

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-27

# IMPLEMENTACIÓN DE UN SISTEMA DE TRANSMISIÓN SIMULTÁNEA DE INFORMACIÓN VÍA BLUETOOTH BASADO EN FPGA

Dagoberto Ramírez Serna

Ingeniería Electrónica

Luis Fernando Castaño Londoño

**INSTITUTO TECNOLÓGICO METROPOLITANO**

**Julio de 2018**

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## RESUMEN

---

En este trabajo se implementa un sistema de comunicación inalámbrico basado en Bluetooth para la recepción de datos usando los módulos PmodBT2 y la tarjeta de desarrollo Zedboard. Los datos recibidos por los periféricos Bluetooth pueden provenir de cualquier dispositivo que utilice este protocolo de comunicación, por ejemplo una red de sensores o un dispositivo móvil. La conexión entre los dos módulos Bluetooth y la Zedboard se realiza mediante los puertos destinados para los módulos periféricos Pmod. La Zedboard implementa un puente USB-UART el cual permite que la información recibida se visualice en un computador usando un emulador de terminal conectado a un puerto serie.

La primera parte de la implementación se realiza en Vivado versión 2015. En esta etapa se crea el diseño de bloques, en el cual, se realizan las conexiones y configuraciones. La asignación de pines se realiza en base al manual de usuario de la Zedboard. Después de creado el proyecto, se exporta al SDK (Kit de Desarrollo de Software) en donde se crea un nuevo proyecto con el programa escrito en lenguaje C con el que se controla el Hardware. En el archivo principal se encuentran las funciones de inicialización de los módulos y de visualización de la información.

Con la implementación este trabajo se busca incrementar la velocidad de adquisición de datos provenientes de los sensores, gracias a la propiedad de concurrencia de la Zedboard y su facilidad para conectarse con dispositivos periféricos como el PmodBT2.

### Palabras clave

Bluetooth, comunicación, datos, FPGA, información, módulo, SDK, Vivado, Zedboard.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## RECONOCIMIENTOS

---

Primero que todo quiero dar infinitas gracias a mi esposa Paola Andrea López por su motivación y su apoyo incondicional durante toda la carrera. También quiero agradecer a todos los profesores con los que tuve la fortuna de ver cada uno de los cursos, por compartir sus conocimientos conmigo, en especial, al profesor Luis Fernando Castaño Londoño quien me asesoró durante todo este trabajo. Por último, a los compañeros del laboratorio de Microelectrónica y Nanotecnología gracias por su gentil atención.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

# ACRÓNIMOS

---

FPGA Arreglos de Compuertas Programables en Campo

SDK Kit de Desarrollo de Software

UART Transmisor-Receptor Asíncrono Universal

IP Propiedad Intelectual

HDL lenguaje de Descripción de Hardware

RN Roving Networks

MIO Multiplexado de Entrada/Salida

PL Lógica Programable

PS Sistema de Procesamiento

Pmod Módulo Periférico

SW Software

HW Hardware

BSP Paquete de Soporte de la Tarjeta

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	6
2. MARCO TEÓRICO .....	9
3. METODOLOGÍA .....	11
4. RESULTADOS Y DISCUSIÓN.....	16
5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO.....	22
REFERENCIAS .....	24
APÉNDICE.....	25

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

# 1. INTRODUCCIÓN

---

## **Generalidades.**

Las redes de comunicación vía Bluetooth han tenido gran acogida debido a la sencillez de implementación, a los bajos costos que representan, a la compatibilidad entre diferentes modelos gracias a la estandarización del protocolo y a la capacidad de actualización a medida que evoluciona a otras versiones.

Una de las principales desventajas de la tecnología Bluetooth es la velocidad de transmisión de información. Esto se debe a que la comunicación maestro esclavo retarda el proceso de transmisión de los demás módulos de la red. La cantidad de información que se transmite se ve afectada en algunas aplicaciones específicas, como en las que se requiere el envío de datos en tiempo real, por ejemplo, de una red de sensores.

Con la combinación entre la tecnología Bluetooth y la capacidad de procesamiento de datos de la tarjeta de desarrollo Zedboard se busca contrarrestar el inconveniente de la baja velocidad de transmisión de datos. Se espera reducir aproximadamente a la mitad los tiempos de recepción y, por consiguiente, aumentar la cantidad de información que se envía a través de las redes de comunicación.

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

### **Objetivo General**

Diseñar un sistema de comunicación basado en FPGA y la tecnología Bluetooth que reduzca el tiempo de transmisión de datos por medio de dos módulos configurados como maestros.

### **Objetivos Específicos**

1. Implementar el hardware basado en FPGA para la configuración y conexión de los módulos Pmod para Bluetooth utilizando módulos IP del Vivado.
2. Implementar el software de control de los módulos en lenguaje C que permita visualizar la información transmitida.
3. Comparar el rendimiento del sistema cuando se cambia la cantidad de módulos configurados como maestros.

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

### **Organización de la tesis.**

Para realizar el diseño de bloques se usó el entorno de desarrollo Vivado, el cual cuenta con los núcleos IP necesarios para conectar y configurar los módulos Bluetooth. El sistema implementado está conformado por el procesador ZYNQ y dos módulos Bluetooth PmodBT2. Se realizan las conexiones entre los núcleos, se valida el diseño, se crean los archivos HDL que son los que conectan los puertos de entrada/salida con los pines físicos de la Zedboard y por último se ejecuta el Bitstream para generar las salidas finales.

Una vez terminado el diseño del Hardware se exporta el proyecto con los archivos creados al entorno de desarrollo SDK para realizar la aplicación con la que se controlarán la tarjeta y los módulos periféricos. Esta transferencia de archivos de un entorno al otro la realiza internamente Vivado.

