

# Sincronización de semáforos vehiculares mediante protocolo LoRaWAN

David Runke  
Centro CODAPLI  
UTN-Facultad Regional La Plata  
La Plata, Buenos Aires, Argentina  
davidrunke@gmail.com

Julietta Sánchez  
Centro CODAPLI  
UTN-Facultad Regional La Plata  
La Plata, Buenos Aires, Argentina  
julysanchez.220@gmail.com

Héctor Hugo Mazzeo  
Centro CODAPLI  
UTN-Facultad Regional La Plata  
La Plata, Buenos Aires, Argentina  
hugo.maz@frlp.utn.edu.ar

Omar E. Rodríguez  
Centro CODAPLI  
UTN-Facultad Regional La Plata  
La Plata, Buenos Aires, Argentina  
orodriguez@frlp.utn.edu.ar

José A. Rapallini  
Centro CODAPLI  
UTN-Facultad Regional La Plata  
La Plata, Buenos Aires, Argentina  
rapalini@frlp.utn.edu.ar

**Abstract**—Este trabajo presenta un desarrollo para la comunicación entre dos nodos en una red inalámbrica de sensores y su utilidad para lograr la sincronización de semáforos vehiculares, buscando facilitar el tránsito en las ciudades, haciendo a las calles y avenidas menos caóticas y evitando así problemas de congestión vehicular. Con la coordinación de los semáforos, se concibe el concepto de “onda verde” que agiliza el tránsito y propone una solución en las horas pico, cuando las calles se atiborran de transeúntes. La solución planteada pretende demostrar que, valiéndose de sensores inalámbricos en red, bajo uno de los múltiples protocolos existentes de transferencia de datos, se puede lograr la sincronización de los semáforos, evitando su posterior desconfiguración y sin la necesidad de gastar grandes cantidades de dinero para resolver el problema. (*Abstract*)

**Keywords**—semáforos, red, sensores, inalámbricos, LoRaWAN (*key words*)

## I. INTRODUCCIÓN

En muchas ciudades del mundo se apunta al paradigma de “Smart City”, que plantea que a través de la automatización y el control se pueden optimizar objetos o servicios para tornarlos más eficientes y funcionales. En este trabajo se pretende utilizar recursos hardware para simplificar la sincronización de los semáforos en avenidas, implementando el concepto de IoT (Internet de las cosas) y apuntando hacia el nuevo paradigma de las ciudades inteligentes.

El tránsito de las grandes urbes es un factor creciente que genera caos, accidentes y retrasos temporales. Muchas personas suelen sufrir este enorme desorden a diario, siendo perjudicados por variables que no pueden controlar. Para evitar estos embotellamientos y sobre todo accidentes [1], se adoptó la medida de regular el paso con semáforos. Estos son muy útiles para evitar colisiones y embotellamientos. Pero suponen una gran pérdida de tiempo si no están sincronizados entre ellos. Si bien, más de una vez, se coordinan los semáforos para que se genere un flujo de tránsito deseable a determinadas horas del día, estos acaban por desconfigurarse al cabo de un tiempo. Esto sucede porque el enlace se basa en el funcionamiento de relojes ubicados en cada controlador. Estos relojes deben mantenerse todos en igual hora, lo cual es prácticamente imposible por las condiciones climáticas y técnicas que afectan al controlador, debiéndose por lo tanto ajustar periódicamente para evitar que se modifique la coordinación.

Se observa que en muchos casos existe un conflicto con la sincronización de los semáforos dado que no deberían entorpecer el flujo de tránsito sino simplificarlo. Esto podría lograrse con la coordinación de los semáforos en las avenidas para lograr el dinamismo necesario para un caudal constante de vehículos, evitando entorpecer con grandes tiempos de espera en cada esquina.

Algunas soluciones para mejorar esta situación son:

- Poner en hora los relojes que llevan dentro los controladores de los semáforos.
- Sugerir una velocidad cruce por la avenida.
- Generar una “onda verde” que facilite el tránsito, sincronizando los relojes.

