

# Capítulo 1

## Introducción

El presente informe busca dar a conocer al lector las tareas y actividades desarrolladas por el autor, 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 del trabajo es diseñar e implementar una interfaz para la transmisión de datos hacia una computadora personal (PC), adquiridos por sistemas desarrollados en arreglos de compuertas de campo programables (FPGA) para aplicaciones científicas, a través del Bus Serial Universal (USB). A lo largo de este documento, se comprenderá la problemática que se resuelve y la configuración, fundamentos y modo de uso del sistema propuesto.

En la sección 1.1 se presentan las motivaciones de este trabajo y se detalla la problemática a resolver. Luego, se detallan los objetivos que persigue este trabajo. Seguido a esto, se otorga un esquema que describe la solución planteada y se justifica el protocolo elegido. Finalmente, se repasan algunos conceptos importantes de la norma USB que luego se utilizan en el trabajo desarrollado.

### 1.1. Motivación

El grado de avance que han experimentado la electrónica y la tecnología en general, gracias a la industria de los semiconductores, permite que la producción científica pueda adquirir una gran cantidad de datos. Para llevar a cabo la producción del conocimiento, es necesario el relevamiento y registro de diferentes tipos de magnitudes físicas y/o químicas sobre el objeto o proceso a investigar. En muchas ocasiones, estas magnitudes resultan difíciles de observar y cuantificar, por lo que es conveniente transformar las variables a conocer en otras más sencillas de medir. Para este propósito, se utilizan transductores.

Se conoce como transductor a cualquier dispositivo que recibe estímulos energéticos de una condición, situación o fenómeno físico y/o químico y los convierte en una señal asociada y definida de otra forma de energía[1][2]. En otras palabras, los transductores son conversores de energías[2][1][3]. Se denomina sensor a una clase particular de transductor que genera, como variable de salida, una señal eléctrica que está especialmente adaptada para ser ingresada en un circuito electrónico, o adecuada al sistema de medida que se utilice [4][5][6].

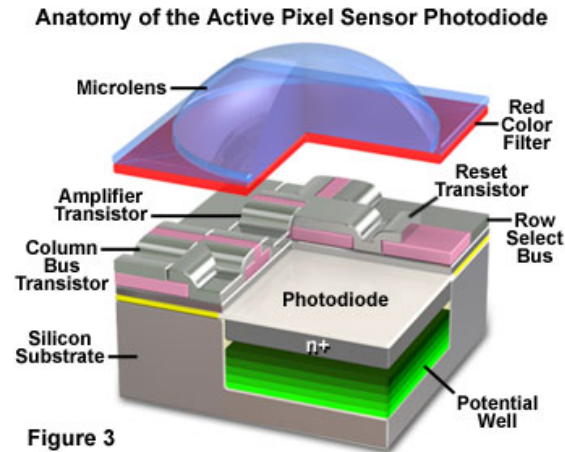


Figura 1.1: Esquema físico de un píxel activo[8]

Las altas escalas de integración de circuitos alcanzadas en la actualidad posibilitan el diseño de sistemas sensoriales cada vez más complejos, en los cuales se logra agrupar miles de sensores en áreas reducidas, obteniendo medidas simultáneas y flujos crecientes de datos. Este trabajo se centrará en la transmisión de datos provenientes de sensores de imagen, uno de los desarrollos que se encuentra en boga.

Una imagen, desde un punto de vista digital, es un arreglo bidimensional de números, los cuales pueden ser exhibidos en una pantalla en forma de intensidad y colores de luz. Cada punto del arreglo que se muestra en pantalla se denomina píxel, acrónimo del inglés *PIcture ELement*, o elemento de imagen. Por esto, un sensor de imagen puede estar compuesto, bien por un arreglo bidimensional de sensores lumínicos (cómo la cámara de un teléfono celular), como por un transductor que es simultáneamente desplazado y medido, (método utilizado, entre otras, para la microscopía de fuerza atómica [7]), o por una combinación de ambos métodos. Por ejemplo, un scanner posee un arreglo lineal de transductores que son desplazados a través de la hoja para generar una imagen digital. En cualquiera de 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 no menor de datos.

Uno de los trabajos que más aportó al desarrollo de sensores de imágenes modernos, fue la introducción de los APS (*Active Pixel Array Sensor*, o sensor matricial de píxeles activos) [9]. Este sensor integra en un proceso CMOS (acrónimo inglés de Metal-Óxido-Semiconductor Complementario, que es el método actualmente más económico para integrar transistores en una única pastilla de silicio), un fotodiodo, un transistor de reset (utilizado para controlar el tiempo de integración, es decir, de exposición a la luz) transistores de selección (utilizados para conectar un píxel determinado dentro del arreglo) y un amplificador seguidor de fuente en cada píxel[8]. El fotodiodo, previamente cargado, transduce la luz en una pequeña descarga eléctrica y el amplificador convierte la carga remanente en tensión para facilitar su lectura. La Figura 1.1 muestra un dibujo de un píxel activo. Se observa el fotodiodo realizado con implantación de iones de boro y fósforo sobre una pastilla de silicio y los diferentes transistores que intervienen en el funcionamiento del APS. Además se incorpora una microlente cuya función es la de enfocar

los fotones sobre el área sensible y un filtro utilizado para identificar los diferentes colores. En el caso de sensores monocromáticos, se omite la colocación del filtro de color durante la fabricación.

A partir del desarrollo de los APS, se fue perfeccionando el método hasta obtener circuitos integrados con mayor cantidad de pixeles y que pueden tener diversas aplicaciones. Por ejemplo, en los trabajos [10] y [11] se presentan sensores CMOS basados en la arquitectura MIMOSA (de *Minimum Ionizing particle MOS Active pixel Sensor*, Sensor con Pixel activo MOS de partículas ionizantes mínimas). Estos sensores se desarrollaron con el objetivo específico de detección de radiación ionizante.