En la herramienta SDK se crea un nuevo proyecto en el que se desarrolla el software de control. En este proyecto se encuentran los archivos del hardware creados en Vivado, sus definiciones y la asignación de las direcciones en la memoria local. El software de control tiene como base el ejemplo que se encuentra en la carpeta drivers del proyecto exportado. En el archivo principal se crean las funciones de inicialización de los módulos Bluetooth y la transmisión UART que permite visualizar la información en el terminal y realizar la configuración avanzada de los módulos RN mediante comandos de código ASCII.

Una vez terminado el software de control, se compila y se programa la FPGA. En ese momento los módulos Bluetooth se energizan y quedan preparados para la conexión Bluetooth y posterior transmisión y recepción de información.

Para analizar el funcionamiento del sistema se realizan pruebas de recepción de datos usando un solo módulo maestro conectado a la Zedboard. Se realiza una conexión inalámbrica por medio de un teléfono celular. A través de una aplicación se envían datos al módulo Bluetooth conectado a la tarjeta. Adicionalmente, se realiza una conexión por medio de un hiperterminal para probar las posibles configuraciones según el protocolo RN42.

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## 2. MARCO TEÓRICO

La realización de proyectos tecnológicos utilizando las comunicaciones inalámbricas ha tenido gran acogida por la facilidad que tienen para su implementación. El protocolo Bluetooth es uno de los más usados por los bajo costos y la simplicidad para el desarrollo del hardware y el software. Este protocolo de comunicación integrado a los sistemas de desarrollo basados en FPGA, es una muy buena opción para el diseño de productos que requieren gran procesamiento de información.

En (Portilla, Riesgo, & de Castro, 2007), los autores usan una FPGA para aprovechar su flexibilidad para reconfigurar los nodos de la red sin necesidad de cambiar el hardware de procesamiento. También destacan la importancia de la tecnología Bluetooth para crear redes dinámicas en las que pueden variar los nodos sin alterar el funcionamiento de dichas redes. Esta tecnología es una de las primeras opciones debido a su velocidad de transmisión y a su bajo consumo de energía. Su principal desventaja radica en la configuración de la red cuando hay cientos de nodos que pertenecen a ella. Este proceso se vuelve inviable.

Una aplicación similar a la que se desarrolla en el presente trabajo, es la que se exponen (Hawayek, Hargrove, & BouSaba, 2013). En este documento los autores presentan el diseño de un proyecto para el área de la salud basado en una FPGA con un PmodBT integrado y una aplicación para un dispositivo Android. La interface UART del módulo Bluetooth permite la comunicación con la FPGA. La base de este sistema es la FPGA, pues, es la que recibe los datos de los sensores, los procesa y los envía al dispositivo móvil a través del PmodBT.

Los autores (Tana, Sazish, Ahmad, Sharif, & Amira, 2010), destacan las FPGA como el hardware ideal para realizar proyectos con tecnología inalámbrica. En este artículo, se exponen las capacidades de paralelismo y las propiedades de reconfiguración y procesamiento de señales que poseen estas tarjetas. También mencionan las ventajas de bajo consumo de energía y alta demanda de la comunicación Bluetooth. Estas dos tecnologías integradas reducen los recursos utilizados como la memoria, la potencia y el costo computacional.

Como puede verse, estos tres trabajos son un ejemplo de la capacidad que tienen las tarjetas de desarrollo combinadas con las cualidades de la tecnología Bluetooth. Ellos coinciden en que el bajo consumo de energía, la velocidad de procesamiento de datos y la

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

facilidad de implementación son factores que se deben tener en cuenta a la hora de diseñar productos tecnológicos.

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

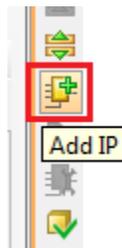
### 3. METODOLOGÍA

La implementación de este producto de laboratorio se basa en la guía “Primeros pasos con módulos periféricos IPs de Digilent”, (Digilent). En esta guía, el fabricante del dispositivo PmodBT2 describe cómo usar el núcleo IP en Vivado con el procesador Microblaze o el Zynq. Allí se especifican las plataformas y módulos periféricos que son compatibles y los requisitos de hardware y software en el cual se crean los proyectos en Vivado y SDK.

Los recursos utilizados para realizar la implementación son dos módulos Bluetooth PmodBT2, la Zedboard y el computador portátil personal.

En la primera parte del tutorial se realiza el diseño y configuración del hardware con el que se crea la red de comunicación Bluetooth. En el ejemplo que se desarrolla en esta guía se usa la versión 2015.4 de Vivado. Para este proyecto se trabaja con la versión 2015.3; debido a esto y a que se usa otra FPGA, cambia la forma en que se adiciona el módulo periférico al diagrama de bloques y cómo se realiza la configuración de pines.

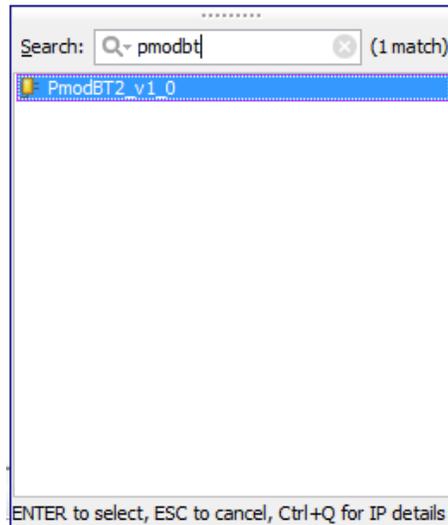
Para adicionar los módulos periféricos se da clic en el botón **Add IP** ubicado al lado izquierdo de la ventana del diagrama de bloques. (Ver Figura 1)



**Figura 1: Botón Add IP**

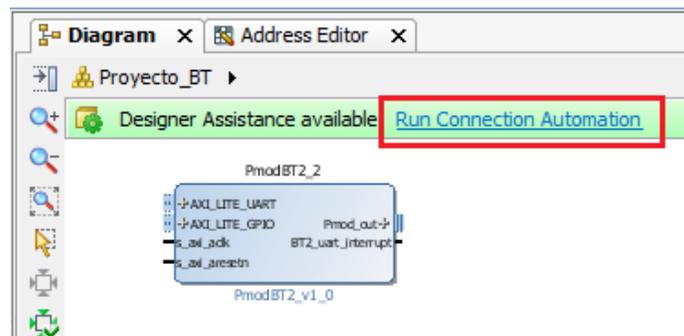
Cuando se despliegue la ventana con el listado de IPs disponibles, en el cuadro de búsqueda se digita **pmodbt**. Aparece resaltado el PmodBT2. (Ver Figura 2).

