

**Instituto Tecnológico de Costa Rica  
Escuela de Computación  
Centro de Investigaciones en Computación**

**Proyecto:  
Implantación del protocolo Internet Cero: I0-CR**

**Informe Final**

**Investigadores:  
Milton Villegas Lemus (coordinador)  
Yuen Law Wan**

**Cartago,  
Mayo 2008**

## Tabla de Contenidos

Resumen.....	4
Abstract.....	5
1 Introducción.....	6
1.1 Antecedentes .....	6
1.2 Problema.....	7
1.3 Objetivos .....	8
2 Revisión de Literatura.....	8
3 Materiales y Métodos.....	15
3.1 Materiales y equipo utilizado. ....	16
4 Resultados y discusión.....	17
4.1 Submódulos de desarrollo .....	18
4.2 Codificación Manchester .....	18
4.2.1 Algoritmo.....	19
4.2.2 Decodificación.....	20
4.2.3 Resultados .....	21
4.3 Comunicación Auto- Sincrónica .....	22
4.3.1 Monitor .....	22
4.3.2 Modelo de comunicación.....	23
4.3.3 PIC.....	23
4.3.4 Limitaciones de modelo.....	24
4.3.5 Nuevo modelo.....	25
4.3.6 Comunicación con Pocket PC.....	26
4.3.7 Resultados .....	26
5 Conclusiones.....	27
6 Recomendaciones .....	29
7 Referencias.....	31

## Índice de Figuras

<b>Figura 1.</b> Mesa de información interactiva usada en el Museo de Arte de Nueva York. .	6
<b>Figura 2.</b> Internet 0 es un protocolo que logra romper barreras de compatibilidad.....	7
<b>Figura 3.</b> Representación de un byte en Manchester .....	12
<b>Figura 4:</b> Segmento de código en Java para la codificación Manchester. ....	19
<b>Figura 2:</b> Segmento de código en Java para la decodificación Manchester. ....	20
<b>Figura 6.</b> Ventana principal del programa monitor .....	23
<b>Figura 7.</b> Diagrama de conexión del MAX232 para la comunicación serial entre computadora y PIC .....	24
<b>Figura 8.</b> Modelo de conexión para pruebas de transmisión con Internet 0. ....	25

## Resumen

Internet 0 es un nuevo protocolo de comunicación que define un conjunto de reglas a seguir para lograr pasar información entre dispositivos que son esencialmente distintos. El objetivo de los creadores de este protocolo es poder llevar Internet a los aparatos menos convencionales, desde bombillos y lámparas hasta pinturas y esculturas. Extender Internet a este nivel tiene potenciales aplicaciones en distintas áreas: arte, salud y arquitectura entre muchos otros.

Muchas implementaciones están empezando a surgir en países desarrollados. Estas implementaciones no toman en cuenta las condiciones de los países en desarrollo, como las variables de costo y acceso a las tecnologías. Esta es la motivación que llevó a plantear el proyecto de implantación de Internet 0 en Costa Rica, con el que se pretende evaluar el desempeño y la factibilidad del protocolo.

En este documento se describen las experiencias y resultados encontrados durante el desarrollo del proyecto y se presentan algunos de los detalles técnicos de aspectos más importantes del protocolo.

## **Abstract**

Internet 0 is a new communication protocol that defines a set of rules to pass information between devices that are essentially different. The goal of the creators of this protocol is to take Internet all the way down to unconventional devices, from lightbulbs and lamps to sculptures and paintings. Extending Internet to this level has potential applications in areas such as art, health and architecture, just to mention a few.

Implementations are starting to appear in developed countries. These implementations do not take into account the conditions of the developing countries, such as cost and access to technology. This is the motivation that took us to propose project of implantation of Internet 0 in Costa Rica, with which we pretend to evaluate the protocol's performance and factibility.

This document describes the experience and results achieved during the development of the project and presents some of the more technical details of relevant issues of the protocol.

# 1 Introducción

## 1.1 Antecedentes

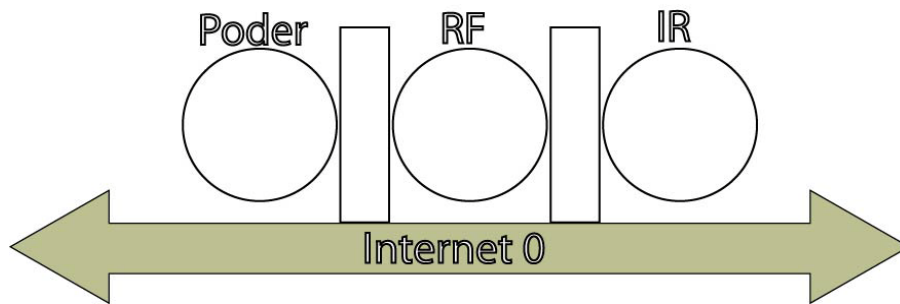
Internet 0 es un protocolo de comunicación entre dispositivos propuesto por el profesor Neil Gershenfeld del Laboratorio de Medios del Instituto Tecnológico de Massachusetts. El protocolo define una serie de reglas para lograr una comunicación sencilla, confiable e independiente del medio de transmisión. Se presentó por primera vez en el Laboratorio de Medios del Instituto Tecnológico de Massachusetts en una conferencia titulada “Internet 0: Past, Present, and Future” (Internet 0: Pasado, Presente y Futuro).



**Figura 1.** Mesa de información interactiva usada en el Museo de Arte de Nueva York.  
Fuente: Media Lab, MIT

La comunicación entre dispositivos no es un tema nuevo. El concepto se ha venido desarrollando desde hace ya varios años, en el Museo de Arte Moderno en Nueva York se utilizó, durante una exposición, muebles para proporcionarles información a los visitantes sobre el evento (ver Figura 1). En arquitectura y construcción, se habla de electrodomésticos inteligentes que le avisan al dueño cuando hay que comprar más leche, por ejemplo. Lo que es innovador acerca de Internet 0 es que trata de integrar todo los esfuerzos existentes en uno solo para tratar de evitar el problema de incompatibilidad que sufrió Internet en sus inicios. Como se aprecia en la figura 2, actualmente existen diversos protocolos de comunicación entre dispositivos, radio

frecuencia, infra-rojo e incluso la línea de alimentación, todos diferentes e incompatibles entre sí. Con Internet 0, se espera romper esas barreras para lograr la compatibilidad deseada.



