

# Capítulo 1

## Introducción

El presente informe busca dar a conocer al lector las tareas y actividades desarrolladas por el autor, a fin de dar cuenta sobre lo realizado en el marco del Trabajo Final de la carrera Ingeniería Electrónica, dictada en la Facultad de Ingeniería de la Universidad Nacional de San Juan.

El objetivo principal de este trabajo es la implementación de un sistema de comunicación mediante la norma USB 2.0 para sistemas científicos y/o tecnológicos desarrollados en FPGA.

A lo largo de este trabajo, se busca que el lector comprenda el problema que se pretende resolver, la configuración del sistema propuesto, los fundamentos que dan base a la dicha configuración y el modo de uso del sistema.

Para ello, este Capítulo explica, en primer lugar, la motivación, es decir, se busca aclarar la problemática que se busca resolver. Luego, se detallan los objetivos que persigue este trabajo. Luego, el lector puede conocer la organización del presente informe. Además, se comparan diferentes protocolos de comunicación con el fin de justificar la elección del protocolo a implementar y finalmente, se brinda un repaso de algunos conceptos importantes de la norma USB que luego se utilizan en el trabajo desarrollado.

### 1.1. Motivación

La información es el resultado de recopilar, ordenar y procesar un conjunto de datos, de forma tal que permitan adquirir mayor conocimiento sobre un asunto determinado y otorguen un significado mayor que cada uno de los datos por separado. Por ejemplo, Si un caprintero quiere medir la distancia de una barra de madera, utilizará una cinta métrica. Se denomina cinta métrica a una lámina metálica que posee marcas graduadas. Esta graduación, se realiza comparando la lamina metal con una barra patrón que indica que la distancia indicada se corresponde con la convención de distancia.

El carpintero compara el tamaño de las patas de la mesa con la cinta métrica, que a su vez, posee registrada su distancia en función del patrón. Esto quiere decir que el dato 1, la longitud del patron, junto al dato 2, escala graduada de la cinta, más el dato 3, la longitud de la cinta

métrica, permiten al carpintero cambiar su estado de desconocido a conocido, con respecto a la longitud del trozo de madera, a través de la información proporcionada por el conjunto de datos.

### **La Ciencia como fuente de datos**

La Ciencia es un conjunto de técnicas y procedimientos que, a través de un método científico, busca adquirir, descubrir y/o desarrollar nuevo conocimiento. Se puede deducir, entonces, que la ciencia produce, de forma fundamental, datos, que luego de un procesamiento se transforman en información. A su vez, el estudio detallado de esta información genera conocimiento.

Cuando se nombra la palabra ciencia, se hace referencia a un conjunto conformado por diferentes objetos de estudio. El objeto de estudio es el sujeto a través del cuál se da la principal clasificación de las diferentes ramas de la Ciencia: las Ciencias Sociales estudian las relaciones humanas y las Ciencias Naturales dedican sus esfuerzos a objetos que se encuentran en la naturaleza. Dentro del último grupo encontramos las Ciencias de la Tierra se enfocan en una rama más particular de la naturaleza, como lo es el estudio de la superficie terrestre; y siguiendo así se puede encontrar muchas más clasificaciones de ciencias, incluso como ramas englobadas por las anteriormente nombradas.

### **Las herramientas que adquieren datos**

Los grados de avance que ha experimentado la tecnología en general, y la electrónica en particular, gracias a la industria de los semiconductores, permite que la producción científica pueda adquirir una gran cantidad de datos.

Actualmente, se encuentra en desarrollo un gran número de sensores que permitan obtener flujos crecientes de datos. Se denomina sensor a los dispositivos que adquieren valores registrables en función de alguna propiedad física. Los sensores constan de, al menos, un transductor, es decir, un elemento que transforma una magnitud física en otra, y de un sistema que convierta esta magnitud en una señal eléctrica útil, si no la otorga el transductor mismo.

Uno de los desarrollos que se encuentran en boga es el de sensores y sistemas que adquieran imágenes de diferentes tipos. Desde el punto de vista digital, una imagen es un arreglo bidimensional de números, los cuales pueden ser exhibidos en una pantalla en forma de intensidad y colores de luz. Por esto, un sensor de imagen puede estar compuesto, bien por un arreglo bidimensional de transductores, por un transductor que es simultáneamente desplazado y leído y/o por una combinación de ambos métodos. En todos los casos, es de suma utilidad que la lectura de imágenes sea realizada en el menor tiempo posible ya que cada imagen conlleva una cantidad de datos que no es despreciable.

