



## **UNIVERSIDAD NACIONAL DE LOJA**

**ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES  
NO RENOVABLES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES**

**TÍTULO**

**“DESARROLLO DE UN PROTOTIPO DE PARQUEADERO INTELIGENTE PARA  
LA AUTOMATIZACIÓN DEL SISTEMA DE APARCAMIENTO SIMERT EN LA  
CIUDAD DE LOJA”**,

**TESIS DE GRADO PREVIO A  
OPTAR POR EL TÍTULO DE  
INGENIERA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**AUTORA:**

**CRISTINA STEFANÍA GODOY RAMÓN**

**DIRECTOR:**

**ING. BENJAMÍN ANDRÉS PUSAY VILLARROEL, M.I.**

**LOJA-ECUADOR**

**2015**

## CERTIFICACIÓN

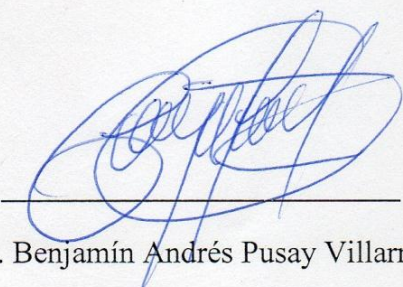
Ing. Benjamín Andrés Pusay Villarroel

DIRECTOR DE TESIS

CERTIFICA:

Haber dirigido, asesorado, revisado y corregido el presente trabajo de tesis de grado, en su proceso de investigación cuyo tema versa en “DESARROLLO DE UN PROTOTIPO DE PARQUEADERO INTELIGENTE PARA LA AUTOMATIZACIÓN DEL SISTEMA DE APARCAMIENTO SIMERT EN LA CIUDAD DE LOJA”, previa a la obtención del título de Ingeniera en Electrónica y Telecomunicaciones realizado por la señorita egresada: Cristina Stefanía Godoy Ramón, la misma que cumple con la reglamentación y políticas de investigación, por lo que autorizo su presentación y posterior sustentación y defensa.

Loja, Octubre de 2015



Ing. Benjamín Andrés Pusay Villarroel

DIRECTOR DE TESIS


CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL, Y PUBLICACIÓN

Yo **CRISTINA STEFANÍA GODOY RAMÓN** declaro ser autora del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicional acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi tesis en el Repositorio Institucional-Biblioteca Virtual.

**Autora:** Cristina Stefanía Godoy Ramón

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información y comunicación del exterior, con las cuales tenga convenio la Universidad.

**Firma:**  .....

**Cédula:** 1104731144

**Fecha:** 26 de Noviembre de 2015

## CARTA DE AUTORIZACIÓN AL PENSAMIENTO

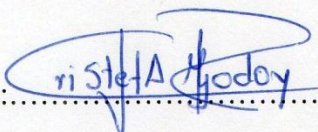
**CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL, Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.**

Yo, **CRISTINA STEFANÍA GODOY RAMÓN**, declaro ser autora de la tesis titulada: **“DESARROLLO DE UN PROTOTIPO DE PARQUEADERO INTELIGENTE PARA LA AUTOMATIZACIÓN DEL ACTUAL SISTEMA DE APARCAMIENTO SIMERT EN LA CIUDAD DE LOJA”** como requisito para optar al grado de: **INGENIERA EN ELECTRÓNICA Y TELECOMUNICACIONES**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los veintiséis días del mes de Noviembre del dos mil quince, firma la autora.

Firma:  .....

**Autor:** Cristina Stefania Godoy Ramón

**Cédula:** 1104731144

**Dirección:** Loja, Av. Chuquiribamba y Diego de Troya

**Correo Electrónico:** crisgodoy92@hotmail.com

**Teléfono:** 2542450 **Celular:** 0995092058

### DATOS COMPLEMENTARIOS

**Director de Tesis:** Ing. Benjamín Andrés Pusay Villarroel, M.I.

**Tribunal de Grado:** Ing. Diego Vinicio Orellana Villavicencio, Mg. Sc.

Ing. Luis Enrique Chuquimarca Jiménez, M.I.

Ing. Diego Fernando Carrera Moreno, Mg.

## **PENSAMIENTO**

**“QUIEN NO ESPERA VENCER, YA ESTÁ VENCIDO”**

José Joaquín de Olmedo

Victoria de Junín. Canto a Bolívar

## **DEDICATORIA**

A todos aquellos que creyeron en mí y me dieron una mano, levantaron mis ánimos y fueron mi apoyo.

A Carlos y a mi hermosa hija Monserrath por ser la columna vertebral de mi existencia.

A mi madre, sin su apoyo no habría podido alcanzar esta meta.

A la familia Morocho Pineda por toda su comprensión y ayuda desinteresada.

## **AGRADECIMIENTO**

A Dios por darme el don de la vida y dejarme disfrutar de las maravillas de la misma.

A mi amor Carlos, por todo el apoyo que me ha brindado, por ser el motivo de mi felicidad, y por darme la mayor inspiración de esta vida: nuestra hija Monserrath, a quien también agradezco por ser el motor que me mueve cada día.

A mis profesores y director de tesis quienes han sido el pilar fundamental de conocimientos

A mi familia, amigos y amigas por siempre estar a mi lado brindándome apoyo e impulsándome a seguir adelante.

## TABLA DE CONTENIDOS

CERTIFICACIÓN.....	I
AUTORÍA .....	II
CARTA DE AUTORIZACIÓN.....	III
PENSAMIENTO.....	IV
DEDICATORIA.....	V
AGRADECIMIENTO.....	VI
TABLA DE CONTENIDOS.....	VII
ÍNDICE DE FIGURAS .....	X
ÍNDICE DE TABLAS.....	XII
SIMBOLOGÍA.....	XIII
a.-TÍTULO.....	1
b. RESUMEN.....	2
SUMARY.....	3
c.- INTRODUCCIÓN.....	4
d.- REVISIÓN DE LITERATURA.....	6
d.1 CAPÍTULO I:.....	6
ESTADO DEL ARTE.....	6
d.1.1 SMART CITIES.....	6
d.1.1.1 ¿QUÉ SON LAS SMART CITIES? .....	7
d.1.1.2 CARACTERÍSTICAS.....	9
d.1.1.3 TECNOLOGÍAS PARA LAS SMART CITIES.....	10
d.1.2 SMART PARKING .....	13
d.1.2.1 ANTECEDENTES.....	13
d.1.2.2 CARACTERÍSTICA DEL SMART PARKING.....	14
d.1.3 SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT) .....	15



d.1.3.1 SITUACIÓN LEGAL .....	15
d.1.3.2 COORDINACIÓN DEL SIMERT .....	17
d.1.3.3 HORARIO DEL SISTEMA Y SANCIONES .....	17
d.1.3.4 COMPONENTES DEL SIMERT .....	19
d.1.3.5 MOVILIDAD Y APARCAMIENTO EN LA CIUDAD .....	20
d.2 CAPÍTULO II: .....	21
COMPONENTES DEL PROTOTIPO.....	21
d.2.1 DETECCIÓN VEHICULAR .....	21
d.2.1.1 LOOP DE PISO .....	23
d.2.2 MÓDULO RFID .....	25
d.2.3 INTERACCIÓN CON EL USUARIO (teclado y módulos de visualización).....	27
d.2.4 ARDUINO MEGA.....	28
d.2.5 DS1307 - RELOJ EN TIEMPO REAL.....	30
d.2.6 COMUNICACIÓN INALÁMBRICA. ....	30
d.2.6.1 ESTÁNDAR ZIGBEE .....	32
d.2.6.2 SEGURIDAD.....	33
d.2.6.3 MÓDULO XBEE.....	33
d.2.7 ALIMENTACIÓN .....	34
d.3 CAPÍTULO III:.....	36
CONSTRUCCIÓN DEL PROTOTIPO Y FUNCIONAMIENTO. ....	36
d.3.1 ELABORACIÓN DEL SOFTWARE Y HARDWARE.....	36
d.3.1.2 UBICACIÓN DE PULSADORES PARA ELECCIÓN DEL TIEMPO DE ESTACIONAMIENTO.....	37
d.3.1.3 SISTEMA DE VISUALIZACIÓN. ....	37
d.3.1.4 DIAGRAMA DEL DISPOSITIVO DE CONTROL. ....	38
d.3.2 FUNCIONAMIENTO.....	39
d.3.2.1 SISTEMA DE VISUALIZACIÓN. ....	39
d.3.2.2 PROCESO QUE DEBE SEGUIR EL USUARIO .....	40
d.3.2.3 PROCESO GENERAL. ....	40
e.- MATERIALES Y MÉTODOS.....	48
e.1 MATERIALES.....	48

e.1.1 MATERIALES PARA LA CONSTRUCCIÓN DEL PROTOTIPO.....	48
e.1.2 LISTA DE SOFTWARES USADOS PARA LA CONFIGURACIÓN DEL PROTOTIPO.....	50
e.2 MÉTODOS.....	50
f.- RESULTADOS .....	58
f.1 RESULTADOS CON EL SISTEMA DE VISUALIZACIÓN.....	58
f.2 RESULTADOS CON EL DETECTOR DE VEHÍCULOS. ....	59
f.3 RESULTADOS CON EL DISPOSITIVO DE CONTROL .....	60
f.4 RESULTADOS CON EL CENTRO DE CONTROL. ....	61
g.- DISCUSIÓN .....	63
h.- CONCLUSIONES .....	64
i.- RECOMENDACIONES .....	65
j.- BIBLIOGRAFÍA .....	67
k. ANEXOS .....	69
k.1 ANEXO 1. TABLA DE PRESUPUESTO DEL PROTOTIPO. ....	69
k.2 ANEXO 2. INSTRUCTIVO PARA LA CONSTRUCCION DEL LOOP .....	71
k.3 ANEXO 3. DESCRIPCIÓN DEL MÓDULO XBEE PRO 900 .....	74
k.4 ANEXO 4. ALGORITMO DEL ARDUINO.....	76
K.5 ANEXO 5. ALGORITMO PARA CONTROL DEL TECLADO MATRICIAL .....	94
k.6 ANEXO 6. UBICACIÓN DE LOS ELEMENTOS QUE COMPONEN EL PROTOTIPO DE PARQUEADERO INTELIGENTE.....	95

## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Actores en el entorno Smart Parking .....	15
<b>Figura 2.</b> SIMERT en los alrededores del Mercado Gran Colombia (El Mayorista).....	16
<b>Figura 3.</b> Campo magnético generado en el devanado del sensor.....	22
<b>Figura 4.</b> Salida del oscilador en presencia de objetos metálicos y sin presencia de estos. ....	23
<b>Figura 5.</b> Loop detector PD132 .....	25
<b>Figura 6.</b> Pantalla LCD 20x4.....	27
<b>Figura 7.</b> Teclado matricial 4x4.....	28
<b>Figura 8.</b> Parte frontal del Arduino Mega 2560. ....	28
<b>Figura 9.</b> Topologías en redes 802.15.4 .....	32
<b>Figura 10.</b> Módulo Xbee y su antena .....	34
<b>Figura 11.</b> Ubicación de los dispositivos de control en la calle Colón. ....	36
<b>Figura 12.</b> Ubicación de los pulsadores.....	37
<b>Figura 13.</b> Simulación en Proteus del sistema de visualización.....	38
<b>Figura 14.</b> Diagrama del dispositivo de control .....	39
<b>Figura 15.</b> Mensaje que aparece cuando el dispositivo está en reposo .....	41
<b>Figura 16.</b> Datos que identifican al usuario del SIMERT .....	41
<b>Figura 17.</b> Mensaje que pide el número de plaza ocupada.....	42
<b>Figura 18.</b> Mensaje que aparece para que el usuario elija el tiempo.....	42
<b>Figura 19.</b> Mensaje para que el usuario confirme o cancele el tiempo que eligió.....	43
<b>Figura 20.</b> Mensaje que aparece cuando ha terminado el registro de un usuario.....	44
<b>Figura 21.</b> Mensaje para que el usuario no olvide retirar su tarjeta .....	44
<b>Figura 22.</b> Mensaje que aparece cuando el usuario no cuenta con saldo .....	45
<b>Figura 23.</b> Mensaje de cancelación del registro de un usuario.....	45
<b>Figura 24.</b> Mensaje que indica tarjeta desautoriza .....	46
<b>Figura 25.</b> Aviso de que no hay sistema.....	46
<b>Figura 26.</b> Mensaje para que el usuario corrija el lugar donde se estacionó.....	46
<b>Figura 27.</b> Mensaje de error cuando se elige mal el lugar donde se aparcó el vehículo. ....	47
<b>Figura 28.</b> Simulación en Proteus del teclado matricial .....	51
<b>Figura 29.</b> Diagrama de flujo (Parte 1).....	53
<b>Figura 30.</b> Diagrama de flujo (Parte 2).....	54

<b>Figura 31.</b> PCB del dispositivo de control. ....	55
<b>Figura 32.</b> Elementos soldados y acoplados en la baquelita. ....	56
<b>Figura 33.</b> Letrero LED .....	57
<b>Figura 34.</b> Sistema de visualización finalizado .....	58
<b>Figura 35.</b> Enrollamiento de alambre (Loop).....	59
<b>Figura 36.</b> Dispositivo de control finalizado .....	60
<b>Figura 37.</b> Interfaz del Centro de Control .....	62
<b>Figura 38.</b> Perforación en el piso para el Loop .....	73
<b>Figura 39.</b> Loop para vehículos livianos. ....	73
<b>Figura 40.</b> Loop para vehículos pesados. ....	73

## ÍNDICE DE TABLAS

<b>Tabla 2.</b> Principales tecnologías inalámbricas.....	11
<b>Tabla 3.</b> Horario y límites del sistema.....	18
<b>Tabla 4.</b> Sanciones a imponer.....	18
<b>Tabla 6.</b> Materiales usados para la construcción del dispositivo de control.....	48
<b>Tabla 7.</b> Materiales usados en la construcción del dispositivo de control.....	49
<b>Tabla 8.</b> Programas.....	50
<b>Tabla 9.</b> Detalle del costo del dispositivo de control.....	69
<b>Tabla 10.</b> Detalle del costo del sistema de visualización. ....	70
<b>Tabla 11.</b> Construcción del Loop en función del tamaño.....	71

## SIMBOLOGÍA

CA	Corriente alterna
CC	Corriente continua
EEPROM	Electrically Erasable Programmable Read-Only Memory – ROM programable y borrrable electricamente
ICSP	In Circuit Serial Programming
IEEE	Institute of Electrical and Electronics Engineers – instituto de ingenieros eléctricos y electrónicos
LCD	Liquid Crystal Display – pantalla de cristal líquido
RFID	Radio Frequency IDentification – identificación por radiofrecuencia
SIMERT	Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado
SRAM	Static Random Access Memory – memoria estática de acceso aleatorio
TTL	Transistor-transistor logic – Lógica transistor a transistor
UART	Universal Asynchronous Receiver-Transmitter – transmisor receptor asíncrono universal.
PWM	Pulse width modulation - Modulación por ancho de pulsos

a.-TÍTULO

**“DESARROLLO DE UN PROTOTIPO DE  
PARQUEADERO INTELIGENTE PARA LA  
AUTOMATIZACIÓN DEL SISTEMA DE  
APARCAMIENTO SIMERT EN LA CIUDAD DE  
LOJA”**

## b. RESUMEN

En este trabajo se presenta una propuesta para la automatización del Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT). La idea nace de la necesidad de optimizar recursos en el Municipio de Loja, así como para un mejor control en el uso del sistema. Existen tres partes claves en la elaboración de este proyecto de tesis que son el dispositivo de control, el centro de control y el sistema visualizador de plazas disponibles.

El **dispositivo de control** tiene como componentes principales los siguientes: el bloque de detección vehicular (sensor y loop de piso), un módulo RFID, un módulo ZigBee, un teclado, pulsadores, una pantalla LCD, la alimentación y un Arduino Mega 2560. La pantalla LCD, los pulsadores y el teclado permiten al usuario del SIMERT interactuar con dicho dispositivo, el resto de elementos están en el interior del mismo.