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22



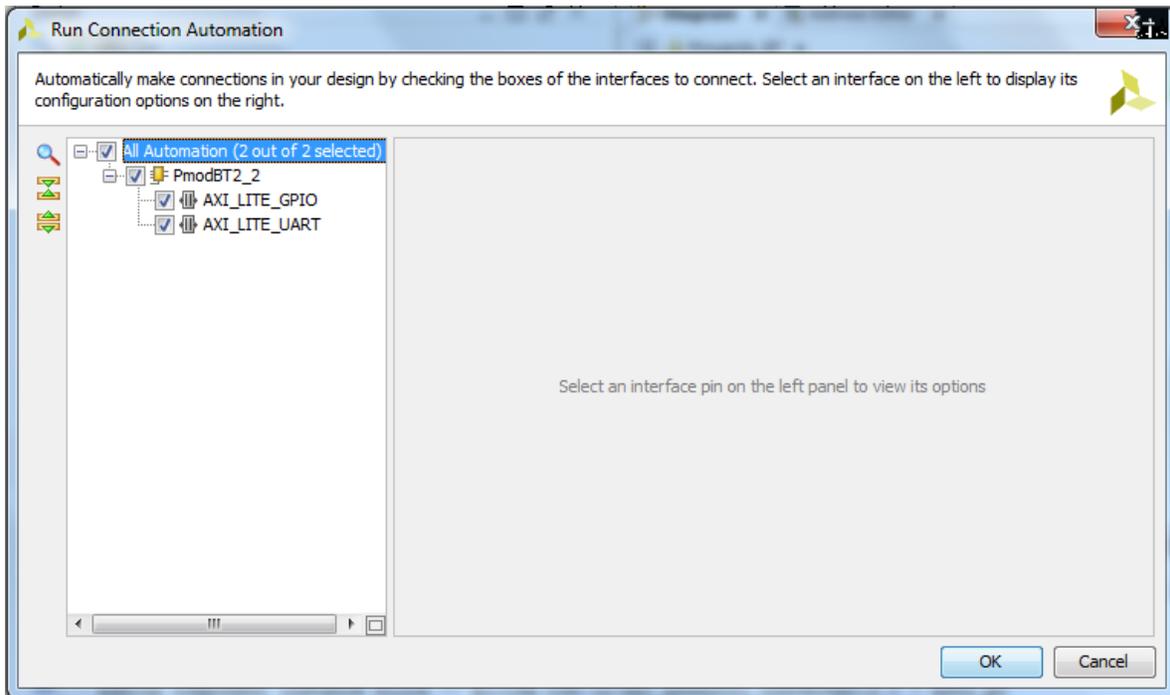
**Figura 2: Selección de Pmod**

Se agrega el bloque al diagrama dando Enter o doble clic sobre el IP resaltado.



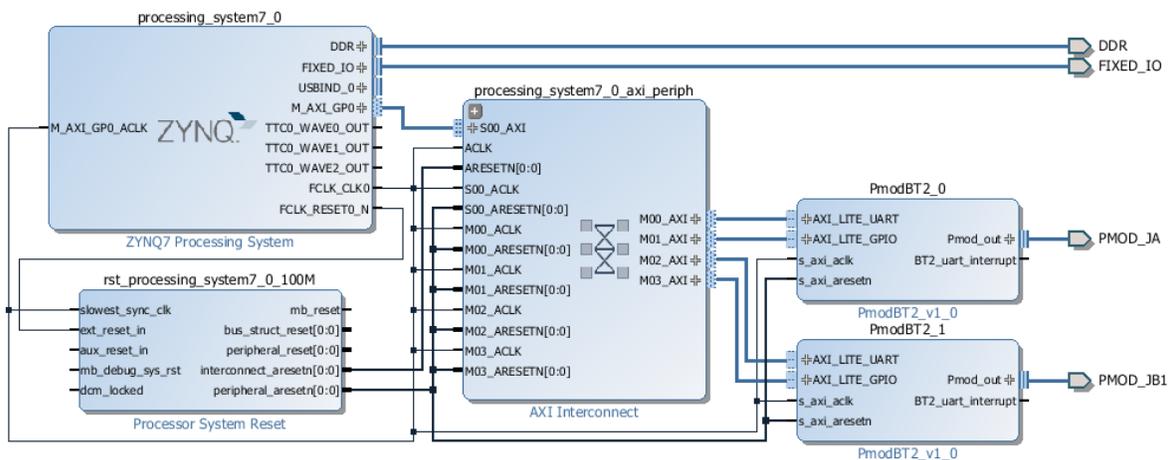
**Figura 3: Bloque PmodBT2**

Una vez agregado el bloque, se habilita automáticamente la asistencia de diseñador **Run Connection Automation** para realizar la conexión del módulo (ver Figura 3). Al dar clic en ella, aparecen las opciones para seleccionar los puertos que se van a conectar como se muestra en la Figura 4.



**Figura 4: Conexión de puertos del IP**

Después de realizar la conexión automática entre los núcleos y el procesador y generar la conexión externa para los Pmods, se tiene el diseño de bloques como se muestra en la Figura 5.