### **Sobre circuitos de alta velocidad**

Desde un punto de vista electrónico, para poder mover grandes volúmenes de datos en forma digital, se requiere de circuitos que sean capaces de manejar niveles de tensión que se modifican con alta velocidad. Esto no es trivial, ya que a frecuencias de funcionamiento elevadas

se evidencian capacidades no contempladas en diseño que perjudican el desempeño de los sistemas.

En gran medida, la incorporación y evolución de microcontroladores permite obtener una captura y obtención cada vez superior de datos. Sin embargo, al poseer una estructura rígida y su capacidad de procesamiento está limitada a una instrucción por vez, no es del todo adecuada para el desarrollo de nuevos dispositivos y técnicas de medida especializadas. En ciertos casos, la posibilidad de realizar cientos de operaciones es excesivo para aplicaciones muy específicas. Una solución óptima, sin considerar los costos asociados a esto, sería el desarrollo de un circuito integrado de aplicación específica (ASIC) que resuelva en forma precisa el problema. Sin embargo, cuando sí se considera el costo asociado a este enfoque, se vuelve una solución totalmente ineficiente, del orden de las decenas de miles de dolares.

El siguiente enfoque, óptimo al criterio de este trabajo, es la utilización de FPGAs para realizar el desarrollo. Al poseer, este tipo de herramientas, la posibilidad de conectar, adquirir y procesar en paralelo y a alta velocidad, incorpora una gran ventaja con respecto a un procesador. A su vez, al ser programable, se adapta mejor a una solución específica. En cuanto al costo de un FPGA, van desde las decenas a las centenas de dolar. Esto es de dos ordenes de magnitud más económico que un ASIC.

Existen diversas publicaciones en donde se observa el uso de FPGAs para la implementación de sistemas que producen imágenes. Por ejemplo, el desarrollo de un sensor de radiación utilizando un sensor CMOS comercial. Para ello, los autores utilizaron un FPGA para configurar y transmitir imagenes a un grabador de video a través de puerto UART. Esto permitió adquirir una imagen a través de un disparador realizado con un pulsador[1].

Se denomina ultrasonografía a la técnica de adquirir imágenes basandose en rebotes de ultrasonido. Sus aplicaciones son múltiple, en las que se destaca el diagnóstico médico, ya que es una técnica no invasiva y sin riesgos de radiación ionizante sobre el paciente. Un trabajo reciente desarrolló un sistema de ecografía médica con bajo costo utilizando un FPGA[2]. El autor también presentó un algoritmo realizado y probado en PC. Luego se implementó e en una FPGA.

Existen sistemas de telescopía implementados con FPGA. Yanagisawa, entre otros, desarrolló un sistema con telescopios pequeños para explorar objetos de campo cercano con la finalidad de monitorear cuerpos celestes que puedan colisionar con el planeta[3]. Los autores utilizan la velocidad de los circuitos implementados en FPGA para minimizar el tiempo de adquisición.

### El procesamiento de datos

La obtención de datos por si misma no otorga información. Para ello, es probable que un gran flujo de datos requiera de un procesamiento y análisis exhaustivo de los mismos. La invención y evolución de las computadoras, como así también el desarrollo de nuevos algoritmos, dan lugar a procesamiento de datos cada vez más complejos en tiempos mucho menores. Las primeras ENIAC, computadora de propósito general desarrollada en el año 1946 para el cálculo de tablas balísticas de las fuerzas armadas estadounidenses, podía ejecutar 20 operación cada 10  $\mu$ s, es decir, a 100 kHz. Hoy por hoy, un procesador moderno corre a unidades de GHz.

Es por esto que la PC *Personal Computer* se ha transformado en la herramienta indispensable en cualquier ámbito, pero en especial en los entornos en donde se requiere el manejo, cálculo, procesamiento y análisis de grandes cantidades de datos e información.

Se torna de suma utilidad, entonces, proveer una comunicación efectiva y robusta que permita mover grandes volúmenes de datos en poco tiempo, y de esta forma facilitar los tiempos de desarrollo, las pruebas y depuración de sistemas.

Desde la inclusión de la norma USB, en el año 1996 a la fecha, se ha convertido en el elemento que no falta en ningún equipo, al punto tal que ha desplazado a cualquier otro conector. Al punto tal es esto, que para requerir algún puerto adicional que no sea de esta norma, cualquier comprador debe especificar que así sea, mas no es necesario especificar que tiene USB como norma de conexión.