Una vez planteada la solución surge otra dificultad que radica en que los relojes de los semáforos no modifiquen sus horarios y permanezcan todos con la hora correcta y uniforme a pesar de la humedad y las variaciones climáticas que suelen afectar de forma directa su funcionamiento.

## II. PROTOCOLO LORAWAN

Para evitar la contingencia de los relojes de los semáforos a deshora, se plantea una solución tendiente a perfeccionar su sincronización. Para lograrlo se propone el uso de una red de sensores inalámbricos (WSN, Wireless Sensor Network) para la interconexión de los nodos (semáforos). A través de esta red se busca generar una comunicación entre los nodos que facilite la puesta en hora de los relojes evitando cambios abruptos en las regulaciones del tránsito. Con esto se pretende constituir un avance hacia el campo de las Smart Cities, consiguiendo la interoperabilidad de estos “semáforos inteligentes”, apuntando a una circulación vehicular más ordenada y segura.

Cada tecnología inalámbrica tiene puntos fuertes y débiles. El estándar WiFi, por ejemplo, puede transmitir grandes cantidades de datos a alta velocidad, pero tiene un alcance limitado. Una red celular (GSM) combina alta velocidad y larga distancia, pero se queda corta en cuanto a lograr bajos niveles de consumo de energía.

Las aplicaciones IoT tales como la adquisición de datos remotos, el control de iluminación urbana, la vigilancia meteorológica y la agricultura, tienen cada una un conjunto

diferente de prioridades. Las cantidades que se miden o controlan en estas aplicaciones tales como las condiciones climáticas, los niveles de humedad en el suelo o la instalación de luminarias cambian muy lentamente durante un período de tiempo prolongado. Además, los nodos de sensores están a menudo a kilómetros de distancia y están operados a batería, por lo que el protocolo inalámbrico óptimo debe ser capaz de enviar pequeños paquetes de datos de forma eficiente a través de largas distancias con el mínimo consumo de energía. El protocolo LoRaWAN (Long Range Wide Area Network) fue diseñado exactamente para estos requerimientos. Es un protocolo de red de capa de enlace que usa la tecnología de radio LoRa para comunicar y administrar dispositivos de IoT.

El término LoRa significa Largo Alcance [2]. Es una tecnología inalámbrica de radiofrecuencia introducida por una compañía llamada Semtech [3]. Esta tecnología se puede utilizar para transmitir información bidireccional a larga distancia sin consumir mucha energía. Esta propiedad puede ser utilizada por sensores remotos que tienen que transmitir sus datos simplemente operando con una batería pequeña. Por lo general, LoRa puede alcanzar una distancia de 15-20 km y funcionar con batería durante años.

En cualquier solución de IoT típica habrá cientos de nodos sensores implementados en el campo que supervisarán los parámetros vitales y los enviarán a la red para su procesamiento. Pero estos sensores deben ser inalámbricos y deben funcionar con una batería pequeña para que sea portátil. Las soluciones inalámbricas como RF pueden enviar datos a larga distancia, pero requieren más potencia para hacerlo, por lo que no pueden funcionar con batería, mientras que BLE (Bluetooth Low Energy), por otro lado, puede funcionar con muy poca potencia, pero no puede enviar datos a larga distancia (Fig. 1).

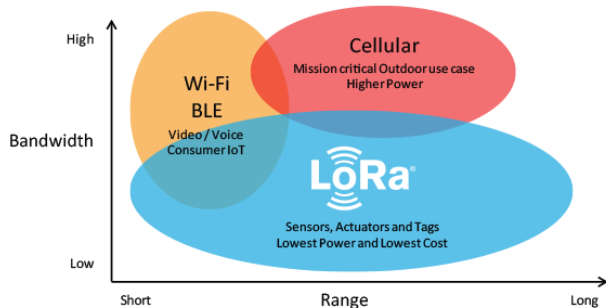


Fig. 1. Comparación de LoRa con otras tecnologías inalámbricas

Teniendo en cuenta el análisis previo, la tecnología LoRaWAN nos daría una buena opción para la solución, dado que su cobertura abarca grandes extensiones, puede tener muchos nodos coordinados, sirve para manejar pequeñas porciones de información y es una tecnología económica respecto de otras existentes [4].