El **centro de control** ha sido simulado en el programa *Visual Basic*, una de las funciones que realiza es recibir los mensajes del dispositivo de control avisando cuando se ha ocupado una plaza, información del usuario del SIMERT (nombre y apellidos), placa de su vehículo, tiempo que estará estacionado y el lugar en que aparca su vehículo; otra función es enviar mensajes al dispositivo de control, estos mensajes avisan si el cliente tiene o no saldo para estacionamiento y por ende si puede hacer o no uso del servicio de estacionamiento, también es el responsable de dar una alarma cuando ha transcurrido el tiempo de estacionamiento y el vehículo aún no se ha retirado, también se produce una alarma cuando un vehículo ha ocupado un lugar de estacionamiento y no realiza el registro del mismo luego de un tiempo de gracia.

El **sistema de visualización** de plazas disponibles está compuesto por un letrero LED y un módulo Zigbee, su función es guiar al conductor hacia espacios disponibles para estacionarse.



## SUMMARY

This work presents a proposal for the SIMERT automation. The idea comes from the necessity to optimize resources in Loja City Hall, as well as for a better control in the use of the system. There are three key issues in the development of this project that are the control device, the center control and the display system of available spaces.

The SIMERT **control device** has the following main components: the vehicle detection block (sensor and loop floor), RFID and Zigbee modules, a module, a keyboard, push buttons, LCD display, power supply and a 2560 “Arduino” Mega. The LCD screen and push bottoms and the keyboard allow the SIMERT user to interact with the device, the other elements are in its inside.

The **central control** has been simulated in the Visual Basic program, one of the functions that it performs is receive the messages of the control device warning when a space is full, information of the SIMERT user (name and last name), license plate, time that is going to be parked and the place that the vehicle is parking; another function is seconds messages to the control device, those messages notified if the client has o not credit for parking and so if it could make use or not of the parking service, also it is responsible to give an alarm when the parking time is over and the vehicle has not yet been removed, also there is an alarm when a vehicle occupied a parking place and it skips the registration after an extra time.

The **display system** of available spaces is compound by a LED signboard and a ZigBee module, the function is drive to users to available spaces for parking.

## c.- INTRODUCCIÓN

En el mundo actual en el que vivimos todo gobierno busca optimizar recursos con el mínimo esfuerzo realizado y llevar a cabo cambios necesarios para el desarrollo sostenible, el concepto de *Smart City* ha tomado más fuerza en los últimos años debido a que los gobiernos están conscientes de que deben cambiar sus modelos de gestión ya que estos nuevos modelos de ciudades es hacia donde se encamina el futuro. Por este motivo muchas empresas trabajan para ofrecer servicios integrables dentro de las *Smart Cities*.

*Smart City* es aquella ciudad en la cual se busca hacer la vida más fácil y cómoda de manera que los ciudadanos apenas puedan darse cuenta. Con la realización de este proyecto se persigue crear un sistema automatizado capaz de ordenar el flujo de vehículos en búsqueda de un lugar para estacionarse y gestionar de forma más eficaz el cobro de las tarifas fijadas por el Municipio de Loja para el uso del “Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado” (SIMERT).

Los municipios, ciudades y áreas metropolitanas han sufrido crisis al momento de brindar servicios de calidad, debido principalmente a la falta de recursos humanos generado por bajas economías y presupuestos; esto ha conllevado al planteamiento de soluciones eficientes para cada uno de estos problemas. En la actualidad existe el planteamiento de que una ciudad más eficiente, debe ser en esencia más inteligente, usando como instrumento para esto las tecnologías de la información y las comunicaciones (TICs).

Con las “*Smart Cities*” se quiere hacer que las ciudades sean más eficientes. En la ciudad de Loja se está hablando de aplicar nuevas tecnologías (medir la calidad del aire, el estado del tiempo, los servicios de alumbrado, la recolección de basura, la gestión del transporte público de bajas emisiones, control de los niveles de contaminación, los niveles del agua en los ríos, videovigilancia, trámites en línea, etc) para dinamizar y potenciar los recursos de la ciudad. Uno de estos avances de los que se habla es la aplicación de parqueaderos inteligentes para sustituir el actual sistema de parqueadero de exteriores “SIMERT”; con el propósito del crecimiento y el desarrollo sostenible de la ciudad.

Loja siempre se ha caracterizado por ser una ciudad limpia y ordenada. Es por ello la importancia de implementar sistemas que ayuden con el ordenamiento de la ciudad, esto se lograría con la automatización del sistema SIMERT. La automatización del SIMERT ayudaría a controlar de manera más eficiente los espacios disponibles para estacionarse en las calles, así como el cobro del tiempo de ocupación de los mismos, el ahorro en mano de obra humano ya que lo que se busca es implementar el nuevo sistema con un bajo coste de implementación y mantenimiento; que sea fácil de manejar y administrar.

Este trabajo deja abierta la posibilidad de implementar más complementos que sirvan para mejorarlo, así como también puede ser referente teórico para nuevas investigaciones. La implementación de este proyecto traería consigo un gran impacto económico ya que se lograría reducir gran número del personal que controla los tiempos de estacionamiento, los encargados de ir a colocar y retirar los “candados” y el personal que vende las tarjetas de estacionamiento.

Con el *Smart Parking* se lograría ahorrar tiempo de las personas al momento de buscar espacios disponibles, minimizar los gastos de energía, las emisiones de elementos contaminantes y el personal no tendría que desplazarse un gran número de veces de un lugar a otro para constatar que todo esté de acuerdo a lo establecido por el Municipio de Loja.

### **OBJETIVO GENERAL**

Desarrollar un prototipo de parqueadero inteligente para la automatización del sistema de aparcamiento SIMERT en la ciudad de Loja

### **OBJETIVOS ESPECÍFICOS**

- ✓ Analizar las condiciones en las que se desenvuelve el tráfico vehicular en los sectores en los que se aplica el sistema tarifado Municipal de aparcamiento en la ciudad de Loja.
- ✓ Diseñar y construir un prototipo de parqueadero inteligente que se ajuste a las condiciones en las que trabaja el SIMERT en la ciudad de Loja.
- ✓ Construir un sistema de señalización práctico, para guiar a los usuarios hacia plazas desocupadas.

## **d.- REVISIÓN DE LITERATURA**

### **d.1 CAPÍTULO I:**

## **ESTADO DEL ARTE**

### **d.1.1 SMART CITIES**

El término *Smart City* se lo ha empleado hace más de veinte años, este término hace referencia a un concepto de ciudad sostenible, la cual ofrece una serie de servicios y prestaciones que elevan la calidad de vida de sus habitantes, y que al mismo tiempo permite a la ciudad incrementar su competitividad y su capacidad para crecer económicamente. Es necesario actuar en diversas dimensiones o ejes para que una ciudad llegue a convertirse en una *Smart City*, pero el elemento común que permite desarrollar cada uno de esos ejes es el uso y aplicación de las Tecnologías de la Información y la comunicación (TICs). Por este motivo, el sector de las TICs debe jugar un papel protagonista en el avance hacia este nuevo modelo de ciudad, ya que estas tecnologías constituyen el principal elemento facilitador para el desarrollo de las *Smart Cities*.

Para lograr que las *Smart Cities* se conviertan en una realidad es necesario que existan los estímulos precisos con el fin de aproximarnos a contar con los recursos que esta gran transformación exige. El principal estímulo es el desarrollo urbano que ha provocado de las grandes ciudades se conviertan en inagotables fuentes de actividad económica y demográfica. En el año 1950, la población urbana en todo el planeta era aproximadamente 750 millones de personas. En el año 2011 esta cifra subió a 3.600 millones de personas lo que se supone que es el 50% de la población mundial. Se calcula que para el 2030 el 60% de la población vivirá en las ciudades.

La concentración de la población, del consumo energético, de las emisiones contaminantes, de la actividad humana en general y particularmente la económica, en una pequeña superficie del planeta, ofrece diversas ventajas y oportunidades, pero también se generan problemas de diversa índole. El modelo de *Smart City* persigue

potenciar las primeras y erradicar los segundos. Las TICs son un pilar básico para el crecimiento sostenible de todos los sectores de la economía.

Las TICs inteligentes (Smart TICs) son tecnologías más verdes y más sostenibles, que no sólo consumen menos energía, sino que ayudan a mejorar el rendimiento y el comportamiento medioambiental de la sociedad. Este desarrollo y aplicación “inteligente” de las TICs en pro de la sostenibilidad constituye la denominada *Green TIC* (TICs verdes), una de cuyas aplicaciones más destacada es precisamente el desarrollo de las Smart Cities. Es decir, este modelo de ciudades constituyen un instrumento esencial para garantizar un futuro sostenible, en el cual los recursos sean aprovechados de manera eficiente y la minimización del impacto en el medio ambiente de las actividades humanas. (Valdés Castro, 2014)

Las *Smart Cities* constituyen un importante nicho de mercado para la industria tecnológica, así como una oportunidad clave para que la industria del sector continúe impulsando el crecimiento, el empleo y la mejora de la calidad de vida. Son también una oportunidad de desarrollo gracias a todos los nuevos productos y servicios que aparecen.

Tomando en cuenta el papel protagónico que están teniendo las ciudades (compitiendo por estar en la punta en tecnología, ordenamiento, etc) en este momento de la historia, el desarrollo de las *Smart Cities* ha pasado en muy poco tiempo a ocupar un lugar destacado en la agenda política y social.

#### ***d.1.1.1 ¿QUÉ SON LAS SMART CITIES?***

Son ciudades capaces de sostenerse por sí mismas **económica, social y medioambientalmente**. Una ciudad inteligente nace de la necesidad de mantener una armonía entre estos aspectos.

*Smart City* es un término que comprende de forma integrada las iniciativas orientadas a mejorar la calidad de vida, la sostenibilidad y la gestión eficiente de los servicios, innovando en materiales, recursos y modelos usando la tecnología de forma exhaustiva. *Smart City* afecta a todos los servicios que se prestan en una ciudad, como: movilidad, producción y distribución de servicios urbanos (energía, agua, etc),

educación, salud, emergencias, seguridad, etc. *Smart City* incorpora innovación, tecnología e inteligencia (en materiales, soluciones y modelos) a las infraestructuras básicas creadas por el hombre para desarrollar una ciudad más eficiente, flexible y menos costosa (Tecnologic, 2012).

Muchas personas coinciden en decir que, a día de hoy no existe una definición oficial para describir qué es una ciudad inteligente. *Smart City* es mucho más que una simple definición; es un concepto, una vía en desarrollo y una forma de enfocar el crecimiento controlado y sostenible de las grandes urbes. Existen numerosos puntos de vista para definir qué es una *Smart City*:

- ✓ **Punto de vista ciudadano:** Se dice que una *Smart City* es aquella ciudad que le hace la vida más fácil y cómoda a sus habitantes sin que ellos apenas se den cuenta. Todo en ella está encaminado a optimizar el tiempo de sus habitantes, evitando pérdidas del mismo en tareas diarias de escaso valor.
- ✓ **Punto de vista industrial:** Es la ciudad en la cual se han minimizado los gastos de energía, las emisiones de elementos contaminantes y se han optimizando las vías de distribución de mercancías para tenerlas controlados en todo momento desde su origen hasta su destino.
- ✓ **Punto de vista tecnológico:** Es quizá el más importante, consiste en el uso de la tecnología en todos sus niveles para obtener nuevos productos innovadores que faciliten la forma de hacer que cualquier servicio o aplicación relativa a la ciudad esté conectada con un mundo global. Se trata de interconectar cada elemento para poder controlar su información y hacer uso de ella, creando de esta manera una “ciudad virtual” capaz de gestionar todos los recursos acorde con la necesidad de uso de los mismos (Herrador Muñoz, 2013).

Una ciudad inteligente usa las infraestructuras, la innovación y la tecnología, pero también requiere de una **sociedad inteligente, activa y participativa:** personas, talento, emprendedores, organizaciones colaborativas, etc. No se puede pensar en el término *Smart City* sin las personas que viven y trabajan en la ciudad, que son el motor y el alma de la misma. Las estrategias de estas ciudades son a la vez una oportunidad de negocio para las empresas innovadoras y de servicios; debido a que este es un negocio en expansión.

### *d.1.1.2 CARACTERÍSTICAS*

Una Ciudad inteligente debe proveer a través de las Tecnologías de la Información y las Comunicaciones lo siguiente:

- ✓ Un desarrollo sostenible
- ✓ Un incremento de la calidad de vida de los ciudadanos
- ✓ Una mayor eficacia de los recursos disponibles
- ✓ Una participación ciudadana activa

El informe *Ranking of European medium-sized cities* define a las *Smart Cities* como aquella ciudad en la que las TIC juegan un papel principal en uno o varios sectores. Por ejemplo si en una ciudad las actividades económicas que en ella se realizan cuentan con una importante presencia de las TIC, sería una *Smart City* desde la perspectiva de su estructura económica. Desde el punto de vista demográfico, la ciudad inteligente es la que cuenta con una población formada y capacitada, especialmente en el campo del manejo y uso de las TIC. Una ciudad será inteligente en el campo de la gobernanza si las TIC son ampliamente usadas en el gobierno de la ciudad y además sirven como canal de comunicación y de desarrollo de políticas de transparencia entre Administración y Ciudadanía. Desde la perspectiva del transporte y la movilidad, que es el tema que nos interesa, esta clase de ciudades aplican las TIC para mejorar la movilidad, el tráfico y la logística en el seno de la ciudad. Estas y las siguientes facetas están agrupadas en las siguientes seis características que debe reunir una ciudad para ser inteligente (AMETIC, 2012):

1. *Smart Economy* (economía inteligente)
2. *Smart Governance* (gobernanza inteligente)
3. *Smart Environment* (entorno inteligente)
4. *Smart People* (personas inteligentes)
5. *Smart Mobility* (movilidad inteligente)
6. *Smart Living*

### *d.1.1.3 TECNOLOGÍAS PARA LAS SMART CITIES*

#### **TECNOLOGÍAS DE SENSORES**

Los sensores permiten recopilar información de múltiples estados de la ciudad, esta información puede ser tratada posteriormente y usada para tomar decisiones de gestión. Los sensores brindan información sobre el tráfico, sobre el consumo de energía, acerca de las necesidades de iluminación, etc. Estos sensores pueden ser integrados ya sea en la infraestructura urbana (como los situados en las vías públicas o en los sistemas de distribución de agua o de energía); o pueden ser aportados por los propios ciudadanos (ya sea en los vehículos, en los terminales móviles o en cualquier otro elemento fijo o móvil de propiedad privada y que se encuentre conectado a la red de comunicaciones de la ciudad).

Los sensores pueden utilizarse para medir diversas variables como consumos o flujos de agua y energía; pueden medir magnitudes asociadas a niveles de seguridad; pueden registrar temperatura, humedad, velocidad y dirección del viento o presión atmosférica; pueden medir velocidades o intensidades de tráfico, o niveles de ocupación de vías públicas o de medios de transporte; pueden medir posiciones o movimientos. Todos estos sensores pueden ubicarse en la infraestructura de la ciudad, en medios públicos de transporte, en vehículos o sistemas privados, o en los teléfonos móviles y *smartphones* que portan la mayoría de los habitantes de la ciudad (AMETIC, 2012).

Los terminales móviles, específicamente los *smartphones*, constituyen una clase de dispositivos de especial interés en el modelo de Smart City. Debido a su popularidad y la elevada penetración de este dispositivo en la ciudad, este dispositivo es capaz de generar grandes volúmenes de datos que, una vez agregados y procesados, pueden ofrecer una información de mucho valor y en tiempo real, de la situación de la ciudad. Por ejemplo los *smartphones* que cuentan con cámaras y capacidad de indicar la posición en la que se encuentran, están en disposición de transmitir, a través de las redes de comunicación inalámbrica, información útil para que la plataforma de servicios de la ciudad tome decisiones orientadas a optimizar servicios como la movilidad, el alumbrado público o la distribución de efectivos de seguridad o de atención sanitaria.



