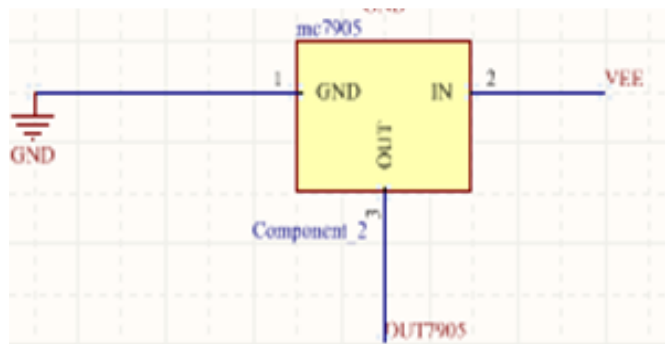


Figura 19. Regulador mc7905 placa de control manual

- *Conector macho de 12 pines* para la distribución de la alimentación y el bus de datos.

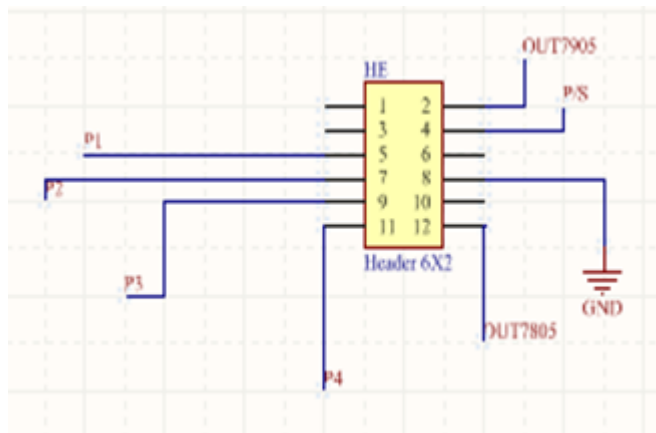


Figura 20. Conector de conexión a la placa RF

- **Clema** para introducir la alimentación de la fuente de alimentación VCC y VEE.

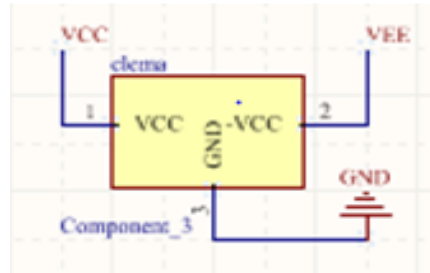


Figura 21. Clema de alimentación

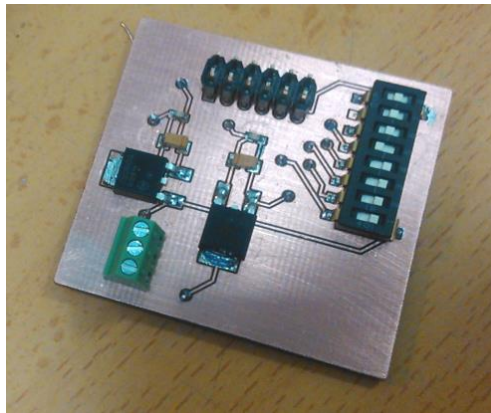


Figura 22. Placa control manual o paralelo

Las salidas de Conector macho de esta placa se conectan a través de una tira plana directamente a la placa de Radiofrecuencia.

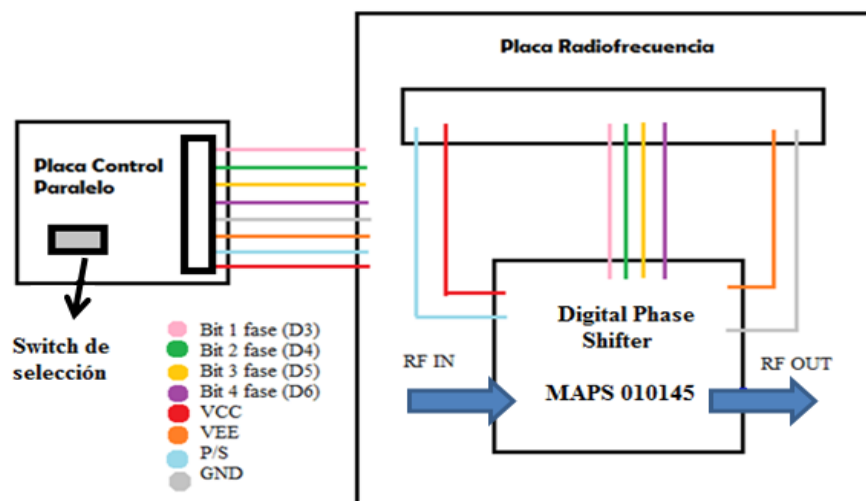


Figura 23. Esquema de conexión entre la placa control paralelo y placa RF

- *P/S*: Controla el Modo de Control de la antena, en este caso la seria 0 (paralelo), en el caso de Control Serie es 1, conectado al Pin 2 del desfasador.
- *P1, P2, P3 y P4*, serán los selectores de fase del desfasador digital de 4 bits, que entrarán por los pines 19, 20, 21 y 22 (D3, D4, D5 y D6) de este.
- OUT 7905 salida de voltaje *VCC*, Pin 18 del desfasador.
- OUT 79 salida de voltaje *VEE*, Pin 1 del desfasador.
- Pin de *GND* necesario, conectado a un plano tierra de la Placa de Radiofrecuencia, en la capa Boton.

4.1.2 Placa de control electrónico

La finalidad de este PFC es el control electrónico o control serie. La placa principal necesaria para dicho control se diseña, construye y verifica en un proyecto[2] previo a este, donde se realizan ligeras modificaciones para extraer las salidas que interesan, en este caso.

El elemento principal de dicha placa principal, es el microcontrolador ATmega32u4 (DataSheet Anexo I). Por lo que es necesario, comprobar que dicho controlador es compatible con los receptores que conforman la antena, es decir, los desfasadores digitales MAPS 010146 (Datashet Anexo II). Para ello, es importante conocer el rango de frecuencias y de tensiones donde dichos componentes funcionan eficientemente.

Tabla 1. Rango de Tensión

	Min.	Máx.
ATMega 32u4	2.7 V	5.5 V
MAPS 010146	3 V	5.5 V

Tabla 2. Rango de Frecuencia

	Min.	Máx.
ATMega 32u4	8 MHz	16 MHz
MAPS 010146	8 MHz	20 GHz

En las tablas extraídas se comprueba que ambos dispositivos comparten un rango de tensión y de frecuencias donde tienen un buen funcionamiento.

Una vez comprobada la compatibilidad entre los elementos que forman el sistema total. Es esencial, conocer las formas de comunicación disponibles por parte del microcontrolador para el envío de datos, en este caso hacia antenas planas.

Estudiando en detalle la hoja de datos del microcontrolador y realizando varios programas de prueba para entender correctamente su funcionamiento (como el encendido de leds o comunicaciones con memorias externas). Se analizan varios protocolos para el envío de datos, con las que es compatible el ATmega32u4.

4.1.2.1 Topologías de conexión entre la placa de control serie y la placa de RF

4.1.2.1.1 Topología Daisy Chain

Es una conexión en cadena. Los datos enviados a través de la placa de Control Serie únicamente serán enviados al primero de los Desfasadores Digitales de la placa de RF, este enviará los datos al segundo de ellos, el segundo al tercero... y así sucesivamente. Esto es posible, ya que el Datasheet del MAPS 010146 (Anexo II) menciona que SER IN es SER OUT tras 6 ciclos de reloj. El bucle no está cerrado puesto que los datos del último desfasador no son retornados a la placa de control.

En este diseño el cableado es muy simple, pues solo se necesitan un conector de 7 pines entre las placas de Control y de Radiofrecuencia. Las conexiones dentro de la placa de RF se hace por medio de pequeños cables, situados en la parte de debajo de dicha placa y que no ocuparán espacio alguno, así se cumplen las limitaciones de tamaño en los soportes para la placa RF, y la intervención con las señales de radiofrecuencia son menores.

Se adjunta los esquemas donde se puede apreciar el envío de datos en la cadena de desfasadores.

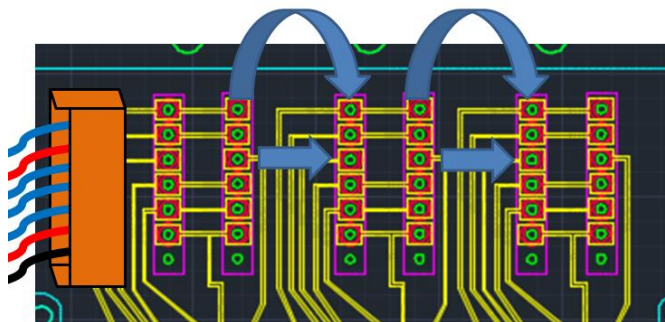


Figura 24. Alternativa 1 Protocolo de Comunicación Array (I)

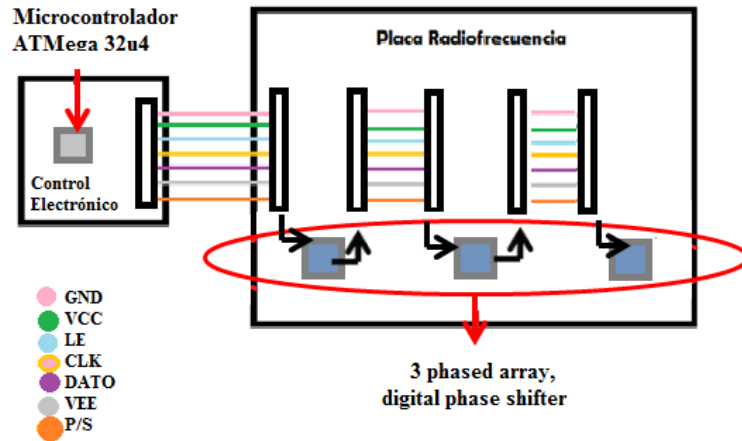


Figura 25. Alternativa 1 Protocolo de Comunicación Array (II)

4.1.2.1.2 Control de la antena mediante el envío de datos de forma independiente a cada desfasador digital

Se envían señales independientes para cada desfasador, por lo tanto, los desfasadores se activarán por señales LE independientes.

El cableado de esta placa es más complicado puesto que aumenta el tamaño de la placa de control, ya que es necesario un mayor número de conectores de salida, y por tanto, aumentará la cantidad de cables de conexión con la placa de RF, pudiendo facilitar la interferencia con las señales de RF y no conseguir limitarnos al espacio disponible.

Se adjunta un esquema donde se puede apreciar el envío de datos a los desfasadores:

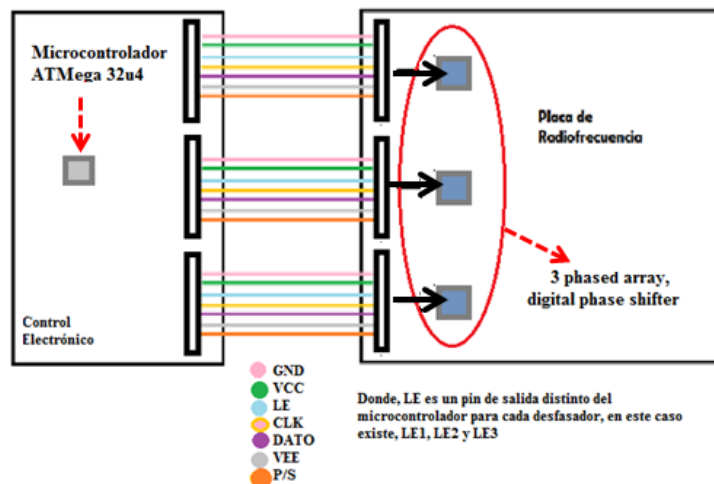


Figura 26. Alternativa 2 Protocolo de comunicación array

Finalmente, se decide utilizar la alternativa daisy chain, puesto que el diseño es mucho más sencillo y eficaz.

4.1.2.2 Diseño placa control electrónica

La placa de control electrónico, es la placa principal, y está formada, por una placa facilitada por el grupo HCTLab, diseñada y construida en un proyecto previo [2], que es la placa que contiene el ATmega32u4, y además la placa interfaz necesaria diseñada y construida en este proyecto para lograr la finalidad de este mismo.

La placa contenedora del microcontrolador, tiene una gran variedad de componentes, pero en este proyecto, se hace uso únicamente de alguno de ellos:

- **Microcontrolador ATmega32u4:** componente principal de control.

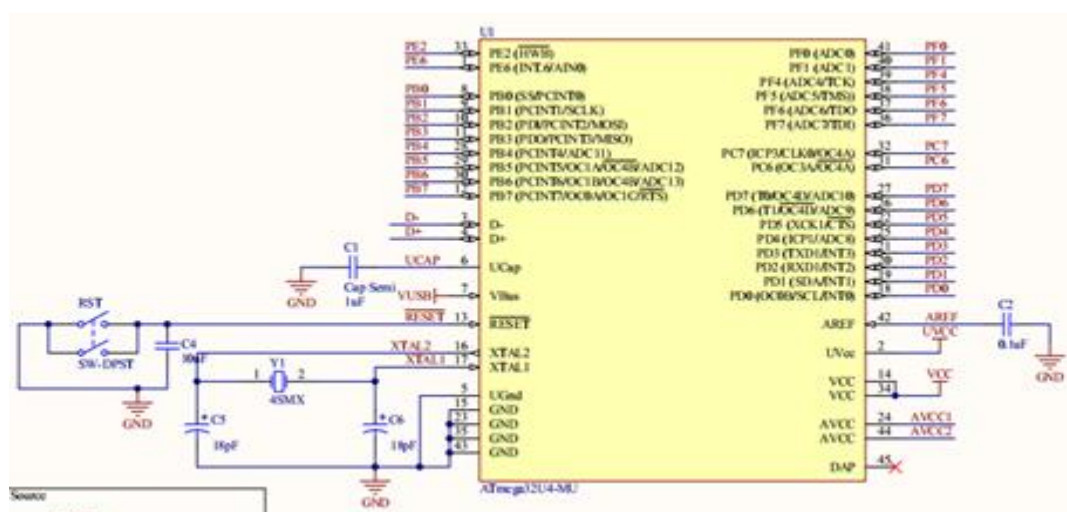


Figura 27. Esquemático ATmega32U4

- **Conector Macho de Entrada del Programador AVRISP mkII:** necesario para la programación del microcontrolador ATmega32u4, está conectado a los pines del PORT B del micro, utilizados para el módulo de comunicación SPI. El AVR, si recibe los datos en grupos de 8 bits.

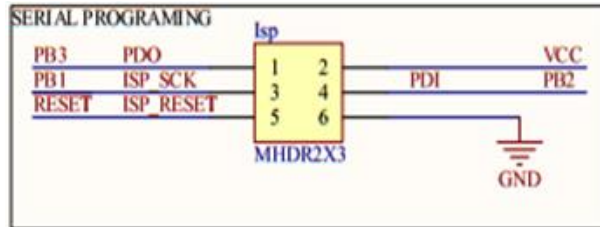


Figura 28. Esquemático conector AVRSPI mkII de la placa principal

- **Conector Mini USB 2.0 SMD:** utilizado para alimentar la placa y para el envío de datos a través del Firmware diseñado finalmente.

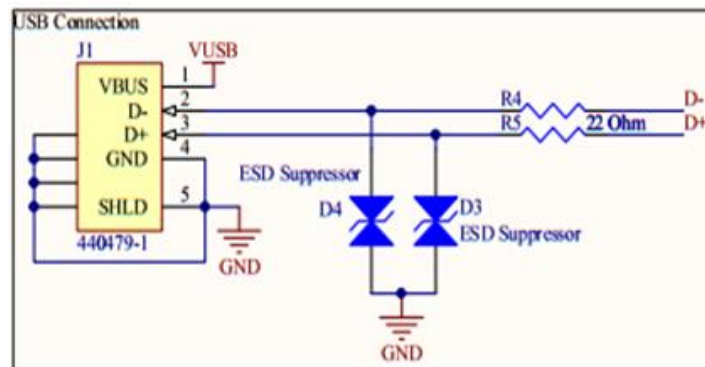


Figura 29. Esquemático conector Mini USB de la placa principal

La placa principal puede ser alimentada por medio del USB o directamente a través de una fuente de alimentación. La elección del modo de alimentación, se selecciona situando los jumpers en la posición correcta del conector.

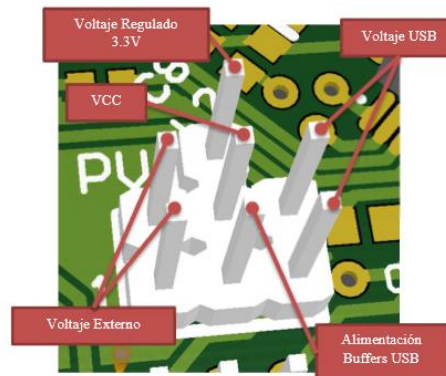


Figura 30. Selector de alimentación de la placa principal

- *LEDs y pulsadores*, utilizados en las pruebas iniciales.

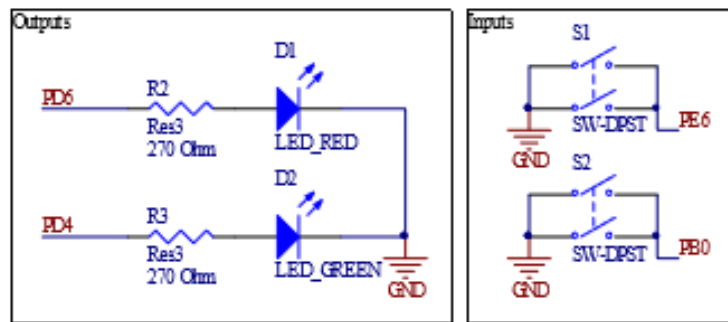


Figura 31. Esquemático LEDs y pulsadores placa principal

- *Reguladores lineales de tensión necesarios.*
- *Conectores genéricos de conexiones a placas externas*

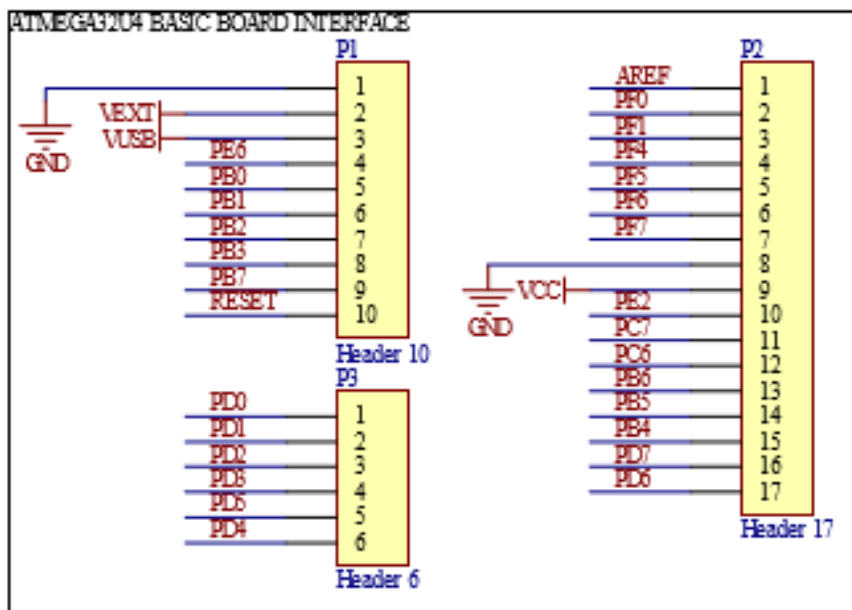


Figura 32. Conectores genéricos placa principal

Además en la placa interfaz que se ha incorporado en este proyecto:

- *Conectores genéricos* necesarios para la conexión con la placa ATMega32u4
- *Clema Alimentación*: Para introducir la alimentación negativa que necesita el Desfasador Digital.
- *Conector Macho de 7 pines*: Es el conector de salida de las señales necesarias para la comunicación con la placa de radiofrecuencia. Estas placas también se han diseñado con el software Altium Designer 13.3.