**Figura 2.** Internet 0 es un protocolo que logra romper barreras de compatibilidad

El proyecto se encuentra aún en desarrollo y los prototipos que se han presentado son productos de países desarrollados como Estados Unidos, España y Noruega. Estos prototipos plantean soluciones a los problemas en condiciones propias de esos países. Lo que se desea, con este proyecto, es evaluar la aplicabilidad de Internet 0 en países en desarrollo, en donde los costos altos y el acceso limitado a ciertas tecnologías se vuelven un factor importante a la hora de implementar iniciativas de este tipo.

## 1.2 Problema

Internet 0 se encuentra en etapa de diseño y define un conjunto de reglas a seguir para lograr comunicar dispositivos que son esencialmente distintos. Pero estas reglas deben ser implementadas en dispositivos reales de manera que se pueda comprobar su efectividad y factibilidad, sobre todo en países en desarrollo, en los cuales los recursos son limitados. Las principales variables a evaluar y observar son: distancia, potencia, ancho de banda y costo.

### 1.3 *Objetivos*

a. Objetivo General:

- Implantar Internet 0 en dispositivos electrónicos inalámbricos para evaluar su desempeño y factibilidad.

b. Objetivos Específicos:

- Comunicar dispositivos con interfaces inalámbricas utilizando Internet 0.
- Lograr comunicación entre dispositivos desarrollados en Costa Rica.
- Comunicar dispositivos desarrollados en Costa Rica con dispositivos desarrollados en Noruega.
- Analizar la difusión y conectividad en diferentes medios de comunicación para diferentes elementos colocados a corta distancia (menos de 1 metro)
- Evaluar la compatibilidad y la conectividad de los dispositivos Internet 0 con el Internet convencional.
- Evaluar las variables de distancia, potencia, ancho de banda y costo.

## 2 **Revisión de Literatura**

A continuación se presentará de manera resumida la propuesta de Internet Cero (I0) realizada por el Profesor Neil Gershenfeld del Laboratorio de Medios del Instituto Tecnológico de Massachusetts (MIT). El documento fue publicado en la revista *Scientific American*.

Debido al rápido crecimiento de la Internet, se le puede encontrar en casi cualquier lugar y dispositivo, por ejemplo: teléfonos, electrodomésticos y cafeterías. Existen razones de peso por las que es deseable conectar dispositivos atípicos a la Internet (por ejemplo un bombillo) que van desde el uso eficiente de energía hasta la manifestación artística y arquitectónica. Pero las implementaciones existentes, las cuales fueron diseñadas para “mainframes” (computadores con gran capacidad de procesamiento), no son apropiadas para conectar todo en todo lugar. El proyecto de Internet Cero (Internet 0) pretende extender el concepto de “internetworking” (conexión entre redes con



protocolos de comunicación y arquitecturas diferentes) a “interdevice internetworking”. Este último concepto se refiere a interconectar dispositivos que son esencialmente distintos, de manera que la comunicación se de independientemente del medio de transmisión.

El objetivo no es reinventar el concepto de Internet ni ningún otro tipo de red, sino, el trabajar en dispositivos que le sirvan a las computadoras actuales como interfaces de comunicación con el medio y por ende, crear formas simples de comunicación a través de dichos dispositivos. Por ejemplo, una presentación sobre futuras tecnologías mostraba como un estante de un baño que se comunica con una computadora puede ayudarle a un adulto mayor a administrar sus medicamentos. En el Museo de Arte Moderno en Nueva York se utilizó, durante una exposición, muebles para proporcionarles información a los visitantes sobre el evento.

Algo que se asume normalmente cuando se trata el tema de redes e Internet es que más rápido es mejor, pero las redes rápidas tienen un costo escondido: la complejidad incrementa también con la velocidad. Muchos proyectos de construcción de “Hogares inteligentes” han tenido dificultades en lograr la conectividad deseada entre los diferentes elementos. Ya que a diferencia de la red eléctrica o a la cañería, las redes de infraestructura requieren de planeación, instalación y operación mucho más especializada. El reto tecnológico aquí no es lograr altas velocidades ni grandes capacidades de almacenamiento, sino configuraciones más simples.

Debido al gran potencial de este mercado, existe una gran cantidad de alternativas de comunicación, cada una con sus propios esquemas: IRDA, USB, Bluetooth, I2C y otros. Esta situación es análoga a la vivida en los inicios de la Internet, cuando los estándares existentes competían unos con otros llevando el desarrollo de la Internet hacia un futuro de incompatibilidad. Y al igual que IP (Internet Protocol) fue la solución para lograr interconectar redes distintas a través de Internet, Internet 0 podrá ser la solución para interconectar dispositivos con esquemas distintos de comunicación.

Los siguientes son los 7 principios que guían y definen la arquitectura de Internet 0:

**IP en los nodos hoja.** En una red de Internet 0 el protocolo IP se lleva hasta los nodos hoja, lo que quiere decir que toda la red “habla” el mismo idioma. Históricamente se ha evitado llevar el protocolo IP hasta los nodos hoja (terminales) por miedo a la excesiva cantidad de recursos que este le puede demandar a los dispositivos y a la red en sí. Esto ha causado incompatibilidad en los bordes de las redes. Lo cierto es que en la actualidad la capacidad de los dispositivos ha crecido lo suficiente para soportar la demanda de recursos que requiere el protocolo.

**Estándares compilados:** Actualmente se utiliza el modelo OSI, el cual define 7 capas, cada una con una función específica. Pero en realidad, todo el conjunto realiza menos trabajo de lo que parecería si se suman todas las partes. Es posible simplificar las capas implementándolas juntas y no separadas. Esto remueve el tiempo requerido para el envío de mensajes entre las diferentes capas.