**Figura 5: Diagrama de bloques del proyecto**

Para realizar la asignación de pines, damos clic en **Implemented Designed** en la ventana de flujo de navegador, luego en la pestaña **I/O Ports**, desplegamos las carpetas de salidas de

los puertos y procedemos a realizar la asignación según el manual de la tarjeta de desarrollo como se muestra en la Figura 6.

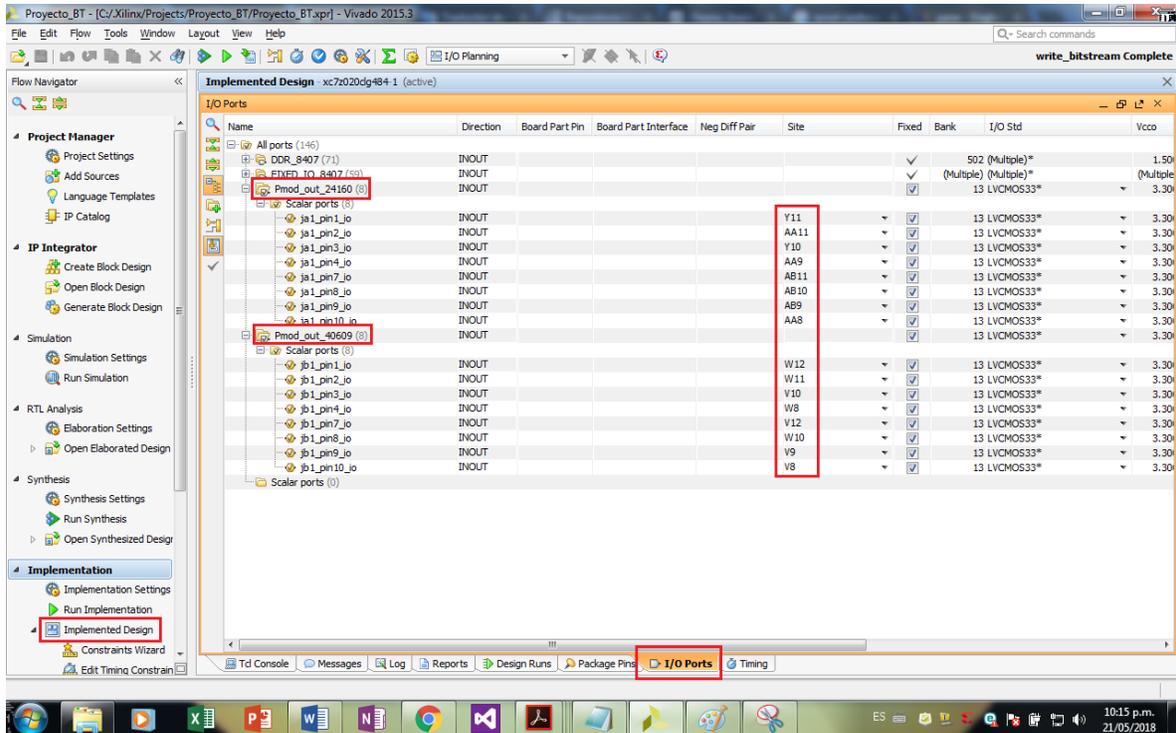


Figura 6: Asignación de pines.

La asignación de pines se realiza basándose en la guía de usuario de la tarjeta de desarrollo. En esta guía se encuentra la descripción de los conectores para los módulos periféricos y los nombres de los respectivos pines en la Zedboard (AVNET Electronics Marketing, 2014).

Después de realizar la asignación de pines, se genera el **Bitstream**. Como se mencionó antes, en este paso se crean los archivos de salida, que entre otras cosas, incluyen las direcciones en la memoria de los núcleos IP utilizados en el diagrama y que se exportarán al SDK.

Cuando se exporta el proyecto al SDK, en la pestaña explorador de proyectos ubicada al lado izquierdo de la ventana, se puede ver la plataforma de hardware en la cual se encuentran todas las definiciones de diseño, las interfaces IP, la información de la señal de salida y las direcciones de memoria local. Estas especificaciones de diseño de hardware y los bloques IP incluidos se muestran en el archivo **system.hdf**. El SDK es independiente de

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Vivado. Desde este punto se creará el software en lenguaje C sobre el diseño hardware exportado.

Los controladores para el dispositivo IP Pmod se encuentran en la carpeta de la plataforma de hardware, en **/drivers**. Para editar estos controladores, se deben usar las versiones que se encuentran en el paquete de soporte de la placa en **libsrc**. Si se modifican los controladores, hay que tener en cuenta que cualquier cambio en el hardware sobrescribirá estos cambios, así como cualquier uso de la opción **Regenerate BSP sources**.

Para verificar el funcionamiento inicial del sistema se procede a adicionar el archivo **main.c** que se encuentra en la carpeta de ejemplos como lo indica la guía. Después de programar la Zedboard y correr el programa, se realiza la conexión entre el Pmod y un celular usando la aplicación Bluetooth Terminal HC-05. En la Figura 7 se muestra la interfaz de la aplicación.