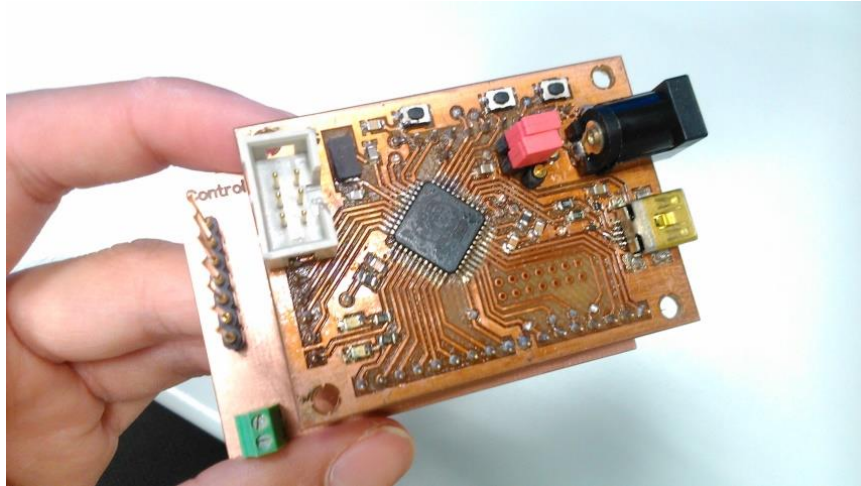


Figura 33. Placa de control electrónico principal

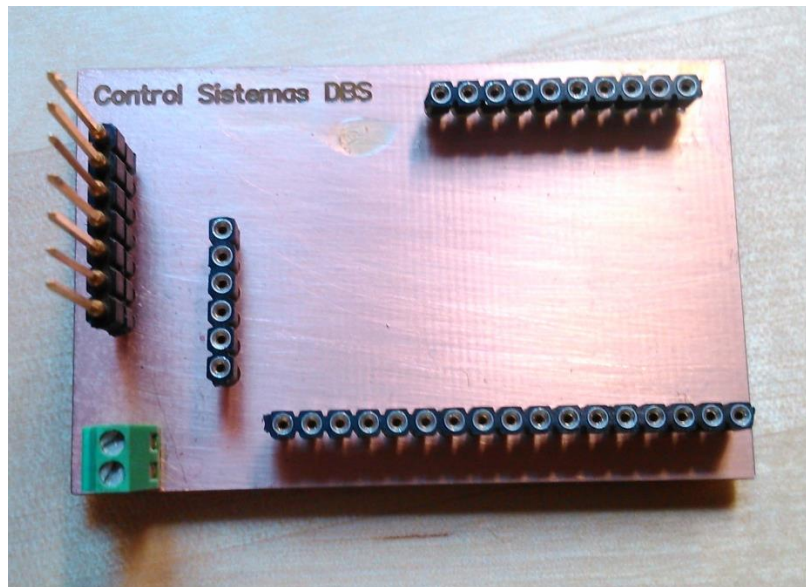


Figura 34. Placa de Interfaz de Control

Las salidas de conector macho de esta placa se conectan a través de una tira plana directamente a la placa de Radiofrecuencia.

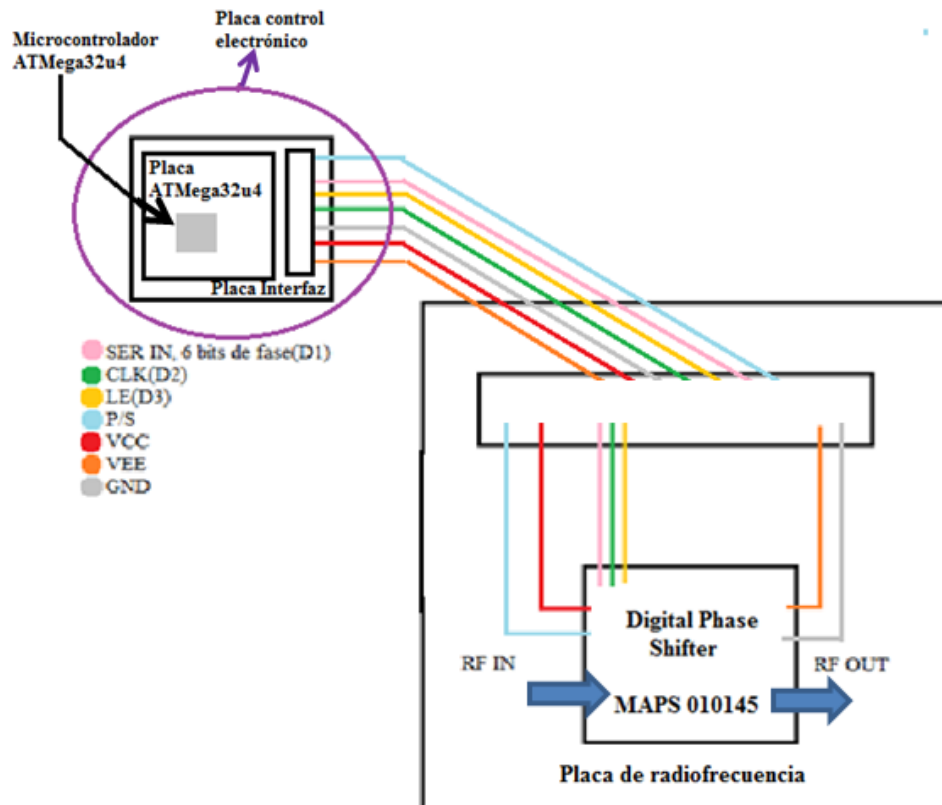


Figura 35. Esquema de conexión entre la placa de control serie y la placa RF

- *P/S*: Controla el Modo de Control de la antena, en este caso la seria 1 (Serie), conectado al Pin 2.
- *VCC*: Coincide con el voltaje de VUSB, conectado al Pin 18.
- *VEE* es de Control de Fase en el Modo Serie, se obtendrá a través del PORT D2 del Microcontrolador, conectado al Pin 24 del Desfasador Digital.: Obtenido de la clema de alimentación, conectado al Pin 1.
- *SER IN*: Pin
- *CLK*: Señal de Reloj, conectado al Pin 23 del Desfasador Digital.
- *LE*: Señal de Enable a bajo, conectada al Pin 22 del Desfasador Digital.
- Pin de *GND* necesario, conectado a un plano tierra de la Placa de Radiofrecuencia, en la capa Boton

4.1.3 Placa de RF

El componente principal de la placa de radiofrecuencia es el desfasador digital de fase MAPS 010146 (Datasheet en Anexo II). La función principal de este desfasador es la de proporcionar 16 estados de fase RF posibles por medio de estados altos y bajos CMOS sobre 6 bits de control, siendo los 4 MSB los que nos proporcionan el desfase.

Tabla 3. Estados digitales respecto a la fase

D6	D5	D4	D3	D2	D1	Phase Shift
0	0	0	0	X	X	0°
0	0	0	1	X	X	- 22.5°
0	0	1	0	X	X	- 45°
0	0	1	1	X	X	- 67.5°
0	1	0	0	X	X	- 90°
0	1	0	1	X	X	- 112.5°
0	1	1	0	X	X	- 135°
0	1	1	1	X	X	- 157.5°
1	0	0	0	X	X	- 180°
1	0	0	1	X	X	- 202.5
1	0	1	0	X	X	- 225°
1	0	1	1	X	X	- 247.5°
1	1	0	0	X	X	- 270°
1	1	0	1	X	X	- 292.5°
1	1	1	0	X	X	- 315°
1	1	1	1	X	X	- 337.5°

Las placas de Radiofrecuencia han sido diseñadas usando el software AutoCAD 2015. Ya que en un principio se intentó diseñarlas con Altium Designer como las placas de control, pero el diseño de las líneas de RF no era nada sencillo y podría empeorar la radiación, así que para evitar problemas se decide utilizar AutoCAD.

Los diseños en AutoCAD no diferencian entre Esquemático y PCB como en el caso de Altium Designer. Únicamente existe una plantilla donde se irán introduciendo las formas que posteriormente imprime la fresadora.

La placa inicial de radiofrecuencia diseñada contiene únicamente un Desfasador Digital MAPS 010146, y no sigue ninguna topología. En esta placa es donde se realizan las pruebas tanto en serie como paralelo para verificar resultados.

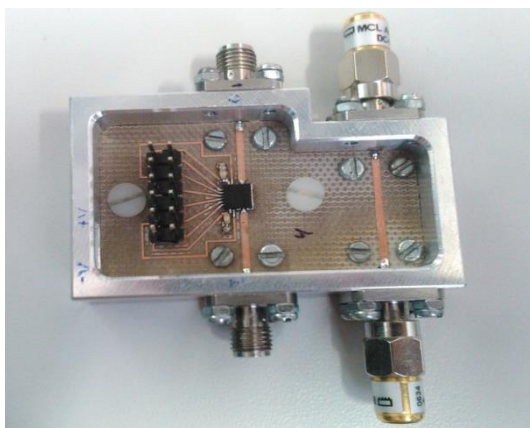


Figura 36. Placa RF en soporte con un desfasador digital

La placa final de radiofrecuencia que se controla en este PFC contiene un array lineal de 3 desfasadores, que como se ha mencionado anteriormente siguen la topología daisy chain, ya que es una cadena de enlaces sin retorno.

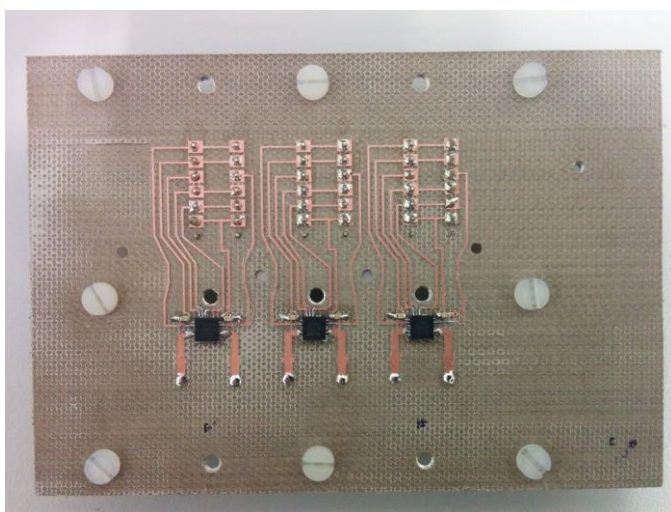


Figura 37. Placa RF con 3 phased array (parte de adelante)

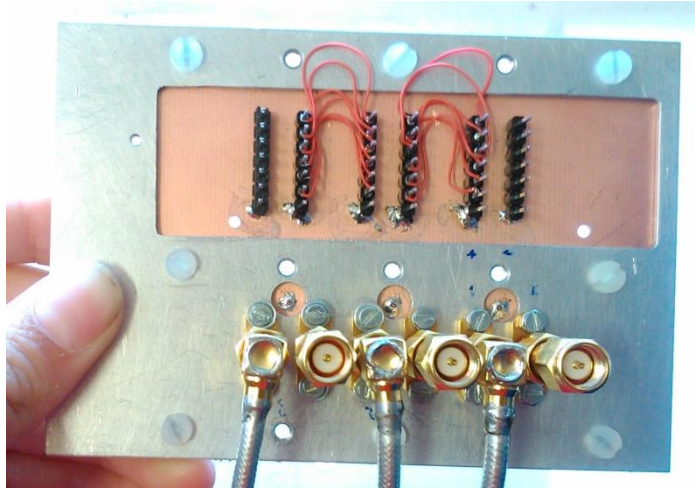


Figura 38. Placa RF con 3 phased array (parte de atrás)

Para la construcción de las placas, se ha utilizado la fresadora disponible en el taller, enviándola a esta los archivos gerber exportados del proyecto Altium (Anexo E) y en el caso de la placa de RF a partir de su único diseño.

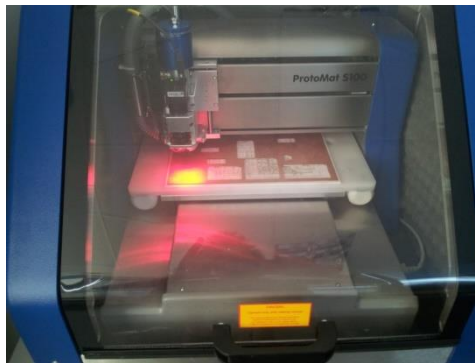


Figura 39. Fresadora circuitos impresos

Posteriormente, se han soldado los componentes manualmente con un soldador básico.

4.2 Programación del microcontrolador en Atmel Studio

En un principio se ha estudiado profundamente el microcontrolador *ATMega32u4* ya que es la parte principal del sistema de control electrónico diseñado, utilizando un Programador AVRISP mkII que es necesario para cargar cualquier programa en el procesador.

Las primeras pruebas se hacen en el software Atmel Studio 6.1 que hay disponible de forma gratuita en la página de Atmel.

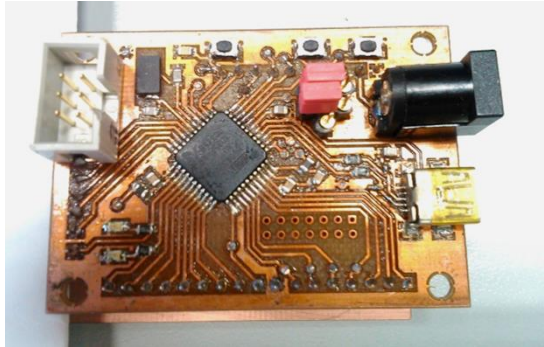


Figura 40. Placa Microcontrolador ATMega32U4 (placa principal)



Figura 41. Programador AVRISP mkII



Figura 42. Logotipo Atmel Studio

4.2.1 Programa de control de placa de RF

Tras entender el funcionamiento de la placa de control y diseñar cada una de ellas, se crea el código en ATmel Studio, que controlará de 1 hasta infinitos desfasadores del phased array. La modificación del número de desfasadores que se quieren controlar, se consigue modificando el número de iteraciones del bucle existente en el código del programa (haciéndolo coincidir con el número de desfasadores) y aumentando el número de bits de la variable “cod” a enviar. Para cargar este código en el microcontrolador, es necesario utilizar el programador AVR mkII del que ya se ha hablado.

```
#include <avr/io.h
#define LE PORTD0
#define CLK PORTD1
#define DATO PORTD2
#define PS PORTD3
#define LED PORTD4
void Enviar_dato(int dat){
    if (dat == 1){ //envio un 1 o 0 dependiendo del dato de entrada
        PORTD |= _BV(DATO);
    }else{
        PORTD &= ~(_BV(DATO));
        PORTD |= _BV(LED); //PS = 1 para configuración serie
    }
    if (dat == 1){ //envio un 1 o 0 dependiendo del dato de entrada
        PORTD |= _BV(LED); //PS = 1 para configuración serie
    }else{
        PORTD |= _BV(LED); //PS = 1 para configuración serie
    }
    PORTD &= ~_BV(CLK); //CLK LOW
    PORTD |= _BV(CLK); //CLK high
}
int main(void)
{
    DDRD = _BV(DATO) | _BV(LE) | _BV(CLK) | _BV(LED) | _BV(PS);
    int i;
    int cod[] = {0,0,0,0,0,0, 0,0,0,1,1,0, 0,0,1,0,0,1}; // D6 D5 D4 D3 D2 D1
    PORTD |= _BV(PS); //Configuración serie
    PORTD &= ~(_BV(LE)); //LE = 0 para que empiece la transmisión
    for( i=0; i < 18; i++){
        Enviar_dato(cod[i]);
    }
    PORTD |= _BV(LE); //LE = 1, cuando se ha transmitido todos los códigos
}
```

Para comprobar que la transmisión de datos coincide con la que la antena RF necesita para modificar su desfase, se utiliza el osciloscopio *Agilent MSO-X-3014A*, disponible en el laboratorio HCTLab, y que nos permite dibujar la forma de las señales de interés para ver que el protocolo de envío se cumple.



Figura 43. Osciloscopio Agilent MSO-X-3014A

Para visualizar las señales correctamente, debemos seleccionar el eje de tiempos adecuado. Los pasos a seguir:

1. Se selecciona el modo de acoplamiento “Normal”.
2. Es útil, poner un trigger cuando la señal LE HIGH (cuando finaliza el envío de datos).
3. Después en la opción “Digital”, seleccionamos las salidas que queremos mostrar por pantalla. Externamente con el *Logic Analyzer Probe Cable* se conecta el osciloscopio en los pines del puerto D de entrada y salida del sistema, para así verificar, que la señal de entrada coincidía con la señal de salida después 6 ciclos de reloj. Las señales obtenidas, coinciden con las facilitadas en el datasheet del dispositivo (Anexo II).

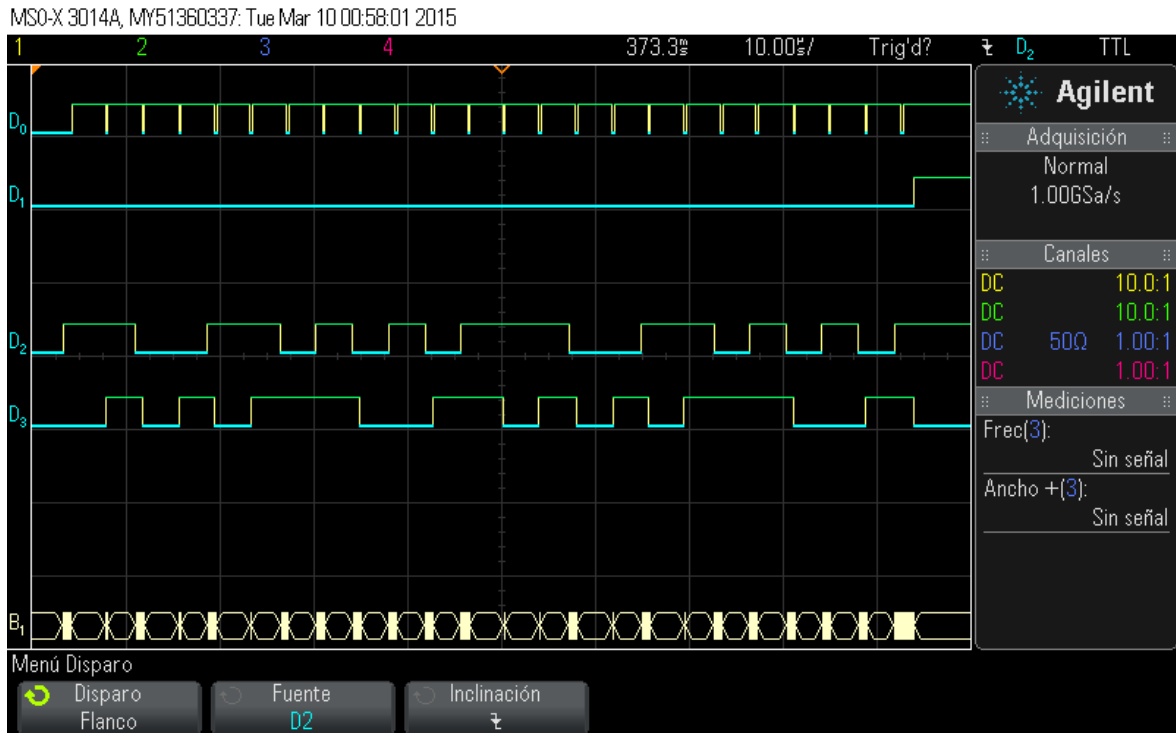


Figura 44. Resultado osciloscopio de envío de datos a la placa RF

Dónde:

- D0 es CLK
- D1 es LE
- D2 es el dato de entrada a la placa RF(SER IN)
- D3 es el dato de entrada a la placa RF(SER IN), pero con un delay de 6 ciclos de reloj

4.3 Desarrollo software

Una vez terminada la parte hardware del sistema de control para la antena de apuntamiento, es necesario crear una Interfaz de Usuario que de usabilidad y accesibilidad al sistema. Consiste en una Interfaz usando un PC que permita el modificar el desfase de la antena a través de la selección de opciones en una pantalla. Como ya se ha mencionado en el apartado de diseño, el protocolo de comunicación entre el PC y el microcontrolador, se hace por medio de USB, aprovechando el conector Mini USB, disponible en la placa facilitada en el proyecto[2].

El protocolo de comunicación USB(Bus Universal en Serie) es un bus estándar industrial que define los cables, conectores y protocolos usados en un bus para conectar, comunicar y proveer de alimentación eléctrica entre computadoras, periféricos y dispositivos electrónicos. Para utilizar este estándar de comunicación con AVR, es necesario utilizar una librería especial. Existen varias librerías que permiten implementar el protocolo de comunicación USB, en este PFC se ha utilizado la librería LUFA, que permite

hacer que los microcontroladores sean compatibles con el protocolo de comunicación USB. En concreto se utiliza la Demo VirtualSerial, ya que como se puede apreciar en el diagrama de flujos adjunto a continuación, la comunicación se realiza a través del envío de datos por un puerto serie, que es lo que se trata en dicha demo. Así, únicamente serán necesarias algunas modificaciones dentro de esta demo, que se mencionan más adelante en este documento.