También existen desarrollos de sensores de radiación a través de APS comerciales. Perez *et al.* identificaron eventos producidos por partículas alfa en campos de radiación mixtos mediante el procesamiento de imágenes adquiridas con sensores comerciales CMOS[12] y desarrollaron detectores de neutrones térmicos con sensores similares a los que se les depositó una capa de  $Gd_2O_3$ [13]. Galimberti *et al.* utilizaron un sensor de imágenes comercial para realizar un detector de gas Rn en el ambiente[14]. En otro trabajo, Hizawa, *et al.* fabricaron un sensor que adquiere imágenes midiendo el pH de cada uno de los pixeles[15], obteniendo imágenes de fenómenos químicos en tiempo real.

Como se mencionó antes, una imagen digital es un arreglo de datos. Esto quiere decir que un sensor de imágenes con  $n$  pixeles de largo y  $m$  de ancho, captura  $n \times m$  datos en cada lectura. A su vez, para digitalizar valores, un circuito debe poseer, al menos, un conversor analógico-digital (ADC) de  $x$  cantidad de bits, lo que implica que cada dato estará compuesto por  $x$  dígitos binarios, es decir, un volumen importante de datos por cada lectura. Como ejemplo, un pequeño sensor comercial VGA, en su configuración más básica, posee 640 líneas horizontales y 480 verticales, con una resolución de 8 bits por cada pixel, lo que otorga 2.457.600 bits por cada lectura del sensor.[16] Si además se incorpora la cantidad de imágenes que se toman en función del tiempo (cuadros por segundo o fps), nos otorga un flujo de datos para nada despreciable.

Desde el punto de vista de la electrónica digital, para poder adquirir y transmitir grandes volúmenes de datos, se requiere de circuitos que sean capaces de operar a altas frecuencias de conmutación. El diseño de dichos circuitos no es trivial, ya que cuando las longitudes de onda de las señales presentes son comparables con las dimensiones físicas de dichos circuitos, debe considerarse el uso de líneas de transmisión[17]. Esto implica que no se puede diseñar utilizando un criterio de uniformidad en los parámetros y exige un análisis mas detallado y preciso.

Otro problema que presentan los circuitos electrónicos digitales tiene que ver con los tiempos de propagación de las corrientes y tensiones que circulan a través de ellos. Cuando se aplica un impulso en un conductor, debido a las capacidades propias de los materiales utilizados, las tensiones pueden demorar unos instantes en establecerse. Puede suceder que varias señales lleguen a los puertos de un dispositivo por conductores con distintas longitudes y generen retardos diferentes. Esto puede ocasionar un comportamiento indeseado si no se toman los recaudos adecuados.

Aún suponiendo un perfecto diseño, los circuitos digitales de alta velocidad se encuentran

limitados en la frecuencia de conmutación por la temperatura que se necesita disipar. La potencia consumida por estos dispositivos es proporcional a la frecuencia de funcionamiento[18]. Parte de esta potencia se transforma en calor y produce un aumento en la temperatura. Si el incremento es indiscriminado, puede destruir los circuitos.

Una posible solución para disminuir la frecuencia de las señales sin perjudicar la tasa de transferencia es la incorporación de varios conductores para enviar datos en paralelo. La cantidad de conductores a través de los cuales circula la información, se denomina ancho de bus. Idealmente, para lograr una tasa de transferencia determinada, se podría disminuir la frecuencia tantas veces cómo conductores se agreguen. Por ejemplo, transmitiendo por cuatro filamentos, se podría enviar la misma información a un cuarto de la frecuencia que se necesitaría con uno solo de iguales características.

La incorporación y evolución de microcontroladores permite capturar y procesar volúmenes crecientes de datos. Sin embargo, este tipo de dispositivos posee una estructura rígida, capacidad de procesamiento se encuentra limitada a una instrucción por ciclo de reloj y ancho de bus definido. Para aumentar los volúmenes de datos que circulan a través de ellos, no es posible aumentar el ancho de bus, sino que se torna necesario incrementar la frecuencia de funcionamiento, generando los problemas anteriormente detallados.

Una solución óptima, sin considerar los costos asociados a esto, sería el desarrollo de un circuito integrado de aplicación específica (ASIC del inglés *Application Specific Integrated Circuit*). En este tipo de circuitos, el diseñador elabora un circuito que puede operar a altas velocidades y, a su vez, obtener un ancho de bus sin restricciones, más que las dimensiones físicas del área donde será realizado el circuito. Sin embargo, cuando sí se considera el costo asociado a este enfoque, se vuelve una solución ineficiente en bajas cantidades. La manufactura de este tipo de dispositivos puede tener un costo de miles hasta cientos de miles de dólares, dependiendo del proceso de fabricación utilizado. Gran parte de estos costos son no recurrentes, es decir, solo se pagan una vez por proyecto. En grandes cantidades de dispositivos, este tipo de soluciones se vuelven más convenientes.

Otro enfoque, es la utilización de Arreglos de Compuertas Programables por Campo (FPGA, acrónimo del inglés *Field-Programmable Gate Array*). Un FPGA es un dispositivo electrónico que posee la capacidad de sintetizar casi cualquier circuito digital. En esencia, es una matriz de bloques lógicos (también llamadas *slices* o celdas lógicas, dependiendo del fabricante), que contienen Tablas de Verdad(LUTs o *Look-Up-Table*) y flip-flops (ff), entre otras cosas, y pueden ser interconectadas entre sí, según el criterio del usuario. Así, permite implementar una solución digital en un circuito físico, a diferencia de los microcontroladores, lo realiza a través de un algoritmo almacenado en una memoria, incorporando la ventaja de definir el ancho de bus necesario para relevar una gran cantidad de datos y transmitirlos a frecuencias de trabajo menores, además de ejecutar tareas en paralelo, disminuyendo los tiempos de procesamiento. A su vez, al ser implementado en un área muy pequeña, debido a la integración del sistema, este tipo de sistemas puede trabajar a frecuencias muy elevadas, lo que implica una mayor tasa de datos aún. A pesar de la gran diversidad de precios existentes en el mercado, una FPGA de costos menores a la centena de dólares suele tener muy buenas prestaciones para la mayor parte de las aplicaciones.