**Eliminación de servidores.** En el esquema cliente-servidor, dispositivos pequeños recuperan información para una máquina más grande. Pero este modelo centralizado posee una grave falla: sin el servidor central, los clientes son inútiles. Los dispositivos actuales, incluso los más simples son capaces de mantener, administrar y comunicar su propio estado. En una red Internet 0, cada dispositivo es responsable de mantener una lista de los dispositivos que debe controlar, o una lista de los dispositivos que lo controlan. Un servidor podría agregar valor a la red con datos y procesamientos más complejos, pero este no sería indispensable para el buen funcionamiento de la red.

**Identidad Física.** En las redes convencionales, cada computadora posee dos direcciones: una lógica y una física. La dirección lógica permite que los enrutadores puedan enviar los paquetes al destino correcto dentro de la red, mientras que la dirección física identifica a cada computadora. Cada dispositivo en una red Internet 0 también posee una dirección física y una dirección lógica. La dirección lógica le permite comunicación con redes convencionales. La dirección física le permite establecer comunicación con otros dispositivos de la red Internet 0: dos dispositivos pueden establecer una conexión lógica si se transmiten primero sus direcciones físicas. Esta comunicación puede además darse por medio de un tercer dispositivo que sirva como

intermediario. Por ejemplo, se puede utilizar un agenda personal (PDA) como un dispositivo móvil para establecer comunicación entre un interruptor y un bombillo que se encuentran en habitaciones diferentes.

**Estándares abiertos.** Para evitar la aparición de sistemas propietarios redundantes que entren en conflicto unos con otros y que amenacen el crecimiento de la Internet, se propone que de existir sistemas propietarios estos deben crearse sobre las bases de los sistemas y estándares abiertos y no en contraposición a los mismos.

**Bits grandes.** Un bit representa una unidad de información (1 o 0) y se representa por algún tipo de excitación (electrones en un cable, fotones en una fibra, ondas electromagnéticas en el aire). A la velocidad actual de las redes Ethernet, que corresponde a un gigabit por segundo, cada bit transmitido tiene una longitud aproximada de un pie. Si la red es más grande que el bit, cabe la posibilidad de que se generen bits espurios si dos terminales inician una transmisión simultánea, por ejemplo. Por eso es que las redes de alta velocidad requieren de cables especiales y otro equipo especializado. Si por otro lado, un bit es más grande que la red este llenará la red independientemente de la configuración. En un edificio de 100 m, la velocidad de transmisión es de aproximadamente 1 millón de bits por segundo, suficiente velocidad para transmisión entre bombillos. Los bits enviados a esta tasa de transmisión tienen tiempo de abarcar toda la red, lo cual simplifica la forma en la que se codifican.

**Modulación de extremo a extremo.** Para que dos dispositivos puedan comunicarse deben estar de acuerdo en la forma en que van a representar la información. Si la velocidad de transmisión no es un factor determinante, como en el caso de Internet 0, entonces los acuerdos que se establecen entre un dispositivo y otro pueden ser simplificados.

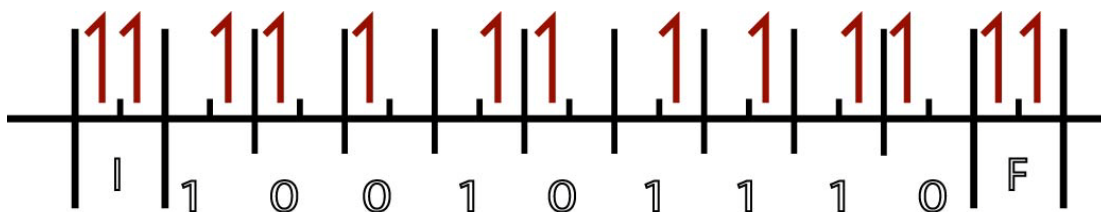
El destino final de Internet 0 no son solamente los bombillos. Una red de Internet 0 no se puede diferenciar de las computadoras a las que se conecta. En realidad, una red de Internet 0 puede verse como un sistema computacional: permite comunicación, procesamiento, almacenamiento, captura y despliegue de información utilizando las mismas formas de representación de datos. La diferencia es que la información se intercambia entre diferentes dispositivos que pueden ser acomodados y ensamblados

dinámicamente para ajustarse a las necesidades de un problema específico, y no están limitados físicamente por una caja.

Para lograr comunicación entre dispositivos utilizando Internet 0, se deben desarrollar los componentes que lo conforman. Tales componentes incluyen la Codificación Manchester para la transmisión asincrónica. Esto es, los módulos que permitan la codificación y decodificación de los datos transmitidos y recibidos. Otro componente que se requiere es el direccionamiento de los datos. Cada dispositivo dentro de la red debe poseer una dirección única para permitir comunicación entre dispositivos específicos.

**Codificación Manchester.** El primer componente corresponde a la codificación Manchester, la cual es utilizada en el protocolo para representar la información transmitida. Esta codificación, como se verá a continuación, tiene características que la hacen elegible ya que se ajusta bien a los principios de diseño que se buscan en Internet 0.

La codificación Manchester es una técnica de codificación auto-sincrónica, es decir, la señal de reloj se codifica junto con los datos. Para lograr esto, los datos binarios transmitidos utilizan un formato diferente en el cual los unos y los ceros lógicos se representan con transiciones de 0 a 1 o de 1 a 0; la transición de 1 a 0 representa un 0 y la transición de 0 a 1 representa un 1. Esta representación da como resultado la división de cada bit de información en dos fases. Entre estas dos fases es que ocurre la transición. Como se observa en la figura 3, donde las flechas rojas representan la señal en alto (1) y los espacios en blanco la señal en bajo (0), el byte '100101110' se representaría como '11 01101001 1001010110 11' si se incluyen los bits de inicio y final.



**Figura 3.** Representación de un byte en Manchester

Este mecanismo ofrece varias ventajas. La principal es el hecho de que la señal se sincroniza a sí misma, permitiendo que la velocidad de transmisión se establezca con cada bit en el momento de la comunicación. Esto reduce la cantidad de errores y por ende, hace la señal más confiable que en otros esquemas. Otra ventaja es que se logra distinguir entre el envío de una secuencia de ceros y ninguna transmisión. La desventaja es que se requiere el doble de bits para enviar una secuencia codificada.