En este proyecto se va a usar un sensor llamado “Loop de piso”, este sensor cumple la misión de detectar masas metálicas. Con este sensor lo que se quiere es poder conocer cuándo una plaza de estacionamiento es ocupada y de esta manera poder informar dicho evento al Centro de Control del SIMERT. El capítulo 2 se detallará el funcionamiento de este sensor y se explicará por qué se lo está usando.

## 🚦 REDES DE COMUNICACIONES

En una *Smart City* las redes de comunicaciones permiten la interconexión de todos los sistemas que las integran, y específicamente facilitan la recolección de datos obtenidos de los sensores. La infraestructura de comunicaciones de la ciudad incluye tanto redes fijas como redes móviles de banda ancha. En las *Smart City* son de mayor importancia las redes móviles o inalámbricas ya que permiten la conexión de vehículos, dispositivos móviles y personas. Como se indicó anteriormente el Smartphone es un dispositivo muy importante y está llamado a convertirse en el sensor por excelencia de las ciudades inteligentes. La tabla 2 muestra las principales tecnologías inalámbricas aplicadas a las redes de estas ciudades:

*Tabla 1.* Principales tecnologías inalámbricas.

	Nivel físico y MAC	Radio de acción	Tasa de bits	Consumo
<b>ZigBee</b>	802.15.4-2003 DSSS CSMA-CA	10-100m Interior 1Km exterior	250Kbps (2.4GHz) 20Kbps (868MHz) 40Kbps (915MHz)	Consumo pico 50mW (2.4GHz) En suspensión <1µW
<b>Wavenis</b>	Propietario	200m interior 1Km LOS	Desde 10Kbps hasta 100Kbps	18mA Rx 45mA Tx y 2µA en suspensión
<b>Wireless MBus</b>	EN 13757-4:2005	60-80m interior 500m exterior sin obstáculos	Desde 16Kbps-66Kbps hasta 100Kbps	22mA Rx 37mA Tx y 0.2µA en suspensión
<b>Z-Wave</b>	Propietario	30m interior 100m exterior	40-100 Kbps	20mA Suspensión: 1µA
<b>WiFi low</b>	802.11b/g	50-70m interior	1/2/5.5/11 Mbps	60mW

<b>power</b>	DSSS CSMA/CD	<300m exterior		Suspensión: 5μW
<b>WiMax</b>	Se basa en IEEE 802.16	Hasta 75Km	Hasta 75 Mbps	230mW-49μW
<b>PLC Watteco</b>	Línea eléctrica	50m (objetivo: 150m)	10Kbps (objetivo: 40Kbps)	Interior a ZibBee y Z-Ware
<b>PLC NEC</b>	Línea eléctrica		100bps-30Kbps	25mW
<b>GSM/GPRS</b>			Hasta 85.6Kbps	Modo en reposo 2.6mA,

[Fuente: (AMETIC, 2012)]

En este prototipo para la automatización del SIMERT es necesaria una red de comunicaciones inalámbrica, por sus excelentes características la tecnología que se va a usar en esta red de comunicaciones inalámbricas es ZigBee.

#### **SISTEMAS DE ALMACENAMIENTO Y TRATAMIENTO DE LA INFORMACIÓN.**

El almacenamiento y la capacidad de procesamiento, permiten explotar los grandes volúmenes de información generados por las redes de sensores. Las técnicas conocidas como *Big Data* permiten agregar los enormes flujos de información que proceden de la vasta red de sensores de una ciudad, y transformarlos en conocimiento útil y aplicable para tomar decisiones de gestión de los servicios de la *Smart City*.

#### **PLATAFORMAS DE SERVICIOS**

Las plataformas de prestación de servicios permiten a los operadores de los diferentes servicios (energía, movilidad, turismo, etc.) prestar los servicios de su competencia de manera eficiente mediante el adecuado aprovechamiento de la información puesta a su disposición por la infraestructura TIC de la *Smart City*.

Por ejemplo la empresa *Living PlanIT*, con su sistema operativo urbano denominado "*Urban Operating System*", dotan a estas plataformas de las capacidades necesarias para acceder a datos, realizar transacciones y pagos, almacenar información, autenticar usuarios, etc.

## d.1.2 SMART PARKING

En toda ciudad, especialmente en las grandes, es común el problema de encontrar aparcamiento tanto en espacios públicos como en los privados. Buscar un aparcamiento produce en el usuario desatención hacia la tarea de conducción, lo que se traduce en un aumento de la inseguridad al volante. Así mismo se produce un incremento en el vertido a la atmosfera de gases contaminantes, y a su vez, incrementa la congestión de vehículos.

De lo anterior podemos decir que no existe nada positivo en la tarea de buscar un aparcamiento. Es por esto que nace la necesidad de ayudar al conductor a encontrar en tiempo real un espacio disponible para estacionar su vehículo.

### d.1.2.1 ANTECEDENTES

Para lograr una movilidad eficiente se requiere de acciones integrales y transversales, que reduzca el uso de energía fósil y la contaminación atmosférica. Diversas ciudades están impulsando acciones al uso integrado del medio de transporte público, el impulso a usar la bicicleta como medio de transporte, se ha dado impulso al uso del vehículo eléctrico, etc.

*Smart Parking* es un sistema de estacionamientos que utiliza soluciones de alta tecnología y que han sido probadas alrededor del mundo; esto va de la mano de un meticuloso servicio al cliente, que mejora la experiencia del usuario al estacionarse. Entre las finalidades del *Smart Parking* tenemos desahogar a las ciudades del tráfico, disminuyendo así la contaminación vial; ayudar al usuario a encontrar lugares de estacionamiento rápidamente, optimizando su tiempo, etc.

A continuación se citan dos proyectos encaminados a mejorar la movilidad en las ciudades con diferentes enfoques (Tecnologic, 2012):

- ✓ **LIVE (Logística per a la implementació del Vehicle Elèctric):** Barcelona es un referente en movilidad urbana, tanto por su red de transporte integrada, como el éxito del proyecto *Bicing* de bicicletas públicas. Y ahora está también liderando

el cambio de modelo de la implantación del vehículo eléctrico con el proyecto **LIVE**. Una plataforma público-privada que pretende respaldar y promover el desarrollo de la movilidad eléctrica en la ciudad y en el área metropolitana de Barcelona. Entre las iniciativas que incluye este proyecto cabe destacar la creación de 234 puntos de recarga repartidos por el territorio geolocalizados a través de un servicio basado en *Google Maps* que se combina con la tarjeta del vehículo eléctrico LIVE que sirve como identificación a la vez que proporciona prestaciones sociales como beneficios fiscales, estacionamiento gratuito o reducción en los peajes.

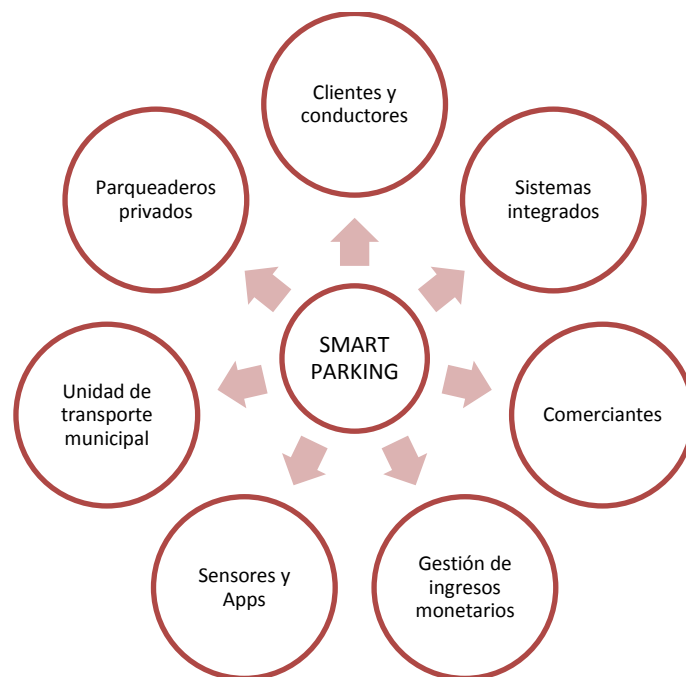
- ✓ **Smart Park:** Otras ciudades abordan el problema del transporte desde otra perspectiva: minimizar el tiempo de búsqueda de aparcamiento libre y, como resultado, reducir la generación de CO<sub>2</sub>. El proyecto piloto SFpark13 en San Francisco proporciona información de las plazas disponibles en la ciudad en tiempo real accesible desde un portal y desde el móvil.

#### ***d.1.2.2 CARACTERÍSTICA DEL SMART PARKING***

Un problema de gran magnitud en las grandes ciudades, es encontrar un lugar para estacionarse, por lo general para encontrar una plaza desocupada es necesario recorrer determinado tiempo hasta lograrlo; como solución a este problema de la movilidad urbana nace el aparcamiento inteligente o *Smart Parking*.

En el congreso mundial de transporte inteligente, se indicó que el futuro de la transportación está en la movilidad y la comunicación entre automóviles, así la gente puede revisar y reservar su propia plaza de aparcamiento y ser llevado directamente a su destino por la ruta más rápida, basándose en datos obtenidos en tiempo real, siempre y cuando se cuente con un dispositivo móvil conectado a la red de comunicaciones de la ciudad. Se plantea un entorno de aparcamiento inteligente que está compuesto por actores claves en el sistema, como lo son: conductores, estacionamientos, ciudades, centros comerciales, universidades y aeropuertos. Esto, con la finalidad de reunirlos con los aspectos técnicos como los sensores, aplicaciones móviles, redes de comunicaciones, bases de datos, etc (OECD, ECMT, 2006).

En la figura 1 se muestra un ecosistema saludable de Smart Parking y los elementos con los que deben converger para considerarlo eficiente.



**Figura 1.** Actores en el entorno Smart Parking

[Fuente: (Yaguana Tapia, y otros, 2014)]

### **d.1.3 SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT)**

#### ***d.1.3.1 SITUACIÓN LEGAL***

El 15 de Marzo de 2002 el Concejo Cantonal en base a la Ley de Régimen Municipal aprueba la Ordenanza de Creación del Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja. Este nuevo sistema de estacionamiento (SIMERT) comenzó a funcionar a partir del 3 de Mayo de 2002. Teniendo como objetivo el siguiente:

“Devolver a todos los usuarios el derecho a utilizar la vía pública en forma organizada y ordenada, a través de la generación de una oferta permanente y continua de espacios libres para estacionamiento, además de reducir la contaminación ambiental provocada por el flujo continuo de vehículos”.

Inicialmente el SIMERT funcionaba en las siguientes zonas de la ciudad de Loja:

- ✓ De manera continua: en el Centro de la Ciudad y Terminal Terrestre (Zona “A”)
- ✓ De manera temporal: en el Complejo Ferial (Zona “B”)

El 30 de Junio de 2014 se reformó en SIMERT en nuestra ciudad y dentro de los cambios realizados se establecen dos zonas de estacionamiento (Diario La Hora, 2015):

- ✓ La zona tipo “A” con los siguientes límites: por el norte hasta la calle Imbabura, por el sur la Lourdes, al oriente la Juan José Peña y al occidente la Avenida Universitaria, en esta zona el tiempo máximo de estacionamiento es de 90 minutos y el costo por hora es de 50 centavos.
- ✓ La zona tipo “B” con los siguientes límites: por el norte desde la calle Imbabura hasta la puerta de la ciudad, por el sur desde la Lourdes hasta la Catacocha, al oeste las calles Lauro Guerrero y Ramón Pinto desde la Mercadillo hasta la Colón, y por el este la Juan José Peña, 24 de Mayo, Emiliano Ortega y Macará; en esta zona el tiempo máximo de estacionamiento es de 120 minutos y el costo por hora es de 25 centavos.



**Figura 2.** SIMERT en los alrededores del Mercado Gran Colombia (El Mayorista)

[Fuente: *Diario la Hora*, 20 de Mayo de 2013]

Para poder diferenciar entre las dos zonas, estas se encuentran pintadas con diferentes colores, color naranja el área con valor de 50 centavos (zona “A”) y de color azul el área con valor de 25 centavos (zona “B”). Esta última zona se localiza al otro

lado del río Malacatos, es decir por la calle Lauro Guerrero y en las inmediaciones del mercado Gran Colombia (El Mayorista). El titular de esta dependencia, Diego Guerrero, Gerente Municipal de la Unidad de Tránsito aclaró en una rueda de prensa que para hospitales y clínicas se otorgará dos espacios de reserva, mientras que un espacio a los parqueaderos que tengan más de 9 vehículos de capacidad, estas zonas están pintadas de color blanco. También mencionó que las motos no deben pagar el SIMERT y que hay espacios de estacionamiento para estas en todas las cuadras (Díaz, 2014).

Cuando arrancó el SIMERT en el 2002 habían 1085 espacios de estacionamiento en las zonas que en ese entonces contaban con este sistema de estacionamiento tarifado, con la nueva reforma existen en la actualidad 1974 espacios de estacionamiento en la zona centro de la ciudad (Díaz, 2014).

#### ***d.1.3.2 COORDINACIÓN DEL SIMERT***

Antes de Enero de 2008 el SIMERT estaba bajo la administración de la Unidad de Tránsito Municipal, pero actualmente funciona como un área independiente. Sus oficinas están situadas en las calles Bernardo Valdivieso y Colón esquina, en el edificio perteneciente a la Policía Municipal.

El SIMERT cuenta con un personal de 50 personas entre los cuales se tiene controladores, inspectores y administrativos; este personal a más de cumplir con su labor de velar por el correcto uso de las tarjetas, colabora con la seguridad de los vehículos, esto lo pueden hacer gracias a que ellos cuentan con un sistema de comunicación permanente que les permite solicitar ayuda en caso de ser necesario.

#### ***d.1.3.3 HORARIO DEL SISTEMA Y SANCIONES***

El tiempo máximo de estacionamiento continuo es de 3 horas. Para poder hacer uso del SIMERT los clientes deben adquirir una TARJETA PREPAGO cuyo valor es de \$ 1.50, más adelante se especificarán los lugares en los cuales se la puede adquirir.

El área del SIMERT está compuesta por 100 manzanas dentro de las cuales existen 174 cuadras aptas para poder aparcarse. Estas zonas están bien señalizadas pintadas con los colores distintivos de cada zona y que previamente ya se los especifico, además existen letreros que indican el horario de funcionamiento del SIMERT. En la

tabla 3 se muestra el horario y límites (calles) del SIMERT de acuerdo a las zonas con las que se trabaja:

**Tabla 2.** Horario y límites del sistema.

ZONA A	ZONA B
LUNES A VIERNES DE 08H00 A 19H00 SÁBADO DE 09H00 A 13H00	LUNES A DOMINGO DE 09H00 A 24H00
<b>LÍMITES DE LA ZONA A</b>	<b>LÍMITES DE LA ZONA B</b>
AL SUR: calle Catacocha	AL SUR: Av. Isidro Ayora
AL ESTE: calle Juan José Peña	AL ESTE: Salvador Bustamante Celi
AL NORTE: calle José Félix de Valdivieso	AL NORTE: calle Miguel Cano Madrid
AL OESTE: calle Ramón Pinto	AL OESTE: calle Velazco Ibarra

Fuente: Alcaldía de Loja

El municipio concede 15 minutos de gracia, establecidos en la misma ordenanza con la que se creó el SIMERT y es aplicable luego de utilizado y finalizado el tiempo que el usuario haya registrado en la tarjeta. Una vez consumidos estos tiempos se procederá a imponer la sanción correspondiente, según se muestra en la tabla 4.

Cuando se ha impuesto una sanción a un usuario, este debe acercarse a la ventanilla de cobranzas del Municipio de Loja con el papel que le entrega la persona quien le impuso la multa y hacer el pago correspondiente según el tipo de infracción que ha cometido. Si la sanción incluyó la inmovilización del vehículo, una vez realizado el pago de la multa se envía al personal hacia el lugar donde se encuentra el vehículo sancionado y se procede a retirar el candado inmovilizador.