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 detector de radiación ionizante utilizando un sensor de imagen CMOS comercial. Para ello, los autores utilizaron una FPGA para configurar diversos parámetros del sensor con el fin de generar estrategias para la identificación de partículas alfa en campos de radiación mixtos y transmitir imágenes a una computadora personal (PC) a través de un puerto UART[19].

Se denomina ultrasonografía a la técnica de adquirir imágenes basándose en reflexiones de ultrasonido. Sus aplicaciones son múltiples, en las que se destaca el diagnóstico médico debido. Un trabajo reciente desarrolló un sistema que mejora la obtención de ecografías médicas con bajo costo utilizando una FPGA[20]. El autor presentó un algoritmo para la supresión de ruido de impulso en tiempo real para imágenes codificadas como JPEG 2000 realizado y probado en Matlab e implementado en una FPGA.

Yanagisawa *et al*, desarrollaron un sistema con telescopios pequeños para explorar objetos de campo cercano con la finalidad de monitorear cuerpos celestes que puedan colisionar con el planeta[21]. En este trabajo, se aprovechó la velocidad de los circuitos implementados en FPGA para minimizar el tiempo de adquisición.

El desarrollo de nuevos sensores brinda a los investigadores un gran volumen de datos. En muchos casos, la obtención de datos por si misma no otorga información, sino que es necesario procesar y analizar 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 operaciones cada 10  $\mu$ s [22], es decir, ejecutaba instrucciones con una frecuencia máxima de 200 kHz. A su vez, tuvo un costo aproximado de U\$S 500.000, pesaba 5 t y consumía 175 kW.

En contraste con aquello, es posible conseguir en el mercado actual, computadoras con tamaño y peso reducido, que ejecutan instrucciones en cuestión de nanosegundos, (5 órdenes de magnitud menos), consumen menos de 1 kW y cuestan algunos cientos de U\$S. A tal punto ha evolucionado esta tecnología, que se cuenta con computadoras muy potentes en casi cualquier laboratorio, oficina u hogar. La capacidad de cálculo que exhiben estos dispositivos, sumada al desarrollo de nuevos métodos y algoritmos de cálculo, permite a los investigadores procesar datos en tiempo reducido, facilitando el análisis y la generación de nueva información.

En todos los casos que se consideran en este trabajo, la generación de datos y el procesamiento de lo mismos se da en sistemas diferentes. Es decir, los datos son relevados por los sensores y adquiridos luego por los FPGAs. Finalmente llegan a una PC para su posterior procesamiento y análisis. Se requiere, por tanto, de una conexión a través de la cual los datos puedan ser transferidos de un sistema, FPGA, al otro, PC y viceversa. Se torna de suma utilidad, entonces, proveer una comunicación efectiva y robusta que permita transmitir grandes volúmenes de datos en poco tiempo, y de esta forma facilitar los tiempos de desarrollo, pruebas, depuración,

procesamiento y análisis.

La implementación de un sistema de comunicación en una FPGA puede ser resuelta de muchas maneras, quedando a criterio del desarrollador utilizar algún protocolo estándar, o bien diseñar uno propio. Sin embargo, en una computadora, las formas de comunicar datos se vuelven un poco más restrictivas y acotadas a los puertos y señales que puede manejar el equipo, conforme el fabricante haya establecido.

Este trabajo busca implementar una comunicación entre una computadora personal y una FPGA, utilizando un protocolo estándar, que esté disponible en cualquier computadora comercial y que posea una tasa de bit suficiente para poder transmitir imágenes.

## 1.2. Protocolos disponibles para la transmisión de datos entre PC y FPGA

Pero, ¿Cuántos datos son suficientes para el propósito de transmitir imágenes? Se toma como base de diseño el sensor que utiliza Pérez en su Tesis de Maestría [23], una cámara para adquirir imágenes monocromáticas de código MT9M001C12STM, comercializado por Actina Imaging [24]. Este sensor posee 1480x1024 pixeles, con profundidad de 10 bits y puede operar hasta a 30 cuadros por segundo. Todo esto, otorga la necesidad de una tasa aproximada de  $393.2 \text{ Mbit s}^{-1}$ .

Esta tasa de bits es más que suficiente. Si la comparamos, por ejemplo con las señales de Televisión Digital. El estándar más exigente de la norma americana de la SCTE (Sociedad de Ingenieros de Comunicación por Cable), posee una tasa de  $38.8 \text{ Mbit s}^{-1}$  [25].

Además, pensando en que la implementación sea compatible con equipos informáticos convencionales, es decir, se encuentren fácilmente en el mercado y no posean especificaciones que escapen al uso de oficina. Esto se cumple hoy en día con tres tipos de puertos: Ethernet, dedicado principalmente a conexión de redes mediante cables; Wi-Fi, utilizado para el accesos a la red de forma inalámbrica; y USB, dirigido a la comunicación de periféricos con la PC.

Al hablar de Ethernet o Wi-Fi, se hace referencia a dos formas diferentes de conectarse a una red de computadoras. En otras palabras, se habla de dos o más nodos, compuestos por PCs o cualquier dispositivo electrónico con capacidad de realizar cálculo binario, que pueden intercambiar datos a través de una trama bastante compleja de componentes diferentes. Ambos protocolos hacen referencia solo a la conexión física de los dispositivos y el control de acceso de cada uno de ellos a la conexión. Quedando a cargo de otros sistemas, con sus protocolos, que los datos enviados puedan ser correctamente recibidos por el usuario de la PC. La gran diferencia entre ellos radica en el medio físico que utilizan: Wi-Fi emplea ondas electromagnéticas emitidas mediante radiofrecuencia, mientras que en Ethernet, estas ondas son acarreadas por uno o más conductores, como ser cable coaxial, cables de par trenzado o fibra óptica.