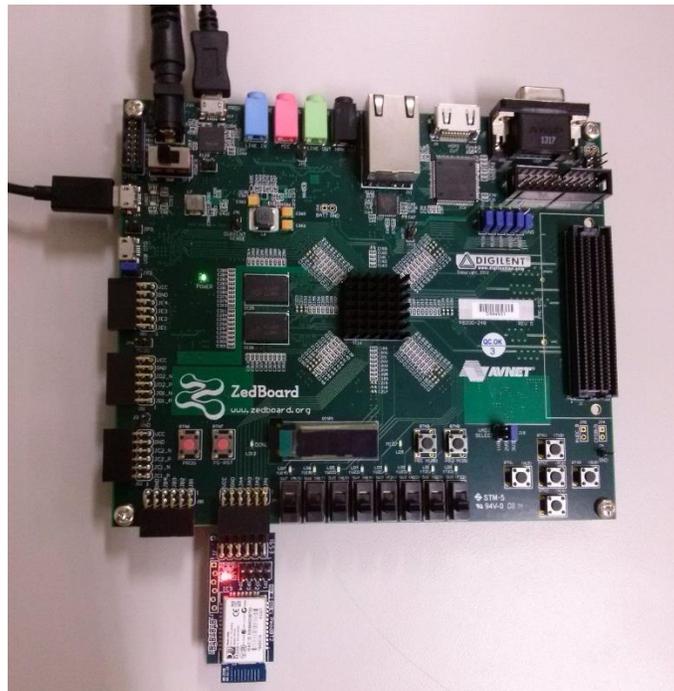


**Figura 7: Aplicación Bluetooth para celular**

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## 4. RESULTADOS Y DISCUSIÓN

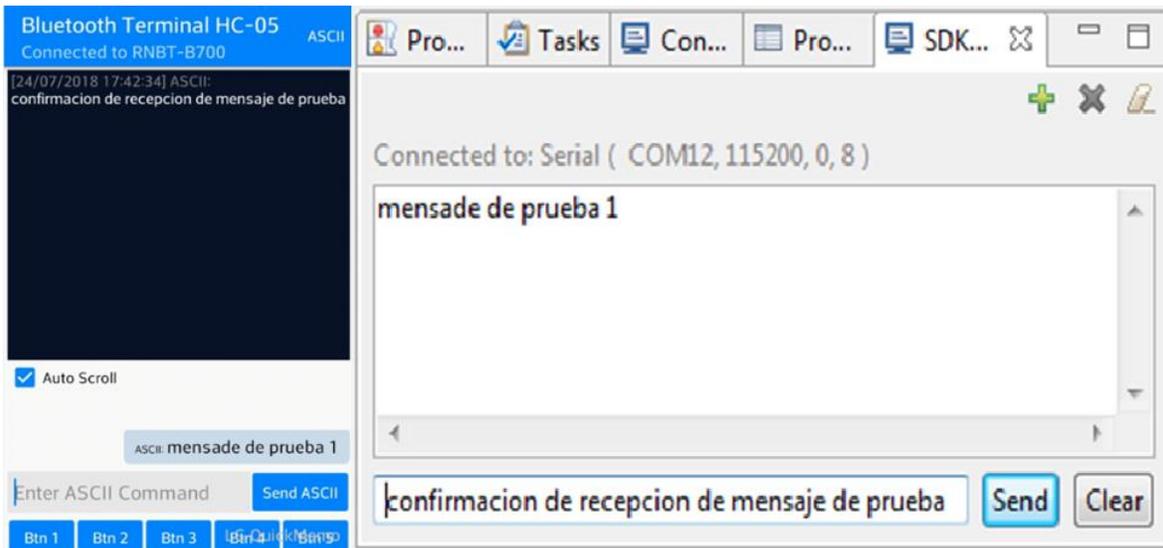
Como resultado se obtuvo una conexión exitosa entre el PmodBT2 y un celular a través de la aplicación Bluetooth Terminal HC-05. En esta primera prueba se tenía un solo módulo conectado a la tarjeta como se muestra en la Figura 8.



**Figura 8: Conexión Zedboard-PmodBT2**

La transmisión de información se verificó enviando un mensaje desde el celular hacia el módulo Bluetooth. El mensaje recibido por el módulo es visualizado en el terminal de SDK. A través de esta misma terminal se envió un mensaje al celular. Con esto se comprobó la transmisión y recepción de datos del módulo Bluetooth. En la Figura 9 se muestra al lado derecho la interfaz gráfica de la aplicación del celular y al lado derecho la terminal de SDK.

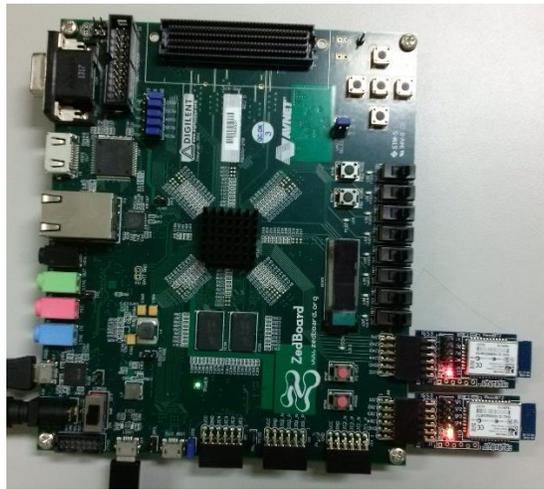
 Institución Universitaria	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22



**Figura 9: Verificación de transmisión de información**

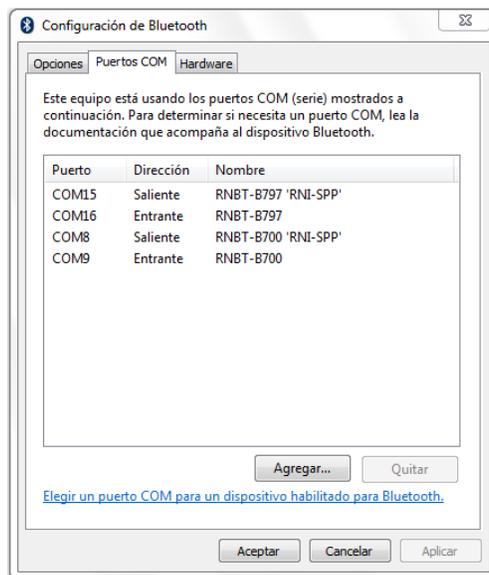
Se envió desde el celular un archivo de texto con un tamaño de 9 Kilobytes y se evidenció que cuando empezaba la transmisión de datos se perdía la conexión. En este caso no se pudo comprobar si esta falla era ocasionada por el PmodBT2 o por la aplicación que se estaba usando en el celular. Se envió el mismo archivo desde la terminal de SDK hacia el celular y la transmisión fue exitosa.