**Tabla 3.** Sanciones a imponer.

<b>SANCIONES</b>	
<b>Motivo</b>	<b>Valor</b>
<b>Ausencia de tarjeta</b>	\$ 10.00
<b>Alteración de tarjeta</b>	\$ 10.00
<b>Retraso de 16 a 30 min</b>	\$ 3.00



<b>Retraso de 31 a 60 min</b>	\$ 5.00 e inmovilización
<b>Retraso de 61 a 120 min</b>	\$ 10.00 e inmovilización
<b>Retraso de 121 min</b>	\$ 40.00 y remolque grúa

Fuente: Alcaldía de Loja

Se puede adquirir las tarjetas del SIMERT y realizar el pago de las multas en las siguientes ventanillas:

- ✓ Ilustre Municipio de Loja, ventanillas de Recaudaciones (calles Bolívar y José Antonio Eguiguren)
- ✓ Distrito el Valle (calle Guayaquil entre Av. Salvador Bustamante Celi y Cuenca).
- ✓ Oficina de recaudación en las inmediaciones del Complejo Ferial
- ✓ Coliseo Ciudad de Loja, Distrito Sur (Av. Iberoamérica y calle Brasil)
- ✓ Terminal terrestre (Av. Isidro Ayora y Av. 8 de Diciembre)

#### ***d.1.3.4 COMPONENTES DEL SIMERT***

- Dispositivos de cobro
  - ✓ Tarjetas
- Medio de distribución de tarjetas
  - ✓ Kioscos atendidos por personal con discapacidad en su mayoría
  - ✓ Tiendas autorizadas
- Equipo de control
  - ✓ Fichas de control
  - ✓ Radios (En el SIMERT usan dos tipos de radios: Motorola pro 5150-terminal portátil y Motorola 5100-terminal móvil)
  - ✓ Motos
  - ✓ Vehículos
- Dispositivos de sanción
  - ✓ Candado inmovilizador
  - ✓ Libretines de multas
- Cobro de multas

- ✓ Ventanillas de recaudaciones del Municipio, se debe acercar con el comprobante de pago que emite el controlador con la muta respectiva.
- Personal operativo
  - ✓ Recaudaciones del Municipio
  - ✓ UMTT-L (Unidad Municipal de Tránsito y Transporte de Loja)

#### ***d.1.3.5 MOVILIDAD Y APARCAMIENTO EN LA CIUDAD***

El parque automotor en nuestra ciudad está creciendo en forma acelerada, y lo que es peor, este se concentra en el centro de la ciudad lo que provoca que se reduzca la capacidad de movilidad de los ciudadanos y de espacios de estacionamiento; lo que se transforma en verdaderos cuellos de botella en las calles, especialmente en las horas pico.

Con el crecimiento del parque automotor de nuestra ciudad se crea un aumento en la demanda de estacionamientos y también se aumenta la centralización de la ciudad, esto se debe a que la mayor parte de locales comerciales y entidades públicas y privadas se encuentran en el casco céntrico, y realmente en nuestra ciudad no se cuenta con una buena infraestructura de estacionamientos ya sean de interiores o exteriores para satisfacer todas las demandas de estacionamiento de los ciudadanos. Lo que provoca una grave incomodidad de los usuarios del SIMERT.

En Loja existen lugares de aparcamiento los cuales se ubican de la siguiente manera: en la vía pública, edificios de parqueo y lotes de aparcamiento públicos o privados.

El hecho de que el SIMERT esté en funcionamiento en nuestra ciudad ha ayudado en parte a disminuir el tráfico, pero no es una solución permanente ni lo suficientemente efectiva, ya que se necesitan reformas del mismo.

## d.2 CAPÍTULO II:

# COMPONENTES DEL PROTOTIPO

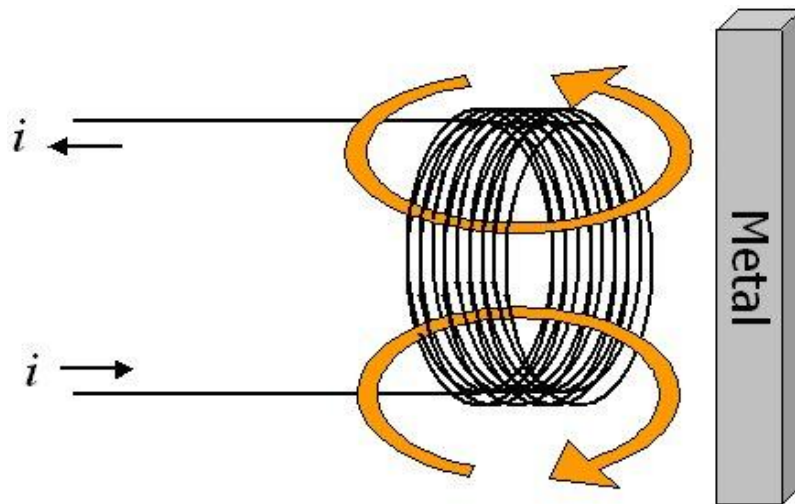
### d.2.1 DETECCIÓN VEHICULAR

La detección del vehículo es un punto clave del Smart Parking, la relevancia que toma esta acción es porque se necesita saber el momento en el que llega un vehículo para de esta manera saber cuándo una plaza ha sido ocupada y empezar a cronometrar el tiempo que ese vehículo está estacionado.

Un detector de vehículos ideal en un sistema inteligente de control de tráfico debe ser autónomo, compacto, fácil de usar e instalar, con bajo consumo de energía, que no necesite mantenimiento y sea robusto a factores climáticos. Los detectores actuales basados en cámara de video, ultrasonido e infrarrojo no cumplen los requisitos mencionados anteriormente, especialmente en aplicaciones al aire libre. Pero los sensores basados en lazo inductivo son una buena opción para trabajar bajo estas condiciones y su aplicación más usual es en la detección vehicular.

Un sensor de Flujo vehicular de Lazo Inductivo se basa en el fenómeno físico de la *inducción electromagnética*, está compuesto de una o más vueltas de un cable aislado que debe estar insertado en el pavimento. Cuando un vehículo lo atraviesa o se detiene encima de este, se inducen corrientes en el vehículo y disminuye la inductancia del lazo, esto trae como consecuencia el desbalance del circuito del cual este inductor es parte (Diseño de un sensor de Flujo Vehicular basado en Lazo Inductivo, 2012).

***Los sensores inductivos son de una clase especial de sensores que sirven para detectar materiales metálicos ferrosos.*** Los sensores de proximidad inductivos contienen un devanado interno. Así que cuando una corriente circula por el mismo, un campo magnético es generado, que tiene la dirección de las flechas naranja (figura 3). Cuando un metal es acercado al campo magnético generado por el sensor de proximidad, éste es detectado (Concepto General de los Sensores, 2009).

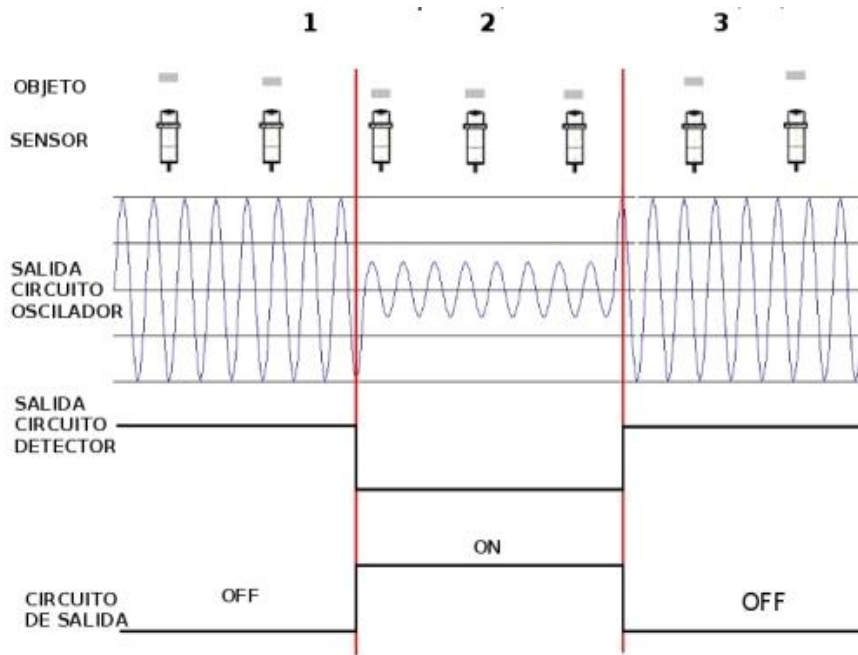


**Figura 3.** Campo magnético generado en el devanado del sensor

[Fuente: (Concepto General de los Sensores, 2009)]

La bobina del sensor inductivo *induce corrientes* en el material a detectar. Estas, a su vez, generan un campo magnético que se opone al de la bobina del sensor, causando una reducción en la inductancia de la misma. Esta reducción en la inductancia de la bobina interna del sensor, trae aparejado una disminución en la impedancia de ésta (Concepto General de los Sensores, 2009).

En resumen, el circuito detector reconocerá el cambio en la impedancia de la bobina del sensor (debido a las corrientes introducidas en el objeto a detectar) y enviará una señal al amplificador de salida, el cual cambiará el estado de la misma. Cuando el metal a detectar es removido de la zona de detección, el oscilador podrá generar nuevamente el campo magnético con su amplitud normal, esto se puede apreciar en la figura 4. Es en este momento en que el circuito detector nuevamente detecta este cambio de impedancia y envía una señal al amplificador de salida para que sea éste quien, nuevamente, restituya el estado de la salida del sensor.



**Figura 4.** Salida del oscilador en presencia de objetos metálicos y sin presencia de estos.

[Fuente: (Concepto General de los Sensores, 2009)]

#### **d.2.1.1 LOOP DE PISO**

El sistema a usarse en la detección del vehículo, es el denominado “detector de metales”, más específicamente un **Loop de piso**. Un detector de metales es un circuito sensible a las variaciones del campo magnético de una bobina, cuando un objeto metálico entra dentro de su área. Para que este principio se pueda aplicar debemos ser capaces de identificar esas alteraciones de campo magnético y procesarlas hacia un componente de salida (por ejemplo el movimiento de una aguja en un medidor), de esta manera podemos ubicar dónde se encuentra un objeto metálico.

El loop de piso está diseñado para detectar en forma eficiente y segura la presencia de vehículos en un área específica. Es apto para integrarse a un sistema o controlar la apertura y cierre de una barrera en forma autónoma. También es ideal para aplicaciones con semáforos de garaje, cortinas/puertas industriales automáticas, detección de sentido de circulación, sistemas de control de acceso, etc.

Este sistema está compuesto de dos partes: el LOOP y el DETECTOR.

El **Loop** es un arrollamiento continuo de cable ubicado en la zona que se desea controlar. Los dos extremos del Loop se conectan al detector para cerrar el circuito. El **detector** genera en el loop un campo magnético que oscila a una frecuencia determinada. Cuando un vehículo se posiciona sobre el loop, la frecuencia de oscilación varía en función de la masa metálica del vehículo y su cercanía al Loop. El equipo está midiendo en forma continua la frecuencia de oscilación, entonces, cuando las variaciones superen el umbral prefijado por la sensibilidad, el equipo produce una variación en su salida de detección.

Este es el método que se usará para realizar la detección del vehículo. Se eligió el Loop de piso para realizar esta acción por las siguientes razones:

- ✓ No necesita fuentes de alimentación, ubicadas en la vía.
- ✓ Es apto para funcionar ante condiciones adversas.
- ✓ Únicamente detectará vehículos.

Los loop de piso poseen dos frecuencias de funcionamiento. Esta opción es útil para esta aplicación, debido a que habrá dos loop muy cercanos entre sí, para que de esta manera no haya interferencias entre los mismos.

La inductancia conseguida debe ser mayor a los  $100\mu\text{H}$ , esto se logra en función del tamaño del loop elegido. Las formas geométricas son varias, y se debe seleccionar la correcta de acuerdo al tipo de vehículo que se vaya a detectar.

El cable a emplear será de 0.75 a 1 mm<sup>2</sup> de sección. Para optimizar su funcionamiento se debe evitar la presencia de malla metálicas debajo del área ocupada por el loop debido a que esto disminuye la sensibilidad de conducción. También se debe evitar el roce de este contra ángulos filosos que pueden dañar su aislación; es necesario no realizar ángulos agudos para evitar la pérdida de aislación. Por último la salida hacia el equipo se hará retorciendo los dos conductores resultantes entre sí, para disminuir las capacitancias parasitas.

Para la detección del vehículo, en este prototipo se va a usar un **Detector** ya fabricado modelo PD132 (figura 5) y se construirá el enrollamiento de alambre (**Loop de piso**). En el anexo 2 se dan más detalles para la construcción del loop de piso.



*Figura 5.* Loop detector PD132

[Fuente: Autor]

### d.2.2 MÓDULO RFID

RFID (Radio Frequency Identification – Identificación por Radiofrecuencia) permite la identificación de objetos de forma inalámbrica, sin necesidad de que exista línea de visión directa entre el lector y el objeto. RFID es una tecnología de identificación automática similar, en cuanto a su aplicación, a la tecnología de código de barras.

Un sistema RFID incluye los siguientes componentes (Blázquez del Toro, 2014):

- ✓ Transponder: contiene un código identificativo (“tag”)
- ✓ Antena: usada para transmitir las señales de RF entre el lector y el dispositivo RFID.
- ✓ Módulo de radiofrecuencia o transceptor: este genera las señales de RF
- ✓ Lector o módulo digital: recibe las transmisiones de RF desde el dispositivo RFID y proporciona los datos al sistema servidor para su procesado.

Para la identificación del vehículo y usuario se eligió este sistema debido a las grandes características que presenta en comparación con otras tecnologías, como por

ejemplo los códigos de barras. La diferencia radica en que RFID usa una señal de radiofrecuencia (RF) en lugar de una señal óptica. Los sistemas de código de barras utilizan un lector de tarjetas impresas, en cambio RFID utiliza un lector y una tarjeta especial. En lugar de utilizar el reflejo de un rayo láser sobre la etiqueta de código de barras, RFID utiliza una señal de radiofrecuencia de baja potencia. Esta señal de radio transmitida por la tarjeta RFID no necesita estar dentro del campo óptico visual del lector debido a las propiedades de las señales de radiofrecuencia (Globatel, 2007).

Además de la identificación del vehículo y usuario, mediante el “tag” se podrá obtener un estado de cuenta del usuario, es decir se verificará si este tiene saldo disponible para poder hacer uso del SIMERT.

RFID es una buena opción en cuestiones de SEGURIDAD. Los chips RFID son extremadamente difíciles de falsificar. Un pirata informático debe tener conocimientos especializados de ingeniería inalámbrica, de algoritmos de codificación y de técnicas de cifrado. Además se puede dar distintos niveles de seguridad a los datos que estén grabados en el tag. Otros factores que lo hicieron el más apropiado para esta aplicación es que funciona óptimamente bajo factores en los que es necesaria la lectura de datos de alta velocidad, la temperatura extrema y la exposición a gases.

Por último es importante señalar que la tecnología RFID y los sistemas de redes inalámbricas se pueden integrar y ofrecer opciones de control a tiempo completo y a gran escala.

La tarjeta RFID deberá ser adquirida en el Municipio de Loja y en los lugares que se autoricen para venderlas, en ella estará grabado:

- ✓ Nombres y apellidos del usuario.
- ✓ Número de su cédula de identidad
- ✓ Placa del vehículo.

Después de que el cliente haya adquirido su tarjeta RFID podrá hacer uso del SIMERT, al acercarse a un dispositivo de control se le descontará el saldo correspondiente por el tiempo que haya elegido para hacer uso de este servicio. Cuando



se haya agotado el saldo de estacionamiento, el usuario se debe acercar a los puntos asignados por el municipio para hacer recargas.