Ethernet, también conocido como IEEE 802.3, es una norma que define cómo se deben



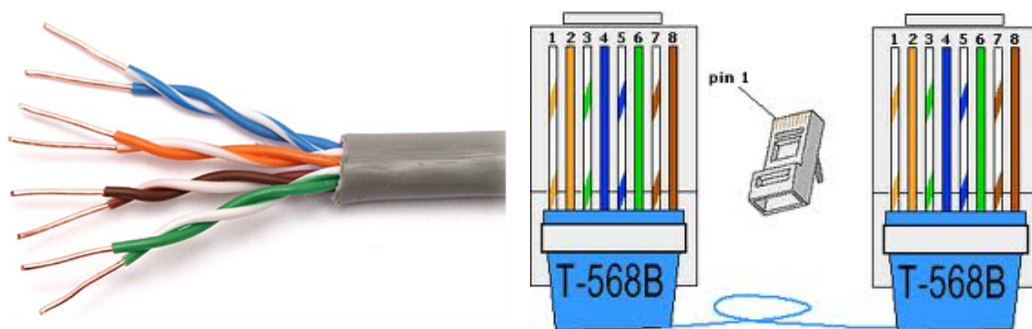
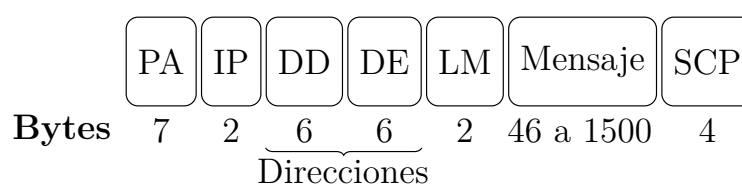


Figura 1.2: Par Trenzado y un dibujo de su ficha de conexión.



### Referencias

**PA:** Preámbulo

**IP:** Inicio de Paquete

**DD:** Dirección de Destino

**DE:** Dirección de Emisión

**LM:** Longitud del Mensaje

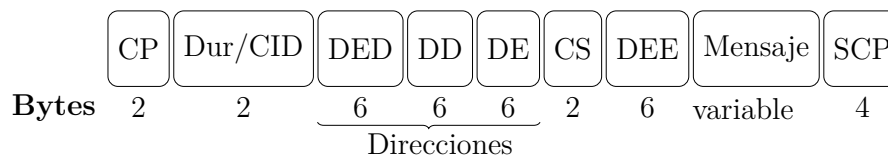
**SCP:** Secuencia de chequeo del paquete

Figura 1.3: Estructura de un paquete Ethernet

conectar nodos a través de conductores para conformar redes de área local (LAN o *Local Area Network*), es decir, redes pequeñas, como ser domésticas, de oficinas o de pequeñas empresas, de forma que puedan transmitir información a velocidades seleccionables entre 1 Mbit/s y 400 Gbit/s [26]. Utiliza una tecnología denominada Acceso Múltiple Sensando la Portadora con Detección de Colisiones (CSMA/CD del inglés *Carrier Sense Multiple Access with Collision Detection*). Esta tecnología se caracteriza por la escucha activa de todos los dispositivos conectados a la red. Solo cuando la dirección del paquete de información que viaja corresponde al dispositivo, este realiza una función. A la hora de transmitir, corrobora que no exista una señal portadora y, si ésta está presente, espera para retransmitir.

Dependiendo de la frecuencia de la portadora y la tasa de transferencia a la que transporta el mensaje, la norma especifica el conector y la distancia máxima a la que debe conectarse una repetidora, es decir, un dispositivo que reciba, reconstruya y emita la señal recibida. Estos conectores pueden ser cable coaxial, fibra óptica o cable de par trenzado. Este último es el más usual en las PCs comerciales y se muestra, junto a su ficha característica en la Figura 1.2.

La información se estructura en paquetes para permitir la comunicación entre muchos nodos de la red. Un paquete, como se observa en la Figura 1.3, se compone de un preámbulo con 7 B que sirve para sincronizar los dispositivos en cada extremo de la conexión, 1 B de inicio,

**Referencias****CP:** Control de Paquete**Dur/CID:** Duración del paquete/Identificación de conexión**DED\*:** Dirección de Enrutador de Destino**DD\*:** Dirección de Destino**DE:** Dirección de Emisión**CS\*:** Control de Secuencia**DEE\*:** Dirección de Enrutador de Emisión**SCP:** Secuencia de chequeo del paquete

\*Pueden no estar dependiendo del tipo de mensaje

Figura 1.4: Estructura de un paquete Wi-Fi

12 B de direcciones, que corresponden 6 al nodo destinatario y 6 al emisor respectivamente, 2 B que indican la longitud del mensaje, entre 46 y 1500 B de datos y 4 B para la verificación de la transmisión. Otra definición importante de la norma, son las características eléctricas de las señales, pero no se detallan en este trabajo porque varían en función de la velocidad del puerto.

Por su parte Wi-Fi, perteneciente a la asociación de compañías denominada Wi-Fi Alliance, se rige por la norma que estableció esta última. Existe una norma equivalente, encuadrada en la especificación IEEE 802.11, referida a las redes de área local inalámbrica, o WLAN (siglas del inglés *Wireless Local Area Network*). Wi-Fi se enfoca en las que se refieren a las comunicaciones de radiofrecuencia con portadora de 2.4 GHz, que se incorporan en las revisiones b, g y n de la norma IEEE. IEEE 802.11 está pensado especialmente para dispositivos portátiles y móviles, que según la norma, los primeros, si bien pueden ser trasladados con facilidad, operan estáticos y los segundos trabajan en movimiento [27]. La principal característica que posee este tipo de comunicación es la falta de conductores para la elaboración de la red, sin contar las conexiones entre los transceptores que emiten y reciben las señales de radiofrecuencias y los nodos, en donde la información es producida y/o consumida. En cuanto al formato del paquete de datos, el cuál se muestra en la Figura 1.4, es bastante similar al de Ethernet. En primer lugar, se envían dos bytes de control que indican el tipo de paquete a enviar. Luego siguen dos bytes que, dependiendo de la etapa de la comunicación puede indicar la duración del mensaje a transmitir o un identificador de una conexión establecida previamente. Siguen entre 6 y 18 bytes de direcciones del enrutador que recibe los datos, el nodo emisor y el destinatario. Continúan, dos bytes de control de secuencia se utilizan para fragmentar transmisiones largas. Continúa un campo más para dirección que corresponde a la red emisora de 6 bytes. Todos los campos de dirección pueden variar en función del tipo de mensaje que se envía. Los últimos dos campos de la trama corresponden a la información que se quiere comunicar (hasta 2312 bytes) y un código de chequeo por redundancia cíclica de 32 bits (4 bytes).