**Comunicación Auto- Sincrónica.** El segundo componente corresponde a la propiedad de auto-sincronización de las transmisiones. Esto se logra utilizando la codificación Manchester y el formato 8N1. Esta propiedad de auto-sincronización permite que los dispositivos en una red Internet 0 puedan transmitir a una velocidad arbitraria, es decir, que no haya sido preestablecida.

**8N1.** El formato 8N1 es básicamente 8 bits de datos puros con un bit “start” al inicio y un bit “stop” al final, sin bit de paridad. Típicamente, los bits “start” y “stop” se representan con 0, pero en este caso, dado a que se utiliza además la codificación Manchester es posible representar estos dos bits de manera que no se confundan con los bits de datos. Ya que los datos son 1 ó 0, y en Manchester estos se representan 01 y 10 respectivamente, se puede utilizar el código 11 para representar el “start” y el “stop”.

**Direccionamiento.** El tercer componente corresponde al direccionamiento de los dispositivos que se encuentran conectados a las redes Internet 0. Los diseñadores de Internet 0 sugieren que deben usar los protocolos existentes siempre que sea razonable, es decir, en los casos que aplique. En caso contrario se debe estudiar el caso e implementar localmente el protocolo adecuado. El direccionamiento es necesario en las redes para que los dispositivos puedan identificarse a sí mismos y a otros dispositivos dentro de la red.

**Sistema Postal.** El sistema de correo postal es talvez de las formas más antiguas de direccionamiento. El primer sistema formal y organizado proviene de Egipto, donde los faraones utilizaban mensajeros para la difusión de sus decretos en el territorio del estado (2400 aC).

En este, las direcciones son de tamaño variable. Consta de direcciones alfa-numéricas descriptivas. Además permite redundancia por medio del uso de códigos postales o códigos ZIP.

Este tipo de direccionamiento permite también sub-direccionamiento por ejemplo detallando el número de apartamento dentro de un edificio. Otra característica importante de las direcciones de correo postal es que son reconocidas mundialmente.

**Red Telefónica.** Las direcciones en una red telefónica se representan por medio de códigos numéricos que típicamente se asignan por posición geográfica. La longitud de estos códigos es variable pero tiene un máximo.

Las redes telefónicas del mundo están conectadas por lo que existen códigos mundiales para permitir comunicación entre teléfonos en diferentes partes del mundo.

**Direcciones IP.** Talvez uno de las más conocidas en el ámbito computacional es el direccionamiento IP. Una dirección IP consta de un número de 32. Hay tres clases de direcciones IP que una organización puede recibir de parte de Internet Assigned Numbers Authority (IANA): clase A, clase B y clase C. En la actualidad, IANA reserva las direcciones de clase A para los gobiernos de todo el mundo y las direcciones de clase B para las medianas empresas. Se otorgan direcciones de clase C para todos los demás solicitantes.

Cada dirección IP es única en Internet, lo que permite identificar una máquina en cualquier parte del mundo. También existen direcciones IP locales que solamente se pueden utilizar dentro de redes privadas, estas máquinas no pueden comunicarse directamente con máquinas fuera de la red privada.

### 3 Materiales y Métodos

El proyecto se realizó en conjunto con el Laboratorio de Fabricación Personal del Centro de Bits y Átomos dirigido por el Profesor Neil Gershenfeld, quien es principal investigador detrás de Internet 0; y con Benn Molund profesor de *Høgskolen i Tromsø*, colaborador del Laboratorio de Fabricación Personal en Noruega, quien realizó una investigación sobre el proyecto Internet 0 aplicado a otras tecnologías. Se realizaron reuniones de planificación, así como algunos talleres, demostraciones de dispositivos existentes y muestras de resultados en cada país. Cada visita a Noruega y a Boston tuvo una duración de 1 semana en la cual se trabajó aproximadamente 40 horas.

El proyecto se desarrolló, en un principio, en paralelo y en colaboración con el equipo de trabajo de Noruega, compartiendo resultados y datos de los estudios de ambos proyectos. Sin embargo, el proyecto en Noruega fue suspendido y aunque se mantuvo el contacto con los investigadores, el proyecto no arrojó mayores resultados.

Se diseñaron y construyeron dispositivos con interfaces de comunicación diferentes. A cada uno se le implantó el protocolo de Internet 0 y se intentó lograr comunicación entre ellos, inicialmente de manera alambrada y después con medios inalámbricos. La implementación del protocolo se realizó en varias etapas según el artículo publicado por el profesor Gershenfeld. Se inició con la programación de la codificación Manchester, seguido del módulo de auto-sincronización y finalmente se estudiaron opciones de direccionamiento.

El proyecto requirió la construcción de dispositivos electrónicos y desarrollo de software para realizar las pruebas de comunicación. Se construyeron dispositivos electrónicos tanto para comunicación alambrada como para comunicación inalámbrica.

Durante la construcción y el desarrollo se observaron las variables de costo (se tomó en cuenta el costo de las partes y componentes requeridas para construir los dispositivos) y la complejidad de implantación (cantidad de horas dedicadas en la construcción, cantidad de partes requeridas, cantidad de horas dedicadas en el desarrollo y cantidad de líneas de código requeridas).

Las pruebas de comunicación alambradas sirvieron para comprobar la eficacia del protocolo Internet 0 para realizar comunicación entre 2 dispositivos similares. En estas pruebas se midió la velocidad de transmisión y el porcentaje de error o falla en la transmisión de datos.

Las pruebas de comunicación inalámbrica se realizaron para comprobar la adaptabilidad del protocolo a diferentes medios de transmisión en dispositivos heterogéneos. En las pruebas de comunicación inalámbrica se observó también la velocidad de transmisión, además se tomaron medidas de los rangos de distancia de transmisión posibles.

### ***3.1 Materiales y equipo utilizado.***

El desarrollo del software se realizó en una computadora de escritorio con procesador Intel Pentium III con 384 Mb de memoria RAM con un Sistema Operativo Windows 2000. El software fue desarrollado utilizando Java, .Net y ensamblador para PIC.