El presente trabajo pretende ofrecer una comunicación basada en la versión 2.0 de la norma USB entre una PC y un FPGA, y otorgue una herramienta adicional a científicos que desarrollen o utilicen sistemas basados en FPGA.

## 1.2. Objetivos

### 1.2.1. Objetivo Principal

El objetivo del presente trabajo es obtener una comunicación USB 2.0 de alta velocidad entre una PC y un FPGA.

Esta comunicación debe realizarse y documentarse de forma tal que pueda ser usado posteriormente en aplicaciones científicas desarrolladas con FPGA's.

### 1.2.2. Objetivos Particulares

Para la consecución del objetivo general, se deben cumplir los siguientes objetivos particulares:

- Comprender el funcionamiento del protocolo USB.
- Seleccionar los componentes a utilizar.
- Configurar los componentes seleccionados.
- Desarrollar un núcleo en VHDL que sirva de interfaz.
- Diseñar e implementar la interconexión de los componentes seleccionados.
- Verificar el sistema desarrollado.
- Desarrollar un documento que explique el modo de uso del código VHDL utilizado.

## 1.3. Estructura del Informe

El presente informe se divide en 2 bloques principales: uno referido al desarrollo del sistema y el siguiente a su forma de uso y verificación.

Dentro del bloque referido al desarrollo del sistema, se encuentran los primeros 5 capítulos:

1. **Introducción:** En este capítulo se intenta exponer lo que motiva el presente trabajo, el objetivo y alcance que el trabajo busca y la estructura del mismo. Se brindan, además, conceptos importantes de la norma USB que son significativos para los objetivos de este trabajo.
2. ??:

## 1.4. El protocolo USB

El protocolo USB es un sistema de comunicación diseñado durante los años 90 por los seis fabricantes de la industria de las computadoras, Compaq, Intel, Microsoft, Hewlett-Packard, Lucent, NEC y Philips, con la idea de proveer a su industria de un sistema que permita la conexión entre las PC's y los periféricos con un formato estandar, de forma tal que permita la compatibilidad entre los distintos fabricantes.

Hasta ese momento, el gran ecosistema de periféricos, sumado a los nuevos avances y desarrollos, hacia muy compleja la conectividad de todos ellos. Cada uno de los fabricantes desarrollaba componentes con fichas, niveles de tensión, velocidades, drivers y un sinnúmero de etc diferentes, lo cuál dificultaba al usuario estar al día y poder utilizar cada componente que compraba. Lo más probable era encontrar que cada vez que uno comparaba una PC, debía cambiar el teclado, el mouse o algún periférico específico. Esto también complicaba a las mismas empresas productoras, por que la introducción de un nuevo sistema requería un mucho soporte extra para poder conectar todo lo ya existente.

Todo esto, quedó saldado con el standar USB, que debido a la gran cuota de mercado de sus desarrolladores, rápidamente fue introducido y se transformó en la norma a la hora de seleccionar un protocolo. Al punto tal esto se cumplió que hoy, más de 20 años después, es muy difícil encontrar PC's con otro tipo de puertos, salvo que en el momento de su compra uno requiera un puerto específicamente. Sin embargo, por norma, cualquier PC nueva disponible en el mercado debe poseer puertos USB para la conexión de los periféricos.

Desde el punto de vista técnico, el protocolo USB es un sistema del tipo maestro-esclavo, donde el maestro, denominado HOST, debe ser necesariamente una PC y cualquier periférico a ella acoplada será un esclavo.

Para describirlo es conveniente tal vez separar el protocolo en tres partes. Una parte física, en donde se definen los componentes que intervienen, una capa de protocolo, en donde se define el formato y el marco en el que son enviados los paquetes, como se direcciona

y como se comunican entre sí, y una parte lógica, en donde cada componente es visto solamente como un extremo y define como fluyen los datos desde un extremo hacia la PC y viceversa.

### 1.4.1. Capa física

En esta sección no se describirán los detalles de las conexiones eléctricas ni mecánicas a las que se refieren las especificaciones de la norma USB debido a dos cuestiones fundamentales. Una de ellas es que toda esta sección de la norma está resuelta ya por los fabricantes del chip de Cypress. Este chip maneja todas las señales, arma y desarma los paquetes que salen hacia la PC y que llegan de ella respectivamente. Por otro lado, no es el objetivo de este trabajo adentrarse en esos detalles. Gracias a la extensión de este tipo de comunicación existen una gran cantidad de fabricantes en el mercado que fabrican cada uno de los componentes, ya sean, cables, conectores en todas sus versiones, adaptadores de un tipo de enchufe a otro, su costo es despreciable con respecto a cualquier tipo de desarrollo en ese sentido y son de una muy buena calidad, en el sentido que todos cumplen con lo que la norma establece.