### **d.2.3 INTERACCIÓN CON EL USUARIO (teclado y módulos de visualización)**

Para interactuar con el usuario se está usando una pantalla LCD 20×4, en la cual se mostrarán diferentes mensajes de acuerdo con los eventos que se vayan desarrollando. En el capítulo 3 correspondiente al funcionamiento del prototipo se detallan estos mensajes. En la figura 6 se muestra esta pantalla.



**Figura 6.** Pantalla LCD 20x4

[Fuente: Autor]

Otro módulo de visualización es un letrero LED que indica la dirección que debe tomar el usuario para encontrar plazas disponibles (en el capítulo 3 se detalla de mejor manera el funcionamiento de este sistema).

Adicionalmente contamos con un teclado matricial 4x4 y 8 pulsadores que permitirán ingresar cierta información al dispositivo de control como el lugar y el tiempo de estacionamiento (más detalles en el capítulo 3). En la figura 7 se muestra el teclado matricial.

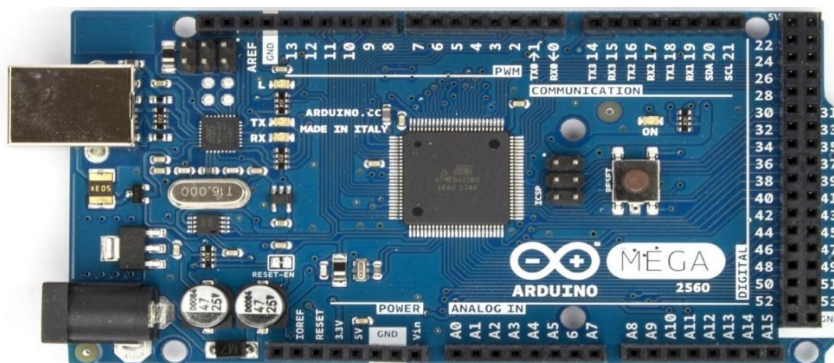


*Figura 7.* Teclado matricial 4x4.

[Fuente: Autor]

#### d.2.4 ARDUINO MEGA

El cerebro de todo el prototipo es el Arduino Mega 2560 (figura 8). Este Arduino es una placa electrónica basada en Atmega2560. Cuenta con 54 pines digitales de entrada / salida (de los cuales 15 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UARTs (hardware puertos seriales), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un header ICSP, y un botón de reinicio. Contiene todo lo necesario para apoyar el microcontrolador; para empezar a funcionar simplemente se debe conectar a un ordenador con un cable USB o el cable de poder con un adaptador de CA (corriente alterna) o la batería de CC (corriente continua) (Pomares Baeza, 2009).



*Figura 8.* Parte frontal del Arduino Mega 2560.

[Fuente: Autor]

**Memoria:** El Atmega tiene 256 KB de memoria flash para almacenar código (de los cuales 8 KB se utilizan para el cargador de arranque), 8 KB de SRAM y 4 KB de EEPROM (que puede ser leído y escrito con la biblioteca EEPROM) (MEDIALAB USAL, 2012).

**Entrada y salida:** Cada uno de los 54 pines digitales en el Mega se pueden utilizar como una entrada o salida, utilizando las funciones *pinMode ()*, *digitalWrite ()*, y *digitalRead ()*. Operan en 5 voltios. Cada pin puede proporcionar o recibir un máximo de 40 mili- amperios (MEDIALAB USAL, 2012).

**Comunicación:** El Arduino Mega2560 tiene una serie de instalaciones para la comunicación con un ordenador, otro Arduino, u otros microcontroladores. El Atmega2560 ofrece cuatro hardware UART para TTL (5V) de comunicación serie (MEDIALAB USAL, 2012).

Una biblioteca *SoftwareSerial* permite la comunicación en serie en cualquiera de los pines digitales del Mega2560.

**Programación:** El Arduino Mega se puede programar con el software de Arduino. El Arduino no necesita de un programador de hardware externo debido a que viene precargado con un gestor de arranque. Se comunica usando el protocolo STK500 (MEDIALAB USAL, 2012).

Se está usando la plataforma Arduino debido a las grandes ventajas que ofrece. Arduino es una plataforma de código y hardware abierto, es decir, se puede acceder a todo el aspecto del funcionamiento circuital y algorítmico de las placas. La programación en Arduino es fácil en comparación con otros lenguajes de programación, ya que ofrece un entorno de desarrollo integrado (IDE) con funciones preestablecidas que reducen la lógica a lectura de entradas, control de tiempos y salidas de una manera intuitiva.

Una de las ventajas más grandes que tiene Arduino es que posee librerías para prácticamente cualquier componente externo que se le quiera acoplar, haciendo innecesario aprenderse el *datasheet* del componente. Para Arduino existen mucha

documentación y tutoriales, prueba de ello es que Internet está plagado de documentos sobre esta plataforma, incluso se puede encontrar videos tutoriales en *Youtube* y la red.

#### **d.2.5 DS1307 - RELOJ EN TIEMPO REAL.**

Cuando el dispositivo de control está en reposo en la pantalla LCD se muestra la hora en tiempo real. Para ello se usa un DS1307, la razón por la que no se calcula la hora directamente del Arduino es porque este tiene que realizar otras tareas que son más importantes.

El DS1307 es un reloj/ calendario en tiempo real que nos suministra segundos, minutos, horas, días, mes y año. Mediante una batería de 3 voltios y un cristal externo puede funcionar independientemente del sistema. La fecha se ajusta automáticamente para los meses con menos de 31 días, además incluye las correcciones para los años bisiestos (Ionescu, 2013).

El DS1307 es de bajo consumo energético, cuenta con 56 *bytes* de memoria SRAM. El reloj funciona tanto con el formato de 24 horas como con el de 12 horas con indicador AM / PM (Ionescu, 2013).

#### **d.2.6 COMUNICACIÓN INALÁMBRICA.**

EL estándar 802.15.4 fue desarrollado en el 2003 por la Alianza ZigBee, cuya finalidad es estandarizar el nivel físico y el control de acceso al medio de redes inalámbricas. Actualmente disponemos de un gran número de estándares inalámbricos (bluetooth, wi-fi, wi-max, etc.) orientados hacia aplicaciones con elevados requerimientos de ancho de banda (redes domésticas y de oficina, videoconferencia, VoIP, etc.), pero hace falta un estándar inalámbrico específico para redes de sensores en aplicaciones industriales y domésticas.

El problema de usar cualquiera de los estándares antes mencionados reside en el gran consumo de energía y ancho de banda que utilizan, frente a la baja tasa de bits enviados por cualquier aplicación sensora o de control. Inicialmente, cada fabricante hizo uso de sus propias soluciones, pero esto trajo problemas de interoperabilidad entre los productos de diversos fabricantes.

De la necesidad de un estándar capaz de enviar datos de baja capacidad y a un bajo costo nace el estándar IEEE 802.15.4. ZigBee es un estándar que define un conjunto de protocolos para el armado de redes inalámbricas de corta distancia y baja velocidad de datos. Opera en las bandas de 868 MHz, 915 MHz y 2.4 GHz. Fue diseñado pensando en las siguientes necesidades; bajo consumo que permita usar equipos a batería, bajo costo de dispositivos y de instalación y mantenimiento de ellos (Dignani, 2011).

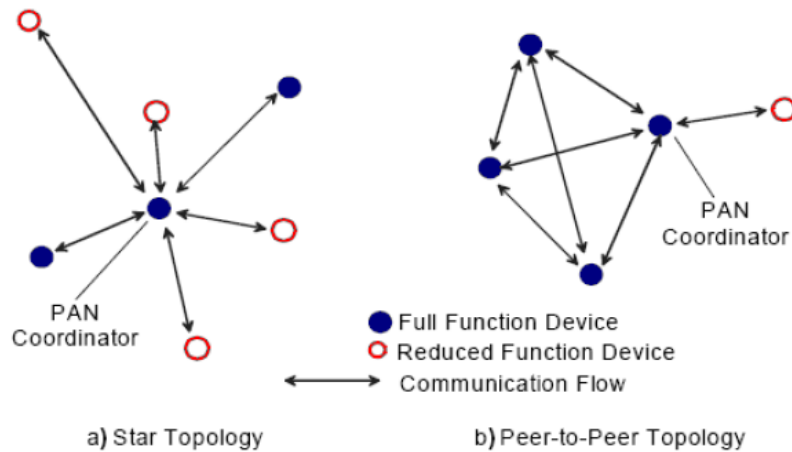
El estándar 802.15.4 dispone de dos tipos distintos de dispositivos (Full function device FFD y Reduced function device RFD), ZigBee soporta dos tipos de topologías para su conexión en red (topología en estrella y topología peer-to-peer), esto se muestra en la figura 9 (Dignani, 2011).

- ✓ **FFD (Full Function Device):** Son dispositivos que tienen funcionalidad completa, pueden operar como Coordinador de Red de Área Personal (coordinador PAN) o como dispositivo de red (end point)
- ✓ **RFD (Reduced Function Device):** Son dispositivos con funcionalidad reducida. Solamente pueden recibir órdenes o informar a un coordinador de red más cercano. Es decir son los dispositivos de red (sensores o actuadores) usados para aplicaciones simples.

✚ **TOPOLOGÍA EN ESTRELLA (STAR):** Todos los nodos de una misma WPAN están coordinados por un único nodo FFD que recibe el nombre de Coordinador de Red y entre sus principales tareas se encuentra la de coordinar el acceso al medio. El coordinador de red, al tener el papel principal de “organizador”, posee unos requerimientos de energía mayores que el resto de nodos y es por este motivo que este suele ser un nodo con una unidad de energía no agotable; es decir debe estar conectado a la red eléctrica.

✚ **TOPOLOGÍA IGUAL A IGUAL (PEER TO PEER):** Todos sus nodos suelen ser FFD, debido a que todos tienen la misma prioridad de acceso al medio, en esta topología existe un coordinador PAN pero este no cumple con funciones relevantes. La diferencia con la topología en

estrella es que cualquier nodo se puede contactar con otro nodo sin permiso del coordinador PAN. Esto la hace similar a arquitecturas de Red Mesh (malla) o Ad-Hoc.



**Figura 9.** Topologías en redes 802.15.4

Fuente: (Arévalo Barazas, 2011)

#### **d.2.6.1 ESTÁNDAR ZIGBEE**

En realidad existen muchos estándares que se pueden usar en redes de corto alcance tales como el 802.11 y *Bluetooth*. Cada uno de estos está desarrollado para una clase de aplicación determinada; sin embargo ZigBee es el estándar más aceptado hoy en día para usarse en redes de sensores y actuadores que deben operar a batería (Barriga Vázquez, y otros, 2006).

Para realizar la transmisión de datos desde cada dispositivo del Smart Parking hacia el servidor era necesario establecer un protocolo de comunicaciones ya sea alámbrico o inalámbrico. En mi caso he elegido el estándar IEEE 802.15.4 o como comúnmente se lo conoce “Zigbee”, el modelo del módulo elegido para trabajar conjuntamente con Arduino es el XBee-PRO 900 (Arévalo Barazas, 2011).

Las razones por las que este protocolo será empleado son las siguientes (la mayoría de los datos son tomados del datasheet del producto, anexo 3):

- ✓ Soporta 65000 dispositivos en la red.
- ✓ Tiene un bajo consumo de energía: en transmisión 30mA y en reposo 3µA.

- ✓ Cobertura de hasta 10Km cuando existe línea de vista y se usa antenas de alta ganancia. De 3Km con línea de vista y el rango para el entorno urbano es de 140m.
- ✓ La velocidad de transmisión de datos es de 156 Kbps.
- ✓ Es un protocolo de comunicaciones que ofrece multisaltos.

#### ***d.2.6.2 SEGURIDAD.***

Dadas las características inalámbricas de la red ZigBee, un mensaje puede ser recibido por cualquier dispositivo cercano. En algunas aplicaciones puede que esto no resulte perjudicial pero en otras un intruso podría violar la privacidad de las personas, producir daños o inhabilitar algún sistema (por ejemplo una alarma) o podría obtener datos privados. ZigBee soporta el uso de protocolos estándar de encriptación y autenticación. El diseñador de la red debe decidir entre el nivel de seguridad, complejidad y el costo de los dispositivos, ya que el aumento de seguridad requiere más capacidad de cómputo y más memoria; y esto es directamente proporcional al gasto energético.

ZigBee utiliza Advance Encryption Standard (AES) como técnica de encriptación. El estándar 802.15.4 soporta autenticación de dispositivos y de datos; el propósito de la autenticación de datos es asegurar que los mismos son válidos y que no sufrieron transformación alguna antes de llegar a su destino.

#### ***d.2.6.3 MÓDULO XBEE***

Los módulos Xbee son soluciones integradas que brindan un medio inalámbrico para la interconexión y comunicación entre dispositivos.

Los módulos XBee utilizan el protocolo IEEE 802.15.4 mejor conocido como ZigBee. Zigbee es un protocolo de comunicaciones, creado por la ZigBee Alliance (Thayer Ojeda, 2015).

Zigbee permite que dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas. Es especialmente útil para redes de sensores en entornos industriales, médicos y sobre todo domóticos.

Cada módulo Zigbee al igual que ocurre con las direcciones MAC de los dispositivos de Ethernet tiene una dirección única. En el caso de los módulos Zigbee cada uno de ellos cuenta con una dirección única de 64 bits que viene grabada de fábrica. Una red con este protocolo utiliza para sus algoritmos de ruteo direcciones de 16 bits. Es por esta razón que el número máximo de elementos que se pueden usar en una red son  $2^{16}=65535$ , que sería el número máximo de direcciones que se pueden asignar. Estos módulos pueden funcionar en configuraciones de red de punto-a-punto, punto-a-multipunto o peer-to-peer (Thayer Ojeda, 2015).

En la figura 10 se muestra el módulo XBee con la antena que se está usando en el prototipo, este módulo funciona a 900 MHz.



*Figura 10.* Módulo Xbee y su antena

[Fuente: Autor]

### d.2.7 ALIMENTACIÓN

Para el funcionamiento del dispositivo de control es necesaria una fuente conmutada de 12 Voltios - 3 Amperios.

Para el funcionamiento del sistema de visualización es necesaria una fuente de alimentación de 24 Voltios – 4 Amperios.

Todos los elementos del dispositivo de control funcionan únicamente con 5 Voltios, es por esto que en cada tarjeta se usó reguladores de voltaje. Estos reguladores son el LM7805 y el LM2596s. También se usan condensadores como filtros para eliminar el ruido. El **dispositivo de control consume 50mA** cuando está funcionando



con alimentación de corriente alterna (AC) y consume 400mA con alimentación de corriente directa (DC).

Y el letrero LED consume 1.5A, este gran consumo de energía se debe a que los LEDs al ser de alto brillo, consumen mucha corriente.

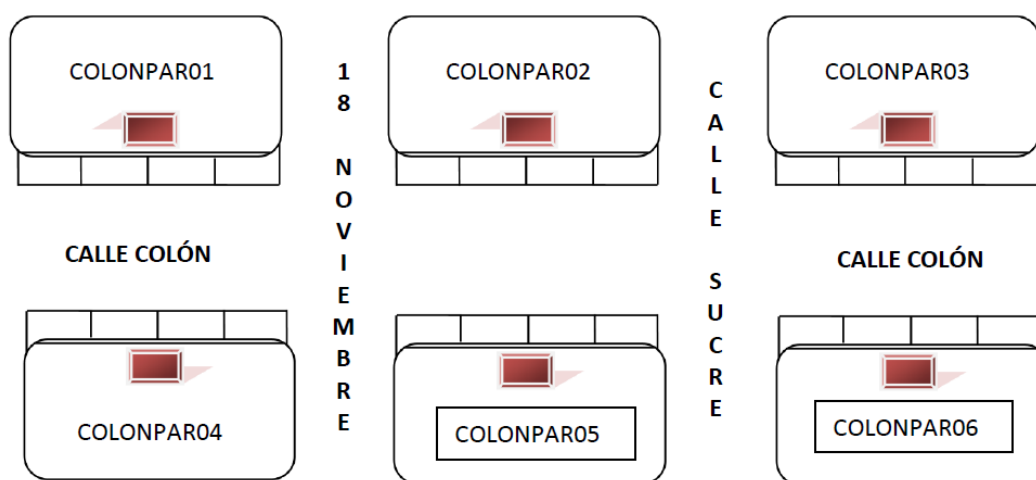
### d.3 CAPÍTULO III:

## CONSTRUCCIÓN DEL PROTOTIPO Y FUNCIONAMIENTO.

### d.3.1 ELABORACIÓN DEL SOFTWARE Y HARDWARE

El programa principal para el funcionamiento del dispositivo de control del SIMERT es realizado en el ID de ARDUINO. También estamos usando microcontroladores PIC, uno para la lectura del teclado matricial y otro para el control del Display numérico. La plataforma que estamos usando para el desarrollo del programa es MikroC para PIC.