Los dispositivos se construyeron con microcontroladores PIC modelo 18F258. Además se utilizaron Pocket PC HP iPaq con módulos de Bluetooth integrados. Los dispositivos desarrollados por los Laboratorios en Noruega y Boston utilizaron microcontroladores AVR modelo AtTiny15.



## 4 Resultados y discusión

Los resultados obtenidos no fueron del todo satisfactorios. El protocolo Internet Cero resultó ser complejo de implantar en los dispositivos electrónicos. El esquema de codificación, decodificación y envío de bytes propuesto por el protocolo no es compatible con las arquitecturas seleccionadas y en consecuencia la implementación de los algoritmos es más compleja, característica que se refleja con una cantidad mayor de instrucciones requeridas.

La mayoría de los dispositivos y microcontroladores incorporan ya algún protocolo de comunicación en su diseño. Al tratar de implantar Internet 0 en estos dispositivos, lo que se hace en realidad es agregar una capa al modelo de comunicación<sup>1</sup>, lo que reduce la eficiencia en la transmisión y recepción de datos, sin obtener los beneficios de abstracción o de simplificación que se esperaría al añadir una capa a cualquier sistema.

En cuanto a la comunicación con Internet convencional es factible pensar en un dispositivo central que trabaje como un portal hacia Internet y que se maneje un protocolo de direccionamiento interno mucho más sencillo entre los dispositivos electrónicos que el protocolo de direccionamiento utilizado en Internet, ya que por los recursos limitados con los que cuentan estos dispositivos, el protocolo IP representa un costo muy alto.

A continuación se detallan algunos aspectos técnicos sobre la implementación del protocolo y se revisan en detalle los resultados obtenidos a partir de las evaluaciones realizadas.

---

<sup>1</sup> En un modelo de comunicación por capas, cada capa se encarga de resolver una parte del problema de comunicación, pero requiere procesamiento extra para pasar información de una capa a otra. En general, entre más capas tenga el modelo más lento es.

## **4.1 Submódulos de desarrollo**

Como se mencionó anteriormente, la implementación del protocolo se dividió en las siguientes etapas:

1. Codificación Manchester.
2. Comunicación auto-sincronización
3. Direccionamiento.

Para cada modulo se describen primero las actividades realizadas más relevantes y al final de cada sección se presentan las conclusiones y los resultados más importantes de la sección.

## **4.2 Codificación Manchester**

Para probar la codificación Manchester en el protocolo Internet 0, se decidió realizar una implementación en el lenguaje de programación Java<sup>2</sup>. La idea general del algoritmo es en realidad muy sencilla. Como se requiere representar cada bit (0 ó 1) con una transición, sólo hace falta representar los '0' como '01' y los '1' como '10'. Como resultado, por cada byte que se recibe, el codificador entrega dos bytes. Se consideraron dos implementaciones diferentes del algoritmo. La primer opción (A) era más sencilla y corta en cuanto a código pero presentaba una pequeña modificación en el formato de representación de los bytes codificados, lo que causaba una incompatibilidad entre los dispositivos con esta implementación y dispositivos con la implementación usual. La segunda opción (B) era más compleja que la opción (A) en cuanto a cantidad de líneas de código utilizadas pero respeta el formato establecido. Se optó por la opción (B) porque no presenta modificaciones en el formato de los bytes codificados. Este algoritmo se explica a continuación.

---

<sup>2</sup> El lenguaje Java fue desarrollado por Sun Microsystems en 1995. Es similar a C y C++, pero el modelado de Objetos es más simple y ofrece menos capacidades para procesamiento a bajo nivel.

### 4.2.1 Algoritmo

El algoritmo (B) es como sigue.

1. Primero se toma el byte y se separa la parte baja de la parte alta en dos bytes  $x$  y  $y$ . Además se tienen 2 bytes más para el resultado:  $resx$  y  $resy$ .
2. Seguidamente se van tomando los residuos de la división entre 2 de cada una de las partes:  $x$  y  $y$ . Se hace un corrimiento de 2 hacia la derecha de  $resx$  y  $resy$  y se les suma  $(64 * (2 - \text{residuo}))$ . De esta manera cuando el residuo es 0, se suma 128 al byte resultado y se suma 64 cuando el residuo es 1.
3. Este proceso se repite 4 veces, al final del ciclo se tienen dos bytes:  $resx$  y  $resy$  con la codificación del byte original.

La figura 4 muestra el segmento de código en Java que produce la codificación.

```
public int[] encode (int b){
    int [] respuesta = new int[2];
    int x = b >> 4;
    int y = b & 15;
    int tmpx = 0;
    int tmpy = 0;
    int resx = 0;
    int resy = 0;

    for(int i = 0; i < 4; i++){
        tmpx = x % 2;
        x = x / 2;
        resx = resx >> 2;
        resx = resx + (64 * (2 - tmpx));

        tmpy = y % 2;
        y = y / 2;
        resy = resy >> 2;
        resy = resy + (64 * (2 - tmpy));
    }

    respuesta[0] = resx;
    respuesta[1] = resy;
    return respuesta;
}
```

**Figura 4:** Segmento de código en Java para la codificación Manchester.

## 4.2.2 Decodificación

De manera similar, el decodificador recibe dos bytes. Recorre los dos bytes leyendo de 2 en 2 los bits para identificar si corresponde a un 1 o a un 0. Finalmente retorna un solo byte con la decodificación correspondiente.

El proceso de decodificación es el siguiente:

1. Se obtienen los dos bytes *x* y *y*, que corresponden a la parte alta y a la parte baja respectivamente.
2. Ambos bytes se procesan en un ciclo de la siguiente manera: se toma el residuo de la división por 2 del número, se multiplica por 8 y se suma a la respuesta. Se hace un shift a la derecha de la respuesta para dar campo al siguiente bit.
3. Esto se realiza 4 veces para cada byte.
4. Al final se realiza un corrimiento hacia la izquierda de *resx* y se hace una operación OR con *resy* para formar el byte decodificado.