El diagrama de flujos, que sigue dicha interfaz:

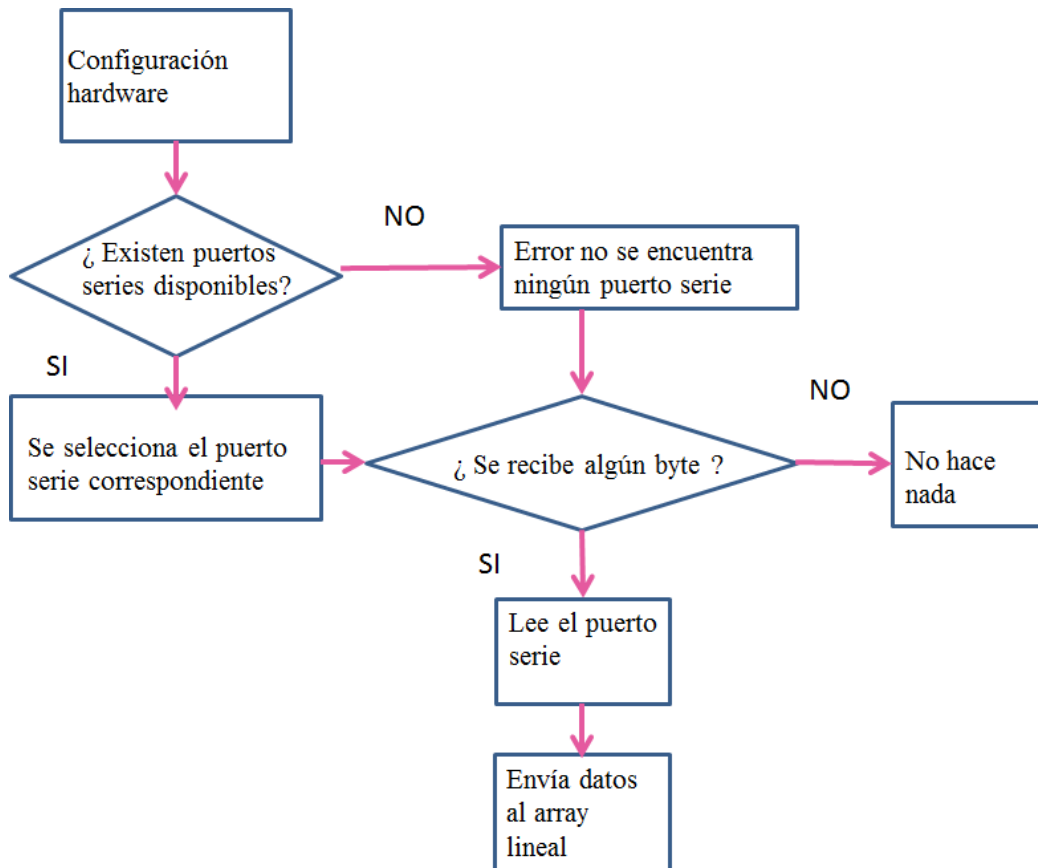


Figura 45. Diagrama de flujo Software

4.3.1 Introducción a LUFA

La comunicación USB por un puerto serie se realiza a través de un Endpoint o buffer. Cuando el Host quiera enviar información al dispositivo escribe en un Endpoint, y el dispositivo leerá la información cuando lo desee.

- Un Endpoint es un canal unidireccional, puede ser IN (dispositivo → Host), OUT(Host → Dispositivo) o IN/OUT (bidireccional).
- Un Endpoint puede diferenciarse también por el tipo de comunicación:
 - Bulk: Para comunicaciones que necesitan de ancho de banda, donde la latencia no es crítica y el ancho de banda no está garantizado.

- Interrupt: La latencia está garantizada, pero el ancho de banda es mejor. El Host atiende estas peticiones periódicamente.
- Isocronus: Garantiza una latencia determinada y dispone de un gran ancho de banda. Pero no garantiza el envío de datos.
- Para un puerto serie se utilizan 4 endpoints:
 - Endpoint 0: es el endpoint de control, encargado de enumerar y controlar la comunicación USB.
 - Endpoint 1: es de tipo Interrupt y es el encargado de la supervisión y control de los parámetros de control serie: conexión, desconexión, velocidad...
 - Endpoint 2 y 3: Endpoint de entrada y salida por donde se reciben y envían los datos del puerto serie.

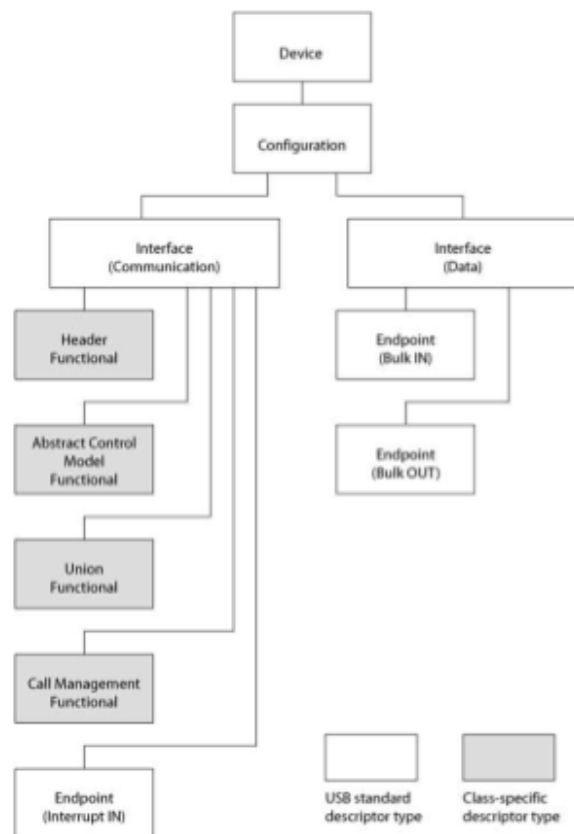


Figura 46. Endpoint puerto serie

4.3.2 Demo VirtualSerial de LUFA en Linux

Antes de poder programar el micro, es necesaria la instalación de los drivers del AVRISP mkII:

- **avr-gcc:** El gcc compilador para AVR
- **avrdude:** Permite utilizar las memorias Flash y EEPROM del AVR.
- **Avr-libc:** Librería para programación C de los microcontroladores con AVR.

Una vez instalados los driver necesarios, a través de los archivos Descriptors.c y Descriptors.h (disponibles en la Demo VirtualSerial), el equipo tiene información acerca del AVR que se utiliza, sus características y los endpoints disponibles. Este proceso conocido como enumeración es inicializado por el host cuando detecta que se ha conectado un nuevo dispositivo adjunto al bus, y habilita su configuración permitiendo la transferencia de datos sobre el este. La parte más importante a tener en cuenta son los códigos VID y PID, localizados en Descriptors.c, que se corresponden con el número de proveedor USB y del producto.

```
.VendorID          = 0x03EB,  
.ProductID         = 0x2044,
```

Figura 47. VID y PID

Los que vienen por defecto en el fichero LUFA descargado se pueden utilizar, ya que son los códigos de LUFA CDC y no se han modificado, pero en el caso de que se quisiera distribuir el producto podrían cambiarse.

En el archivo VirtualSerial.c si se ha modificado el código, ya que es el archivo donde se encuentra el main y donde se tendrá que definir cómo se envían los datos. Para la recepción de datos del PC se utiliza la función **ReceiveByte** que retorna el byte enviado, y en caso de que no se haya enviado nada retorna un valor negativo. El código utilizado se adjunta a continuación.

Nota: no olvidar definir los puertos utilizados como entradas y salidas.

```
int main(void)  
{  
    DDRD = _BV(DATO) | _BV(LE) | _BV(CLK) | _BV(LED) | _BV(PS);  
    char ReportString[256];  
    int16_t byteRecibido;  
    int16_t byteRecibido2;  
    int i = 0;  
    SetupHardware();  
    CDC_Device_CreateStream(&VirtualSerial_CDC_Interface, &USBSerialStream);  
    LEDs_SetAllLEDs(LEDMASK_USB_NOTREADY);
```

```

GlobalInterruptEnable();
for (;;)
{
    byteRecibido=-1;
    byteRecibido2=-1;
    while(byteRecibido<0) //Mientras no se reciba ningún bytes
    {
        byteRecibido2 = byteRecibido;
        byteRecibido =CDC_Device_ReceiveByte(&VirtualSerial_CDC_Interface);
        if(byteRecibido<0)
        {
            CDC_Device_USBTask(&VirtualSerial_CDC_Interface);
            USB_USBTask(); }
        byteRecibido=CDC_Device_ReceiveByte(&VirtualSerial_CDC_Interface);
        //Configuración para comenzar la transmisión de datos
        PORTD |= _BV(PS); //Configuración serie
        PORTD &= ~(_BV(LE)); //LE = 0 para que empiece la transmisión
        PORTD &= ~(_BV(LED)); //LE = 0 para que empiece la transmisión
while(byteRecibido2 != byteRecibido & i<3){
    if ((byteRecibido & 0x20) != 0x00){ PORTD |= _BV(DATO);
    }else{ PORTD &= ~(_BV(DATO));}
    PORTD &= ~_BV(CLK); PORTD |= _BV(CLK);
    if ((byteRecibido & 0x10) != 0x00){PORTD |= _BV(DATO);
    }else{PORTD &= ~(_BV(DATO));}
    PORTD &= ~_BV(CLK); PORTD |= _BV(CLK);
    if ((byteRecibido & 0x08) != 0x00){PORTD |= _BV(DATO);
    }else{PORTD &= ~(_BV(DATO));}
    PORTD &= ~_BV(CLK); PORTD |= _BV(CLK);
    if ((byteRecibido & 0x04) != 0x00){PORTD |= _BV(DATO);
    }else{PORTD &= ~(_BV(DATO));}
    PORTD &= ~_BV(CLK); PORTD |= _BV(CLK);
    if ((byteRecibido & 0x02) != 0x00){ PORTD |= _BV(DATO);
    }else{PORTD &= ~(_BV(DATO));}
    PORTD &= ~_BV(CLK); PORTD |= _BV(CLK);
    if ((byteRecibido & 0x01) != 0x00){PORTD |= _BV(DATO);
    }else{PORTD &= ~(_BV(DATO));}
    PORTD &= ~_BV(CLK); PORTD |= _BV(CLK);
    byteRecibido2 = byteRecibido;
    CDC_Device_USBTask(&VirtualSerial_CDC_Interface);
    USB_USBTask();
    i++;
}
    if(i == 3){
        PORTD |= _BV(LE);i=0;
    }
}

```

También es necesario la modificación del archivo Makefile, porque que se necesita definir el nombre del Micro utilizado (MCU), así como del AVR (ARCH), la frecuencia de CPU (F_CPU) y la frecuencia USB (F_USB).

```
MCU      = atmega32u4
ARCH     = AVR8
BOARD    = USBKEY
F_CPU    = 1000000
F_USB    = $(F_CPU) // Significa que F_USB = F_CPU
OPTIMIZATION = s
TARGET   = VirtualSerial
SRC      = $(TARGET).c Descriptors.c $(LUFA_SRC_USB) $(LUFA_SRC_USBCLASS)
LUFA_PATH = ../../../../LUFA
CC_FLAGS = -DUSE_LUFA_CONFIG_HEADER -IConfig/
LD_FLAGS =
# Default target
all:
# Include LUFA build script makefiles
include $(LUFA_PATH)/Build/lufa_core.mk
include $(LUFA_PATH)/Build/lufa_sources.mk
include $(LUFA_PATH)/Build/lufa_build.mk
include $(LUFA_PATH)/Build/lufa_cppcheck.mk
include $(LUFA_PATH)/Build/lufa_doxygen.mk
include $(LUFA_PATH)/Build/lufa_dfu.mk
include $(LUFA_PATH)/Build/lufa_hid.mk
include $(LUFA_PATH)/Build/lufa_avrdude.mk
include $(LUFA_PATH)/Build/lufa_atprogram.mk
```

El resto de archivos disponibles en la demo no han sido modificados.

La ejecución de esta demo permite programar el micro por medio del programador AVR, y que así posteriormente pueda ser utilizado en un PC sin necesidad de estar conectado al programador.

Esta programación podría hacerse en Windows pero resultaría más complejo, ya que como se facilita el archivo makefile en Linux es bastante sencillo.

4.3.3 Control del sistema de apuntamiento con una terminal windows

Una vez cargado el programa en el ATmega32u4 con la librería LUFA en la plataforma Linux, por medio del programador AVR, es en la plataforma Windows donde se diseña la interfaz para la comunicación del usuario directamente con el micro.

Para que el PC reconozca el micro sin el programador, se configuran las propiedades del dispositivo en Panel de Control.

Para ello se debe actualizar el controlador, esto se consigue modificando la ruta del controlador a la carpeta donde se encuentre el archivo .INF facilitado en la Demo VirtualSerial (es el archivo que tiene un icono de setup).

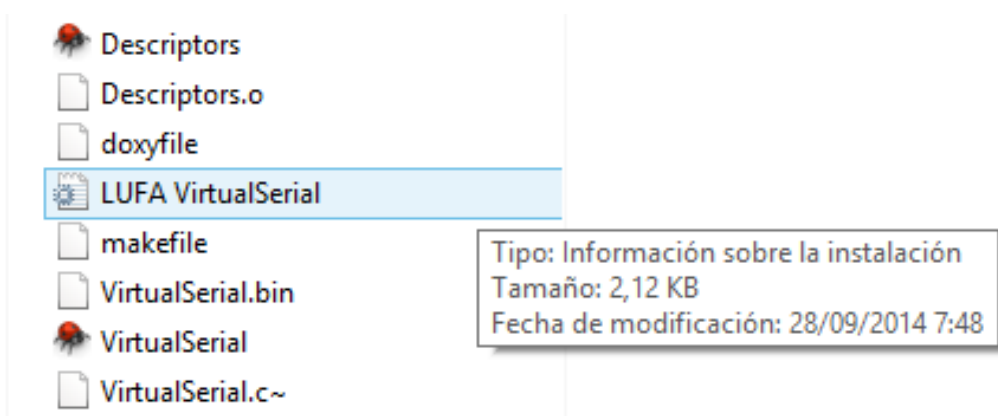


Figura 48. Icono archivo .inf

Cambiar configuración → Actualizar controlador, y seleccionaríamos la ruta a la carpeta donde se encuentre el archivo.inf. Es recomendable, desarrollar esto en el sistema operativo Windows 7.

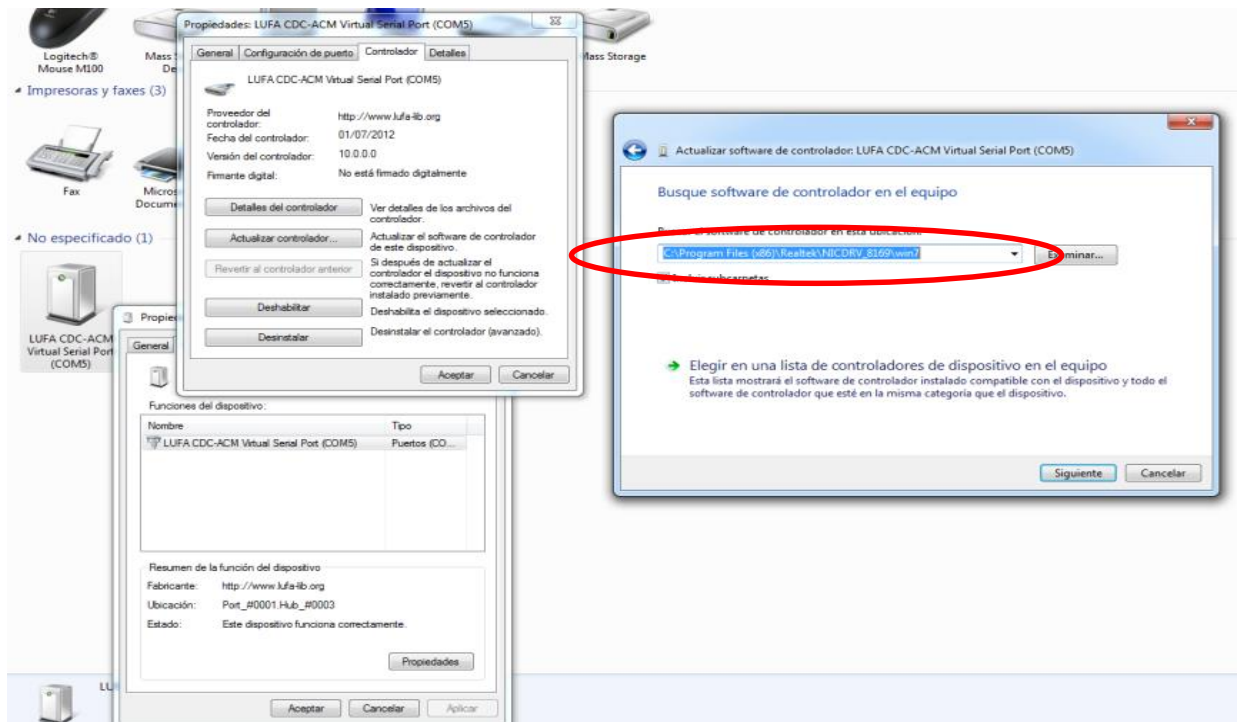


Figura 49. Actualización controlador

4.3.3.1 Control mediante una terminal disponible en la red

En un dis se utiliza la terminal RealTerm disponible de forma gratuita en internet, para comprobar que el modo de transmisión se ha programado correctamente, esta es una terminal cualquiera, se podría a ver utilizado la disponible en Windows.

En la terminal RealTerm se configura la información del puerto serie. Esta información se obtiene de las propiedades del dispositivo dentro de Panel de Control, pinchando con el botón derecho sobre él. Posteriormente en la pestaña hardware (dentro de sus propiedades), configuramos el puerto. Esta información de configuración es la que posteriormente se introduce en la terminal.

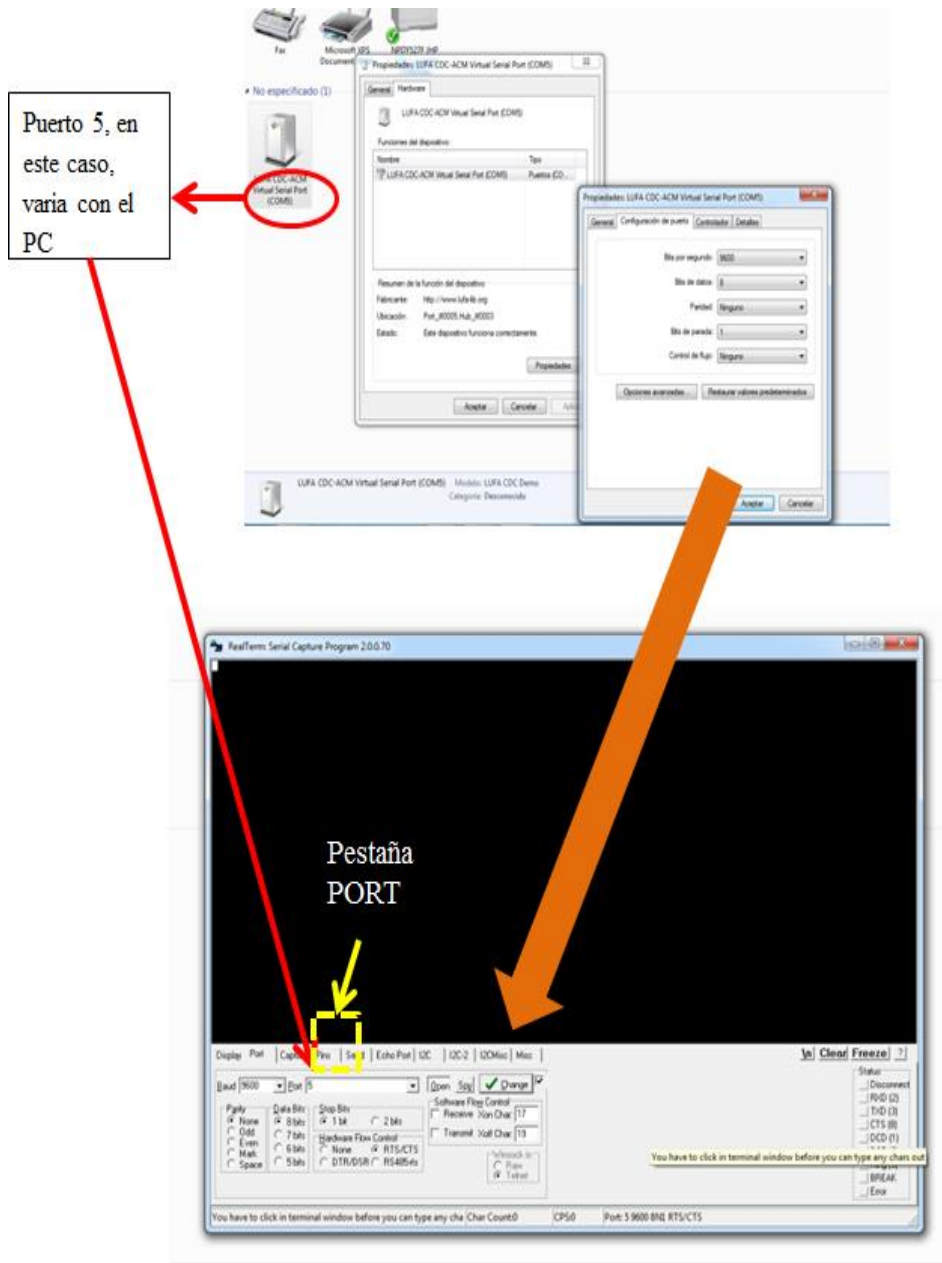


Figura 50. Terminal RealTerm (I)

- Número de puerto serie, dependerá del ordenador.
- Velocidad en baudios(9600)
- Bits de stop(1)
- Bits de paridad(Ninguno)

Y se abre el puerto con el botón “Open”.