Este protocolo está enmarcado en el paradigma IoT, concepto que se refiere a la interconexión digital de objetos cotidianos con Internet. El concepto fue propuesto en 1999 en el Instituto Tecnológico de Massachusetts (MIT), donde se investigaba la identificación por radiofrecuencia en red (RFID) que permite a los objetos conectarse e intercambiar datos y tecnologías de detección por sensores.

La adopción de la IoT se está acelerando en el mundo debido principalmente al crecimiento exponencial de la red, lo

que ha generado el interés para el desarrollo de aplicaciones en todos los campos, la reducción de tamaños y precios de procesadores y sensores y la reducción de costos en almacenaje y procesamiento de grandes volúmenes de información (Big Data).

Una serie de líderes de la industria IT crearon “LoRa Alliance” con el objetivo de desarrollar una nueva tecnología para lograr una comunicación de dispositivos IoT de forma más eficiente. LoRa Alliance es una organización abierta y sin ánimo de lucro cuyo objetivo es estandarizar una tecnología de comunicación de red que cumpla con una serie de características necesarias para IoT.

LoRaWAN es un protocolo de comunicación de radio de baja potencia para el IoT. Su propósito es conseguir establecer conexiones de largo alcance. Estas comunicaciones están enfocadas en pequeños dispositivos IoT o M2M que desean transmitir poca información y por lo tanto sin necesidad de grandes velocidades, pero con la intención de emplear el menor consumo de energía para así poder tener un mayor tiempo de vida con pequeñas baterías.

Su intención es formar una red de dispositivos conectados (nodos), los cuales establecerán un enlace inalámbrico con un elemento más potente capaz de comunicarse con todos estos elementos de manera gestionada. Este elemento toma el nombre de Gateway quien a su vez será capaz de comunicarse por otro protocolo de red -con mayor ancho de banda- transmitiendo toda la información de estos dispositivos a aquellos que la soliciten.

Desde el punto de vista tecnológico LoRa proporciona receptores con una alta sensibilidad, lo que hace que no sea necesario transmitir con potencias altas (<20dBm).

La tecnología LoRa tiene un enorme atractivo tanto a nivel de desarrollo tecnológico como de inmediata aplicación industrial. Sin embargo, una de las limitaciones de dicha tecnología es que se orienta a aplicaciones en las que la tasa de transmisión es baja, es decir, se utiliza típicamente para controlar una alarma anti-incendios, posibles malfuncionamientos de electrodomésticos, etc. En este tipo de aplicaciones, el dispositivo se encuentra normalmente en un estado de bajo consumo, sin actividad ni acceso a red.

Todo esto hace a la tecnología ideal para conexiones a grandes distancias y para redes de IoT que se pueden utilizar en ciudades inteligentes, lugares con poca cobertura celular o redes privadas de sensores o actuadores.

La arquitectura de red LoRaWAN [5] utiliza una topología de estrella en la que cada nodo final se comunica con varias puertas de enlace que a su vez se comunican con el servidor de red.

LoRaWAN tiene cuatro elementos de red (Fig. 2):

- Los nodos finales recopilan datos del sensor, lo transmiten “upstream” y “downstream” y reciben la comunicación desde el servidor de aplicaciones. Los dispositivos de “endpoint” usan comunicación inalámbrica “single hop” con una o varias compuertas.
- El concentrador/coordinador/Gateway, actúa como un puente transparente y retransmite los datos en forma bidireccional entre los nodos finales y los servidores “upstream”.