La figura 5 muestra el segmento de código en Java que produce la decodificación.

```
public int decode (int [] b){
    int x = b[0];
    int y = b[1];
    int tmpx = 0;
    int tmpy = 0;
    int resx = 0;
    int resy = 0;

    for(int i = 0; i < 4; i++){
        tmpx = x % 2;
        x = x / 4;
        resx = resx >> 1;
        resx = resx + (tmpx * 8);

        tmpy = y % 2;
        y = y / 4;
        resy = resy >> 1;
        resy = resy + (tmpy * 8);
    }

    return (resx << 4) | resy;
}
```

**Figura 2:** Segmento de código en Java para la decodificación Manchester.

### 4.2.3 Resultados

La implementación de la codificación fue exitosa en el sentido de que se logró codificar los bytes al formato especificado por el protocolo Internet 0, aunque no se ha realizado ninguna transmisión real para probar su efectividad en la auto-sincronización de la comunicación entre 2 dispositivos.

La programación con Java presentó ciertos problemas con el manejo de los bytes, puesto que esto se manejan, en Java, con signo por lo que algunas operaciones dan resultados adecuados cuando se trabaja a nivel de byte, pero inadecuados cuando se trabaja a nivel de bit, como es en este caso. Finalmente se optó por manejar los datos con el tipo *int*. Aunque estos utilizan dos bytes, se adaptan mejor a las necesidades del programa. La programación de este algoritmo resulta aún más sencilla en dispositivos tales como un microcontrolador PIC, por la capacidad de estos de manejar la información a nivel de bit y no de byte.

Durante la implementación de la recepción de datos con Internet 0 se encontró un problema con el cálculo de la frecuencia de transmisión. Inicialmente se pensó en utilizar el siguiente algoritmo. Al iniciar la rutina de recepción se entra a un ciclo de espera. Una vez que se detecta una señal (1) se asume que se está iniciando una transmisión desde el otro dispositivo, la cual debe iniciar con el bit "start" (11), y se toma el tiempo que dura la señal en alto. Este tiempo se divide entre 2 para determinar la duración de cada fase, entonces se puede utilizar par hacer las lecturas del resto de los bits. El tiempo de la fase se calcularía al inicio de cada byte.

El problema con este algoritmo es que si el primer bit de datos es un 0 (10) entonces la transmisión inicia con 3 impulsos en alto y no con 2, y no existe manera de determinar cual de los 2 casos es. Existen varias maneras de solucionar el problema. Actualmente se está estudiando el caso para tratar de llegar a una solución óptima.

### **4.3 Comunicación Auto- Sincrónica**

Este módulo se puede dividir en dos partes. Primero se requiere lograr comunicación, segundo, se requiere que esta comunicación se auto-sincronice sin necesidad de relojes externos, ya que en principio, los dispositivos podrían transmitir a velocidades diferentes y estos debe ajustarse y adaptarse entre sí.

Para probar la comunicación se utilizaron computadoras de bolsillo y dispositivos con microcontroladores<sup>3</sup>. Se requirió un sistema para monitorear las transmisiones que se dan desde y hacia el dispositivo. A este sistema le llamamos Monitor.

Para probar la comunicación auto-sincrónica se determinó que era necesario implementar la transmisión y recepción de datos en serie a través de un dispositivo como un microcontrolador. Esto debido a que la transmisión en serie por el puerto que usualmente se encuentra en las computadoras pre-establece una velocidad de transmisión y recepción que imposibilita poder probar la auto-sincronización.

#### **4.3.1 Monitor**

Para evaluar la conexión entre la computadora y el dispositivo a desarrollar, se implementó un programa monitor para enviar y recibir mensajes a través del puerto serie. Este programa consta de tres áreas de texto. En el primer área se digitan los mensajes a enviar. En la segunda área se muestran los mensajes recibidos y en la tercera área se muestran, a manera de bitácora, todos los eventos ocurridos desde el inicio del programa. La figura 6 muestra la ventana principal del programa monitor.

---

<sup>3</sup> Un microcontrolador es un circuito integrado que ofrece las posibilidades de una computadora.



**Figura 6.** Ventana principal del programa monitor

### 4.3.2 Modelo de comunicación

El programa monitor, para transmitir, recibe un mensaje de texto, el cual codifica utilizando Manchester y envía este mensaje a través del puerto. Este mensaje debe ser recibido por el microcontrolador y decodificado. La velocidad de transmisión se establece en el computador pero debe ser calculado por el microcontrolador durante la recepción. Entonces el microcontrolador reenvía el mensaje decodificado a la computadora para verificar que este fue recibido correctamente. En un principio se pensó que la recepción en ambas partes (computadora y microcontrolador) debía auto-sincronizarse con su respectivo transmisor, pero nuevamente, en la computadora este proceso es difícil de implementar. Por esto se decidió, finalmente, por dejar la auto-sincronización solamente en la recepción del microcontrolador.

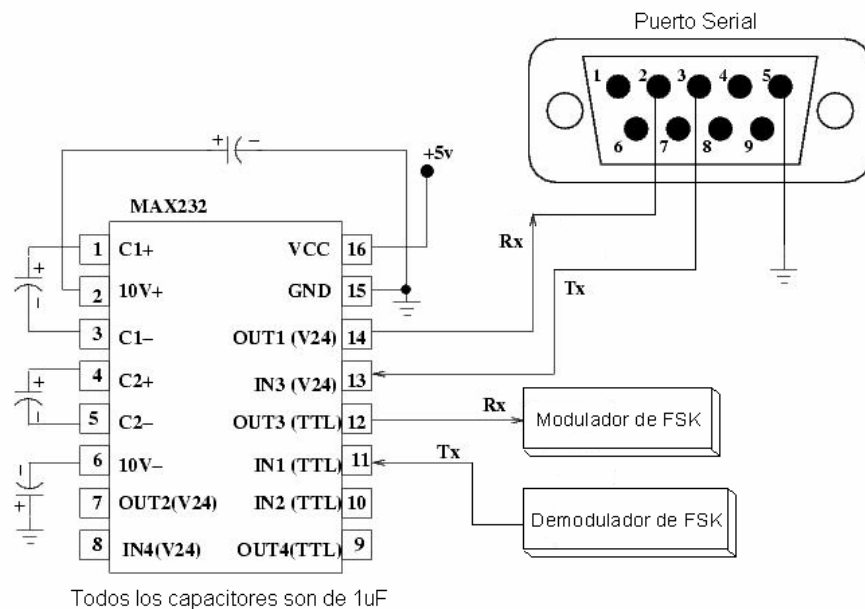
### 4.3.3 PIC

Por otro lado, el dispositivo seleccionado para realizar esta implementación fue el microcontrolador PIC18F258 de Microchip. Principalmente se eligió este dispositivo por la experiencia que se tiene en el desarrollo de aplicaciones, además ya se cuenta con un conjunto importante de herramientas y documentación sobre este PIC en particular.