Después de realizar esta prueba se modificó el software de control para realizar la conexión de dos PmodBT2 a la Zedboard. Esta modificación consistía en agregar una nueva instancia del PmodBT2 en el programa como se muestra en la Figura 10. También se agregó su respectiva inicialización y la transmisión y recepción de datos. Debido a que el diseño de bloques fue creado con dos módulos Bluetooth, las direcciones de memoria del hardware ya se habían importado desde Vivado. Se programó la tarjeta y se comprobó que ambos módulos estaban energizados y listos para establecer una comunicación.



**Figura 10: Conexión Zedboard – 2 PmodBT2**

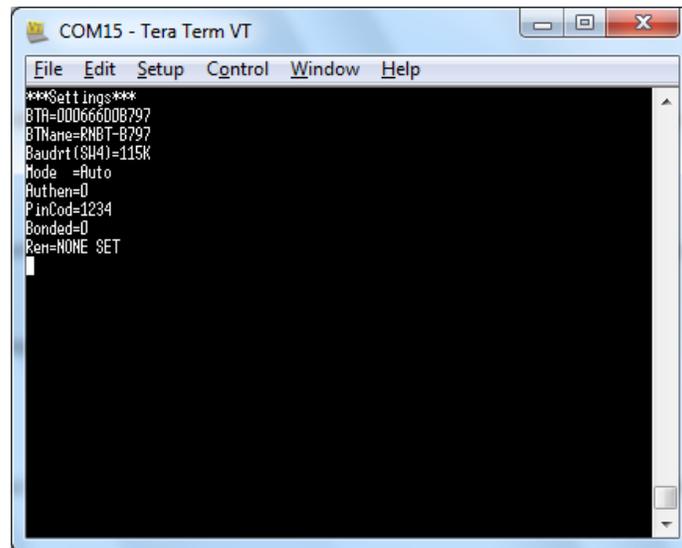
Utilizando los recursos del computador portátil se realizó una conexión Bluetooth con ambos módulos. Esto con el objetivo de probar las configuraciones que se pueden realizar según el manual de usuario de los módulos RN42. Cuando se establece el enlace computador-PmodBT2 se asignan los puertos COM (serie) para comunicación saliente o entrante. A través de estos puertos se realiza la conexión con los módulos usando un emulador de terminal.



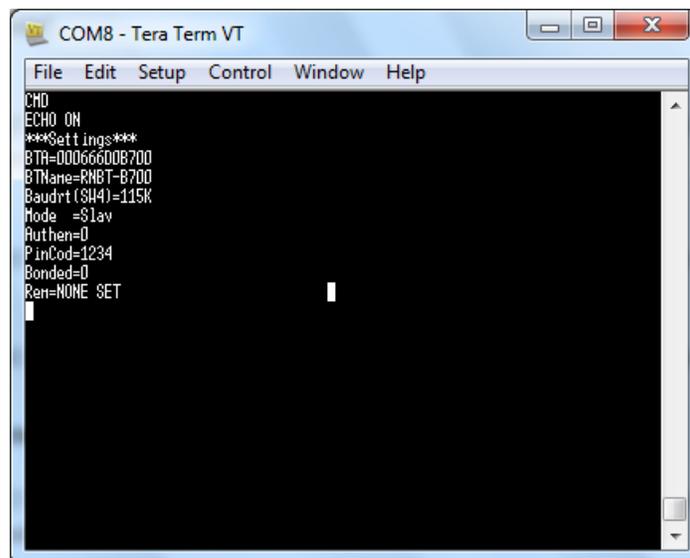
**Figura 11: Asignación de Puertos COM para los PmodBT2**

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Usando los puertos asignados para comunicación saliente y mediante el emulador de terminal **Tera Term VT**, usando los comandos respectivos se realizó una comunicación entre un módulo configurado como maestro y otro como esclavo. Los comando y la información detallada sobre este proceso se encuentra en el manual de usuario mencionado, (Roving Networks Inc., 2013).



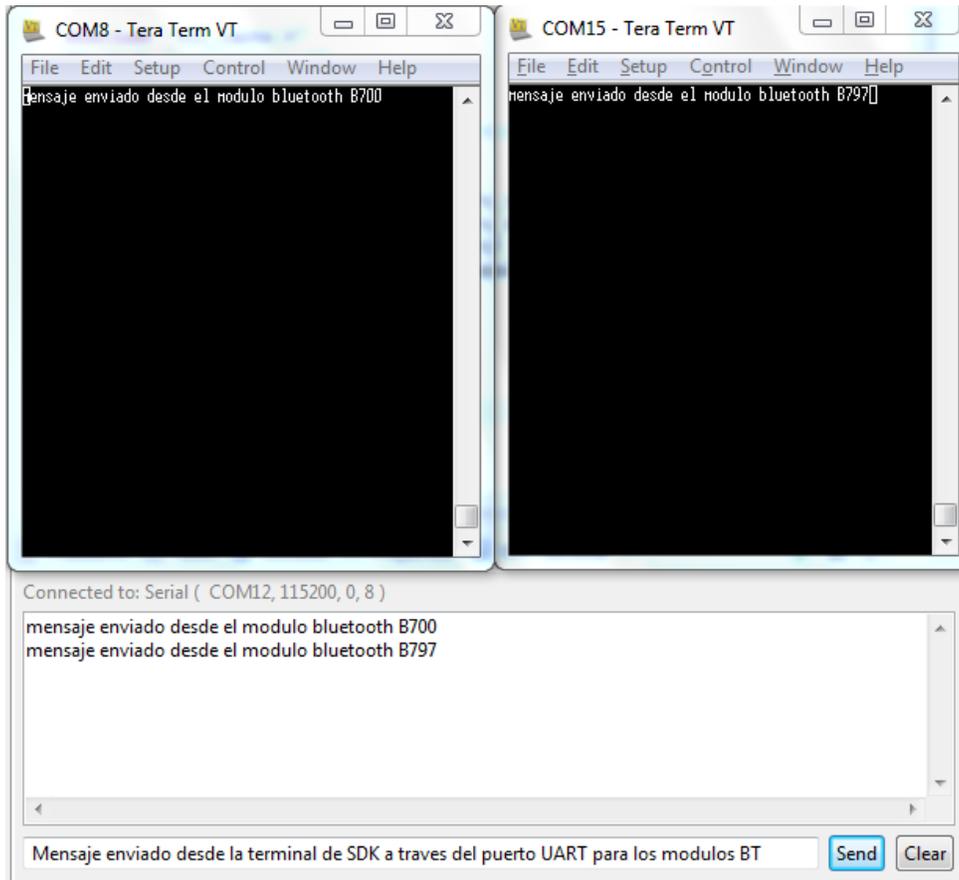
**Figura 12: Configuraciones Módulo Maestro.**