En la actualidad cada espacio de estacionamiento tiene una longitud de 5.8 metros, en cada cuadra existen aproximadamente 10 lugares para estacionarse. Se propone que por cada cuadra exista un dispositivo de control (ver anexo 6), el cual por motivos de reconocimiento debe estar nombrado y numerado tal como se muestra en la figura 11. Es importante que el Municipio ponga una numeración a cada uno de los lugares de estacionamiento ya que como se indicará más adelante, el usuario debe ingresar manualmente el número de la plaza que su vehículo está ocupando.

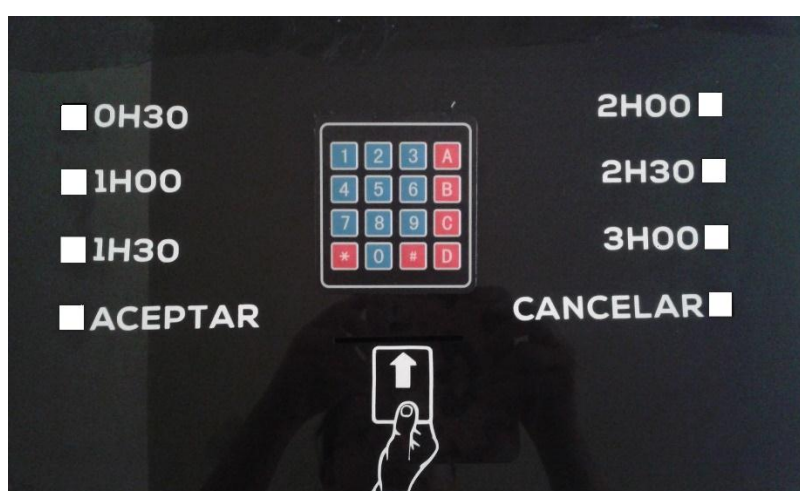


*Figura 11.* Ubicación de los dispositivos de control en la calle Colón.

[Fuente: Autor]

### ***d.3.1.2 UBICACIÓN DE PULSADORES PARA ELECCIÓN DEL TIEMPO DE ESTACIONAMIENTO***

En la figura 12 se muestra como estarán ubicados los pulsadores para elegir el tiempo de estacionamiento, el último botón, de cada columna, como se puede observar tienen otro propósito. El botón “ACEPTAR” es para confirmar alguna acción y el botón “CANCELAR” debe ser presionado en caso de que el usuario quiera realizar alguna corrección. En el medio de los pulsadores se encuentra ubicado el teclado matricial y la ranura para ingresar la tarjeta RFID.



***Figura 12.*** Ubicación de los pulsadores

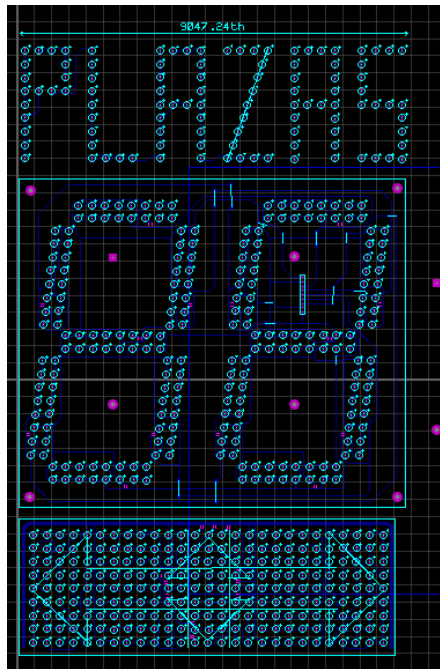
[Fuente: Autor]

### ***d.3.1.3 SISTEMA DE VISUALIZACIÓN.***

Con el propósito de disminuir el tiempo de búsqueda de plazas de estacionamiento disponibles al usuario y por ende reducir las emisiones de elementos contaminantes, se colocarán letreros LED al inicio de cada calle, estos letreros indicarán el número de espacios disponibles en la misma.

El sistema de visualización es simple ya que la finalidad es que el usuario se pueda guiar de manera gráfica y de esta manera pueda captar rápidamente el mensaje y no desconcentrarlo de su actividad de manejo del vehículo.

El esquema del sistema de visualización se puede observar en la figura 13. Este sistema servirá de guía al usuario en su búsqueda de una plaza disponible.

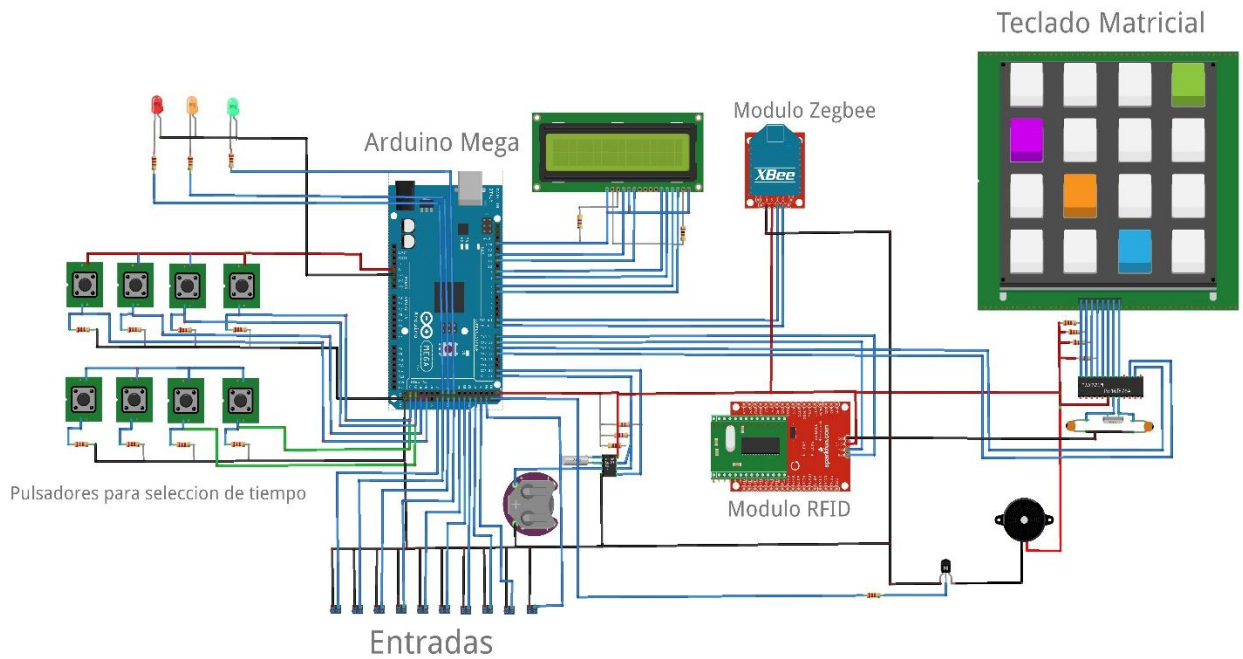


*Figura 13.* Simulación en Proteus del sistema de visualización.

[Fuente: Autor]

#### *d.3.1.4 DIAGRAMA DEL DISPOSITIVO DE CONTROL.*

En la figura 14 se puede observar el diagrama del dispositivo de control, en él se puede observar el Arduino Mega así como los dispositivos y elementos que están conectados a este, también se indican los pulsadores para la elección del tiempo de estacionamiento, la pantalla LCD, el módulo Zigbee, el módulo RFID, el teclado matricial, el reloj en tiempo real, indicadores LEDs, un buzzer o chicharra que suena al momento en que el usuario presiona alguno de los pulsadores y finalmente se observan las entradas para el Loop de piso.



**Figura 14.** Diagrama del dispositivo de control

[Fuente: Autor]

### d.3.2 FUNCIONAMIENTO

El funcionamiento que se propone del sistema tiene como actores principales a los usuarios, el sistema de recolección de datos (centro de control), el sistema de visualización (letreros LED) y los dispositivos de control.

Para una fácil interacción con el usuario la estructura del dispositivo de control se asemeja a un cajero automático. Donde el cliente tiene una ranura en la cual debe colocar su tarjeta RFID, el dispositivo de control indica cuándo ha terminado el proceso y puede retirar su tarjeta, también existen pulsadores para la elección del tiempo de estacionamiento y para aceptar (ACEPTAR) o corregir alguna acción (CANCELAR) (ver figura 12).

#### d.3.2.1 SISTEMA DE VISUALIZACIÓN.

El sistema de visualización se compone por un letrero LED y por un módulo de comunicaciones inalámbrico (módulo Zigbee).

Cuando existan plazas desocupadas el letrero indicara mediante números arábigos la cantidad de espacios disponibles, y mediante las flechas de la parte inferior se mostrará la dirección a la cual se debe dirigir para encontrarlos (figura 13).

Cada letrero LED posee su propio módulo Zigbee el cual recibe la información de los espacios disponibles desde el centro de control.

#### ***d.3.2.2 PROCESO QUE DEBE SEGUIR EL USUARIO***

Cuando un cliente ocupa una plaza disponible, debe acercarse al dispositivo de control correspondiente a su cuadra e insertar su tarjeta RFID en la ranura del mismo. Acto seguido debe confirmar los datos que el dispositivo lee en su tarjeta.

Una vez confirmados sus datos el sistema pregunta el lugar en el que aparcó su vehículo, el usuario indicará el número de plaza que ocupó presionando el correspondiente número en el teclado del dispositivo.

Después el usuario seleccionará el tiempo que va a ocupar esa plaza y finalmente debe retirar su tarjeta RFID.

Este es todo el proceso que debe seguir un usuario, como se puede apreciar es un proceso muy simple y rápido.

En el siguiente acápite se describe detalladamente todo el proceso que se realiza tanto a nivel de software como de hardware.

#### ***d.3.2.3 PROCESO GENERAL.***

Todo el proceso empieza cuando un cliente tiene la necesidad de aparcar su vehículo. Cuando el usuario ha encontrado una plaza disponible guiándose por medio del visualizador de plazas (letrero LED) y la ocupa, se activa el loop de piso, este envía la variación de frecuencia hacia el dispositivo de lectura de los datos. Acto seguido se avisa al servidor que una plaza ha sido ocupada, por ejemplo si se ha ocupado un lugar en la calle Colón el mensaje de aviso sería el siguiente:

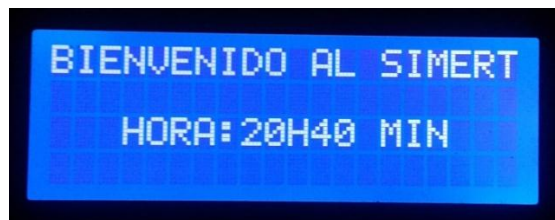
COLONPAR01#OCUPADO#01

Los 10 primeros bits de esta secuencia indican el nombre de la calle donde se encuentra ubicado el dispositivo y el número del mismo, los 7 bits siguientes indican el estado del parqueadero, en este caso indica que un lugar se ha ocupado. Cabe recalcar que esta transmisión inalámbrica de datos se realiza mediante módulos Zigbee. El último bit le indica al centro de control el número del espacio en donde se ha aparcado el vehículo.

Para indicar que un espacio se ha desocupado, el dispositivo de control envía el siguiente mensaje de aviso:

COLONPAR01#DESOCUPADO#01

Cuando el conductor se ha estacionado debe acercarse a uno de los dispositivos de control ubicados en cada cuadra, este dispositivo estará situado en la mitad de la cuadra. Al colocar el usuario su tarjeta RFID en la ranura del dispositivo de control en la pantalla LCD se podrá visualizar el siguiente mensaje:



*Figura 15.* Mensaje que aparece cuando el dispositivo está en reposo  
[Fuente: Autor]

Luego de colocar la tarjeta RFID, los datos de la tarjeta serán leídos y en la pantalla LDC del dispositivo de control aparecerán los nombres del usuario y la placa de su vehículo. Por ejemplo:



*Figura 16.* Datos que identifican al usuario del SIMERT  
[Fuente: Autor]

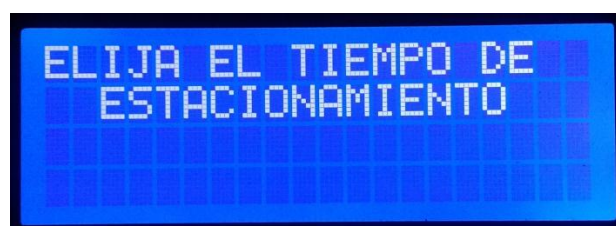
El usuario debe presionar el pulsador “ACEPTAR” para corroborar estos datos. Una vez realizada la identificación del usuario el sistema le pregunta el lugar en el que se encuentra estacionado, mediante el siguiente mensaje:



*Figura 17.* Mensaje que pide el número de plaza ocupada  
[Fuente: Autor]

El usuario debe realizar esta acción presionando el lugar donde se estacionó en un teclado numérico que tiene implementado el dispositivo de control. Cuando el cliente declara el lugar en el que se estacionó el dispositivo le pide confirmación de ese lugar para que de esta manera el usuario pueda hacer la corrección del lugar en caso de haberse equivocado al indicar la plaza que ocupó, en este caso el usuario debe presionar “CANCELAR” y el dispositivo le volverá a preguntar el lugar donde se encuentra ubicado su vehículo.

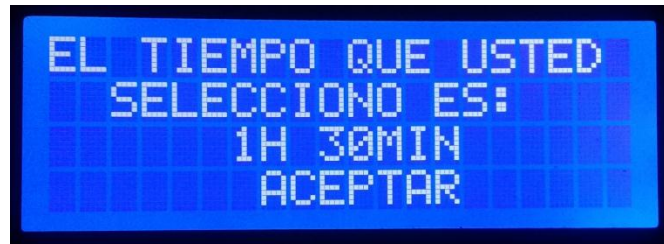
Una vez que el usuario señaló el lugar donde aparcó su vehículo, el dispositivo preguntara el tiempo que el cliente desea quedarse ocupando la plaza en la que se estacionó, mediante el siguiente mensaje:



*Figura 18.* Mensaje que aparece para que el usuario elija el tiempo  
[Fuente: Autor]

La elección de este tiempo se lo realiza presionando cualquiera de los pulsadores que indican el tiempo de estacionamiento (figura 12), cuando el cliente elige el tiempo el dispositivo pregunta si está seguro de ese tiempo, con el siguiente mensaje:





**Figura 19.** Mensaje para que el usuario confirme o cancele el tiempo que eligió.  
[Fuente: Autor]

Si el cliente está seguro debe presionar “ACEPTAR”, caso contrario debe presionar “CANCELAR” para que el dispositivo le permita elegir un nuevo tiempo de estacionamiento.

Es necesario poner a consideración del usuario que el pulsador para la elección de 2h00, 2h30 y 3h00 estará desactivado en las zonas en las que no se permite estos tiempos, es decir, en la zona “A”; por lo tanto el usuario no podrá seleccionarlos. En la zona **A**, el tiempo máximo de estacionamiento es de 1h30 minutos y en la zona **B** es de dos horas. Existe un caso especial que es en los alrededores del complejo ferial en los cuales se aplica el SIMERT, en este caso el tiempo máximo de estacionamiento es de tres horas.