Otra razón que es igual de importante es que este modelo de PIC es bastante general y permite un amplio rango de aplicaciones, lo que permitiría ampliar fácilmente la investigación posterior.

#### 4.3.4 Limitaciones de modelo

La primera limitante que se encontró al tomar esta decisión es que se requiere, además de implementar el software, crear el hardware que permita la comunicación en serie con la computadora. La conexión entre el dispositivo y la computadora requiere un componente adicional de hardware que sirve de intermediario, debido a la diferencia en los niveles de tensión del PIC y la computadora. El diagrama de conexión se muestra en la figura 7. Esta parte no estaba contemplada en el cronograma desde el inicio del proyecto por lo que no se pudo completar es su totalidad.



**Figura 7.** Diagrama de conexión del MAX232 para la comunicación serial entre computadora y PIC

Se pensó también aprovechar el módulo USART que tiene el PIC para facilitar la implementación de las transmisiones, pero al igual que en la computadora, este módulo ofrece la transmisión automática de información por medio de registros especiales que envían y reciben de byte en byte. El envío de los bits de “start” y “stop” también son

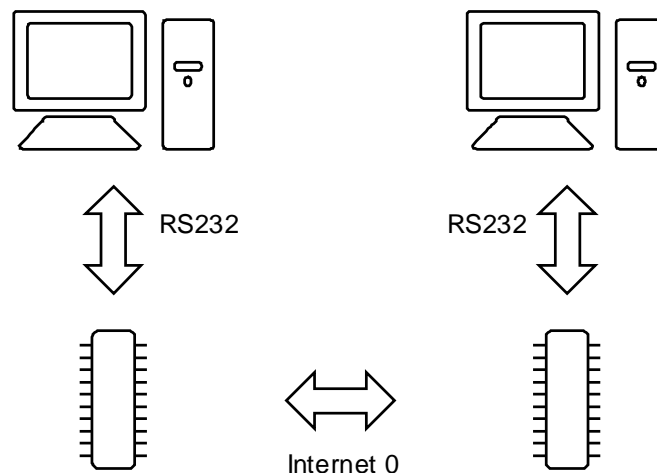


transparentes para el desarrollador. Por esta razón toda la comunicación y los componentes que se requieren para lograrla se debían implementar en su totalidad.

Además este modelo (comunicación Computadora-PIC) solamente permite la evaluación del protocolo Internet 0 en un sentido: desde la computadora hacia el PIC. En donde el PIC recibe, decodifica y sincroniza la señal utilizando los componentes descritos para el protocolo. Pero en sentidos contrario, desde el PIC a la computadora, no se puede hacer ninguna evaluación real del protocolo.

#### 4.3.5 Nuevo modelo

Finalmente se llegó a la conclusión de que el modelo óptimo para realizar las pruebas de transmisión debía ser como se muestra en la figura 8. Para este modelo se requieren dos computadoras conectadas a través del puerto serie con un dispositivo PIC cada una. La comunicación entre computadora y PIC se realiza utilizando los protocolos de transmisión serial conocidos. La comunicación PIC a PIC se realiza con Internet 0. Las computadoras permiten enviar y recibir los mensajes fácilmente, son solamente una interfaz de usuario.



**Figura 8.** Modelo de conexión para pruebas de transmisión con Internet 0.

Para utilizar este modelo, se requiere entonces que el programa del PIC implemente el protocolo Internet 0 para comunicarse con su homólogo. También debe implementar comunicación Serial (RS232) para lograr comunicarse con la computadora. Por otro lado, el programa en cada computadora no requiere implementar ninguno de los componentes de Internet 0.

#### **4.3.6 Comunicación con Pocket PC**

Adicionalmente se trabajó con computadoras de bolsillo (Pocket PC) para tratar de lograr comunicación entre éstas y dispositivos con microcontroladores PIC. Se realizaron pruebas para verificar la factibilidad de la comunicación entre estos dos con resultados positivos se trabajó en la implantación del protocolo Internet 0. Las pruebas hechas se pueden dividir en 2 partes. La primera consistió en enviar datos de la Pocket PC al microcontrolador para que éste respondiera de alguna manera. La segunda parte consistió en enviar datos del microcontrolador a la Pocket PC y representarlos gráficamente en la pantalla.

#### **4.3.7 Resultados**

Aunque se logró establecer comunicación con los dispositivos se encontró que la implantación de Internet 0 no es la más adecuada por razones similares a las mencionadas anteriormente. El protocolo Internet 0 debe programarse sobre los protocolos incluidos en los sistemas actuales lo que lo hace poco viable para un uso extendido. Esta situación cambiaría si los fabricantes de los dispositivos incluyeran las facilidades necesarias para la implantación del protocolo en los sistemas comerciales. Los protocolos actuales son más robustos, están listos para utilizar, se pueden adaptar fácilmente a las necesidades de las aplicaciones y son más o tan eficientes como Internet 0.