**Figura 13: Configuraciones Módulo Esclavo**

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Por último, se realiza una conexión con los módulos Bluetooth usando el emulador de terminal **Tera Term VT**. Se comprueba la recepción de datos enviando mensajes a través de dichas terminales y se verifica la recepción de los datos en el terminal de SDK.



**Figura 14: Prueba de Recepción de Datos.**

De igual manera, se envía un mensaje a través del terminal SDK para probar la transmisión de datos de los módulos. El mensaje enviado se visualiza en las terminales conectadas a los puertos COM salientes. Debido a que solo se ha activado el puerto UART1 en vivo, la información que se envíe desde el terminal SDK será transmitida por ambos módulos. Es por eso que el mensaje enviado se ve al mismo tiempo en las respectivas terminales conectadas a los puertos COM salientes.

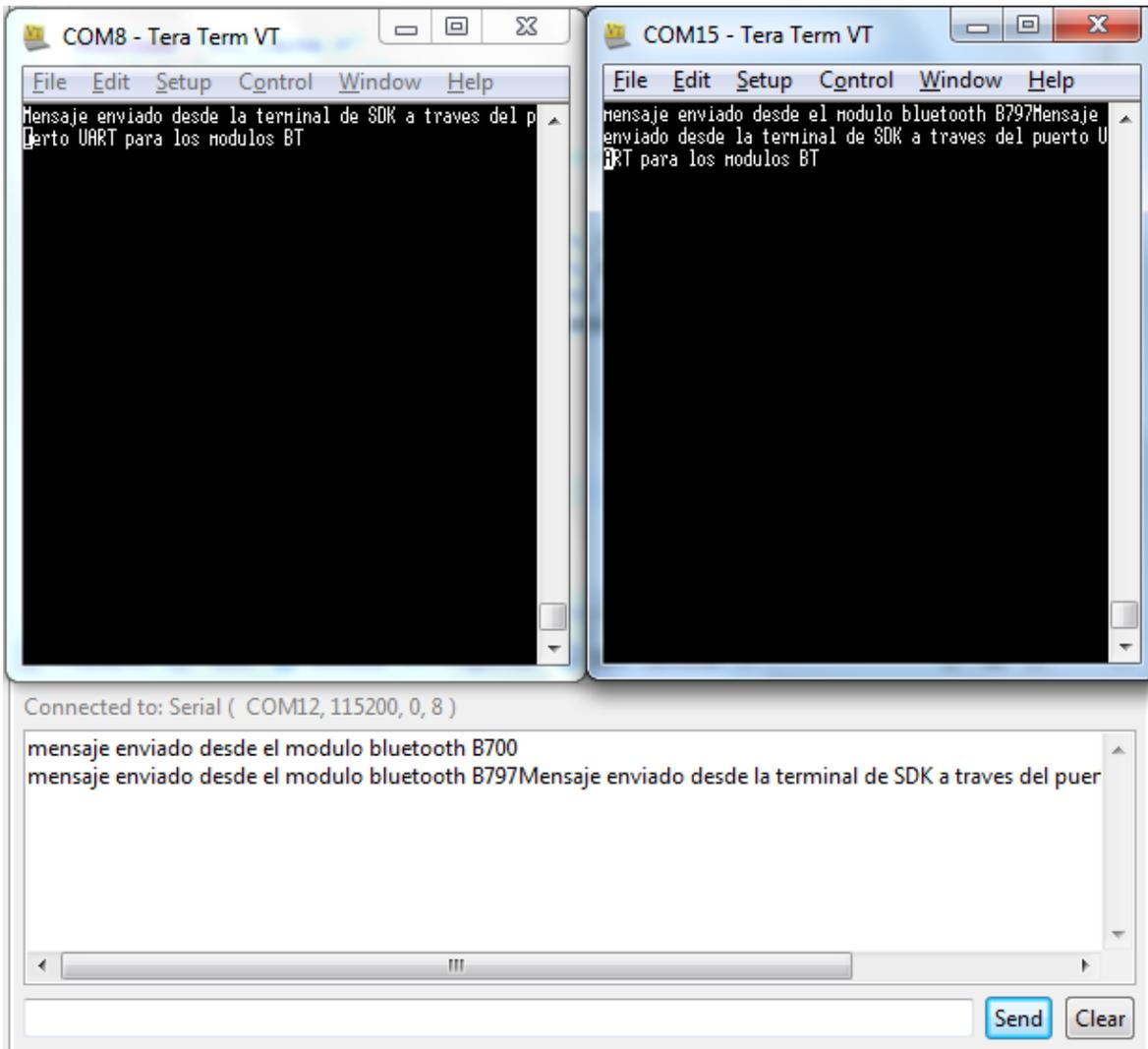


Figura 15: Prueba de Transmisión de Datos.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## 5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