Después de que el usuario ha realizado todas estas acciones, debe esperar unos milisegundos hasta que el dispositivo envíe un mensaje al centro de control, el mensaje que se envía es el siguiente:

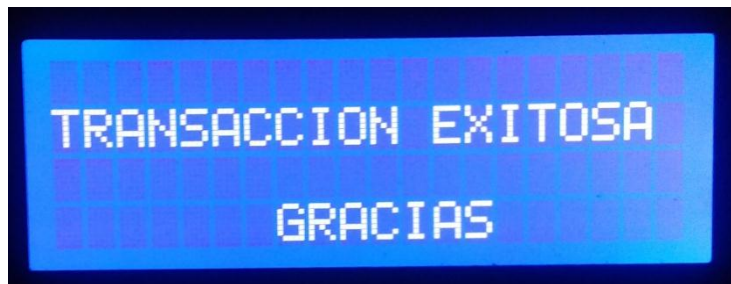
COLONPAR01#TIME#LBA-1092#1h00#01

En este caso los 10 primeros bits corresponden al nombre de la calle donde el cliente se estacionó, los 4 bits siguientes son el nombre del evento que en este caso es “TIME”, los siguientes bits corresponden al número de placa del vehículo estacionado, los 4 bits posteriores indican el tiempo que el cliente eligió para quedarse estacionado (en este ejemplo se indica 1h00 hora estacionamiento) y el último bit corresponde al número de plaza ocupada.

Con esta información se averigua si el usuario cuenta con el saldo suficiente para hacer el uso del SIMERT. Si se ha confirmado que el usuario tiene saldo el centro de control envía el siguiente mensaje al dispositivo:

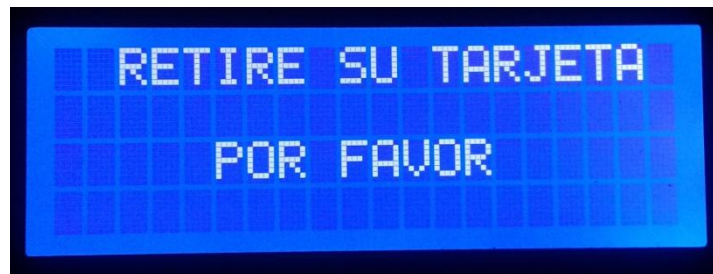
COLONPAR01#TIME# LBA-1092#1

Al igual que el mensaje anterior los 10 primeros bits corresponden al nombre de la calle donde el cliente se estacionó, los 4 bits siguientes son el nombre del evento que en este caso es “TIME”, los siguientes bits corresponden al número de placa del vehículo estacionado y el último bit representa el estado de su cuenta, como en este caso el cliente si contaba con saldo el bit que llega es el “1” y en la pantalla LCD al cliente se le muestran los siguientes mensajes:



*Figura 20.* Mensaje que aparece cuando ha terminado el registro de un usuario

[Fuente: Autor]



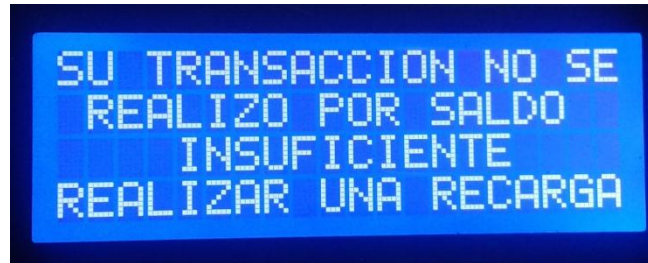
*Figura 21.* Mensaje para que el usuario no olvide retirar su tarjeta

[Fuente: Autor]

En cambio si el cliente no cuenta con el saldo suficiente se envía el siguiente mensaje al dispositivo:

COLONPAR01#TIME# LBA-1092#0

Lo único que cambia en este mensaje con el anterior es el último bit que en este caso es “0” debido a que el usuario no tiene saldo y los mensajes que se visualizaran en la pantalla LCD son los siguientes:



**Figura 22.** Mensaje que aparece cuando el usuario no cuenta con saldo

[Fuente: Autor]

Es importante tener en cuenta que el usuario no debe retirar su tarjeta RFID hasta que el dispositivo se lo indique. También se debe tener presente que el dispositivo acepta dos correcciones, cuando un cliente presiona el pulsador “CANCELAR” por tres ocasiones seguidas en un mismo evento, el dispositivo le presentará el siguiente mensaje:



**Figura 23.** Mensaje de cancelación del registro de un usuario

[Fuente: Autor]

Y el dispositivo se reiniciara. Es decir aparecerá el mensaje por defecto (BIENVENIDO AL SIMERT y se indicara la hora).

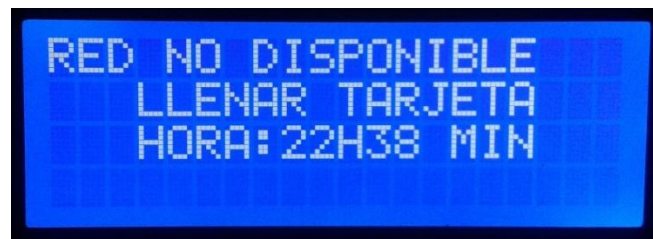
Si una persona ingresa una tarjeta que no sea una RFID (tarjeta de crédito, tarjeta de algún supermercado o centro comercial, etc.), o En caso de que algún individuo inserte una tarjeta RFID pero que no sea la emitida por el Municipio se muestra el siguiente mensaje:



**Figura 24.** Mensaje que indica tarjeta desautoriza

[Fuente: Autor]

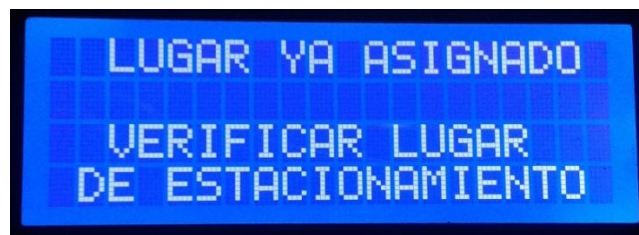
Como ya se indicó anteriormente, cuando un cliente va a registrarse, el dispositivo de control le envía un mensaje al centro de control con el evento “OCUPADO”, el dispositivo espera la respuesta del centro de control por un tiempo de 50 milisegundos. Si al término de este tiempo no se obtiene una respuesta, el dispositivo muestra en la pantalla lo siguiente:



**Figura 25.** Aviso de que no hay sistema

[Fuente: Autor]

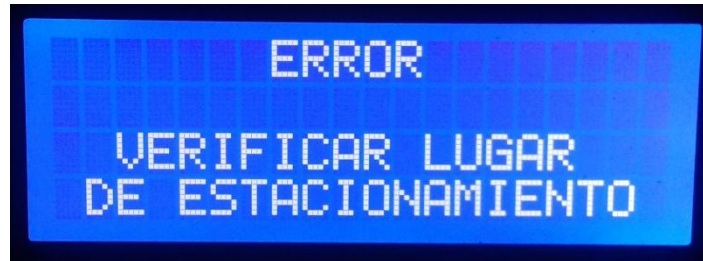
Si un cliente se estacionó por ejemplo en el espacio de aparcamiento número 1 y ya realizó su registro en el centro de control y, viene otro cliente y por error o por no fijarse bien donde estacionó su vehículo le indica al dispositivo de control que está ubicado en el espacio 1. En caso que esto ocurra, el centro de control envía un mensaje al dispositivo diciéndole que ese lugar ya está ocupado; y en la pantalla LCD aparece lo siguiente:



**Figura 26.** Mensaje para que el usuario corrija el lugar donde se estacionó.

[Fuente: Autor]

Otro caso que puede suceder es que un cliente ingrese el número de una plaza de estacionamiento que no existe, ejemplo 70; en este caso el dispositivo se lo advierte al cliente con el siguiente mensaje:



*Figura 27.* Mensaje de error cuando se elige mal el lugar donde se aparcó el vehículo.  
[Fuente: Autor]

## e.- MATERIALES Y MÉTODOS

### e.1 MATERIALES

A continuación se muestra una lista de todos los materiales usados para la construcción del Hardware del prototipo y también se nombran los programas necesarios para la configuración del mismo.

#### e.1.1 MATERIALES PARA LA CONSTRUCCIÓN DEL PROTOTIPO

*Tabla 4.* Materiales usados para la construcción del dispositivo de control.

CANTIDAD	MATERIAL
2	2n3904
1	Arduino
7	borneras de 2 pines
1	Caja
1	capacitor 100 uf
1	capacitor 470 uf
1	capacitor 68 uf
4	conector AC-DC para chasis
2	conector molex 2 pines
1	conector molex de 10 pines
1	conector molex de 12 pines
1	conector molex de 5 pines
3	conector molex de 8 pines
1	crystal 8 MHz
2	disipador de calor to220
1	ds1307 reloj en tiempo real
1	fuentes 12V-3A
2	I7805
1	lcd 20x4
3	led smd
3	loop de piso
1	metro cable multicolor
2	metros cable rojo negro #14
1	módulo Xbee
1	pic16f876a
1	pila cr2032
7	postes 20 mm
8	Pulsadores
2	pulsadores smd
3	regleta de male 40 pines

1	resistor 110 ohm
10	resistor 3.9 Kohm
3	resistor 360 ohm
1	resistor 470 ohm
1	Riel
8	sujeta cables
3	switch on/off
1	teclado 4x4
1	tip110
14	Tornillos
1	zócalo 28 pines
1	zócalo para lipa cr2032
1	zócalo para tarjetas
2	zócalos 8 pines
1	Soporte para módulo Xbee

Fuente: Autor.

**Tabla 5.** Materiales usados en la construcción del dispositivo de control.

CANTIDAD	MATERIAL
2	2n3904
3	baquelita fibra de vidrio 20x30
18	bd135
3	bornera 2 pin
1	Caja
1	conector AC-DC
3	conectores moles de 4 pin
1	fuentes 24V - 4,2A
1	L7805
580	leds verde alto brillo
1	metro de cable rojo negro #14
1	módulo Xbee
1	pic16f876a
1	pic16f877a
10	postes de 30mm
20	resistores 150 ohm
20	resistores 2,2 Kohm
2	tip42
22	tornillos
1	uln2803

Fuente: Autor

## e.1.2 LISTA DE SOFTWARES USADOS PARA LA CONFIGURACIÓN DEL PROTOTIPO.

*Tabla 6.* Programas.

	Software
1	Smrfid
2	Fritzing
3	Visual Basic 6.0
4	Proteus
5	Software de Arduino
6	MikroC PRO for PIC
7	PicKit2
8	X-CTU

Fuente: Autor.

## e.2 MÉTODOS

La construcción de este proyecto de tesis inició con la idea de “construir un prototipo de parqueadero inteligente para la automatización del SIMERT” con el propósito de contribuir con el desarrollo de la ciudad.

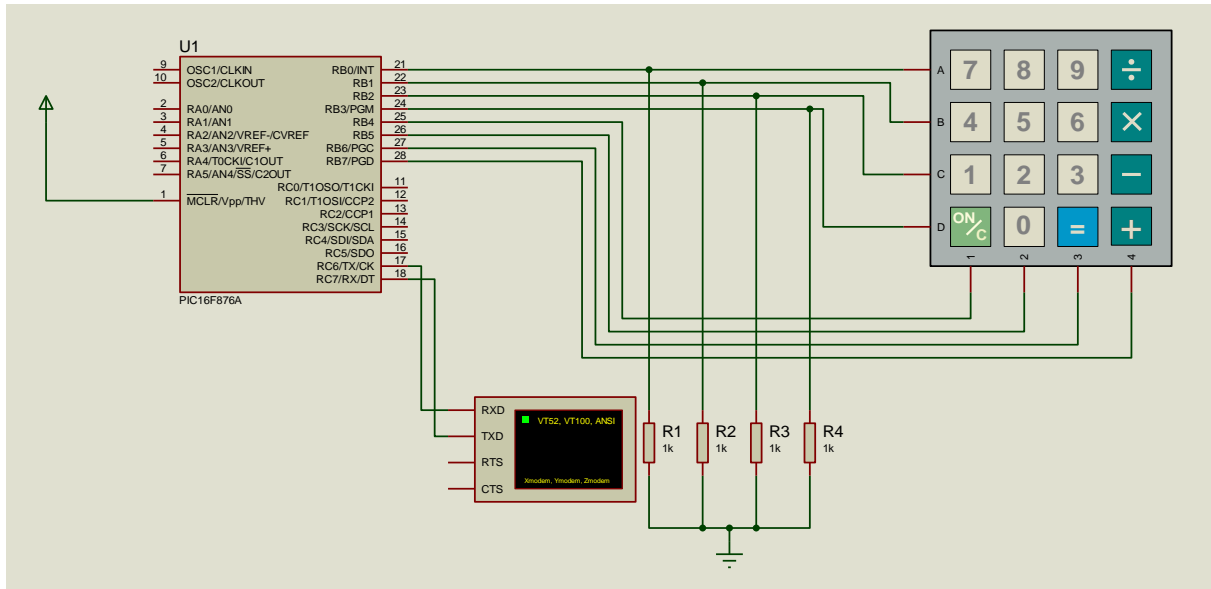
Una vez que la idea estuvo clara, se empezó a desarrollarla. El primer paso que se dio fue seleccionar los materiales, dispositivos, programas, etc. que serían usados para la elaboración de las distintas partes (dispositivo de control, letrero LED y centro de control) del prototipo.

Una vez adquiridos los materiales necesarios para la elaboración del prototipo se procedió a hacer los diagramas necesarios para poder establecer cómo conectar y comunicar los dispositivos. Cuando se tuvo el diagrama listo empecé a realizar la programación en Arduino y MikroC. El diagrama de flujo seguido para realizar el programa para el funcionamiento del Arduino se muestra en las figuras 29 y 30, y el programa en sí consta en el anexo 4.

Se realizaron dos programas en MikroC, uno es para el control de teclado matricial y el microcontrolador que se usó para este propósito es el PIC16F876A, y el



otro programa es para el control del letrero LED y se usó el PIC16F877A. En la figura 28 muestra una simulación en Proteus del control del teclado matricial y en el anexo 5 consta el algoritmo para el teclado matricial.



**Figura 28.** Simulación en Proteus del teclado matricial

[Fuente: Autor]

La **elaboración del diagrama del dispositivo de control** se realizó usando el programa de automatización de diseño electrónico “Fritzing”. Este programa brinda ayuda para poder pasar de prototipos a productos finales, es decir, permite a los diseñadores de circuitos documentar sus prototipos basados en Arduino y crear esquemas de circuitos impresos para su posterior fabricación. El resultado de esta simulación en Fritzing se puede observar en la figura 14.

Este esquema fue de gran ayuda y sirvió de guía para saber de forma clara cómo conectar todos los elementos al Arduino. También facilitó la corrección de algunos errores.