Donde D0 es LE, D1 es CLK y D2 los datos de salida del microcontrolador.

Se comprueba que al enviar en la interfaz:

- Un 5 (000101)
- Un 2 (000010)
- Un 9 (001001)

Como los bits de desfase son los 4 MSB, y el primer byte que se envía corresponde al último desfasador de la cadena, desfasarían según la tabla de desfases del MAPS 010146:

- El primer desfasador 45°.
- El segundo desfasador 0°.
- El tercer desfasador 22.5°.

4.3.3.2 Interfaz gráfica personalizada en Java Eclipse

1. En primer lugar, para utilizar Java Eclipse es necesario descargarse la librería .JRE o .JDK. Disponible en Oracle. En este caso, se ha descargado el fichero .jre1.8.0_45.

The screenshot shows the Oracle Java SE Downloads page. The main content area is titled 'Java SE Downloads' and features two primary download options: 'Java Platform (JDK) 8u45' and 'NetBeans with JDK 8'. Below these, the 'Java Platform, Standard Edition' section is expanded to show 'Java SE 8u45', which includes a security update. A list of links is provided for installation instructions, release notes, licenses, and product information. The 'Readme Files' section is also expanded, showing links for 'JDK ReadMe' and 'JRE ReadMe'. The 'JRE' download button is highlighted with a red circle.

Figura 53. Descarga Java

2. Se descarga el Programa Eclipse disponible de forma gratuita, este programa nos permite crear aplicaciones de usuario, en distintos lenguajes de programación. Existe varias versiones, en este PFC se ha utilizado:

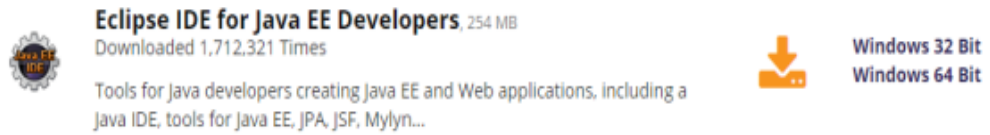


Figura 54. Logotipo Java Eclipse

3. Si queremos utilizar la herramienta de Windows Builder, que nos facilita la creación de ventanas de interfaces sin necesidad de entender la programación java en profundidad. En el sector de herramientas dentro de la página de Eclipse se selecciona Windows Builder y se pone a descargar. Después dentro del programa Eclipse Help -> Install New Software

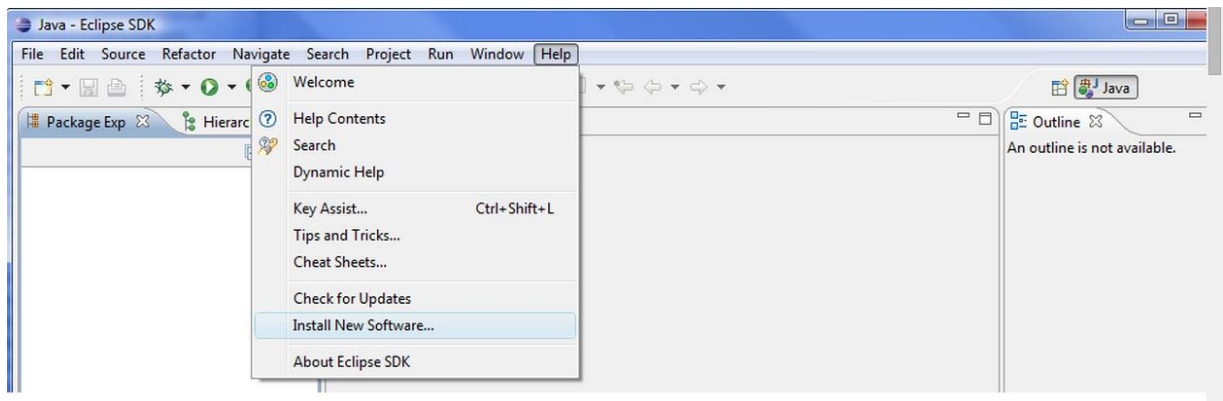


Figura 55. Instalación Windows Builder (I)

Y copiamos la URL de la página de descarga donde se indica.

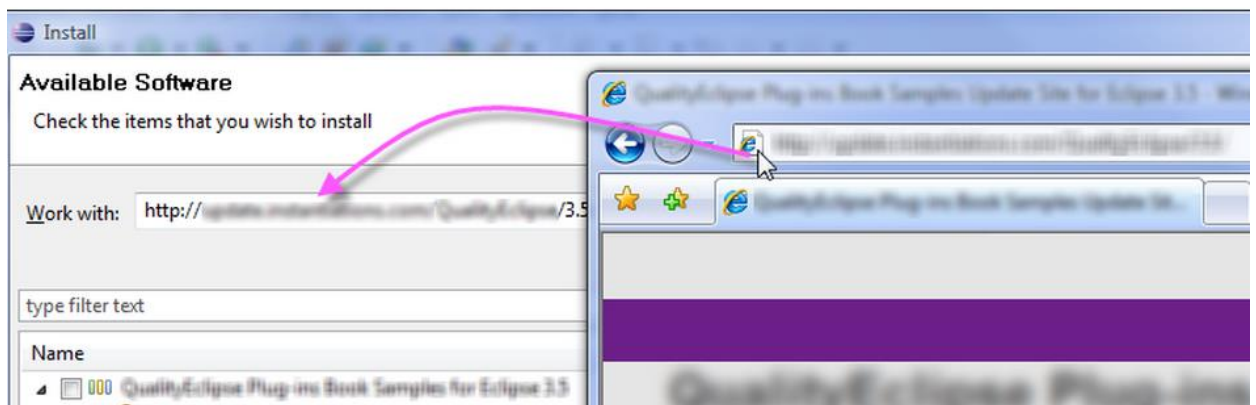


Figura 56. Instalación Windows Builder (II)

Para utilizar las funcionalidades del puerto serie dentro de Java Eclipse, es necesario:

1. Descargarse el fichero javax.comm.
2. Copiar el archivo “win32com.dll” en la carpeta “bin” dentro del fichero JRE donde se encuentre direccionado java en el PC, se puede comprobar en el panel de control.
3. Copiar el archivo “javax.comm.properties” en la carpeta “bin” dentro del fichero JRE donde se encuentre direccionado java en el PC, se puede comprobar en el panel de control.
4. Copiar el archivo “comm.jar” en la carpeta “lib” dentro del fichero JRE donde se encuentre direccionado java en el PC, se puede comprobar en el panel de control. Configurar Built Path, dentro de las opciones del proyecto.

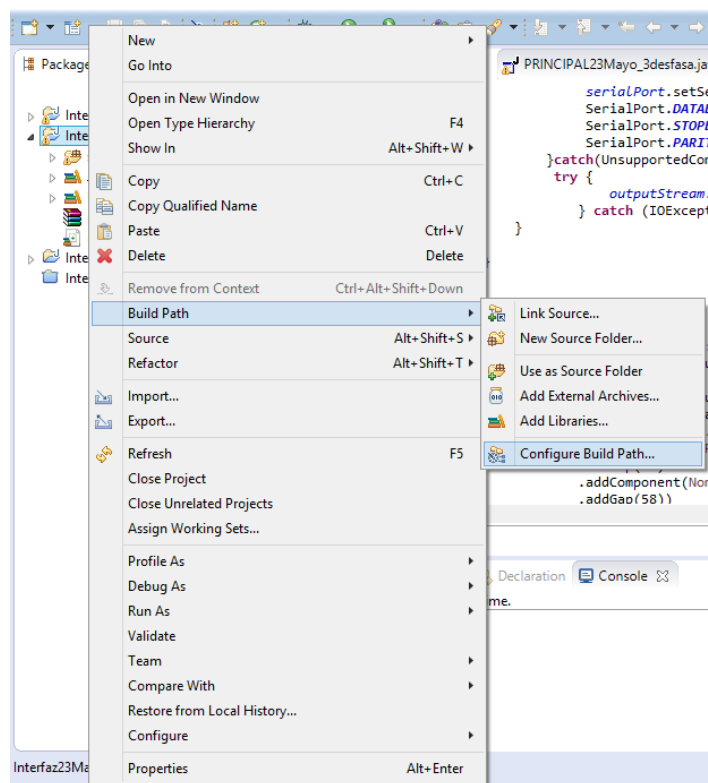


Figura 57. External JARs Eclipse(I)

5. En la opción “Add External JARs” de la pestaña Libraries seleccionar la librería externa, en este caso comm.jar:

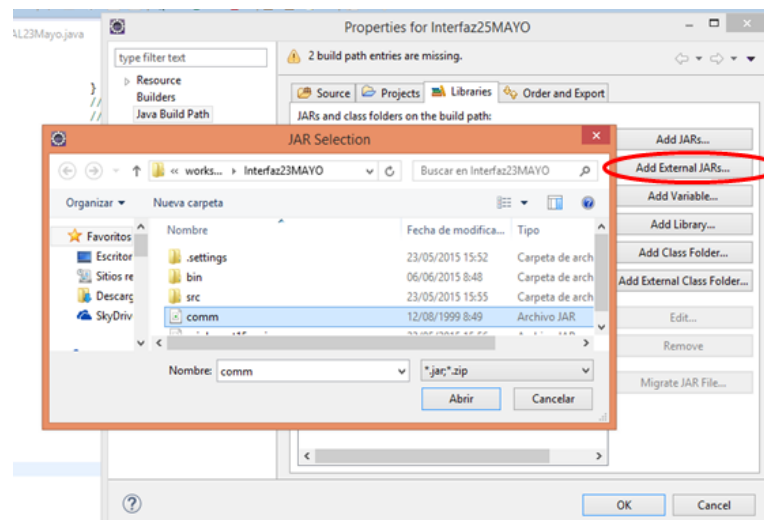


Figura 58. External JARs Eclipse(II)

6. Se exporta la librería seleccionándola en la pestaña “Order and Export”, ya que no se hace automáticamente y se da a OK.

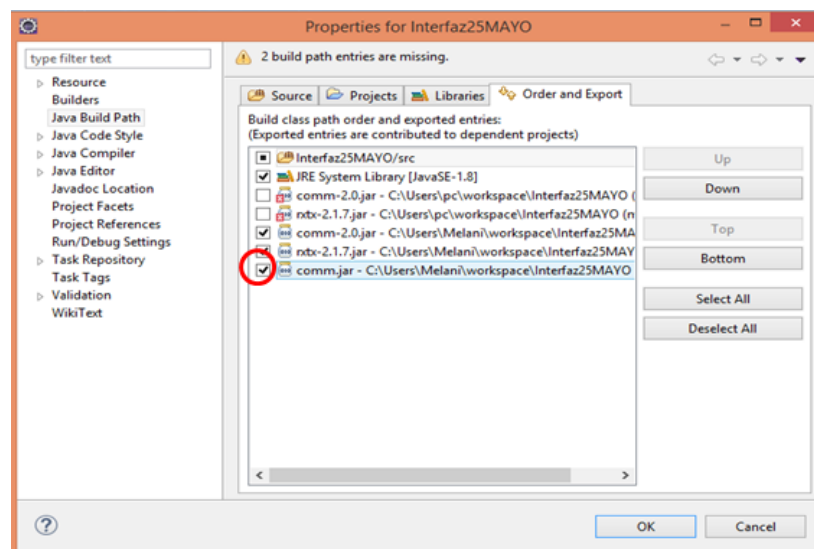


Figura 59. External JARs Eclipse(III)

La interfaz de usuario contiene un menú principal, donde se puede seleccionar el número de desfasadores con el que se quiere trabajar.

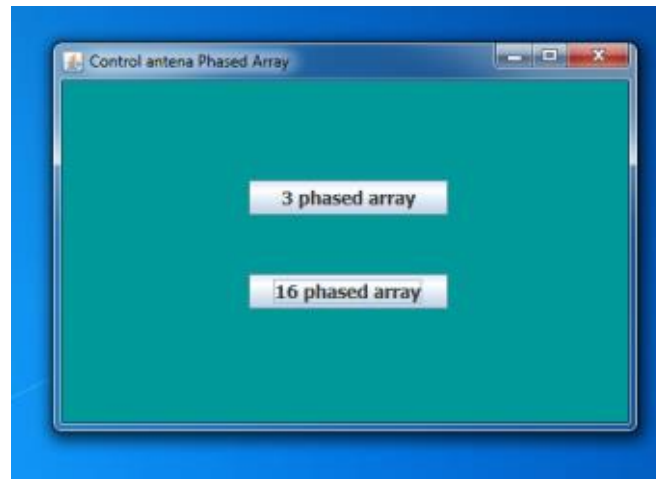


Figura 60. Interfaz usuario, menú principal

En este PFC, el objetivo es el control de un array lineal de 3 desfasadores, por lo que solo se utiliza la primera opción del menú. Si seleccionamos esta opción, aparece una ventana de configuración de desfase para 3 elementos:

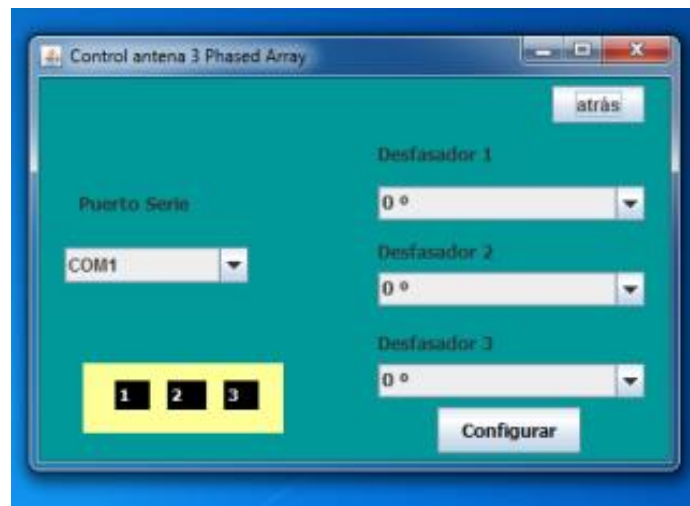


Figura 61. Interfaz usuario, 3 phased array

En la ventana anterior se puede observar:

- Un menú desplegable, donde se debe seleccionar el puerto serie del microcontrolador. El puerto serie donde está conectado, se puede verificar en el panel de control de dispositivos.
- Tres menús desplegables para seleccionar el desfase introducido en cada elemento.

Una vez completadas todas las opciones, pulsando el botón “Configurar” la antena apuntará a la dirección correspondiente.

En el menú inicial de esta interfaz también existe la opción de control de 16 desfasadores, cuya ventana de configuración es:



Figura 62. Interfaz usuario, 16 phased array

Para utilizar esta ventana de configuración, antes deberíamos modificar algunos datos de programación, tanto en la programación del micro modificando la Demo VirtualSerial en el entorno LUFA, como el código de diseño de la interfaz en Java Eclipse.

Las modificaciones necesarias se mencionan en el último punto del apartado de desarrollo, ya que el grupo de investigación RFCAs, busca controlar un array de 16 desfasadores en un futuro cercano. Est

Para comprobar que la interfaz funciona correctamente, usando las funciones de envío del puerto serie serie de la librería exportada comm.jar, se finaliza el diseño de la interfaz y por tanto la parte software del PFC. Los resultados vistos en el osciloscopio que verifica el funcionamiento viendo que envía los datos correctamente, se ven en las figuras adjuntas.

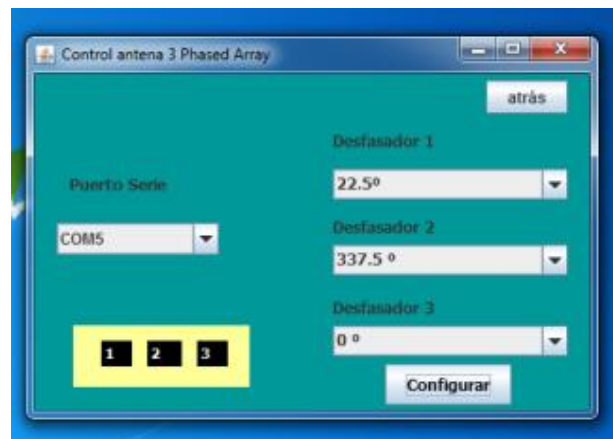


Figura 63. Interfaz usuario envío a la placa RF 3 phased array

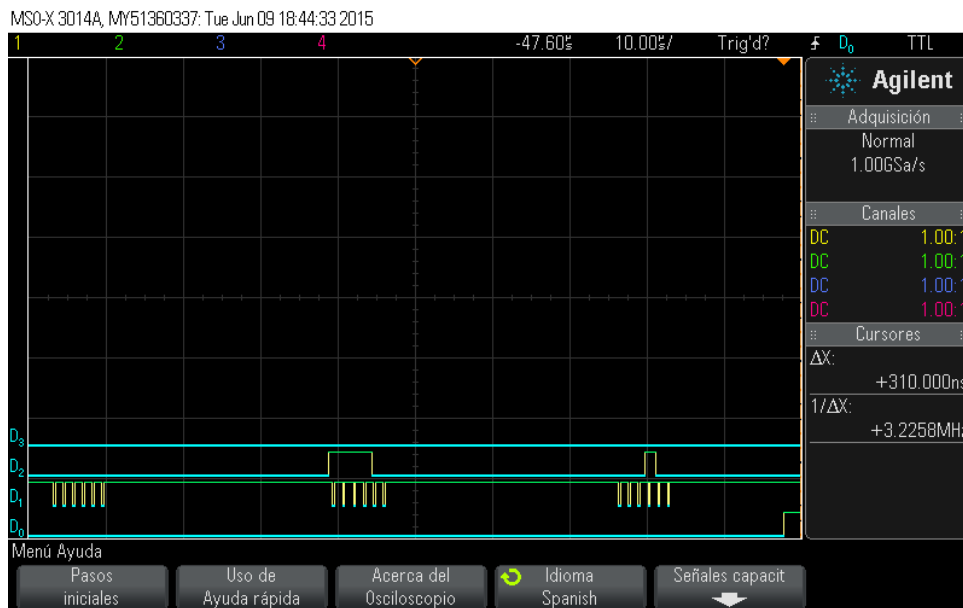


Figura 64. Señales osciloscopio envío de datos con interfaz personal

Donde D0 es LE, D1 es CLK y D2 los datos de salida del microcontrolador.

Observando que el primer dato enviado, es el correspondiente al último desfasador del array.

Para exportar el archivo .jar correspondiente a la interfaz, y así poder acceder a ella desde el escritorio o cualquier otro PC. Es muy sencillo:

1. Archivo → Export...
2. En la ventana que aparece seleccionar Java → Runnable JAR File.
3. Finalmente, se muestra una ventana donde se selecciona la clase principal del proyecto a convertir a .jar, y la ubicación donde se quiere guardar este archivo.

Es muy importante conocer que para utilizar dicho archivo .jar en otro PC, debemos descargar la misma versión de Java que disponemos en el PC donde se ha desarrollado el proyecto y borrar las versiones anteriores, o direccionar las opciones java al fichero JRE necesario, en este caso es “**jre1.8.0_45**”. Este archivo se debe encontrar en la carpeta de Java del PC, suele ser Archivos de Programa.