Existen múltiples ventajas de utilizar radiofrecuencias para conectarse a la red, tales como la libertad de mover el punto de trabajo y la economía a la hora de armar redes con muchos



nodos. Sin embargo, posee algunas desventajas notorias, propias del medio de propagación, que lo hacen no tan óptimo para los fines del presente trabajo. Las redes inalámbricas tienen la característica de que no es del todo confiable: posee múltiples fuentes de interferencia, ya que varias tecnologías que utilizan la misma frecuencia (Bluetooth, Zig-Bee, WUSB, microondas). Esto hace que la señal por momentos presente cierta interferencia. A su vez, suele presentar variaciones temporales y asimetrías en las propiedades de propagación, lo que puede provocar interrupciones en la comunicación.

Ambos protocolos proporcionan una solución de conexión de redes de nivel físico y ejecutan tareas de control de acceso al medio (MAC) a fin de evitar colisión en los datos, es decir, que dos dispositivos transmitan en forma simultánea e interfieran la comunicación. Sin embargo, para establecer una red, faltan componentes físicos y lógicos tales como un sistema de control enlace lógico (Logic Link Control), un sistema de direccionamiento, como el Protocolo de Internet (IP), una capa de transporte de datos, (como el protocolo TCP) y las capas de software que permiten acceder a los protocolos anteriormente mencionados.

A pesar de lo anterior, es posible establecer comunicaciones punto a punto con ambos protocolos, simplificando mucho el sistema de transmisión de datos. Sin embargo esta solución presenta un inconveniente no menor: se le quita a la PC un acceso a la red, que en la mayoría de los casos es el único. Esto no es deseable ya que la conectividad es un requisito fundamental en cualquier hogar u organización, ya sea empresarial, gubernamental, científica o de cualquier tipo.

Por su parte el protocolo USB (acrónimo de *Universal Serial Bus*), es una norma desarrollada por seis de las empresas más grandes de la industria informática, pensada y desarrollada para la conexión de teléfonos y periféricos a PCs [28]. En la versión original, USB poseía conectores cableados de 4 conductores, y presenta una topología de bus, es decir todos los dispositivos conectados a una misma conexión física, que es manejada por una PC y en la cual solo transmite y recibe un dispositivo a la vez. Este tipo de conexión tuvo tal penetración en el mercado que se transformó en una norma de facto y actualmente es incorporada casi por defecto y necesaria a la hora de comprar e instalar periféricos.

USB presenta diferentes versiones de su norma, cada cual con una o más tasas de transmisión y señalización. La versión 1 posee dos revisiones, 1.0 fue lanzada al mercado en el año 1996 y 1.1 que se presentó en Agosto de 1998. La primera alcanza una tasa máxima de  $1.5 \text{ Mbit s}^{-1}$  y la segunda hasta  $12 \text{ Mbit s}^{-1}$ . USB 2.0 fue presentado en Septiembre del 2000 y es capaz de transmitir a  $480 \text{ Mbit s}^{-1}$ . La tercera versión, USB 3.0, fue lanzada al mercado en 2011 y transmite  $5 \text{ Gbit s}^{-1}$ . Esta última versión fue revisada en julio de 2013 y en septiembre de 2017, ofreciendo la primera de ella  $10 \text{ Gbit s}^{-1}$  y  $20 \text{ Gbit s}^{-1}$  la segunda. Resulta suficiente para el propósito del siguiente trabajo, la implementación de una comunicación USB 2.0, con tasa de transferencia de  $480 \text{ Mbit s}^{-1}$ . En el Capítulo 2 se profundizarán en conceptos específicos de la norma USB.

El alumno sabe que es posible implementar una comunicación USB completa a través de una FPGA. Sin embargo, esto sería muy costoso en términos de tiempos de desarrollo y de recursos

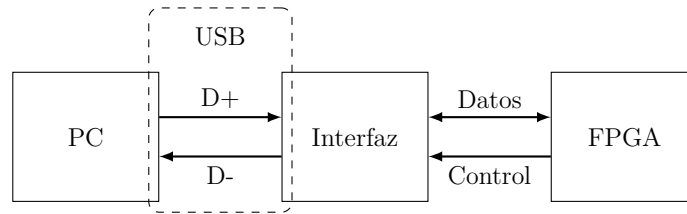


Figura 1.5: Esquema propuesto para implementar la comunicación

de FPGA disponibles para la implementación de otros sistemas, los cuales son el objetivo de la comunicación.

Se plantea, entonces, un esquema como el que se observa en la Figura 1.5 en la cual se utiliza una interfaz externa al FPGA. La comunicación USB propiamente dicha será efectuada entre la interfaz y la PC, mientras que se plantea una comunicación diferente entre la interfaz y el FPGA. Este último, por su parte, tendrá la tarea de realizar el control de esta comunicación.

## 1.3. Objetivos

### 1.3.1. 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.

### 1.3.2. 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.

## 1.4. 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, la propuesta que da solución a la motivación, 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. **??:** Se describe aquí todas las herramientas de las que se vale este trabajo para cumplir con los objetivos propuestos.
3. **??:** Se presenta la arquitectura, configuración y código desarrollado para el presente trabajo, como así también las herramientas específicas provistas por el fabricante, que facilitan el desarrollo.
4. **??:** Este capítulo detalla lo desarrollado para implementar la comunicación entre la FPGA y la interfaz. Se expone una máquina de estados descrita en VHDL y sintetizada en FPGA. También se describe un circuito impreso realizado para conectar ambas partes.
5. **??:** Se desarrollan las tareas desarrolladas a fin de realizar las depuraciones del sistema y la verificación del cumplimiento de las especificaciones.