Si se hará a continuación una descripción de los dispositivos físicos y su categoría, según la norma, en función del rol que cumplen.

La comunicación USB posee una topología maestro-esclavo. Es decir, Existe un dispositivo que dirige todas las transferencias de datos y otros dispositivos que responden luego de que el maestro emite una directiva. Por esto, el elemento central de cualquier comunicación USB es el HOST (director o anfitrión, por su traducción de la voz inglesa). Este dispositivo que en la mayoría de los casos es la PC, aunque también puede ser algún dispositivo inteligente como un smartphone, es el que posee un Host USB Controller. El HOST se encarga de enviar los tokens a todos los periféricos, con la dirección del dispositivo, el sentido de la comunicación, el tipo de transferencia que se espera y todas las acciones de control que el sistema requiera.

En el otro extremo de la comunicación, se encuentran los dispositivos. Los dispositivos son todos los periféricos que actúan como fuente o sumidero de información. Es decir, en caso de periféricos de entrada, serán una fuente de información hacia el Host. Si los periféricos son de salida, serán un sumidero de la información que proporciona la PC. Los casos de periféricos de entrada/salida, se denominarán periféricos compuestos.

Existe también, a los fines de la norma, un elemento intermedio, denominado HUB (concentrador o distribuidor, según la traducción del inglés). Este dispositivo se encarga de conectar dos o más dispositivos, ya sea de entrada o salida, de recibir y enviar las direcciones y de regenerar las señales que el host envía y deben ser recibidas por los dispositivos, o bien, los datos que fluyen por el sistema.

### 1.4.2. Capa de protocolo

En la capa de protocolo, las especificaciones detallan cómo se compone el marco y cómo los paquetes deben ser armados para que sean efectivamente enviados a través del sistema.

Cada mensaje que se intercambia por el bus se denomina paquete. Cada paquete posee en su inicio lo que se denomina PID. el PID (del inglés packet ID o identificador del paquete) puede ser de 4 tipos, definidos por cada uno de los tipos de paquetes que puede haber:

- en prime lugar se encuentran los paquetes token, a través del cual el host envía las directivas a los distintos periféricos. Estas directivas pueden ser IN, cuando solicita datos a un periférico; OUT, cuando va a enviar datos a un dispositivo; SOF, que indica el inicio de cada cuadro, para que cada dispositivo se sincronice y SETUP, cuando va a enviar un paquete de configuración a algún dispositivo.
- el segundo tipo de paquetes es el paquete de datos. Este tipo de PID puede ser emitido por un dispositivo, si es que envía datos al host o bien por el mismo host cuando el flujo de datos es a la inversa.
- el tercer tipo de paquetes es un paquete de reconocimiento, denominado ACK, (por acknowledge o reconocimiento). Este paquete es enviado por los periféricos y le da idea al host de cual es el estado del dispositivo, es decir, si se encuentra operativo o no, si se encuentra ocupado o si recibió la transferencia de forma correcta.
- finalmente existen paquetes especiales, a través de los cuales el protocolo se comunica con los hubs, emite directivas intermedias, envía señales de polling para conocer el estado del bus, entre otras.

Como se menciono anteriormente, el host envía un token SOF que sirve para sincronizar los dispositivos al bus. En un sistema USB, el host provee la base de tiempo y envía cada 1 ms un SOF (Start of frame, o su traducción, inicio de cuadro) seguido de un numero de 11 bits que sirve para contar cada uno de los marcos. Además, en sistemas de alta velocidad, cada cuadro se divide en ocho microcuadros de 125  $\mu$ s, que también son marcados por un SOF, sin embargo, el contador no se actualiza por cada microcuadro.

Luego de esto, el sistema puede comenzar con la transferencia de datos. USB dispone 4 tipos de posibles transferencias, que se detallan un poco más adelante, y que pueden ser usadas conforme a los diferentes requerimientos del sistema.

Cada transferencia de datos está compuesta por un primer paquete de token, emitido por el host, que posee el tipo de transferencia que se espera, sea de entrada, de salida, de control o especial; la dirección del dispositivo que debe responder o escuchar el mensaje enviado por el bus y un código de detección de errores del tipo CRC (código de chequeo redundante cíclico).

Luego, el siguiente paquete posee los datos transferir, precedido por un PID de datos, y otro código CRC para detectar errores. Este paquete es transmitido por el host o el dispositivo dependiendo del sentido de la transferencia.