## 5 Conclusiones

1. La codificación Manchester se puede realizar de una manera sencilla pero lo óptimo es utilizar lenguajes que permitan el manejo de la información a nivel de bit, como los lenguajes de bajo nivel.
2. La codificación Manchester no es asíncrona, es en realidad auto-sincrónica.
3. Los bytes en Java utilizan signo, por lo que resulta difícil la utilización de los mismos para el manejo de bits.
4. Aunque el tipo int en Java toma el doble de espacio que el tipo byte, se adapta mejor a las necesidades del programa implementado.
5. La codificación Manchester se adapta bien al protocolo Internet 0, ya que su implementación se hace de una manera más sencilla para microcontroladores, para los cuales se creó el protocolo.
6. Es difícil implementar la transmisión en serie desde el inicio, en dispositivos y computadores ya que la mayoría de los fabricantes supone que los desarrolladores utilizarán los protocolos existentes y ofrecen las herramientas necesarias para utilizar los mismos, pero no ofrecen facilidades para los desarrolladores que desean implementar sus propios protocolos.
7. Los lenguajes de alto nivel como Java facilitan la comunicación a través de los puertos brindando herramientas que realizan la transmisión y la recepción de los datos automáticamente.
8. Como el protocolo Internet 0 fue pensado para dispositivos diferentes a una computadora, su implementación y evaluación se hace más sencilla en este tipo de dispositivos que en computadoras multipropósito y sistemas operativos que utilizan protocolos de comunicación existentes.

9. El uso de computadores de bolsillo, “Pocket PC”, que aún son sistemas con recursos de almacenamiento y procesamiento limitados, además de permitir movilidad y ubicuidad, hace más evidente la necesidad de crear un protocolo de comunicación que se adapte a estas características, como lo es Internet 0.
10. El desarrollo en Pocket PC es un área relativamente nueva en el ámbito costarricense y no se tiene mucha experiencia alrededor de este tema.
11. Existen diferentes esquemas de direccionamiento que son utilizados en redes mundiales, cada uno con características particulares.
12. Los sistemas de direccionamiento han evolucionado lentamente hasta llegar a lo que son ahora, han sufrido varias modificaciones que normalmente responden al crecimiento y a la necesidad de asignar más direcciones (sub-direccionamiento y direcciones de longitud variable).
13. Como se quiere que Internet 0 se comuniquen con las redes existentes este debe soportar alguno de los sistemas de direccionamiento actuales.
14. El tema de direccionamiento es un tema complejo ya que existen varios elementos que se deben tomar en cuenta para su desarrollo. Por ejemplo, se debe considerar los mecanismos de asignación y el control de las direcciones e identificadores de los diferentes dispositivos, ya que estos deben ser únicos en la red y debería ser posible asignarlos dinámicamente.
15. La relación costo/beneficio no es favorable para Internet 0 por la complejidad de su implantación en dispositivos electrónicos y portables actuales.

## 6 Recomendaciones

1. Para las pruebas de implantación del protocolo Internet 0 y sus componentes, lo óptimo es hacerlo en dispositivos que soporten el manejo de información a nivel de bits, ya que el protocolo lo requiere. Al utilizar lenguajes de alto nivel, se tiene un nivel de complejidad más que se puede evitar con lenguajes de bajo nivel.
2. Al utilizar la codificación Manchester se facilita la sincronización de los dispositivos en el momento de envío y recepción de mensajes.
3. Al trabajar con lenguajes de alto nivel en este tipo de aplicaciones, se requiere seleccionar las estructuras de datos más adecuadas, aunque se deba sacrificar la optimización de los recursos.
4. Las pruebas son más fáciles de ejecutar si éstas son realizadas directamente en los dispositivos para los cuales el protocolo fue diseñado.
5. Al implantar un protocolo de comunicación nuevo se debe hacer el desarrollo desde el inicio sin el apoyo de las herramientas y módulos de comunicación predefinidos en las computadoras y dispositivos, ya que estas no necesariamente se adaptan a las especificaciones del protocolo que se desea implantar.
6. Al utilizar lenguajes de alto nivel se pueden crear fácilmente aplicaciones de prueba que se comuniquen con los dispositivos y microcontroladores.
7. Las fases de evaluación e implantación del protocolo Internet 0 se deben realizar en los dispositivos para el cual fue diseñado en donde se tienen mayor acceso en el bajo nivel a los mecanismos de comunicación y administración de los datos.
8. Las herramientas que ofrecen los lenguajes de alto nivel se pueden aprovechar para crear rápidamente programas de prueba que se comuniquen con los

- dispositivos. De esta manera se puede visualizar, almacenar, analizar mejor los datos obtenidos de los dispositivos.
9. La utilización de computadoras de bolsillo permite movilidad y ubicuidad, características que resaltan la importancia y la necesidad de crear un protocolo como Internet 0.
  10. Se requiere más desarrollo e investigación sobre Pocket PC para obtener los resultados deseados con este producto.
  11. Es necesario una evaluación de los diferentes mecanismos de direccionamiento existentes para hacer la selección del que más se adapte a las necesidades del proyecto.
  12. El tipo de direccionamiento que se debe utilizar en los dispositivos debe ser compatible con IP para que éstos tengan acceso a las redes de computadoras convencionales y para tener acceso a los dispositivos desde estas redes.
  13. Al tratar de establecer comunicación entre dispositivos resulta menos complejo utilizar los protocolos incluidos que tratar de implementar Internet 0.

## 7 Referencias

[1] Fairhurst, G. Manchester Encoding. 01/09/2001. Recuperado el 01/09/2005 en <http://www.erg.abdn.ac.uk/users/gorry/course/phy-pages/man.html>

[2] Peacock, C. Interfacing the Serial / RS-232 Port. 15/06/2005. Recuperado el 08/11/2005 en <http://www.beyondlogic.org/serial/serial1.htm#40>

[3] Neil Gershenfeld, Raffi Krikorian, Danny Cohen, Internet 0: Interdevice Internetworking. *Scientific American*.

[4] Sin autor. Correo. Sin fecha. Recuperado el 11/11/2005 en [http://es.wikipedia.org/wiki/Servicio\\_postal](http://es.wikipedia.org/wiki/Servicio_postal)

[5] Sin autor. Dirección IP. Sin fecha. Recuperado el 11/11/2005 en [http://es.wikipedia.org/wiki/Direcci%C3%B3n\\_IP](http://es.wikipedia.org/wiki/Direcci%C3%B3n_IP)