- El servidor de red se conecta a varias puertas de enlace a través de una conexión TCP/IP segura, ya sea por cable o inalámbrica, elimina los mensajes duplicados, decide qué compuerta debe responder a un mensaje de nodo final y gestiona el nodo final, las velocidades de transmisión de datos con una velocidad de datos adaptable (ADR), destinadas a maximizar la capacidad de la red y extender la vida útil de la batería del nodo final.
- El servidor de la aplicación recopila y analiza los datos de los nodos finales y determina las acciones del nodo final.

La comunicación de extremos normalmente es bidireccional, pero LoRa también admite el funcionamiento de multidifusión para funciones tales como actualizaciones de software. Muchos de los protocolos existentes, tales como ZigBee, emplean una topología de malla en la que cada uno de los nodos finales recibe y retransmite información desde otros nodos finales. Este enfoque aumenta el rango y el tamaño de la celda de la red, pero la sobrecarga de comunicación adicional añade complejidad, reduce la capacidad de la red y aumenta el consumo de energía de cada uno de los nodos.

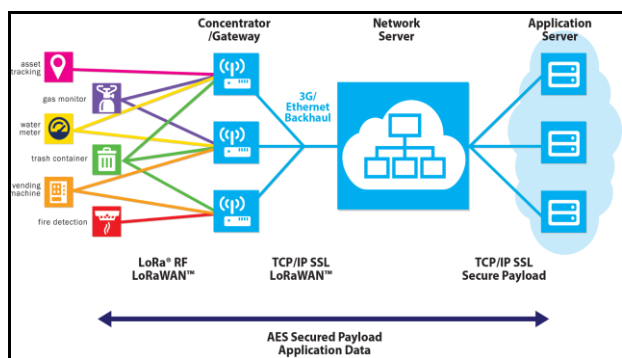


Fig. 2. Elementos de una red LoRaWAN

LoRa está pensado para aplicaciones de baja potencia de red de área amplia (LPWAN). Tiene un rango de más de 15 kilómetros y una capacidad de hasta 1 millón de nodos. La combinación de baja potencia y largo alcance limita la velocidad de datos máxima a 50 kilobits por segundo.

LoRa es una tecnología exclusiva y patentada de propiedad de Semtech Corporation que funciona en la banda ISM. La asignación de frecuencias y los requisitos reglamentarios para ISM varían por región. Dos de las más populares son las frecuencias de 868 megahercios (MHz) utilizada en Europa y 915 MHz. utilizada en América del Norte. Otras regiones, especialmente Asia, tienen diferentes requisitos.

La capa física LoRa utiliza modulación de espectro ensanchado (SSM), codificando la señal base con una secuencia de alta frecuencia que deliberadamente propaga la señal base a través de un mayor ancho de banda, reduce el consumo de energía y aumenta la resistencia a las interferencias electromagnéticas.

En este tipo de redes la seguridad forma un papel fundamental, sobre todo cuando hablamos de despliegue en ciudades (Smart Cities), donde puede existir información sensible que debe ser protegida. Por esta razón LoRaWAN proporciona varias capas de cifrado que hacen uso del

algoritmo AES-128 para proteger las comunicaciones de datos.

### III. PROPUESTA Y DISEÑO DEL PROTOTIPO

Luego del estudio del protocolo seleccionado para el desarrollo de una solución aplicable en la sincronización de las vías semaforizadas, es importante precisar que por cuestiones económicas se decidió realizar una implementación sencilla del protocolo LoRaWAN, sustituyendo la conexión convencional en estrella a través de un Gateway, por una más simple como es la conexión punto a punto. Con esta conexión se pretende simular la comunicación de dos semáforos entre sí. De esta forma se reducen los costos de implementación y se alcanza una solución factible, ejemplificando de esta forma la utilidad del uso de una red de sensores para controlar la sincronización.