Finalmente, el dispositivo devuelve un paquete de reconocimiento, indicándole al Host si la transferencia fue efectiva o no y por qué esta no fue efectiva, siendo ese el caso.

## **Transferencias por paquetes (Bulk transfers)**

Este tipo de transferencias puede ser dispuesta para transmitir un gran flujo de datos. No posee pérdida de datos gracias a un sistema de chequeo y retransmisión de datos. El inconveniente que presenta este tipo de transferencias es que en un nivel de prioridades se presenta en el final del sistema. Es decir, el bus solo va a ser usado para transferir este tipo de datos siempre que se encuentre desocupado, o bien, se le asignará una proporción ínfima de ancho de banda para poder transmitir con este modo. Es comunmente usado para transmitir datos que no son críticos en tiempo, por ejemplo para scanners e impresoras.

## **Transferencias de interrupción**

Este tipo de transferencias sirve para enviar y recibir paquetes de datos que requieren un buen sistema de control de errores, pero que, son más restrictivos en tiempos. El sistema siempre destinará un intervalo fijo de tiempo para transmitir los datos que estén pendientes para transferencias de interrupción.

## **Transferencias Isocrónicas**

Este tipo de transferencias está destinado a datos que son realmente críticos en tiempo. Es usado, principalmente para enviar datos a chorro, como ser el caso de streaming de audio o video, en donde los datos producidos deben ser rápidamente llevados al usuario.

No posee un control de errores muy sofisticado, más que un simple código CRC, pero no existe mecanismo de retransmisión de datos ni handshaking entre los dispositivos y el host.

Como el tiempo es el requerimiento crítico en este tipo de datos, el controlador le asigna una determinada cantidad de tiempo de bus, o en otras palabras, una determinada cuota de ancho de banda.

## **Transferencias de control**

Este tipo de transferencias solo las emite el host y el sistema las utiliza para configurar cada dispositivo. Debido a su criticidad, el controlador dispondrá en cada cuadro de una fracción de ancho de banda para las transmisiones de control. Es el tipo de transferencias que posee el sistema de detección de errores más sofisticado, de forma tal de asegurar la integridad de los datos de control.

A cambio de esto, solo muy poca información puede ser transmitida por cada cuadro, de hasta 64 bytes en sistemas de alta velocidad.



### 1.4.3. Capa lógica

Desde el punto de vista lógico, cada dispositivo es visto por el host como un extremo independiente, que posee un modo de comunicación, es decir, con ese dispositivo el protocolo se comunicara solo por un tipo de transferencia; y un solo sentido. En otras palabras, USB notará como separado un dispositivo de entrada y otro de salida, independientemente de si físicamente el dispositivo es un periférico de entrada y salida.

Esta independencia brinda la posibilidad de configurar cada extremo de forma diferente y obtener el ancho de banda necesario para la subida y bajada de datos, los tiempos de acceso al bus, la dirección y todo lo relacionado a los modos de comunicación conforme a los requerimientos.

El protocolo entiende que entre el host y cada uno de los extremos existe un tubo (la norma en inglés habla de *pipes*) en donde la información es colocada y transferida. Luego, cada tubo posee la configuración establecida por el controlador del host y se comunica con cada extremo por medio de estos tubos. A los fines del usuario, esto es lo importa, por cuanto uno solicita acceso al bus y define en que buffer va a contener los datos a enviar o transmitir y el protocolo se encarga de el empaquetado, el armado de los cuadros, el acceso al bus y el posterior envío de datos.



## Bibliografía

- [1] M. Perez, F. Alcalde, M. S. Haro, I. Sidelnik, J. J. Blostein, M. G. Berisso, and J. Lipovetzky, “Implementation of an ionizing radiation detector based on a FPGA-controlled COTS CMOS image sensor,” in *2017 XVII Workshop on Information Processing and Control (RPIC)*, pp. 1–6, IEEE, sep 2017.
- [2] R. Biswas, *An Embedded Solution for JPEG 2000 Image Compression Based Back-end for Ultrasonography System*. PhD thesis, IIT, Kharagpur, 2018.
- [3] T. Yanagisawa, T. Ikenaga, Y. Sugimoto, K. Kawatsu, M. Yoshikawa, S.-i. Okumura, and T. Ito, “New NEO search technology using small telescopes and FPGA,” in *2018 IEEE Aerospace Conference*, vol. 2018-March, pp. 1–7, IEEE, mar 2018.