## 1.5. Sumario del capítulo

En el presente capítulo se expuso la necesidad de la elaboración de un sistema de comunicación que permita la transferencia de datos entre una PC y un FPGA para ser utilizados por sistemas implementados con este último dispositivo. Se planteó una solución utilizando una interfaz comercial que sirve de intermediario entre estas herramientas y se brindó una justificación del empleo del protocolo USB 2.0 de alta velocidad como la implementación óptima del sistema. Se presentó también la estructura del presente informe y se dieron algunos detalles relevantes para este trabajo de la norma USB.



## Capítulo 2

# Bus Serial Universal 2.0

El Bus Serial Universal, o USB por sus siglas en inglés, es un sistema de comunicación diseñado durante los años 90 por seis fabricantes vinculados a la industria informática, Compaq, Intel, Microsoft, Hewlett-Packard, Lucent, NEC y Philips, con la idea de proveer a su negocio de un sistema que permita la conexión de PCs con teléfonos y periféricos con un formato estándar, fácil de usar y que permita la compatibilidad entre los distintos fabricantes.

Hasta ese momento, el gran ecosistema de periféricos, sumado a los nuevos avances y desarrollos, hacía muy compleja la interoperatividad 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, cuando se comparaba una PC, se debía cambiar el teclado, el mouse y/o algún periférico específico. Esto también complicaba a las mismas empresas productoras, por que la introducción de un nuevo sistema requería de mucho soporte extra para poder conectar todo lo ya existente.

Todo esto, quedó saldado con el aparición de la norma USB, que debido a la gran cuota de mercado de sus desarrolladores, fue adoptado en forma rápida y se transformó en la especificación por defecto 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, se solicite especialmente un puerto determinado. Así, cualquier PC nueva disponible en el mercado debe poseer puertos USB para la conexión de los periféricos.

El presente capítulo intenta ser un breve resumen con los aspectos más relevantes de la norma en cuanto a su composición física, su topología, los dispositivos que intervienen, la importancia de los mismos y como los datos son transmitidos desde y hacia una PC.

### 2.1. Objetivos y metas de la norma USB 2.0

El diseño de la norma USB busca resolver tres problemáticas interrelacionadas, que son: La conexión de teléfonos con las PC, la facilidad de uso, es decir, que el usuario solo conecte su dispositivo y pueda utilizarlo, y la expansión de puertos disponibles para conectar periféricos.

[28]

Para satisfacer estas tres demandas, la norma USB 2.0 busca alcanzar un conjunto de metas:

- Expandir los puertos de PC destinados a periféricos y que posean facilidad de uso.
- Brindar una solución de bajo costo que permita tasas de transferencias de hasta 480 Mbps.
- Ser totalmente compatible con datos de voz, audio y video en tiempo real, es decir, que pueda transmitir una conversación o video-llamada sin intermitencias.
- Poseer un protocolo flexible en el que convivan transferencias isocrónicas y mensajes asincrónicos.
- Integrar la norma en las tecnologías de dispositivos básicos.
- Comprender diferentes configuraciones de Pc y factores de forma.
- Proveer una interfaz estandar capaz de difundir rápidamente en los productos existentes en el mercado.
- Habilitar nuevas clases de dispositivos que aumenten las posibilidades de las PCs.
- Compatibilizar completamente con los dispositivos fabricados con versiones anteriores de la misma especificación.

## 2.2. Descripción de un sistema USB

USB posee un esquema de bus maestro-esclavo, en forma de árbol cuyo nodo principal es el host. Es decir, la comunicación se realiza siempre a través de una sola línea de comunicación a la que se conectan todos los dispositivos que se necesite (dada el campo de direcciones provisto por la norma, 128 dispositivos como máximo). De esta manera, solo puede transmitir un dispositivo a la vez.

El acceso al bus, es decir, el acceso a la línea única compartida de comunicación, es administrado por un maestro. El maestro se encarga de solicitar a cada uno de los dispositivos su intervención. Posteriormente, el dispositivo debe responder al pedido del maestro. Este esquema es lo que se conoce como maestro-esclavo.

En un sistema USB no cualquier dispositivo puede ser maestro. Este rol lo cumple solo uno: una PC, o cualquier dispositivo con capacidad de llevar a cabo las tareas asignadas (que se detallan más adelante); denominado Host por la norma. La palabra *HOST* proviene del habla inglesa y se traduce como anfitrión, aunque en la jerga se conoce comunmente por su nombre en inglés.

La topología del bus, cómo se observa en la Figura 2.1, posee forma de árbol, es decir, puede ser pensada como una comunicación vertical, donde en el punto más alto se encuentra el Host. Siguiendo hacia abajo, el bus puede encontrar dos tipos diferentes de dispositivos: Funciones,



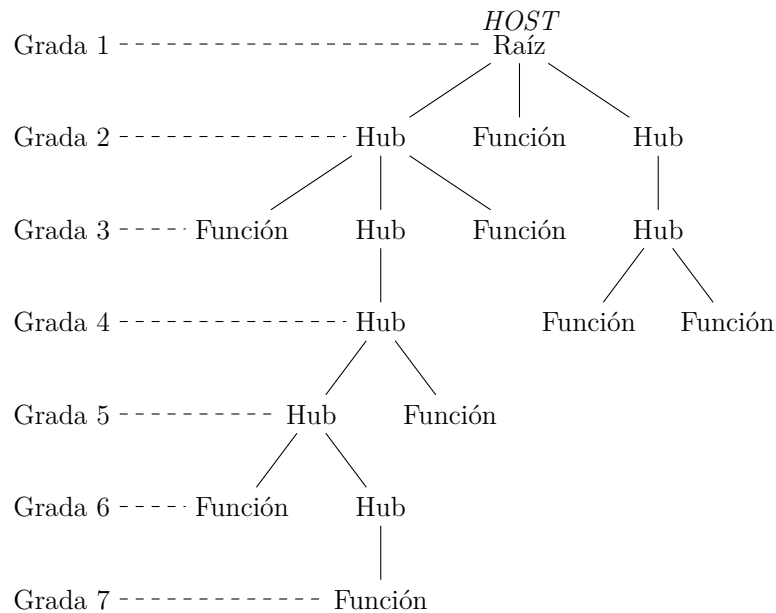


Figura 2.1: Topología de un sistema USB

cuyo rol es el de proveer una utilidad al sistema, como ser la de captura de imagen, reproducción de audio o el ingreso de comandos; y Hubs (concentradores o distribuidores), que se encargan de conectar una o más funciones al sistema. La norma USB establece gradas, en donde cada Hub introduce una nueva grada que contiene a las Funciones conectadas. Por cuestiones de restricciones temporales y tiempos de propagación en los cables, no se permiten más de 7 gradas, incluyendo al Host en la primera. Es decir, no se puede conectar más de 5 Hubs en cascada. La grada 7 sólo puede contener Funciones[28].