Para la construcción del prototipo se utilizaron dos conjuntos de placas similares a fin de implementar la comunicación entre un nodo transmisor y un receptor. Cada conjunto está compuesto por una Arduino Nano [6] y una placa LoRa SX1278 de 433 MHz y cobertura de hasta 4 km (Fig. 3).

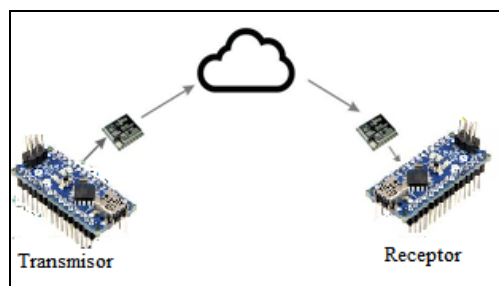


Fig. 3. Diagrama del prototipo implementado para la sincronización de dos nodos de la red LoRaWAN

La conexión de las placas Arduino con las placas LoRa se realiza a través del bus SPI. Este bus tiene una arquitectura de tipo “maestro-esclavo”. El dispositivo maestro (master) puede iniciar la comunicación con uno o varios dispositivos esclavos (slave) y enviar o recibir datos de ellos. Los dispositivos esclavos no pueden iniciar la comunicación, ni intercambiar datos entre ellos directamente. En el bus SPI la comunicación de datos entre maestro y esclavo se realiza en dos líneas independientes, una del maestro a los esclavos y otra de los esclavos al maestro. Por lo tanto, la comunicación es Full Duplex, es decir, el maestro puede enviar y recibir datos simultáneamente.

Según la documentación del standard LoRa se deben establecer las siguientes asignaciones de pines para establecer una conexión correcta entre el Arduino Nano y la placa Sx1278 (Fig. 4).

Pins on component LoRa Ra-02	Pins component on Arduino board
Vin/voltage regulators	3.3 Volt
GND	GND
MISO	D12 / Digital 12
MOSI	D11/ Digital 11
SLCK	D13/ Digital 13
Nss	D10/ Digital 10
RESET	D0/ Digital 0
DIO0	D2/ Digital 2

Fig. 4. Pines del bus SPI a usar para la conexión entre las placas

Otra característica de SPI es que es bus síncrono. El dispositivo maestro proporciona una señal de reloj que mantiene a todos los dispositivos sincronizados. Esto reduce la complejidad del sistema frente a los sistemas asíncronos. Por tanto, el bus SPI requiere un mínimo de 3 líneas.

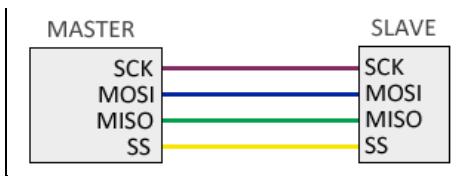


Fig. 5. Líneas necesarias para establecer una conexión SPI full-duplex

En este caso se usa el bus SPI de Arduino Nano que está establecido del puerto 10 al 13. El conexionado final entre las dos placas del prototipo utilizado, incluyendo las conexiones a los leds del semáforo, se muestra en la Fig. 6. Para el control de cada semáforo se requerirá una implementación similar conformando un nodo de la red inalámbrica.

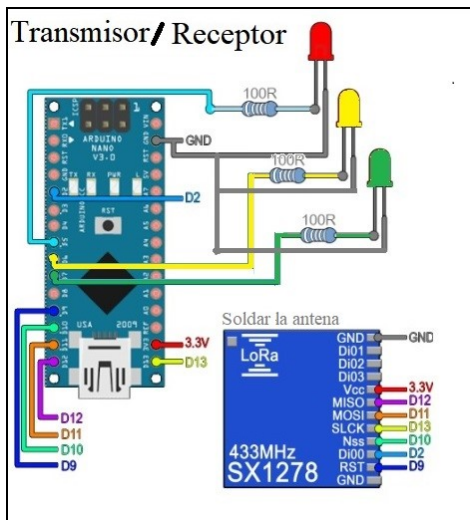


Fig. 6. Conexionado de placas Arduino Nano y LoRa para cada semáforo

#### IV. CONSTRUCCIÓN Y PROGRAMACIÓN DEL PROTOTIPO

Se construyó un prototipo con el cual se realizó una simulación del funcionamiento del protocolo LoRaWAN aplicado a la sincronización de dos semáforos (Fig. 7).