Además en la carpeta bin de este fichero JRE se debe añadir los archivos “win32com.dll” y “javax.comm.properties”.

Finalmente en la carpeta lib del fichero JRE, se añade la librería comm.jar, comm 2-0.jar y rxtx.jar.

4.3.3.3 Modificaciones necesarias para utilizar un array de 16 desfasadores

En la demo VirtualSerial de LUFA de la que se ha hablado anteriormente en esta memoria, es necesario programar el micro aumentando el número de iteraciones de los bucles de 3 al número de desfasadores que se busca controlar, en este caso 16.

```
PORTD |= _BV(LED); //Configuración de LED
PORTD &= ~(_BV(LE)); //LE = 0 para que empiece la t
PORTD &= ~(_BV(LED)); //LE = 0 para que empiece la t

while(byteRecibido2 != byteRecibido & i<3){
    if ((byteRecibido & 0x20) != 0x00){
        PORTD |= _BV(DATO);
    }else{
        PORTD &= ~(_BV(DATO));
    }
}
//Formación del array
```

Figura 65. Modificación archivo VirtualSerial.c para antena de 16 phased array (I)

```
    i++;
}
if(i == 3){
    PORTD |= _BV(LE); //LE = 1, cuando se ha transmitido todos los códigos
    i=0;
}
```

Figura 66. Modificación archivo VirtualSerial.c para antena de 16 phased array (II)

Después de hacer estas modificaciones, se programa el micro por medio del AVR.

Una vez programado el micro, en el entorno Windows se modifica el tamaño del array dentro del código de la interfaz de usuario en Eclipse :

```
int ComboIndice1 = ComboDesfase1.getSelectedIndex();
int ComboIndice2 = ComboDesfase2.getSelectedIndex();
int ComboIndice3 = ComboDesfase3.getSelectedIndex();
byte [] desfases = new byte[3];
try{
    //Parte Envío en Array

    switch(ComboIndice1){
    case 0:
        System.out.print("Seleccione un desfase para el Desfasador 1");
        desfases = null;
        break;
    }
}
```

Figura 67. Captura array de envío, código java

Es en este momento, cuando en el menú de la interfaz se pueden controlar 16 desfasadores.

5 Integración, pruebas y resultados

Para probar finalmente si el funcionamiento de todo el sistema de control desarrollado en el proyecto es correcto, se hacen pruebas de comunicación entre dicho sistema y los desfases que conformarán la antena plana.

Las pruebas se realizan utilizando el analizador de redes AGILENT E5071C que trabaja en un rango de frecuencias de 300KHz hasta 20GHz, cubriendo así el rango de frecuencias para el que se diseña la antena, este es de 11GHz a 12GHz.

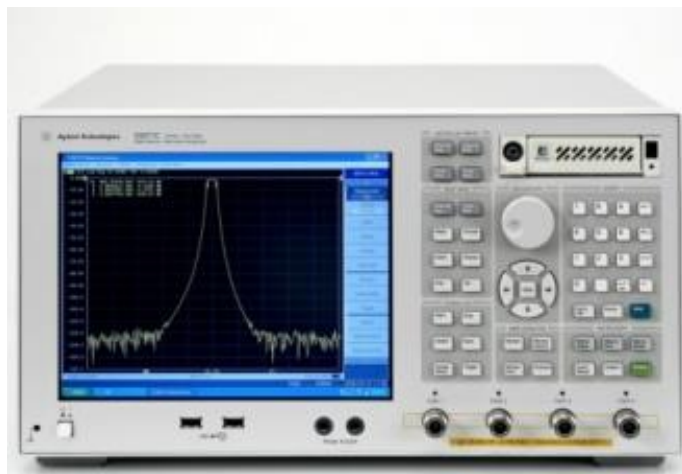


Figura 68. Analizador de Redes Agilent E5071C

5.1 Calibración

La calibración del analizador es una parte imprescindible ya que si no los resultados no serían óptimos. Este es un proceso de alta precisión donde se debe tener en cuenta la impedancia en la que se está operando (50 ohm) y las condiciones en las que está operando el equipo.

Para calibrar el analizador se han utilizado 3 dispositivos:

1. Se conecta a los puertos del analizador el dispositivo SHORT (red en corto circuito), primero a un puerto y luego a otro.
2. Se conecta a los puertos del analizador una carga de 50 Ohm, primero a un puerto y luego a otro.
3. Se conecta el dispositivo THRU (red conectada) para calibrar la transmisión, en este caso, el dispositivo se conecta a los dos puertos a la vez.

Nota: se debe tener en cuenta a la hora de la calibración, realizarla con cualquier transición que se vaya a utilizar posteriormente en las pruebas.

Después, se han hecho pruebas en una Cámara anecoica. Una cámara anecoica es una habitación que anula cualquier sonido del entorno exterior. Los materiales y la estructura con la que está construida, son diseñados específicamente para anular cualquier tipo de interferencia necesaria para realizar estudios científicos y de desarrollo tecnológico principalmente, anulando la reflexión de cualquier tipo de onda.

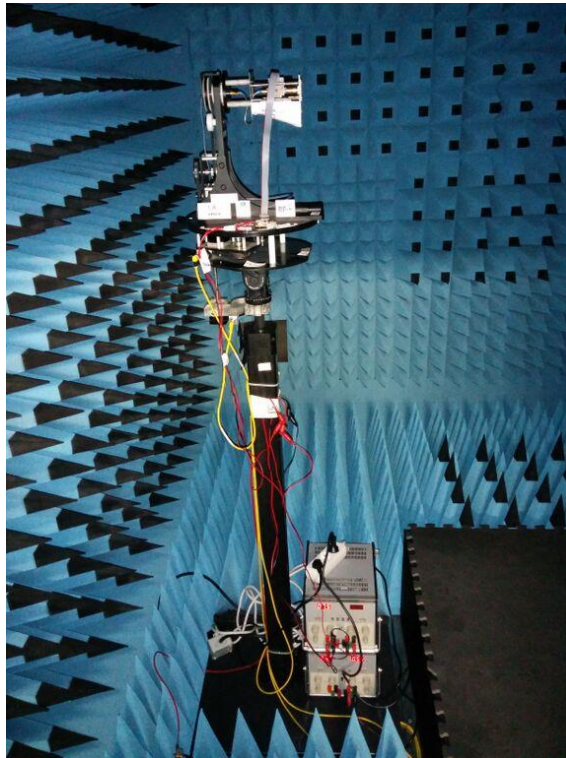


Figura 69. Cámara Anecoica

Los estados de la antena óptimos, que deberían ser obtenidos respecto a la fase, están representados en la hoja de datos del desfasador (Anexo II).

5.2 Pruebas

5.2.1 Pruebas control manual en analizador de redes

El montaje necesario se adjunta a continuación, realizando el desfase por medio de la placa de control manual, utilizando:

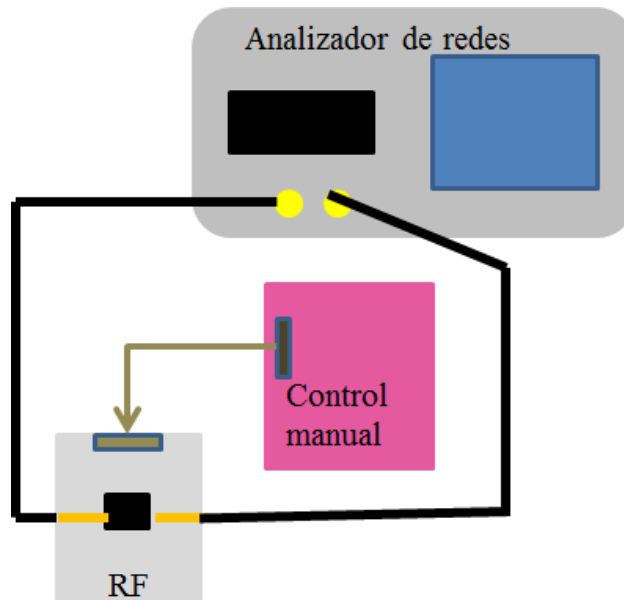


Figura 70. Esquema de montaje control manual en analizador de redes

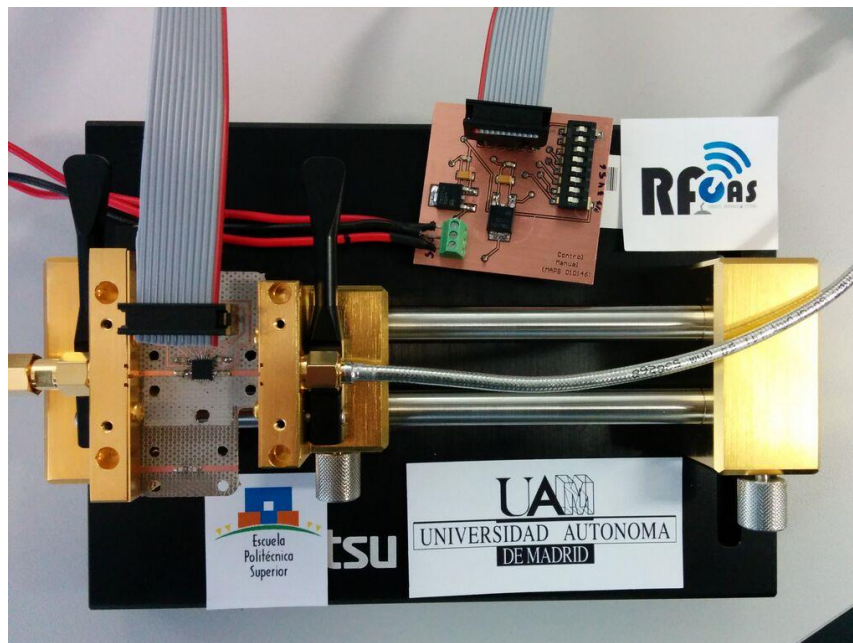


Figura 71. Montaje analizador de redes con control paralelo y placa RF con un desfasador digital.

5.2.1.1 Resultados obtenidos con control manual

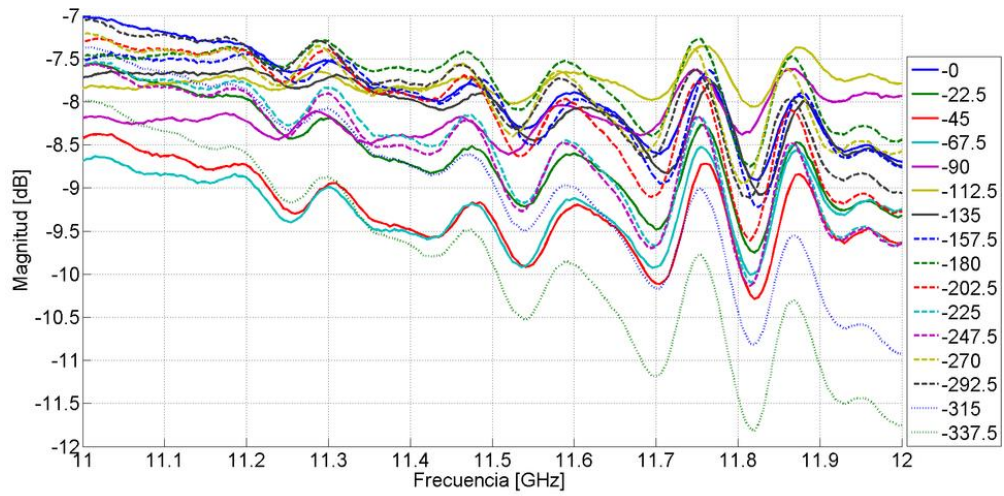


Figura 72. Transmisión (S21) con placa de control paralelo

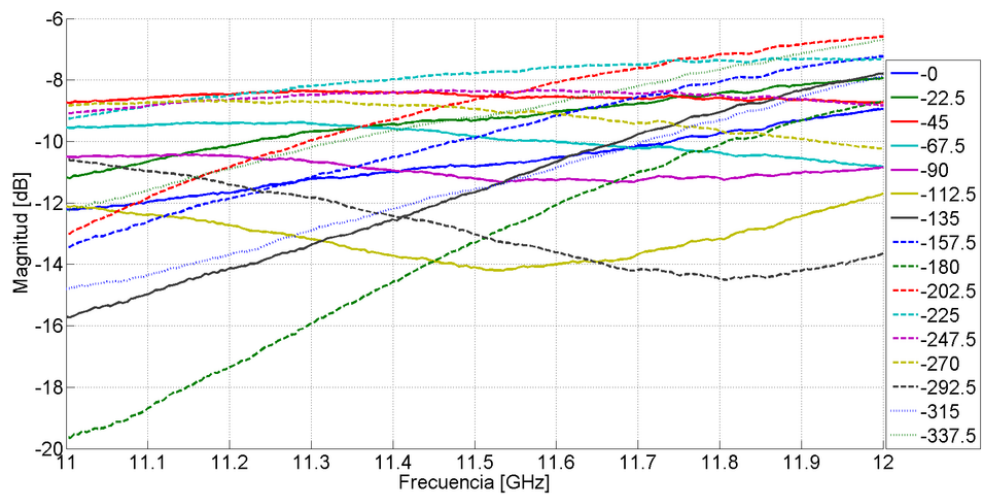


Figura 73. Reflexión (S11) con placa de control paralelo

Respecto a la transmisión, los resultados no son muy buenos puesto que no se aprecia ganancia ninguna.

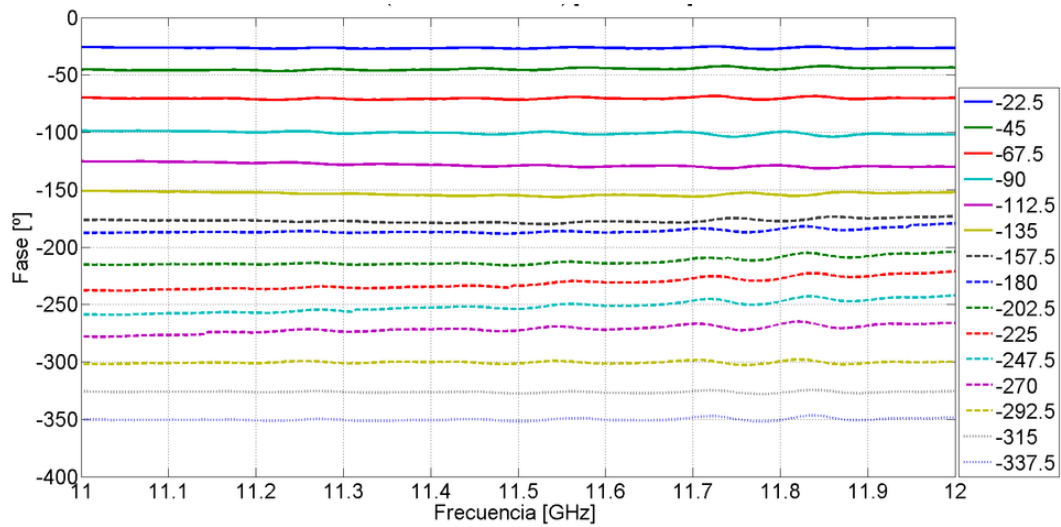


Figura 74. Desfase con referencia al estado 0 de la placa de control paralelo

Los resultados de desfase son más o menos los esperados y vistos en el datasheet del desfasador digital, es cierto que no son correctos del todo puesto que el desfase entre un estado y otro no es de 22.5° exactamente como debería de ser, pero si se asemeja a ese valor.

A continuación, se hacen las pruebas con el montaje serie, que deben ser resultados similares a lo obtenido con este montaje, si es así y tampoco se logra el perfecto funcionamiento, se descarta que el problema no es de las placas de control.

5.2.2 Pruebas control serie en analizador de redes

5.2.2.1 Control serie para placa con un desfasador digital

El montaje utilizado corresponde con el adjunto a continuación, realizando el desfase por medio de la placa de control serie:

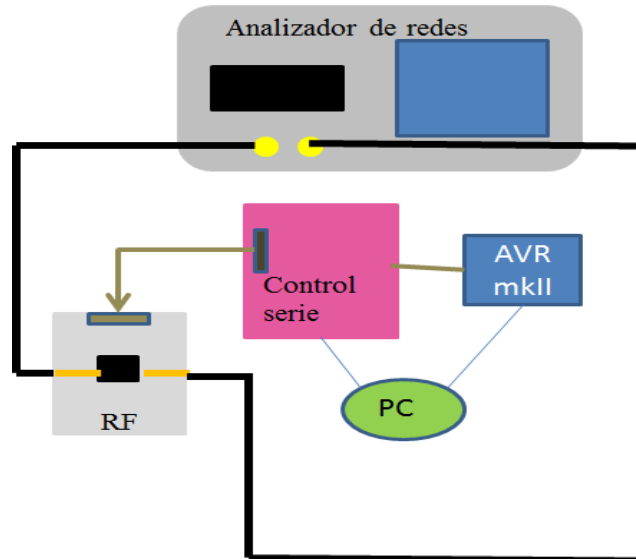


Figura 75. Esquema de conexión control serie y analizador de redes

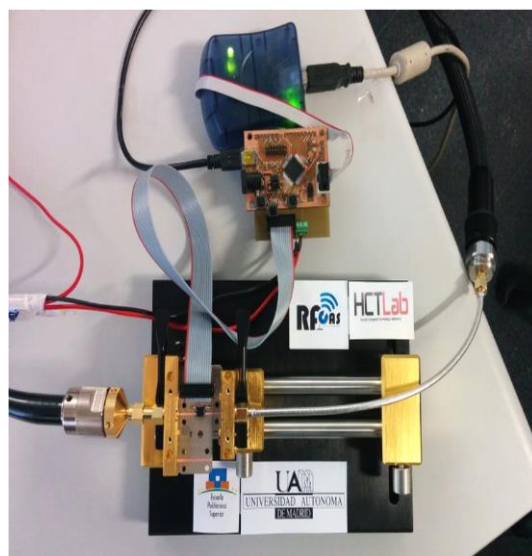


Figura 76. Montaje analizador de redes con control serie y placa RF con un desfasador digital.

5.2.2.1.1 Resultados obtenidos con el control serie y un desfasador digital

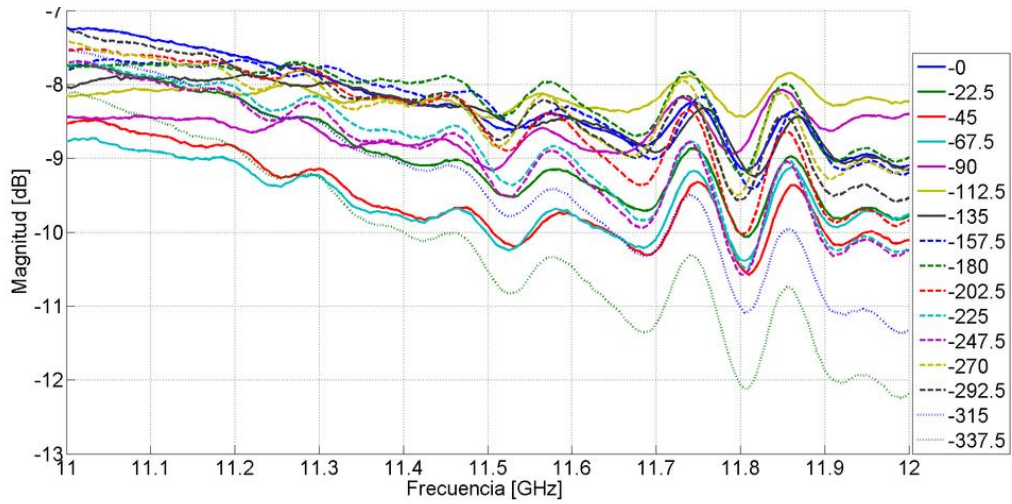


Figura 77. Transmisión (S21) con placa de control serie y un desfasador digital

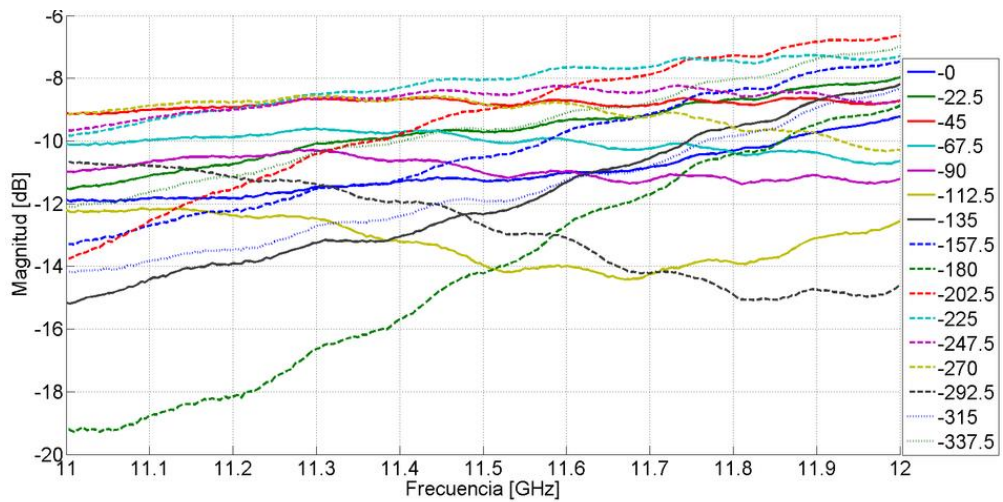


Figura 78. Reflexión (S11) con placa de control serie y un desfasador digital

Tampoco en este caso los resultados de transmisión y recepción son buenos, pero son similares a los obtenidos en el control paralelo por lo que el problema está en la placa de RF, posiblemente el desfasador digital utilizado no tenga muy buenas propiedades.