Cada uno de estos dispositivos diferentes, se interconectan entre sí a través de cables y conductores específicos, diseñados en forma tal que no sea posible conectarlos en forma equivocada. Para cumplir con la norma, el Host debe tener siempre un zócalo compatible con conectores tipo A y los periféricos para enchufes de tipo B. Se observan las diferencias entre uno y otro en la Figura 2.2

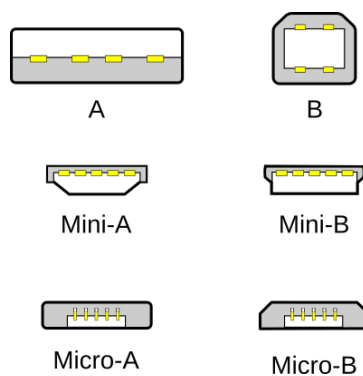


Figura 2.2: Tipos de conectores USB. Los tipo A deben ser usados en el extremo del Host y los tipo B hacia los periféricos[29]

## 2.3. Dispositivos que componen un sistema USB

Dentro de un sistema USB existen tres tipos diferentes de dispositivos: el host, los hubs y las funciones. Cada uno de ellos tiene asignado un rol específico dentro de la comunicación. Se detallan a continuación las tareas pertinentes a cada uno de ellos.

### 2.3.1. Host USB

El host es quien comanda las comunicaciones. Este dispositivo debe tener la capacidad de memoria y de procesamiento necesario para almacenar y ejecutar el software de control. A su vez, debe tener un hardware necesario que le permita hacer un monitoreo y control de los eventos que suceden en el bus.

Entre las tareas que debe llevar a cabo, se encuentran:

- Detectar la conexión y desconexión de dispositivos.
- Administrar el flujo de los comandos de control con los diferentes dispositivos.
- Administrar el flujo de la información entre él (el Host) y los diferentes dispositivos.
- Registrar la estadística de la actividad y el estado del bus.
- Proveer potencia a los dispositivos conectados, cuando estos así lo requieran.

Debido a que las funciones del host necesitan una cantidad mínima de recursos de almacenamiento y procesamiento, lo más normal es que el dispositivo que cumpla esta función sea una PC, o posea algún sistema de cómputo incorporado.

El host es quien inicia, siempre, la comunicación con las funciones. Las funciones, a su vez, responden a lo que fue solicitado por el host, cuando él lo indique.

### 2.3.2. Hubs USB

Un hub USB tiene la función de proveer puertos al bus. El primer hub está incorporado en el host y camino abajo, cada vez que se requieran más puertos a los cuales incorporar periféricos, se pueden ir agregando más hubs.

Una función importante llevada a cabo por un hub, para la norma USB, es la de servir como interfaz entre dispositivos con diferentes velocidades, optimizando así el ancho de banda disponible para la comunicación.

### 2.3.3. Funciones USB

Se nombra como función a todos los periféricos que se conectan al bus. Estos periféricos son los que brindan una funcionalidad extra al host. Por ejemplo, un teclado le brinda un método de entrada a la PC, Un mouse permite manejar un puntero de la interfaz gráfica, un parlante otorga la posibilidad de emitir sonidos y un micrófono para ingresarlos. Cada una de estas utilidades, componen una función USB.

Cada función debe informar, al momento de su conexión al bus, una descripción de sus características y sus requerimientos al host. De esta forma, el host conoce qué tipo de dispositivo es, cómo puede ser configurado y cuál es la forma de comunicarse. La descripción se lleva a cabo a través de una secuencia de datos, denominados descriptores.

Desde el punto de vista lógico, cada periférico posee canales únicos de comunicación con el host, llamados tuberías (*pipes* en el idioma inglés de la norma). Existen dos tipos de tuberías, las de control, por donde circulan mensajes propios del protocolo y sirven para la administración, configuración y gestión de las comunicaciones ; y las tuberías de “chorro” (*stream*) a través de las cuales circulan los mensajes con la información que se desea transmitir de un dispositivo a otro. El final de la tubería se llama extremo (*endpoint*) en el periférico y conectan cada extremo a un buffer en el host. Los periféricos poseen uno o más extremos. Cada extremo de un periférico, posee un tipo de transferencia asociado con una dirección de la información determinada. Esto quiere decir que un dispositivo de entrada y salida, debe poseer al menos dos extremos lógicos diferentes, uno para enviar datos al host y otro para recibirlos. Los tipos de transferencia, a su vez, determinan el ancho de bus asignado por el protocolo, la latencia, la tolerancia a errores en los datos enviados y el tamaño de los paquetes a enviar.

## 2.4. Flujo de la comunicación

## 2.5. Tipos de Transferencias

## 2.6. Conexión física de USB

Una de las características más asociadas y conocidas por los usuarios de USB está relacionada a sus puertos, conectores y cables. Es decir, El usuario reconoce una comunicación USB fácilmente por su conexión mecánica.

La norma USB reconoce y admite solo una conexión hacia la computadora y ha sido un poco más flexible con el diseño de los conectores que se conectan a los periféricos debido a la necesidad de la miniaturización de los productos, específicamente, teléfonos celulares y dispositivos inteligentes.

Entre lo común a periféricos y PC, USB utiliza para su conexión un cable compuesto por dos pares de conductores, de los cuales uno sirve para transportar energía hacia los dispositivos

y otro para datos.