Los semáforos se simbolizaron mediante 3 leds con los colores correspondientes.

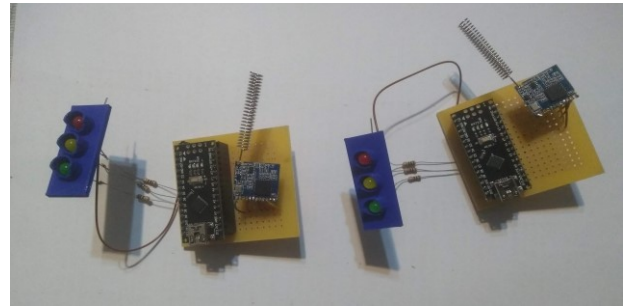


Fig. 7. Prototipo final para la sincronización de semáforos

A manera de ejemplo se agrega el código de comunicación (Figs. 8 y 9), para observar lo simple que es la utilización del protocolo para realizar aplicaciones más complejas y sus posibilidades de ampliación a una red de sensores inalámbricos. El código correspondiente a la comunicación entre transmisor y receptor está escrito en lenguaje C y es relativamente fácil de comprender inclusive para quien no esté interiorizado en el tema.

Vale mencionar que para el desarrollo se utilizaron dos librerías: LoRa.h (para enviar y recibir datos) y SPI.h (librería de Arduino SPI).

```

Código del transmisor
#include <SPI.h> // arduino spi library
#include <LoRa.h> // arduino libraries for ra02 lora
int verde = 7; // puerto al que se le asigna el led verde =7
int amarillo = 6; // puerto al que se le asigna el led amarillo =6
int rojo = 5; // puerto al que se le asigna el led rojo =5

void mandarMensaje() //establecer comunicación entre dos dispositivos: Tx y Rx
{
  LoRa.beginPacket();
  LoRa.print('q'); //envia una 'q'
  LoRa.endPacket(); //finaliza
}

void semaforizar()
{
  digitalWrite(verde,HIGH); // se enciende el led verde, HIGH envia un pulso de 5 v
  delay(5000); // espera 5 segundos
  digitalWrite(verde,LOW); //se apaga el led verde, low envia un pulso de 0 volts
  delay(10); // espera 0.01 segundos
  digitalWrite(amarillo,HIGH); //repite el proceso anterior, con los otros leds
  delay(2000);
  digitalWrite(amarillo,LOW);
  delay(10);
  digitalWrite(rojo,HIGH);
  delay(4000);
  digitalWrite(rojo,LOW);
  delay(10);
}

void setup() {
  pinMode(verde,OUTPUT); //Se define la función del puerto a OUTPUT
  pinMode(amarillo,OUTPUT); //Se define la función del puerto a OUTPUT
  pinMode(rojo,OUTPUT); //Se define la función del puerto a OUTPUT
  Serial.begin(9600); //Se inicializa el serial en ese valor

  while (!Serial); //Si existe Serial

  Serial.println("LoRa Receiver");
  //puede ser 915E6
  if (!LoRa.begin(433E6)) { //datos irrelevantes, considerados basura, se desc
    Serial.println("Starting LoRa failed ");
    while (1);
  }

  LoRa.setSpreadingFactor(10); // se le asigna al SF un valor de 10
  LoRa.setSignalBandwidth(62.5E3); // Se le asigna el ancho de banda

  LoRa.crc(); //procedimiento propio de la librería de LoRa
  semaforizar(); // Se llama al procedimiento que enciende y apaga las luces
  mandarMensaje(); // se envia un mensaje al otro dispositivo

}

void loop() {
  char mensaje= ' '; //se crea una variable mensaje, vacia.
  int packetSize = LoRa.parsePacket(); //guarda el tamaño del paquete, si es que exist
  if (packetSize) { //leer paquete

    while (LoRa.available()) {
      Serial.print((char)LoRa.read()); //convierte la salida en un char
      mensaje+=(char)LoRa.read(); // le asigna a mensaje, la salida
    }
    if (mensaje=='q') //Si mensaje es igual a 'q', se le dió el aviso de inici
    encendido
    {
      semaforizar(); //se prende el led verde y comienza la secuencia
      mandarMensaje(); //Terminó de semaforizar, Tx le avisa con una 'q' a Rx que fina
    }
    else
    { Serial.println("no llego nada"); // no existe la comunicación
    }
  }
}