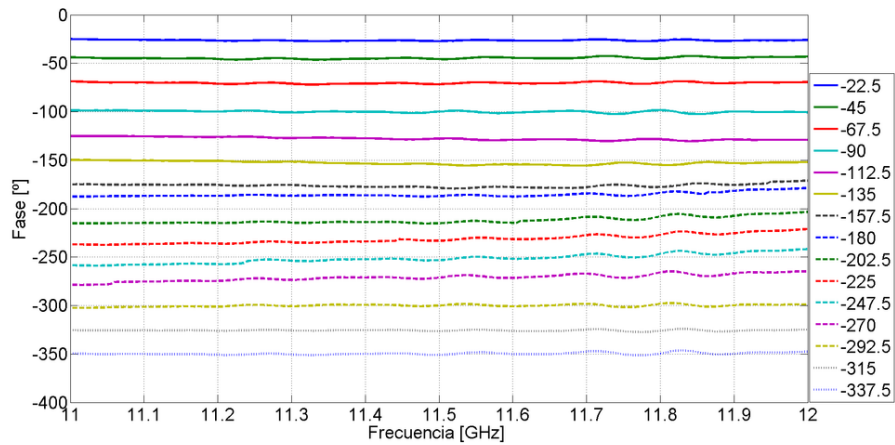


Figura 79. Desfase con referencia al estado 0 de la placa de control serie y un desfasador digital

Los resultados de desfase, son más o menos los esperados y vistos en el datasheet del desfasador digital, es cierto que no son correctos del todo, puesto que el desfase entre un estado y otro no es de 22.5° justos, como debería de ser, pero si son similares, como ocurría con el control manual, por lo que el control se realiza correctamente.

Se comprueba que el resultado tanto en amplitud como en fase es similar con ambos tipos de controles, por lo que se descarta la realización de pruebas futuras con la placa de control manual.

5.2.2.2 Control serie para placa phased array

El montaje con el que se adjunta a continuación, realizando el desfase por medio de la placa de control serie y con la placa RF de 3 desfases:

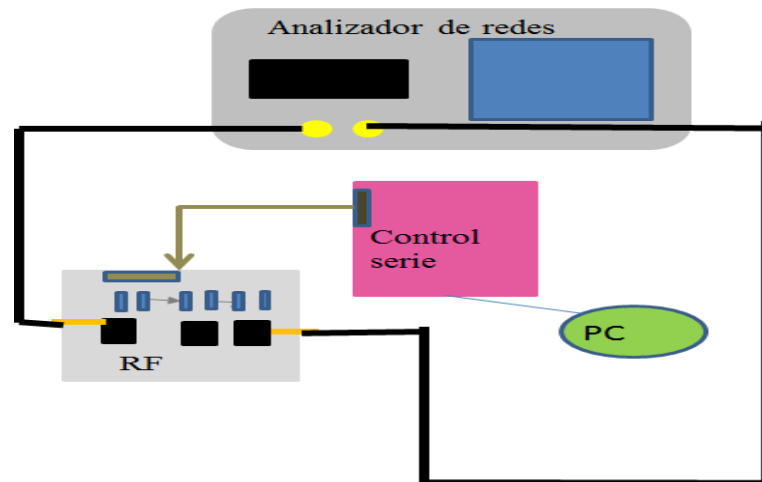


Figura 80. Esquema de conexión de control serie con analizador de redes y 3 phased array

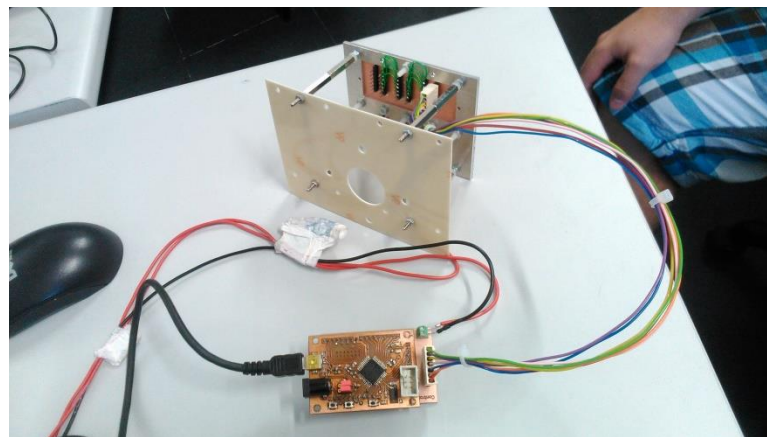


Figura 81. Montaje analizador de redes con control serie y placa RF con un 3 phase array

5.2.2.2.1 Resultados obtenidos con el control serie y el phased array

- En atenuación, se percibe que ha empeorado respecto a cuándo solo teníamos un desfasador digital, las pérdidas son mayores hasta ¡!!!.

- Resultados desfasador 1 del Array

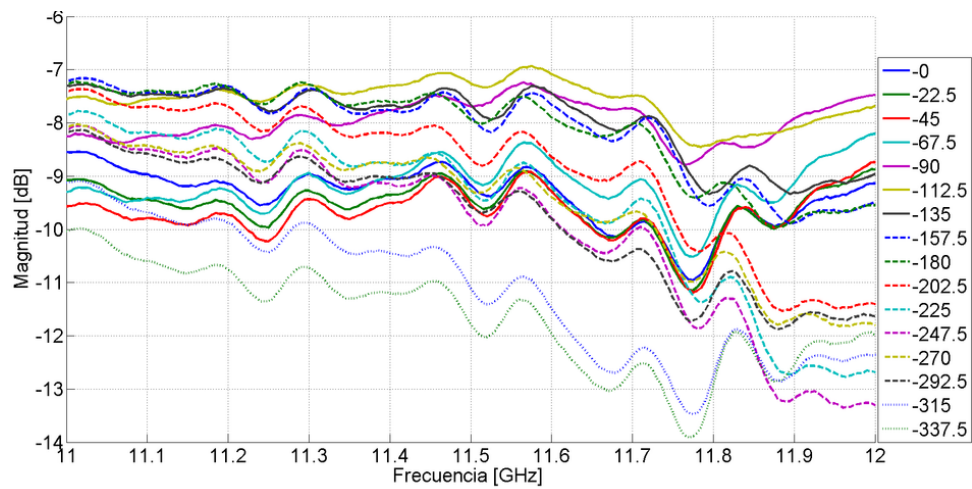


Figura 82. Transmisión (S21) con placa de control serie en array (desfasador 1)

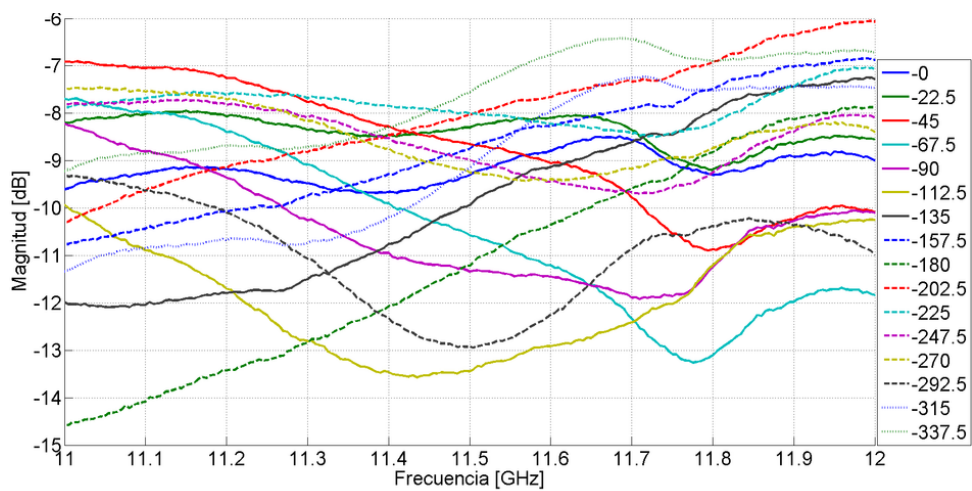


Figura 83. Reflexión (S11) con placa de control serie en array (desfasador 1)

○ Resultados desfasador 2 del Array

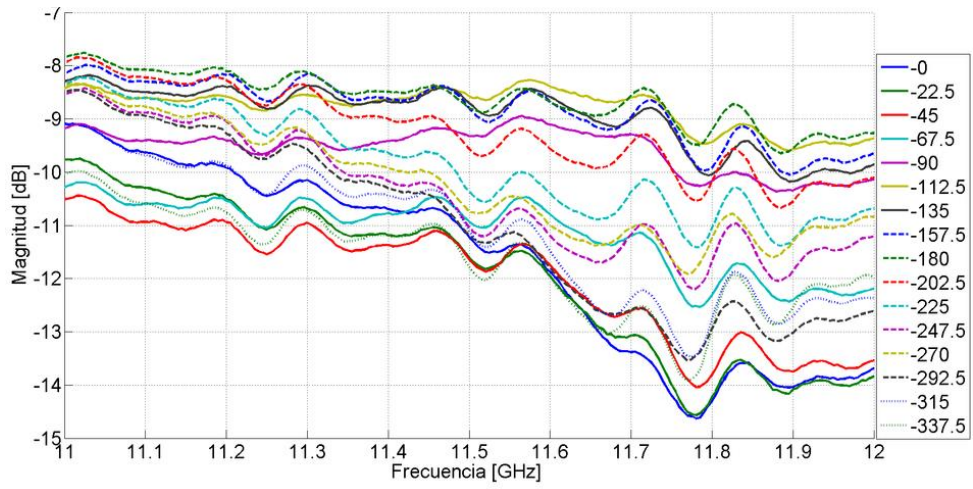


Figura 84. Transmisión (S21) con placa de control serie en array (desfasador 2)

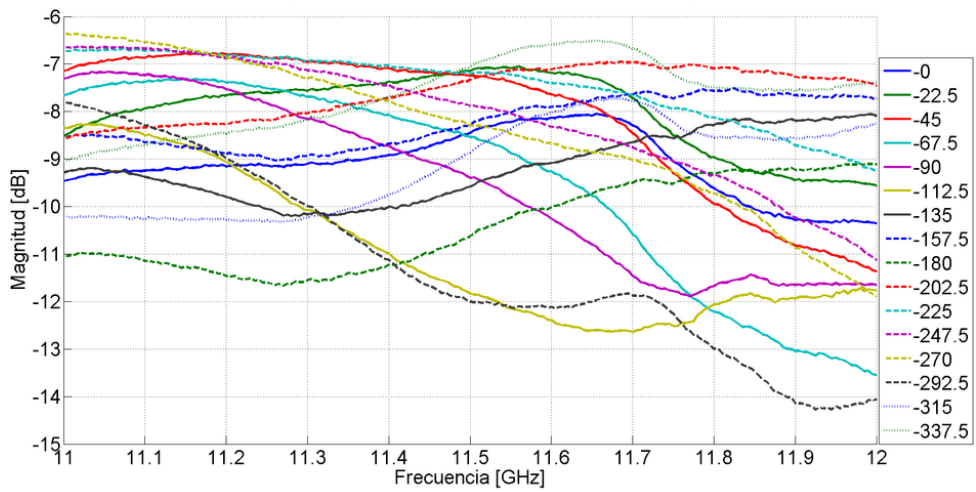


Figura 85. Reflexión (S11) con placa de control serie en array (desfasador 2)

○ Resultados desfasador 3 del Array

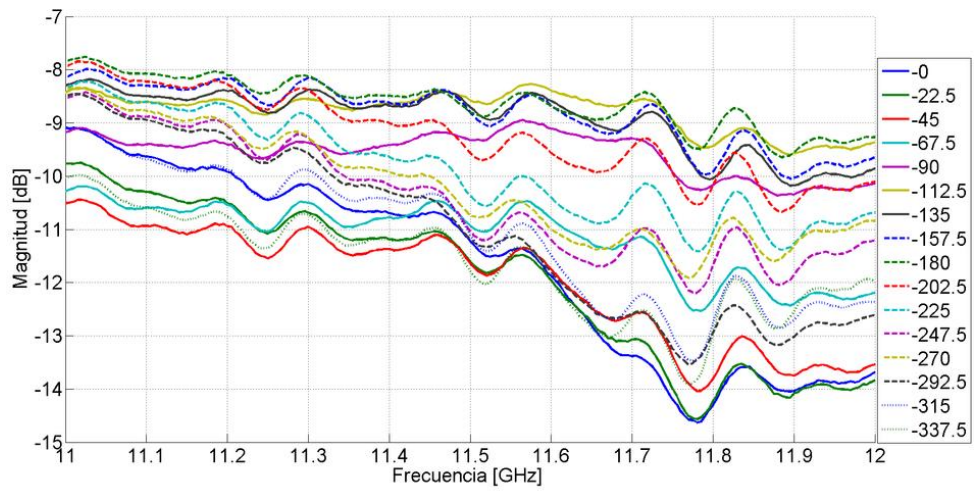


Figura 86. Transmisión (S21) con placa de control serie en array (desfasador 3)

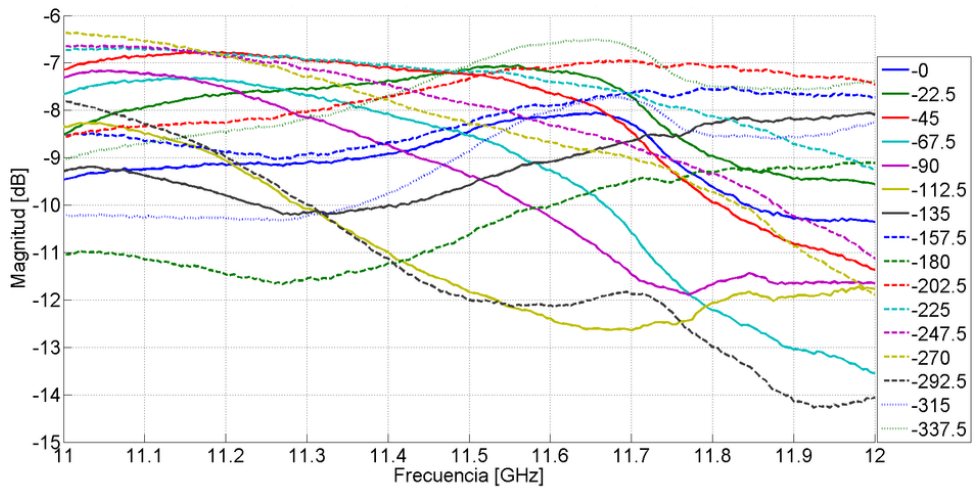


Figura 87. Reflexión (S11) con placa de control serie en array (desfasador 2)

- En fase, los resultados de desfase, son más o menos los esperados y vistos en el datasheet del desfasador digital, es cierto que no son correctos del todo, puesto que el desfase entre un estado y otro no es de 22.5° justos, como debería de ser. Y coincidentes a los obtenidos con un único desfasador.

- Resultados desfasador 1 del Array

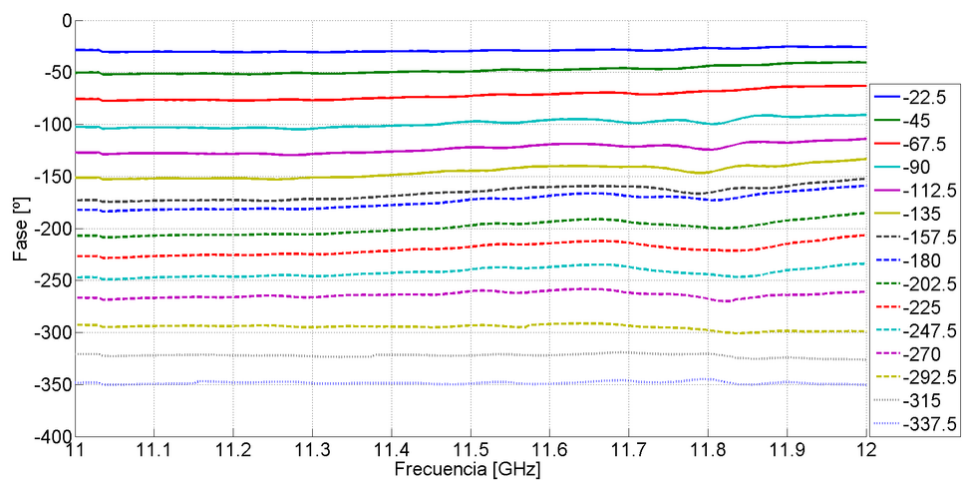


Figura 88. Desfase para desfasador 1 del array

- Resultados desfasador 2 del Array

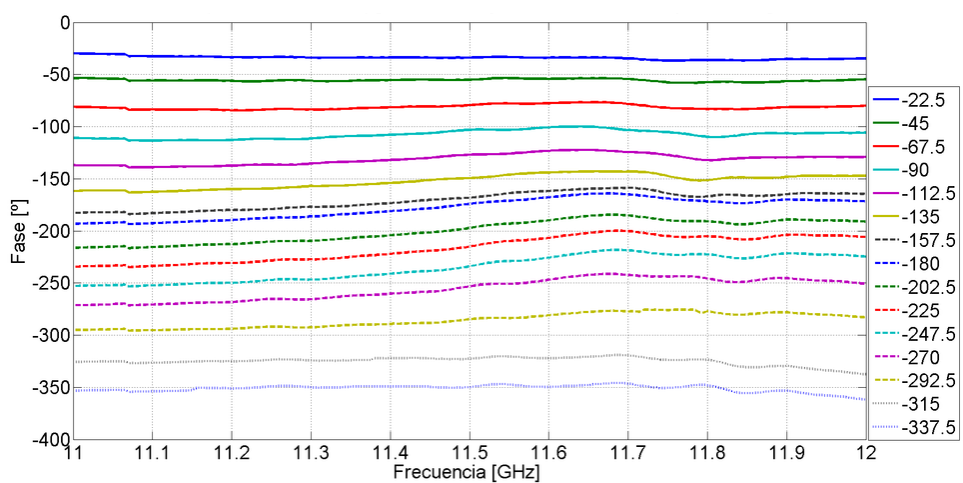


Figura 89. Desfase para desfasador 2 del array

○ Resultados desfasador 3 del Array

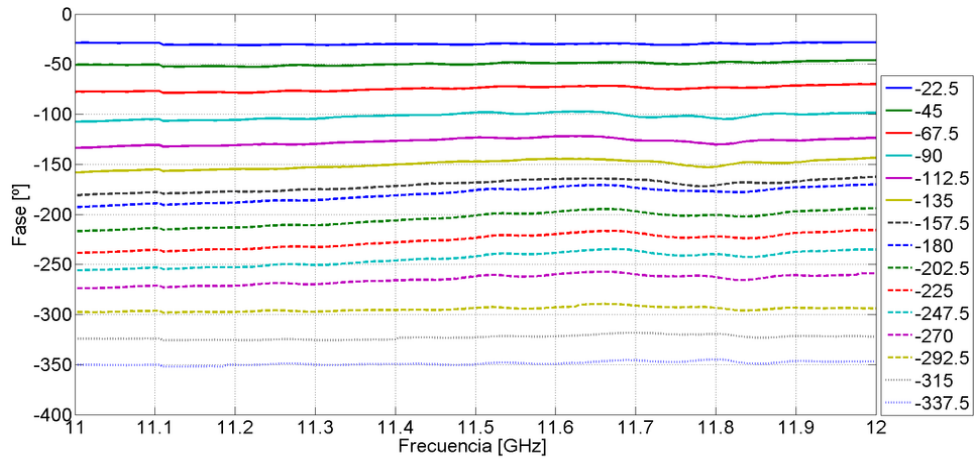


Figura 90. Desfase para desfasador 3 del array

6 Conclusiones y trabajo futuro

6.1 Conclusiones

El objetivo de conseguir controlar de forma automatizada la antena plana formada por un array de 3 parches lineales se ha conseguido, mediante el uso del microprocesador ATmega32u4, utilizando una topología Daisy chain mediante el control de salidas electrónicas en dicho micro.