### CONCLUSIONES:

- En este trabajo se presenta la implementación de un sistema de comunicación diseñado en este proyecto es solo un comienzo para crear redes Bluetooth con aplicaciones reales en diversos campos. En las pruebas realizadas se verificó la correcta transmisión de pequeños archivos de texto. Se debe someter el sistema a pruebas más rigurosas como la transmisión de grandes paquetes de datos o señales de sensores. En teoría la configuración de los PmodBT2 es simple, pero para crear una red completa con un módulo maestro y varios módulos esclavos se requiere buena práctica y conocimiento de la guía de usuario. Lamentablemente no se logró determinar el tiempo de transmisión de archivos y debido a esto no es posible calcular la velocidad y el rendimiento del sistema.
- El software implementado para controlar los módulos Bluetooth está diseñado para que los datos transmitidos vayan a través del puerto UART1 a un solo terminal de SDK. Esto genera que la información que se visualiza no sea muy clara cuando ambos módulos están recibiendo al mismo tiempo. De igual manera, al enviar datos desde el terminal, estos son transmitidos simultáneamente por los dos módulos. Este es un punto en contra pero que se puede mejorar activando el puerto UART0 desde el diseño en Vivado.
- Como se mencionó antes, no es posible comparar el rendimiento del sistema cuando se cambia la cantidad de módulos maestros ya que no se logró implementar un temporizador para medir la velocidad de transmisión de datos. Esta parte queda expresada en las recomendaciones.

	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

**RECOMENDACIONES:**

- Agregar un temporizador al diseño ayudaría a llevar un control preciso de los tiempos de transmisión de información. Con esto se podría calcular exactamente el rendimiento del sistema.
- Utilizar la capacidad de los puertos para Pmod de la Zedboard para ampliar el número de módulos maestros. Esto aumentaría la red y por consiguiente la cantidad de información recibida. Aunque tal vez la configuración de los módulos se convierta en un proceso un poco engorroso.

**TRABAJO FUTURO:**

- Como trabajo futuro, se puede integrar el sistema desarrollado al trabajo de redes de sensores acelerómetros con los que se están experimentando actualmente en el laboratorio de robótica.

 Institución Universitaria	<b>INFORME FINAL DE TRABAJO DE GRADO</b>	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## REFERENCIAS

---

- AVNET Electronics Marketing. (2014). *ZedBoard (Zynq Evaluation and Development) Hardware User's Guide*. Phoenix, Arizona. Recuperado el Febrero de 2018, de [http://zedboard.org/sites/default/files/documentations/ZedBoard\\_HW\\_UG\\_v2\\_2.pdf](http://zedboard.org/sites/default/files/documentations/ZedBoard_HW_UG_v2_2.pdf)
- Digilent. (2016). *PmodBT Reference Manual*. Pullman, WA. Recuperado el Noviembre de 2017, de [https://reference.digilentinc.com/\\_media/reference/pmod/pmodbt2/pmodbt2\\_rm.pdf](https://reference.digilentinc.com/_media/reference/pmod/pmodbt2/pmodbt2_rm.pdf)
- Digilent. (s.f.). *Getting Started with Digilent Pmod IPs*. Pullman, WA. Recuperado el Noviembre de 2017, de <https://reference.digilentinc.com/learn/programmable-logic/tutorials/pmod-ips/start>
- Hawayek, S., Hargrove, C., & BouSaba, N. (2013). Real-time bluetooth communication between an FPGA based embedded system and an Android phone. *2013 Proceedings of IEEE Southeastcon* (pág. 4). Jacksonville, FL.: IEEE. Obtenido de <https://ieeexplore-ieee-org.itm.elogim.com:2443/document/6567418/>
- Microchip Technology Inc. (2015). *RN42/RN42N - Class 2 Bluetooth Module with EDR Support*. Chandler, Arizona. Recuperado el Abril de 2018, de <http://ww1.microchip.com/downloads/en/DeviceDoc/50002328A.pdf>
- Portilla, J., Riesgo, T., & de Castro, Á. (2007). A Reconfigurable Fpga-Based Architecture for Modular Nodes in Wireless Sensor Networks. *2007 3rd Southern Conference on Programmable Logic* (pág. 4). Mar del Plata: IEEE. Obtenido de <https://ieeexplore-ieee-org.itm.elogim.com:2443/document/4234347/>
- Roving Networks Inc. (2013). *Bluetooth Data Module Command Reference & Advanced Information User's Guide*. Los Gatos, California. Recuperado el Noviembre de 2017, de [http://ww1.microchip.com/downloads/en/DeviceDoc/bluetooth\\_cr\\_UG-v1.0r.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/bluetooth_cr_UG-v1.0r.pdf)
- Tana, H., Sazish, A., Ahmad, A., Sharif, M. S., & Amira, A. (2010). Efficient FPGA implementation of a wireless communication system using Bluetooth connectivity. *Proceedings of 2010 IEEE International Symposium on Circuits and Systems* (pág. 4). Paris: IEEE. Obtenido de <https://ieeexplore-ieee-org.itm.elogim.com:2443/document/5537610/>

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

## APÉNDICE

---

### A. Proyecto Completo en Vivado 2015.3

Ver archivo adjunto.

FIRMA ESTUDIANTES Dago R.

FIRMA ASESOR San Fernando Castro Lora

FECHA ENTREGA: 01-08-2018

FIRMA COMITÉ TRABAJO DE GRADO DE LA FACULTAD \_\_\_\_\_

RECHAZADO\_\_\_      ACEPTADO\_\_\_      ACEPTADO CON MODIFICACIONES\_\_\_

ACTA NO. \_\_\_\_\_

FECHA ENTREGA: \_\_\_\_\_

FIRMA CONSEJO DE FACULTAD \_\_\_\_\_

ACTA NO. \_\_\_\_\_

FECHA ENTREGA: \_\_\_\_\_