```

Fig. 8. Código utilizado para el transmisor

Es importante destacar que la sincronización aquí planteada es ideal, pues lo relevante del proyecto es resolver la comunicación entre los semáforos, el modelo de sincronismo es un aditamento que podría variar sin modificar la solución al problema.

#### Código del Receptor

```
#include <SPI.h> // arduino spi library
#include <LoRa.h> // arduino libraries for ra02 lora
int verde = 7;
int amarillo = 6;
int rojo00 = 5;

void mandarMensaje()
{
  LoRa.beginPacket();
  LoRa.print('q');
  LoRa.endPacket();
}

void semaforear()
{
  digitalWrite(verde,HIGH);
  delay(5000);
  digitalWrite(verde,LOW);
  delay(10);
  digitalWrite(amarillo,HIGH);
  delay(2000);
  digitalWrite(amarillo,LOW);
  delay(10);
  digitalWrite(rojo,HIGH);
  delay(4000);
  digitalWrite(rojo,LOW);
  delay(10);
}

void setup() {
  pinMode(verde, OUTPUT);
  pinMode(amarillo, OUTPUT);
  pinMode(rojo, OUTPUT);
  Serial.begin(9600);
  while (!Serial);
  Serial.println("LoRa Receiver");
  //puede ser 915E6
  if (!LoRa.begin(433E6)) {
    Serial.println("Starting LoRa failed ");
    while (1);
  }
  LoRa.setSpreadingFactor(10);
  LoRa.setSignalBandwidth(62.5E3);
  LoRa.crc();
}

void loop() {
  char mensaje=' ';
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    // read packet
    while (LoRa.available()) {
      Serial.print((char)LoRa.read());
      mensaje=(char)LoRa.read();
    }
    if (mensaje=='q')
    {
      semaforear();
      mandarMensaje();
    }
    else
    { Serial.println("no llego nada");
  }
}
```

Fig. 9. Código utilizado para el receptor

En este caso, se plantea el ciclo de semáforos como el tiempo mínimo que puede tardar un vehículo en atravesar una avenida, este lapso está pautado en 6 segundos a una velocidad crucero de 60 km/h (máximo establecido por la ley para circular por avenidas). El ciclo comienza con el vehículo en la esquina, tarda 6 segundos en recorrer la cuadra, al llegar pasa y empieza a transitar los siguientes 100 metros (Fig. 10).

Una vez que transcurre el intervalo de tiempo, se avisa al otro semáforo que cambie su estado a verde y el anterior cambiará a rojo. Así el rojo y el verde durarán 6 segundos cada uno, permitiendo el paso del auto a velocidad crucero.