La parte software, también se ha finalizado de forma eficiente, pudiendo controlar dicho array a través del PC, siguiendo los pasos indicados en esta memoria, para la instalación correcta de los archivos y ficheros necesarios para el control.

Se ha dejado diseñada e implementada, un firmware que permitirá en un proyecto futuro, controlar un array lineal de 16 desfasadores digitales, para utilizar dicha interfaz, también se especifican las modificaciones necesarias en este documento.

He conocido la existencia de los microcontroladores ATmega 32u4, aprendiendo distintos tipos de funcionalidades de estos microcontroladores, a través de programaciones por medio de un programador, como es el AVRISP mkII, viendo por tanto, que son muy útiles en múltiples y variadas aplicaciones de automatización, dando flexibilidad y usabilidad a variedad de procesos.

En este PFC, he podido trabajar con distintos tipos de tecnologías, permitiéndome conocer cada uno de ellos con detalle. Tanto la parte hardware, como una parte software, e incluso ligeros detalles de la parte de RF. Por lo que me ha permitido aumentar mis conocimientos, cosa que agradezco.

6.2 Trabajo futuro

Posibilidad de recibir e interpretar datos recibidos a partir del desfasador a la firmware, esto sería modificar la parte software necesaria dentro del código de la librería LUFA implementado en caso de seguir utilizando USB como protocolo de comunicación, entre el PC y la placa de control.

Diseño y construcción de una única placa, que implemente tanto la funcionalidad de radiofrecuencia como el control de esta.

Diseño y verificación de una firmware programable, donde se pueda seleccionar de forma dinámica, el número de desfasadores que se quiere controlar en cada momento.

Referencias

- [1] Jesús Vazquez Ruiz. Diseño de un phased array de parches sobre guía de onda para comunicaciones con satélite DBS. [Publicación]. PFC: Universidad Autónoma de Madrid; 2014 Julio; Escuela Politécnica Superior.
- [2] Alfredo Manuel Castro Cuartero. Sistema de control de apuntamiento electrónico de una agrupación de antenas planas reconfigurables operando en el sistema WiMAX a 3.5 GHz. [Publicación] PFC: Universidad Autónoma de Madrid; Septiembre 2014; Escuela Politécnica Superior
- [3] Colino RJL. Integración de una antena plana TX/RX de apuntamiento variable con desfasadores digitales y matrices de Butler para comunicaciones WiMAX a 3.5 GHz [Publicación]. PFC. Madrid: Universidad Autónoma de Madrid; 2011 marzo Escuela Politécnica Superior.
- [4] IEE Standards Boards, “IEE Standards Definitions off Term for Antennas”, IEE Std 145-1993, Mayo 1993
- [5] R.Garg, P.Bhartia et. Al., “Microstrip Antenna Design Handbook”, Artech House, London 2001.
- [6] WA8SME, Mark specer, en su proyecto publicado en “The Amset jamel” num 3/2006, ofreciendo el programa PIC.
- [7] Dennis Xabier Arroba Tijerino. Sistema de control automático para una antena satelital. [Publicación] PFC: Escuela de Ingeniería Electrónica del Instituto Tecnológico de Costa Rica. Cartago, Agosto de 2011.

A Datasheet

Anexo I: Microcontrolador ATMEGA32U4

Features

- High Performance, Low Power AVR[®] 8-Bit Microcontroller
- Advanced RISC Architecture
 - 135 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-Chip 2-cycle Multiplier
- Non-volatile Program and Data Memories
 - 16/32K Bytes of In-System Self-Programmable Flash (ATmega16U4/ATmega32U4)
 - 1.25/2.5K Bytes Internal SRAM (ATmega16U4/ATmega32U4)
 - 512Bytes/1K Bytes Internal EEPROM (ATmega16U4/ATmega32U4)
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/ 100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - All supplied parts are preprogrammed with a default USB bootloader
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- USB 2.0 Full-speed/Low Speed Device Module with Interrupt on Transfer Completion
 - Complies fully with Universal Serial Bus Specification Rev 2.0
 - Supports data transfer rates up to 12 Mbit/s and 1.5 Mbit/s
 - Endpoint 0 for Control Transfers: up to 64-bytes
 - 6 Programmable Endpoints with IN or Out Directions and with Bulk, Interrupt or Isochronous Transfers
 - Configurable Endpoints size up to 256 bytes in double bank mode
 - Fully independent 832 bytes USB DPRAM for endpoint memory allocation
 - Suspend/Resume Interrupts
 - CPU Reset possible on USB Bus Reset detection
 - 48 MHz from PLL for Full-speed Bus Operation
 - USB Bus Connection/Disconnection on Microcontroller Request
 - Crystal-less operation for Low Speed mode
- Peripheral Features
 - On-chip PLL for USB and High Speed Timer: 32 up to 96 MHz operation
 - One 8-bit Timer/Counter with Separate Prescaler and Compare Mode
 - Two 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
 - One 10-bit High-Speed Timer/Counter with PLL (64 MHz) and Compare Mode
 - Four 8-bit PWM Channels
 - Four PWM Channels with Programmable Resolution from 2 to 16 Bits
 - Six PWM Channels for High Speed Operation, with Programmable Resolution from 2 to 11 Bits
 - Output Compare Modulator
 - 12-channels, 10-bit ADC (features Differential Channels with Programmable Gain)
 - Programmable Serial USART with Hardware Flow Control
 - Master/Slave SPI Serial Interface



8-bit **AVR[®]**
Microcontroller
with
16/32K Bytes of
ISP Flash
and USB
Controller

ATmega16U4
ATmega32U4

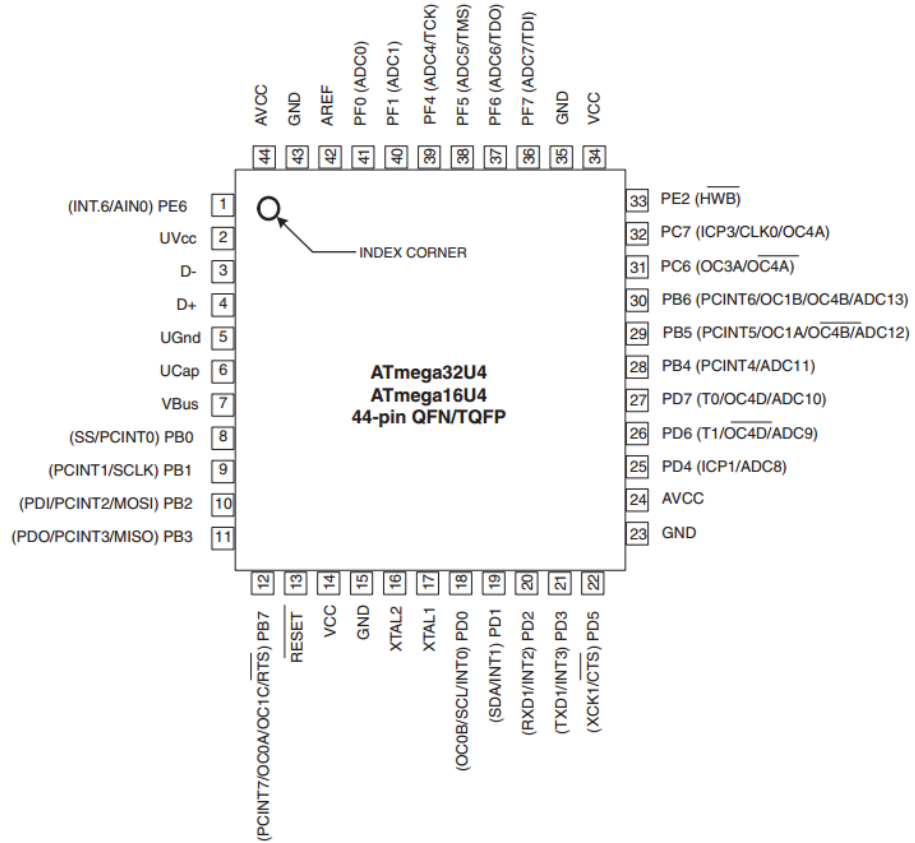
Preliminary
Summary

7766FS-AVR-11/10



1. Pin Configurations

Figure 1-1. Pinout ATmega16U4/ATmega32U4



2. Overview

The ATmega16U4/ATmega32U4 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16U4/ATmega32U4 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Anexo II: Digital Phase Shifter 4bits, MAPS 010146

MAPS-010146



Digital Phase Shifter
4-Bit, 8.0 - 12.0 GHz

Rev. V3

Features

- 4 Bit Digital Phase Shifter
- 360° Coverage with LSB = 22.5°
- Integrated CMOS Driver
- Serial or Parallel Control
- Low DC Power Consumption
- Minimal Attenuation Variation over Phase Shift Range
- 50 Ω Impedance
- EAR99
- Lead-Free 4 mm 24-Lead PQFN Package
- RoHS* Compliant

Description

The MAPS-010146 is a GaAs pHEMT 4-bit digital phase shifter with an integrated CMOS driver in a 4 mm PQFN plastic surface mount package. Step size is 22.5° providing phase shift from 0° to 360° in 22.5° steps. This design has been optimized to minimize variation in attenuation over the phase shift range.

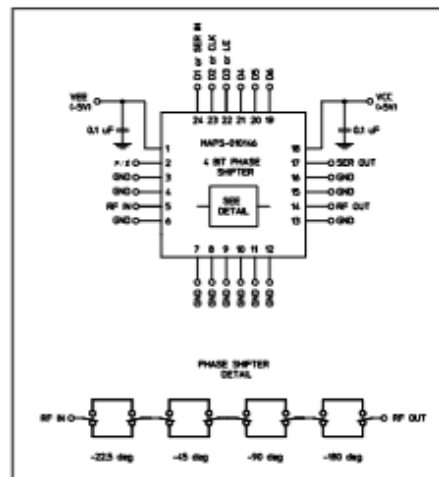
The MAPS-010146 is ideally suited for use where high phase accuracy with minimum loss variation over the phase shift range are required. The 4 mm PQFN package provides a smaller footprint than is typically available for a digital phase shifter with an internal driver. Typical applications include communications antennas and phased array radars.

Ordering Information ¹

Part Number	Package
MAPS-010146-TR0500	500 piece reel
MAPS-010146-001SMB	Sample Test Board

1. Reference Application Note M513 for reel size information.

Functional Schematic



Pin Configuration ²

Pin No.	Function	Pin No.	Function
1	VEE	13	GND
2	P/S	14	RF OUT
3	GND	15	GND
4	GND	16	GND
5	RF IN	17	SER OUT
6	GND	18	VCC
7	GND	19	D6
8	GND	20	D5
9	GND	21	D4
10	GND	22	D3 or LE
11	GND	23	D2 or CLK
12	GND	24	D1 or SER IN

2. The exposed pad centered on the package bottom must be connected to RF and DC ground.

* Restrictions on Hazardous Substances, European Union Directive 2002/95/EC.

1

ADVANCED: Data Sheets contain information regarding a product M/A-COM Technology Solutions is considering for development. Performance is based on target specifications, simulated results, and/or prototype measurements. Commitment to develop is not guaranteed.

PRELIMINARY: Data Sheets contain information regarding a product M/A-COM Technology Solutions has under development. Performance is based on engineering tests. Specifications are typical. Mechanical outline has been fixed. Engineering samples and/or test data may be available. Commitment to produce in volume is not guaranteed.

• North America Tel: 800.366.2266 • Europe Tel: +353.21.244.6400
• India Tel: +91.80.43537383 • China Tel: +86.21.2407.1588
Visit www.macomtech.com for additional data sheets and product information.

M/A-COM Technology Solutions Inc. and its affiliates reserve the right to make changes to the product(s) or information contained herein without notice.

MAPS-010146



Digital Phase Shifter
4-Bit, 8.0 - 12.0 GHz

Rev. V3

Modes of Operation: Serial and Direct Parallel

Serial Mode

The serial control interface (SERIN, CLK, LE, SEROUT) is compatible with the SPI protocol. SPI mode is activated when P/S is kept high. The 8-bit serial word must be loaded with the MSB first. After shifting in the 8 bit word, a rising edge on LE will set the phase shifter to the desired state. While LE is high the CLK is masked to protect the data while implementing the change. SEROUT is SERIN delayed by 8 clock cycles.

When P/S is low, the serial control interface is disabled. When P/S is set high, Pins 22, 23, and 24 have the LE, CLK, and SER IN function.

In serial mode operation, the outputs will stay constant while LE is kept low.

Direct Parallel Mode

The parallel mode is enabled when P/S is set low. In the direct parallel mode, the phase shifter is controlled by the parallel control inputs directly. When P/S is set low, Pins 22, 23, and 24 have the D3, D2, and D1 function.

Mode Truth Table ^{8,9}

P/S	LE	Mode
1	X	Serial
0	N/A	Direct Parallel

8. There are two dummy bits (D1 & D2), that must be sent in the serial mode. This is because the 4 bit phase shifter uses the same driver as the 6 bit phase shifter.
9. In the parallel mode, D1 and D2 should be tied to ground or to V_{cc}.

Truth Table (Digital Phase Shifter) ¹⁰

D6	D5	D4	D3	D2	D1	Phase Shift
0	0	0	0	X	X	Reference Phase
0	0	0	1	X	X	22.5 deg
0	0	1	0	X	X	45 deg
0	1	0	0	X	X	90 deg
1	0	0	0	X	X	180 deg
1	1	1	1	X	X	337.5 deg

10. 0 = CMOS Low; 1 = CMOS High, X is CMOS Low or High

Serial Interface Timing Characteristics

Symbol	Parameter	Typical Performance			Units
		-40°C	25°C	+85°C	
t _{clk}	Min. Serial Clock Period	100	100	100	ns
t _{cs}	Min. Control Set-up Time	20	20	20	ns
t _{ch}	Min. Control Hold Time	20	20	20	ns
t _{ls}	Min. LE Set-up Time	10	10	10	ns
t _{lv}	Min. LE Pulse Width	10	10	10	ns
t _{lh}	Min. Serial Clock Hold Time from LE	10	10	10	ns
t _{lss}	Min. LE Pulse Spacing	630	630	630	ns

5

ADVANCED: Data Sheets contain information regarding a product M/A-COM Technology Solutions is considering for development. Performance is based on target specifications, simulated results, and/or prototype measurements. Commitment to develop is not guaranteed.
PRELIMINARY: Data Sheets contain information regarding a product M/A-COM Technology Solutions has under development. Performance is based on engineering tests. Specifications are typical. Mechanical outline has been fixed. Engineering samples and/or test data may be available. Commitment to produce in volume is not guaranteed.

• North America Tel: 800.366.2266 • Europe Tel: +353.21.244.6400
• India Tel: +91.80.43537383 • China Tel: +86.21.2407.1588
Visit www.macomtech.com for additional data sheets and product information.

M/A-COM/Technology Solutions Inc. and its affiliates reserve the right to make changes to the product(s) or information contained herein without notice.

MAPS-010146

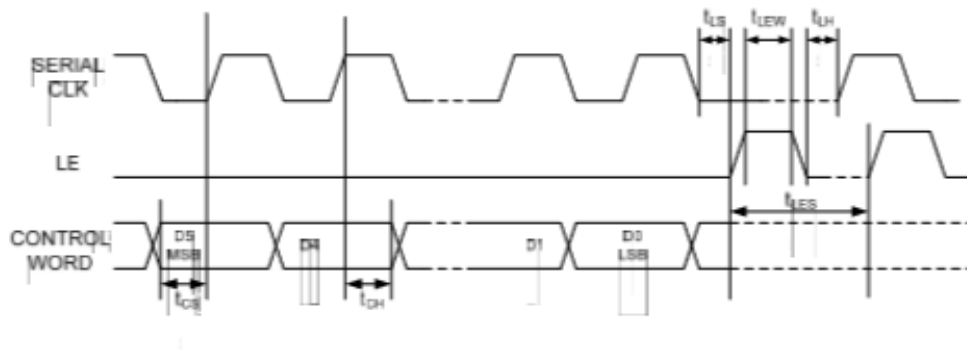


Digital Phase Shifter
4-Bit, 8.0 - 12.0 GHz

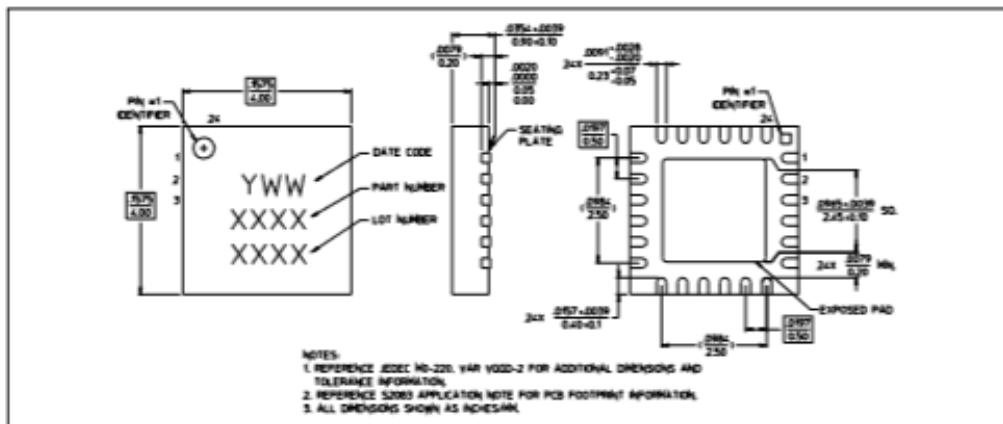
Rev. V3

Functionality Modes of Operation: Serial and Direct Parallel

Serial Input Interface Timing Diagram



Lead Free 4 mm 24-Lead PQFN †



† Reference Application Note S2083 for lead-free solder reflow recommendations.
Meets JEDEC moisture sensitivity level 1 requirements.
Plating is 100% matte tin over copper.