## **2.7. Conexiones eléctricas**

# Bibliografía

- [1] R. Pallàs-Areny and J. G. Webster, *Sensors and signal conditioning*. Wiley-Interscience, 2001.
- [2] D. M. Considine, *Encyclopedia of instrumentation and control*. McGraw-Hill, Inc., 1971.
- [3] A. Perez Garcia, “Curso de instrumentación,” p. 261, 2008.
- [4] J. Fraden, *Handbook of modern sensors: physics, designs, and applications*. New York, NY: Springer New York, 2010.
- [5] E. Slawiński and V. Mut, *Humanos y máquinas inteligentes: conocimiento educativo sobre el comportamiento interno de robots que actúan junto y para el hombre*. Saarbrücken, Alemania: Editorial Académica Española, 2011.
- [6] K. Ogata, *Modern control engineering*. Aeeizh, 2002.
- [7] G. Binnig and H. Rohrer, “Scanning tunneling microscopy,” *Surface Science*, vol. 126, pp. 236–244, mar 1983.
- [8] R. Turchetta, K. R. Spring, and M. W. Davidson, “Digital Imaging in Optical Microscopy - Introduction to CMOS Image Sensors,” (accessed in July 2019).
- [9] S. Mendis, S. Kemeny, and E. Fossum, “CMOS active pixel image sensor,” *IEEE Transactions on Electron Devices*, vol. 41, pp. 452–453, mar 1994.
- [10] C. Hu-Guo, J. Baudot, G. Bertolone, A. Besson, A. S. Brogna, C. Colledani, G. Claus, R. D. Masi, Y. Degerli, A. Dorokhov, G. Doziere, W. Dulinski, X. Fang, M. Gelin, M. Goffe, F. Guilloux, A. Himmi, K. Jaaskelainen, M. Koziel, F. Morel, F. Orsini, M. Specht, Q. Sun, I. Valin, and M. Winter, “CMOS pixel sensor development: a fast read-out architecture with integrated zero suppression,” *Journal of Instrumentation*, vol. 4, pp. P04012–P04012, apr 2009.
- [11] J. Baudot, G. Bertolone, A. Brogna, G. Claus, C. Colledani, Y. Değerli, R. De Masi, A. Dorokhov, G. Doziere, W. Dulinski, M. Gelin, M. Goffe, A. Himmi, F. Guilloux, C. Hu-Guo, K. Jaaskelainen, M. Koziel, F. Morel, F. Orsini, M. Specht, I. Valin, G. Voutsinas, and M. Winter, “First test results of MIMOSA-26, a fast CMOS sensor with integrated zero suppression and digitized output,” *IEEE Nuclear Science Symposium Conference Record*, pp. 1169–1173, 2009.

- [12] M. Pérez, J. Lipovetzky, M. Sofo Haro, I. Sidelnik, J. J. Blostein, F. Alcalde Bessia, and M. G. Berisso, "Particle detection and classification using commercial off the shelf CMOS image sensors," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 827, pp. 171–180, aug 2016.
- [13] M. Pérez, J. J. Blostein, F. A. Bessia, A. Tartaglione, I. Sidelnik, M. S. Haro, S. Suárez, M. L. Gimenez, M. G. Berisso, and J. Lipovetzky, "Thermal neutron detector based on COTS CMOS imagers and a conversion layer containing Gadolinium," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 893, pp. 157–163, jun 2018.
- [14] C. L. Galimberti, F. Alcalde Bessia, M. Perez, M. G. Berisso, M. Sofo Haro, I. Sidelnik, J. Blostein, H. Asorey, and J. Lipovetzky, "A Low Cost Environmental Ionizing Radiation Detector Based on COTS CMOS Image Sensors," in *2018 IEEE Biennial Congress of Argentina (ARGENCON)*, pp. 1–6, IEEE, jun 2018.
- [15] T. Hizawa, J. Matsuo, T. Ishida, H. Takao, H. Abe, K. Sawada, and M. Ishida, "32 × 32 pH image sensors for real time observation of biochemical phenomena," *TRANSDUCERS and EUROSENSORS '07 - 4th International Conference on Solid-State Sensors, Actuators and Microsystems*, pp. 1311–1312, 2007.
- [16] ON Semiconductor, "NOIP1SN0300A Global Shutter CMOS Image Sensors," 2014.
- [17] N. Ida, *Engineering Electromagnetics*. Cham: Springer International Publishing, 3th ed., 2015.
- [18] J. F. Wakerly, *Digital Design: principles and practices*, vol. 1. Pearson, 1999.
- [19] 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.
- [20] R. Biswas, *An Embedded Solution for JPEG 2000 Image Compression Based Back-end for Ultrasonography System*. PhD thesis, IIT, Kharagpur, 2018.
- [21] 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.
- [22] H. H. Goldstine and A. Goldstine, "The Electronic Numerical Integrator and Computer (ENIAC)," *Mathematical Tables and Other Aids to Computation*, vol. 2, p. 97, jul 1946.
- [23] M. Perez, *DETECCIÓN DE RADIACIÓN IONIZANTE UTILIZANDO SENSORES DE IMAGEN CMOS COMERCIALES*. PhD thesis, 2018.
- [24] I. Micron Technology, "1 / 2-Inch Megapixel CMOS Digital Image Sensor MT9M001C12STM (Monochrome)," pp. 1–35, 2004.



- [25] S. of Cable Telecommuniocations Engineers, *American National Standard ANSI/SCTE 07 2006. Digital Tansmission Standard for Cable Television*. Society of Cable Telecommunica-tions Engineers, Inc., 2006.
- [26] IEEE Computer Society, *IEEE Standard for Ethernet*, vol. 2018. 2018.
- [27] IEEE Computer Society, *Part 11 : Wireless LAN Medium Access Control ( MAC ) and Physical Layer ( PHY ) Specifications IEEE Computer Society Specific requirements Part 11 : Wireless LAN Medium Access Control ( MAC ) and Physical Layer ( PHY ) Specifications*, vol. 2012. 2016.
- [28] Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, and Philips, *Universal Serial Bus Specification*, vol. Revision 2.0. 2000.
- [29] “Usb hardware.” [https://en.wikipedia.org/wiki/USB\\_hardware](https://en.wikipedia.org/wiki/USB_hardware). Ingreso: 8 de agosto del 2019.