Fig. 10. Sincronización de la onda verde

#### V. CONCLUSIONES Y OBJETIVOS FUTUROS

En este trabajo se ha desarrollado una posible solución para la sincronización de semáforos vehiculares, utilizando el protocolo que mejor satisfaría la problemática planteada e implementando un prototipo para su puesta en práctica. Se propone así una posible mejora a aplicar en las ciudades actuales donde el tránsito es abundante y los accidentes son habituales. El beneficio consiste en una solución inteligente y eficiente para el ordenamiento del tránsito. También existe provecho en reducir la emanación de gases contaminantes para el ambiente. Está comprobado que el parque automotor incluye un numeroso y activo conjunto de vehículos propulsados por la combustión de hidrocarburos (ciclomotores, automóviles y camiones). Las emisiones procedentes de los escapes de estos vehículos contienen monóxido de carbono, hidrocarburos y óxidos de nitrógeno que son liberados a la atmósfera en importantes cantidades; son los componentes del "smog oxidante fotoquímico". Por esta razón, las zonas urbanas más pobladas son las que sufren la mayor contaminación de este tipo. Cada vez que se arranca un auto, la emanación de gases es mayor, pues la combustión necesaria para el primer impulso, también lo es. Entonces, al evitar las acciones de frenar y volver a arrancar en cada esquina, fruto de la sincronización de semáforos planteada, se reduce la polución ambiental.

Por último, es importante destacar que el prototipo operacional montado para las pruebas funcionó de forma correcta. Se puede inferir que, al aplicar el modelo a una escala mayor, también debería operar de manera adecuada. Esto supone un enorme beneficio y una solución innovadora, con la aplicación del nuevo concepto de Internet de las cosas, a la altura de las grandes urbes del mundo que día a día están migrando hacia un nuevo paradigma con eje en las nuevas tecnologías.

Cabe también destacar que este desarrollo forma parte de un proyecto más ambicioso que lleva a cabo el Centro CODAPLI a partir de un PID para la implementación de sistemas de comunicación en las redes inalámbricas de sensores y actuadores [7]. La disponibilidad concreta de este prototipo posibilitará la realización de un escenario de pruebas real para experimentación, con la interacción de software en tiempo real. Permitirá además implementar simulaciones con sistemas complejos de tránsito urbano, previendo la circulación de vehículos en caso de emergencias, evitando aglomeraciones de tránsito y teniendo el control sobre la circulación de los móviles en toda la ciudad. Asimismo, será

posible ensayar con sistemas para el mejoramiento de características ambientales, ahorro de energía eléctrica, etc.

Para la continuación de este proyecto actualmente disponemos de becarios trabajando en dos líneas de investigación, una profundizando el tema de utilización de las redes inalámbricas en el mejoramiento de los sistemas de control inteligente del tránsito en las ciudades y otro en el estudio y desarrollo de gateways para la interconexión de la red inalámbrica a Internet, proveyendo la posibilidad de construir aplicaciones de IoT.

#### REFERENCIAS

- [1] Luchemos por la vida - Organización No Gubernamental (<http://luchemos.org.ar>)
- [2] Digi-Key -Distribuidor de componentes electrónicos (<https://www.digikey.es/es/articulos/techzone/2017/jun/develop-lora-for-low-rate-long-range-iot-applications>)
- [3] Semtech -Proveedor líder de semiconductores analógicos y de señales mixtas de alto rendimiento y algoritmos avanzados-. (<https://www.semtech.com/lora/ecosystem/lora-alliance>)
- [4] Blog: "Alltime IOT"-Redes de Conectividad inalámbrica IoT- (<http://iot.alltimetech.com.co/blog/blog4/>)
- [5] Blog: "Medium" (<https://medium.com/beelan/haciendo-iot-con-lora-cap%C3%ADtulo-1-qu%C3%A9-es-lora-y-lorawan-8c08d44208e8>)
- [6] Aprendiendo Arduino (<https://aprendiendoarduino.wordpress.com/category/sigfox/>)
- [7] Sistema de comunicación para la implementación de redes inalámbricas de sensores, WICC 2018, pág. 982. Archivo digital: descarga y online ISBN 978-987-3619-27-4