6

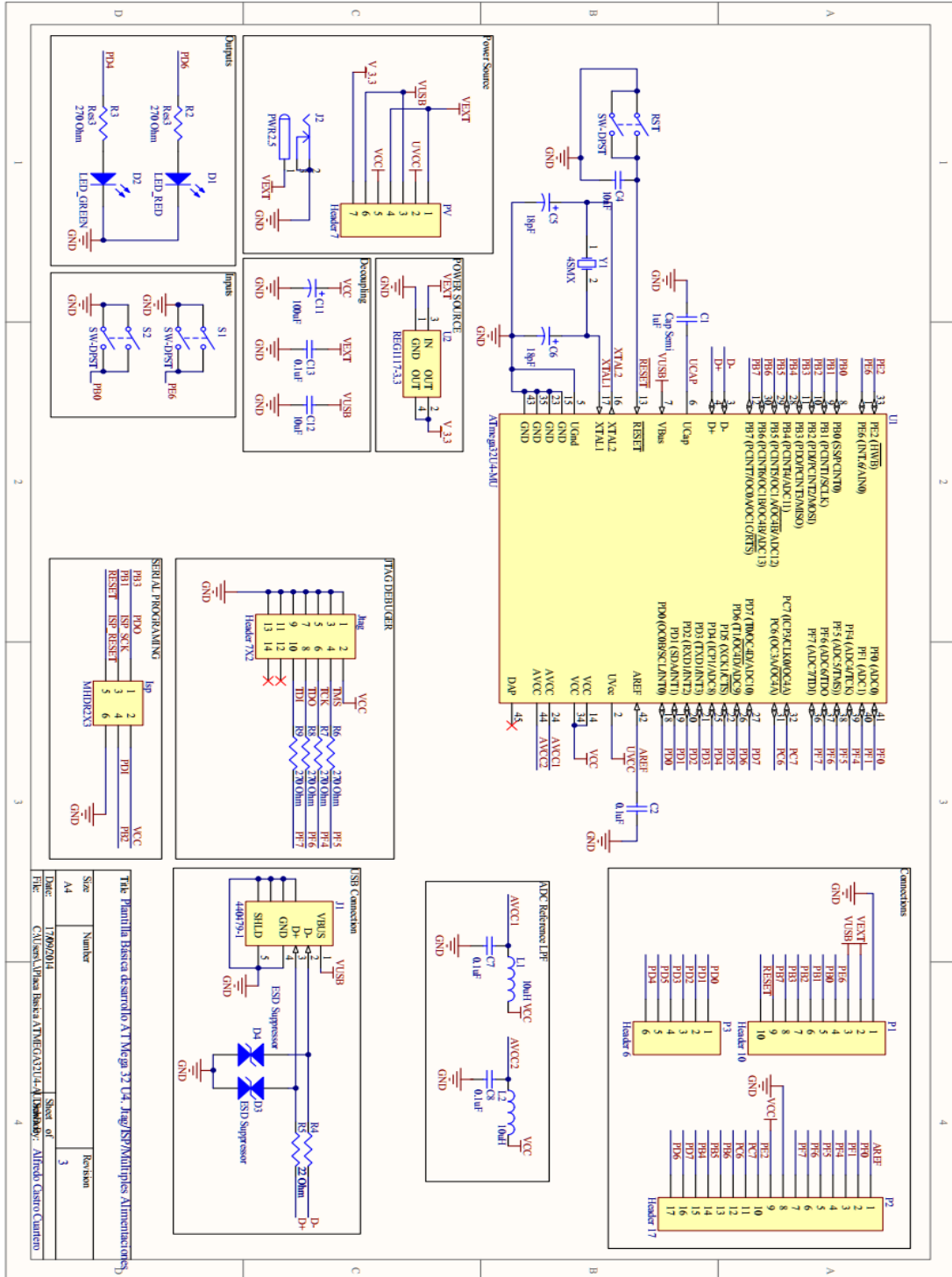
ADVANCED: Data Sheets contain information regarding a product M/A-COM Technology Solutions is considering for development. Performance is based on target specifications, simulated results, and/or prototype measurements. Commitment to develop is not guaranteed.
Preliminary: Data Sheets contain information regarding a product M/A-COM Technology Solutions has under development. Performance is based on engineering tests. Specifications are typical. Mechanical outline has been fixed. Engineering samples and/or test data may be available. Commitment to produce in volume is not guaranteed.

• North America Tel: 800.366.2266 • Europe Tel: +353.21.244.6400
• India Tel: +91.80.43537383 • China Tel: +86.21.2407.1588
Visit www.macomtech.com for additional data sheets and product information.

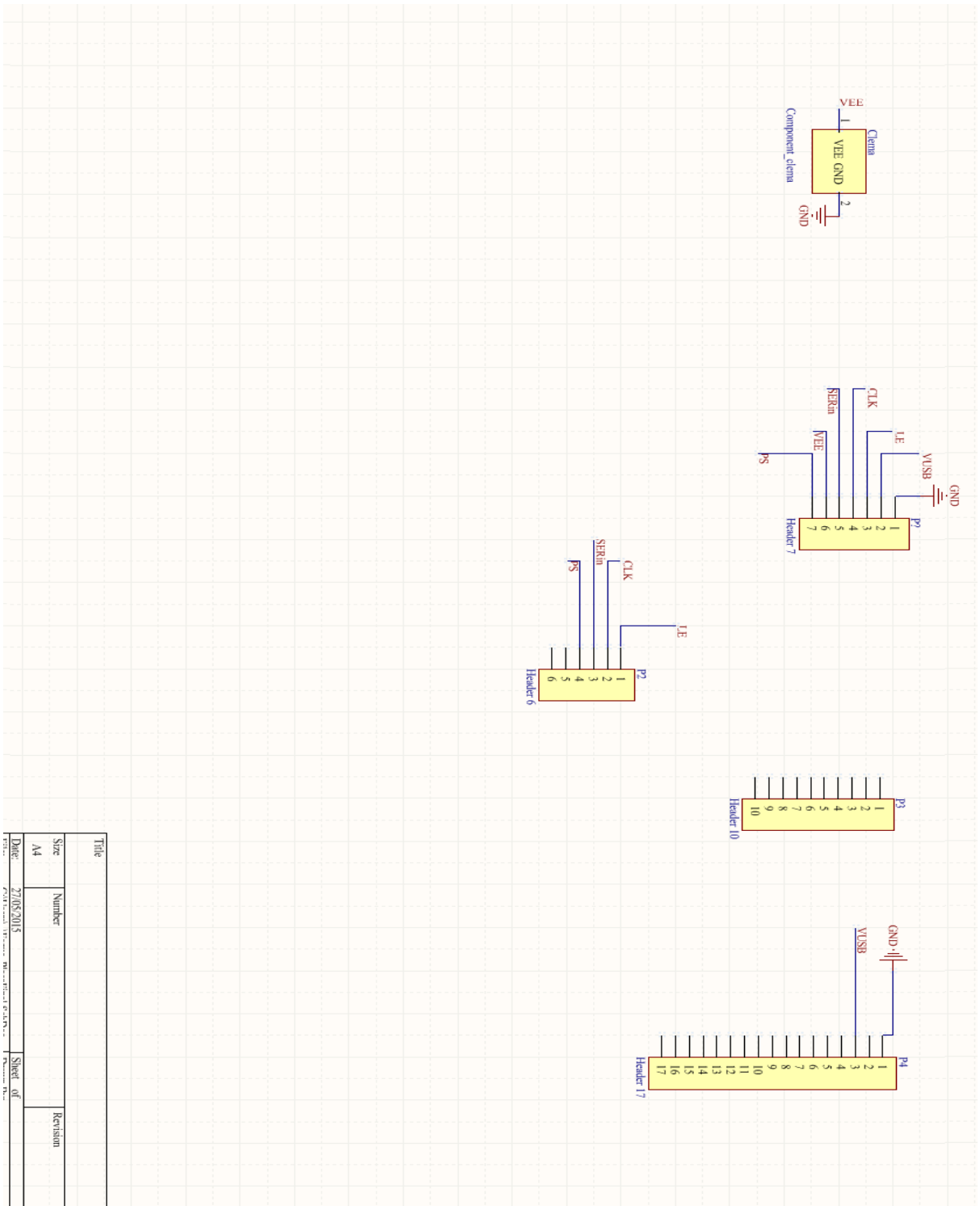
M/A-COM Technology Solutions Inc. and its affiliates reserve the right to make changes to the product(s) or information contained herein without notice.

B Esquemáticos

Anexo III: Esquemático Placa de desarrollo ATMEGA32U4 control serie

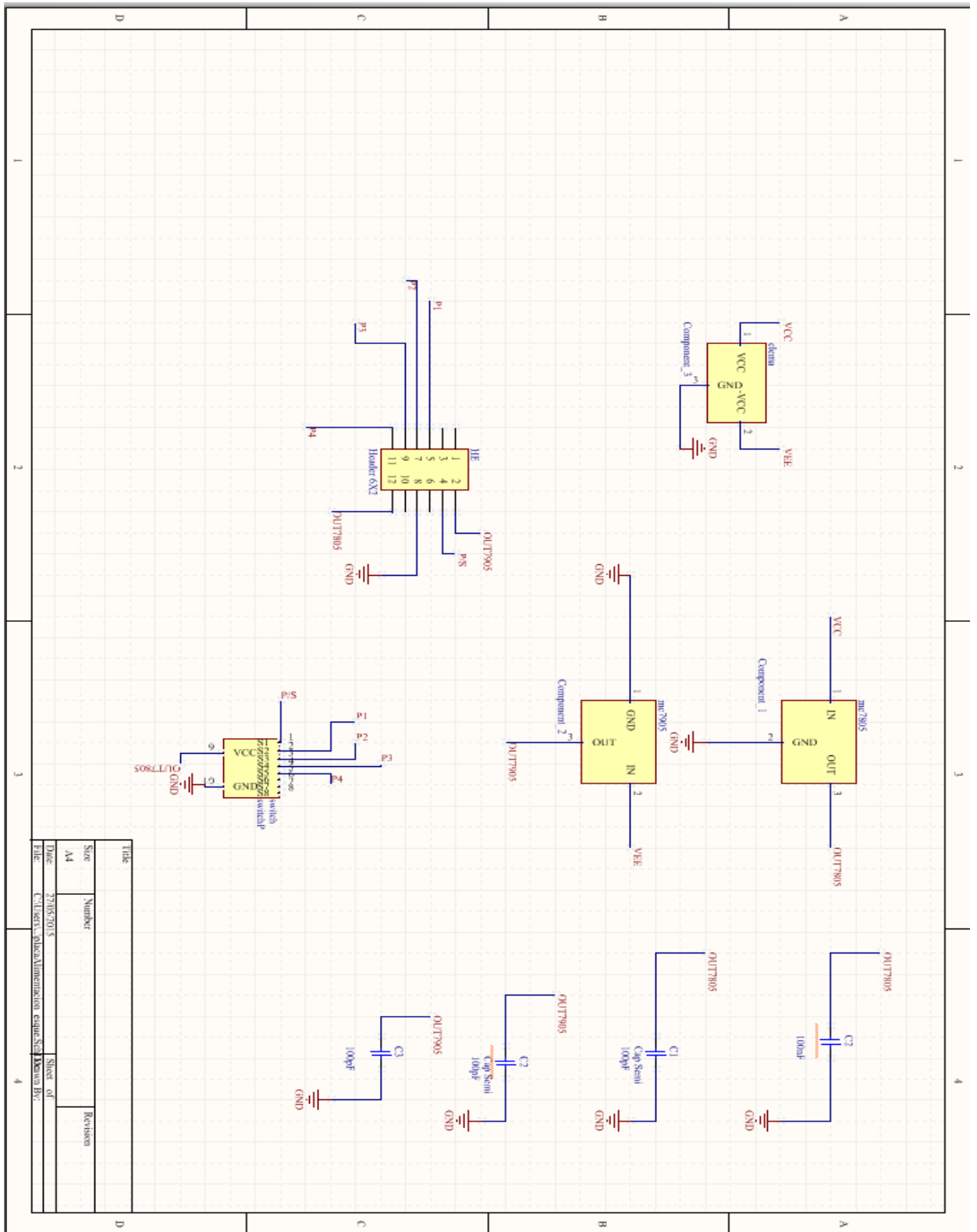


Anexo IV: Esquemático Placa interfaz control serie



Title	
Size	Number
A4	
Date:	Revision
27/05/2015	
Sheet of	
17	

Anexo V: Esquemático Placa control paralelo



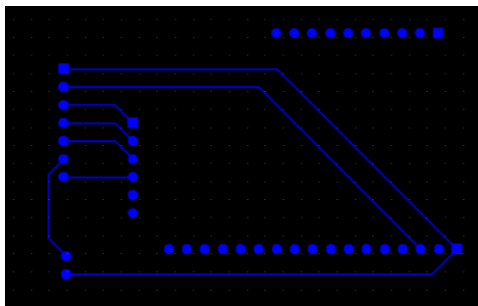
C Archivos GERBERS

Anexo VI: Gerber Placa de desarrollo ATMEGA32U4 control serie

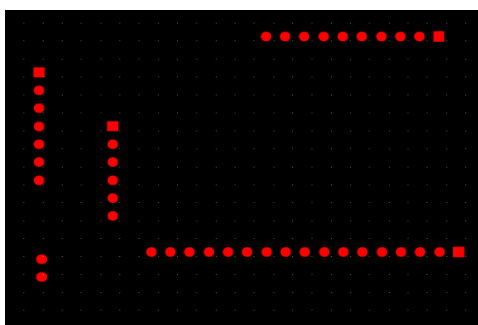
Top Layer:



Bottom Layer:

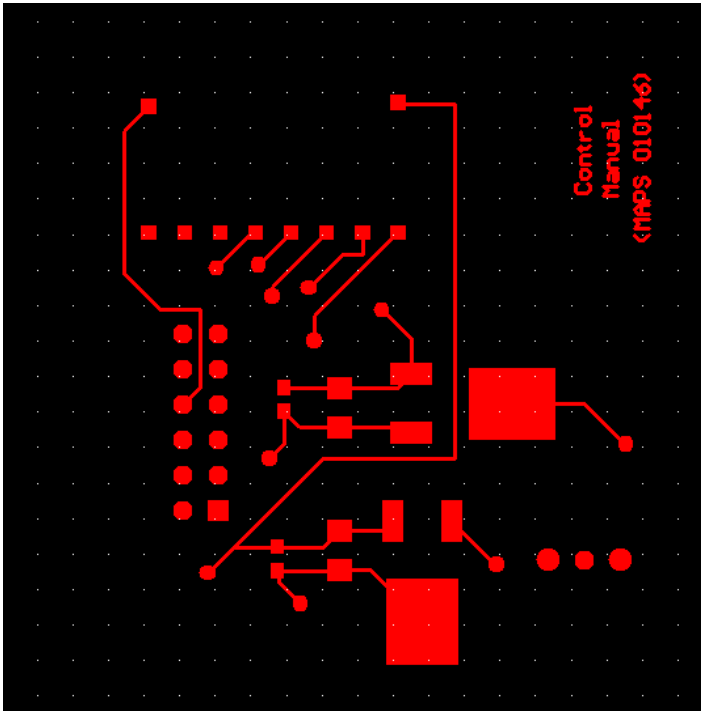


Drill Guide:

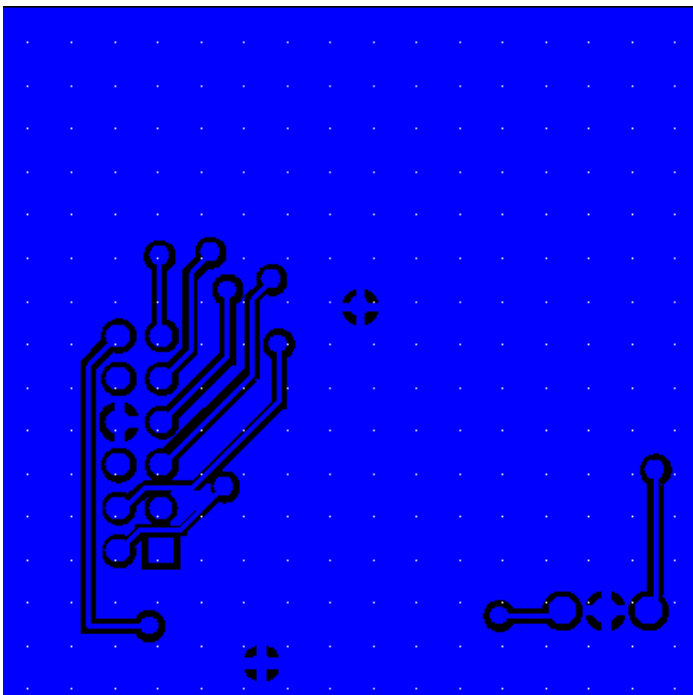


Anexo VII: Gerber Placa de control paralelo

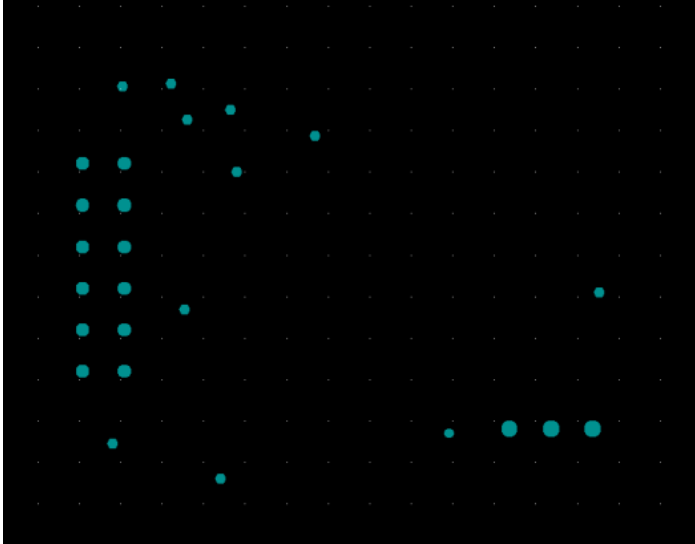
Top Layer:



Bottom Layer:

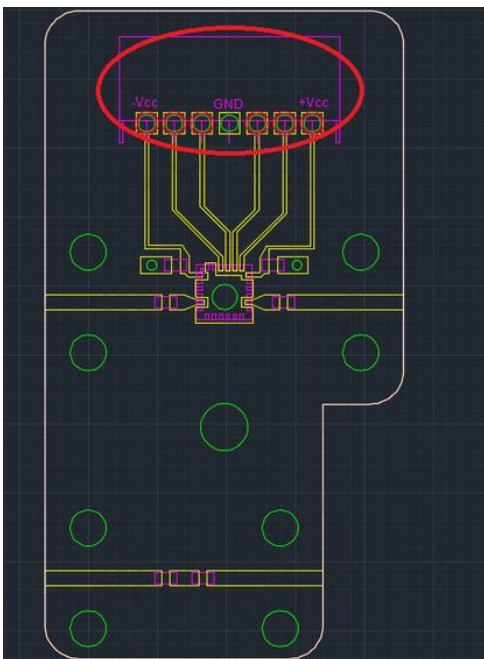


Drill Guide:

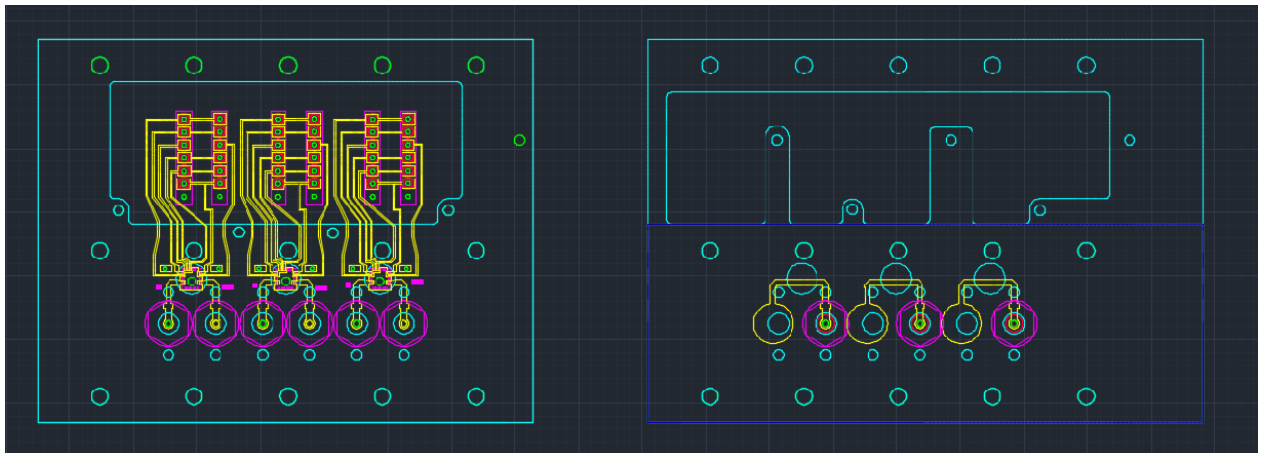


Anexo VIII: Diseño Placa RF

Prototipo un desfaseador:



Prototipo 3 phased array:



PRESUPUESTO

1) Ejecución Material

- Compra de ordenador personal (Software incluido)..... 2.000 €
- Alquiler de impresora láser durante 6 meses.....50 €
- Medida Analizador de Redes(13 h).....910 €
- Medidas en cámara Anecoica.....800 €
- Alquiler fresadoras PCBs1000 €
- Material prototipos
 - Componentes electrónicos325 €
 - Placas de cobre.....45 €
- Material de oficina150 €
- **Total de ejecución material** 5280 €

2) Gastos generales

- 16 % sobre Ejecución Material 845 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material 318 €

4) Honorarios Proyecto

- 740 horas a 15 € / hora 11.100 €

5) Material fungible

- Gastos de impresión 60 €
- Encuadernación 200 €

6) Subtotal del presupuesto

- Subtotal Presupuesto 17.803 €

7) I.V.A. aplicable

- 21% Subtotal Presupuesto3.738,63 €

8) Total presupuesto

- Total Presupuesto 21541,63 €

Madrid, Julio de 2015

El Ingeniero Jefe de Proyecto

Fdo.: Melani Corral Baltasar

Ingeniero de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un SISTEMA DE CONTROL SERIE PARA APUNTAMIENTO ELECTRÓNICO DE ANTENAS PLANAS PARA RECEPCIÓN DE SATÉLITE DBS. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las

condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.