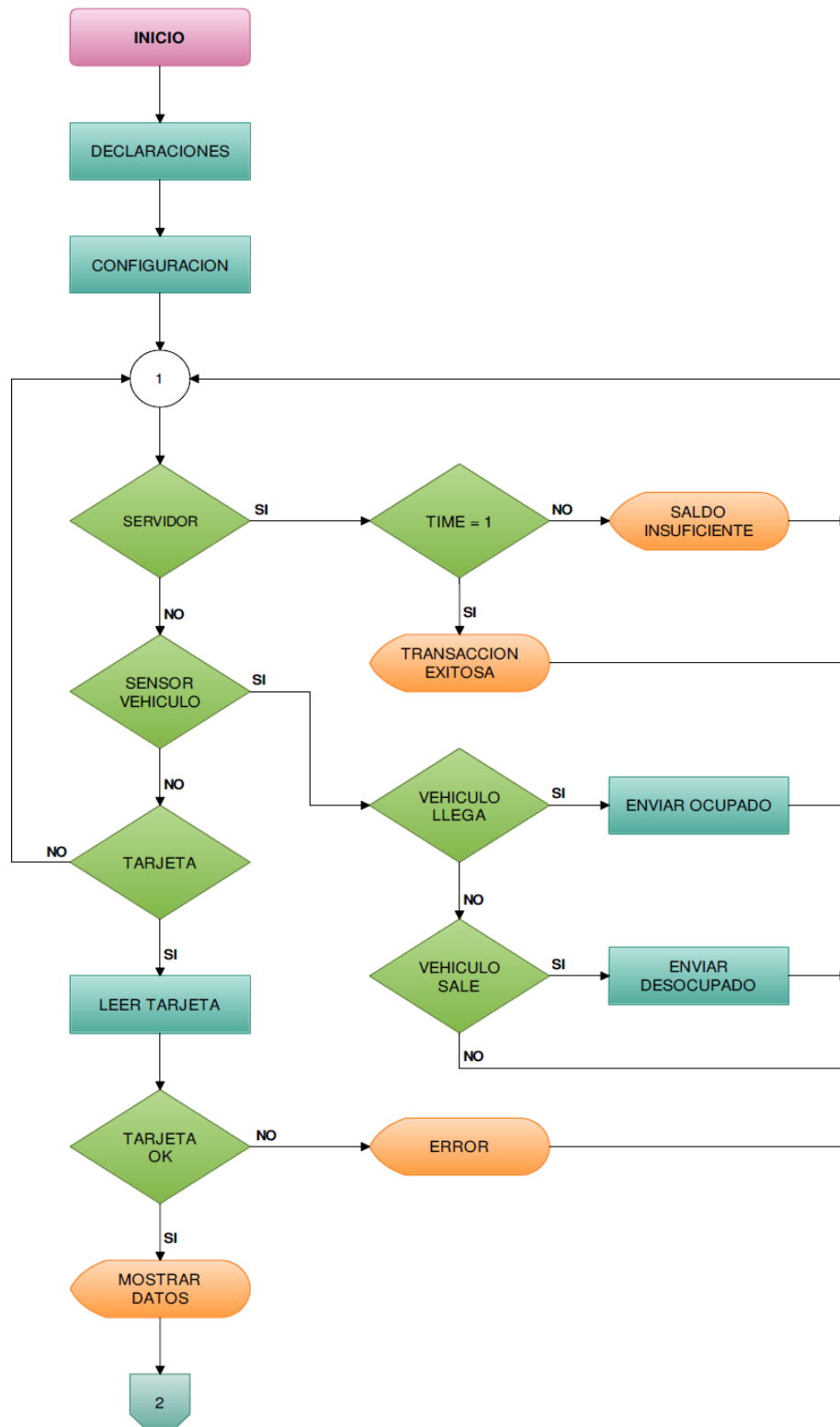
## DESCRIPCIÓN DEL DIAGRAMA DE FLUJO

El programa tiene tres partes fundamentales o principales. Primero tenemos la declaración de todas las librerías, variables y los mensajes que se muestran en la LCD, a continuación tenemos la configuración de los pines de entrada/salida y finalmente el ciclo repetitivo que constituye el programa propiamente dicho.

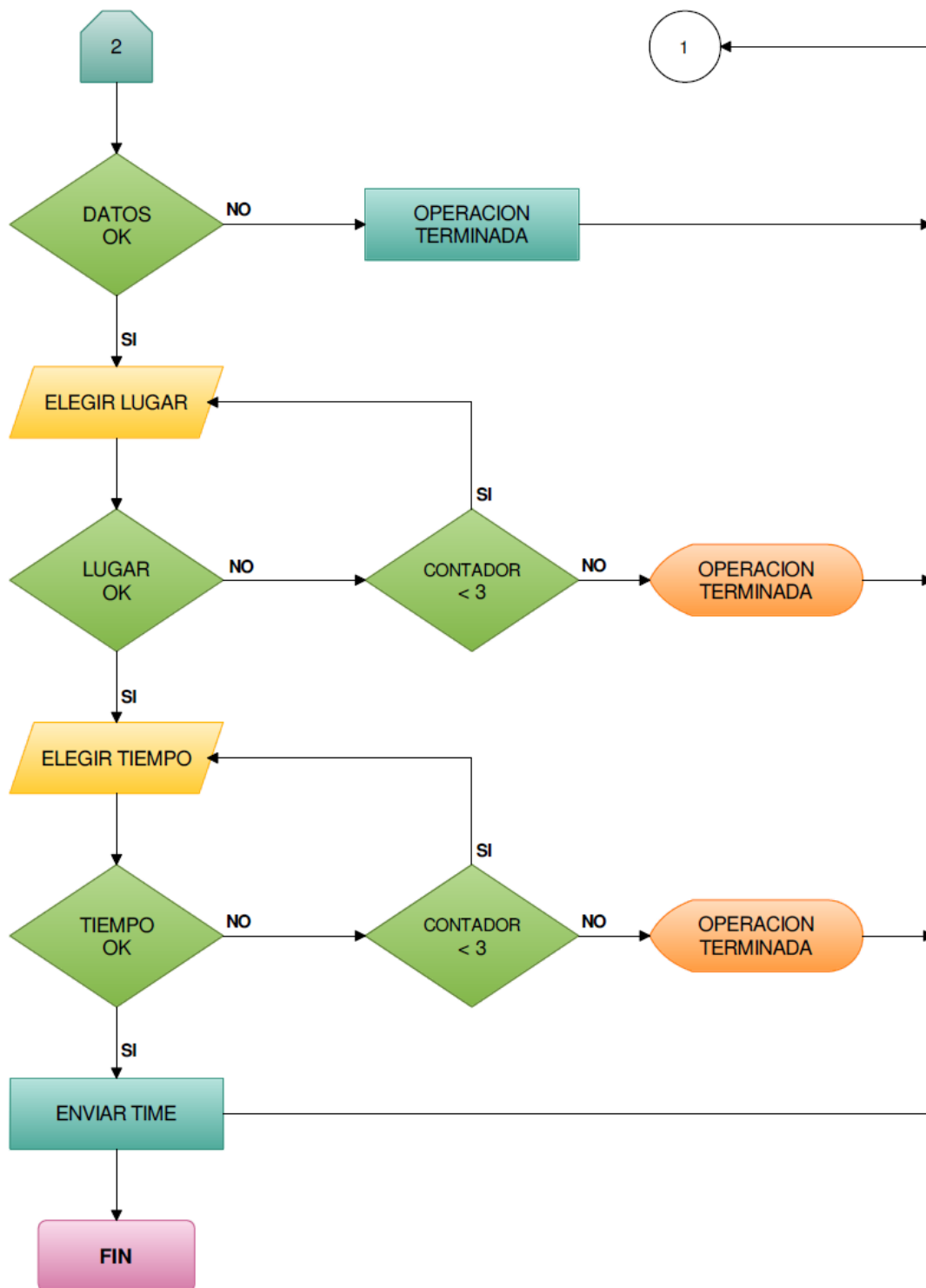
En el ciclo repetitivo el programa está monitoreando permanentemente el centro de control, si hay datos provenientes del dispositivo de control se revisa el comando TIME para saber si la tarjeta tiene o no saldo para estacionar.

A continuación se revisa si se activa o se desactiva algún sensor de vehículo, si se desactiva se envía el comando **desocupado**, caso contrario se envía el comando **ocupado** y se da paso para la detección y la lectura de la tarjeta RFID.

Una vez realizada la detección de la tarjeta se lee los datos de la misma; si la tarjeta no corresponde al sistema se presenta un mensaje de error, caso contrario se procede a la verificación de los datos con el usuario y si estos datos son correctos se da la opción para escoger el lugar y el tiempo de estacionamiento. Si el lugar o el tiempo son erróneos se da tres oportunidades para la rectificación de los mismos, si en las tres oportunidades no se ha escogido el lugar y tiempos correctos se termina la operación y el proceso se reinicia. Si el tiempo y el lugar son confirmados por el usuario se envía el comando TIME para la verificación de los datos en el centro de control.



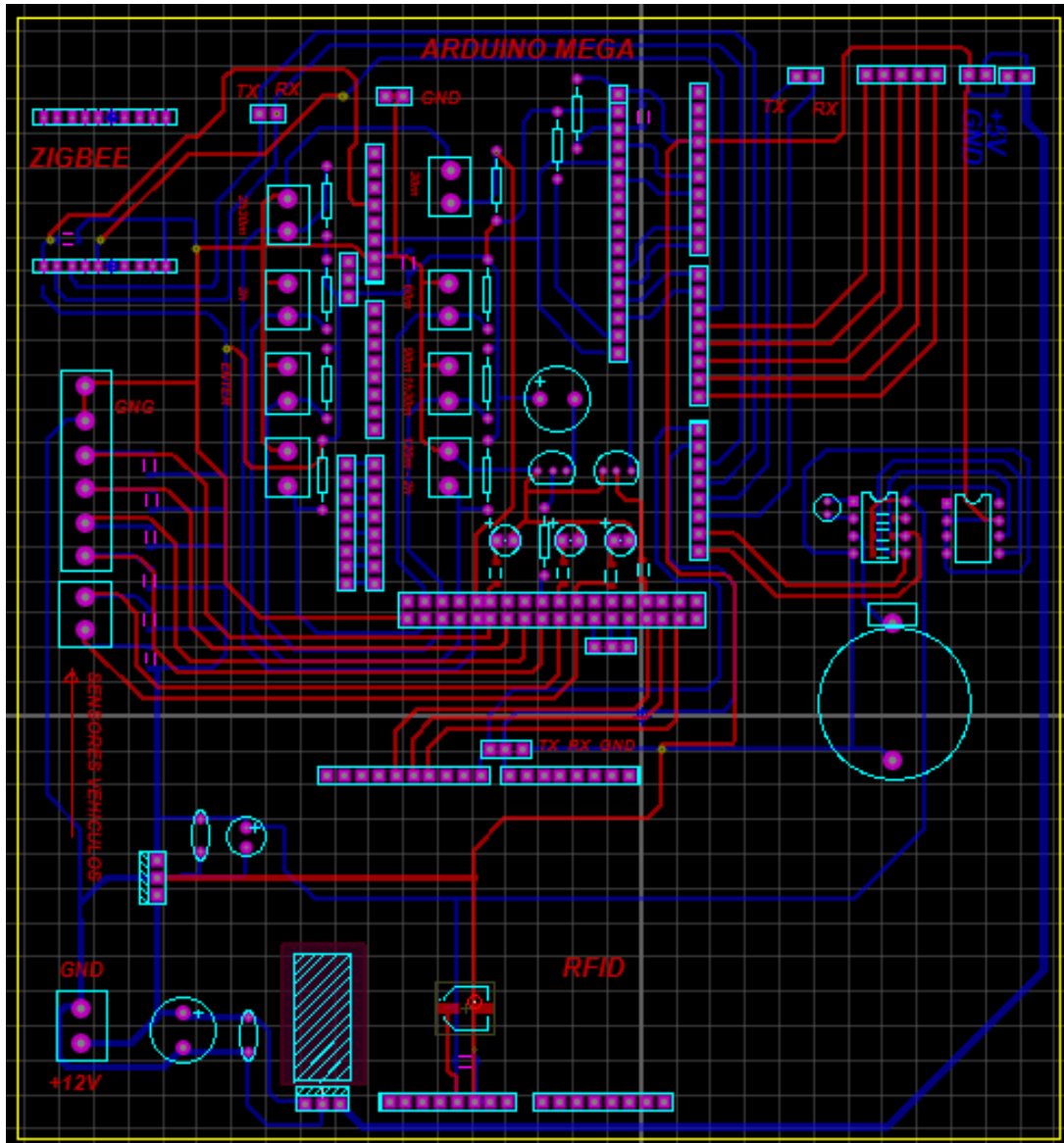
**Figura 29.** Diagrama de flujo (Parte 1)  
[Fuente: Autor]



*Figura 30.* Diagrama de flujo (Parte 2)

[Fuente: Autor]

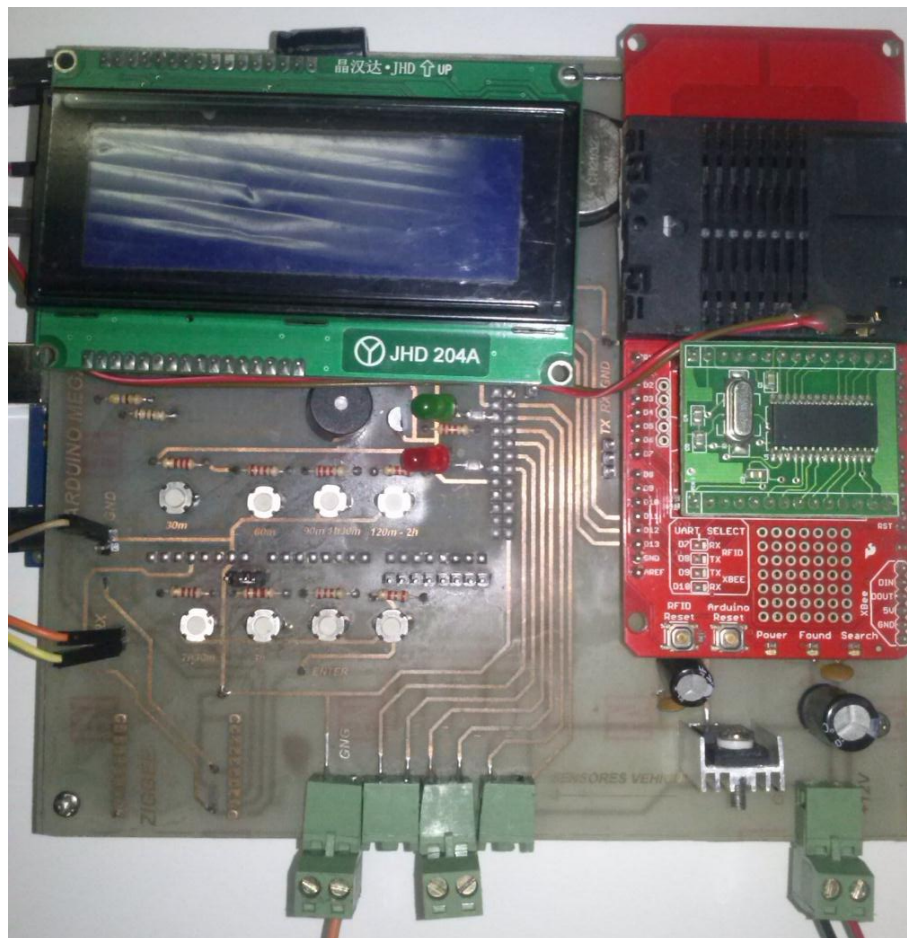
Una vez realizado el diseño en Fritzing y corregidos los errores, se empezó con la construcción física del prototipo. Para esto fue necesario diseñar el PCB en el programa Proteus; imprimir el PCB del dispositivo de control; quemar la baquelita, perforar y finalmente soldar los elementos. En la figura 31 se muestra el PCB del dispositivo de control.



*Figura 31.* PCB del dispositivo de control.

[Fuente: Autor]

En la figura 32 se puede apreciar la baquelita con todos los elementos soldados y acoplados a la misma.



**Figura 32.** Elementos soldados y acoplados en la baquelita.

[Fuente: Autor]

A la baquelita de la figura 32 se le hicieron pruebas, como medición de voltajes, comprobar que haya continuidad, etc. Luego de las pruebas realizadas, se corrigieron los errores encontrados.

A continuación se cargó la información (nombres del usuario y placa de su vehículo) en la tarjeta RFID con la utilización del programa “smRFID”. También se cargó el programa en el Arduino y se configuró los módulos XBee.

Para poder realizar la configuración de un módulo XBee se necesita del programa X-CTU, que es un programa que presenta menos problemas en comparación

con otros programas existentes en el mercado. Esta configuración es necesaria ya que el módulo XBee usado trabaja a una frecuencia de 900 MHz y es necesario configurar el canal por el cual se van a comunicar los módulos del prototipo.

**La construcción del letrero LED** para indicar las plazas disponibles se inició realizando la simulación en Proteus (ver figura 13) y posteriormente se realizó el PCB, para finalizar con la construcción física del mismo. En la figura 33 se muestra el letrero LED finalizado.



*Figura 33.* Letrero LED

**Para la elaboración** del programa que simula el **centro de control** se empezó por realizar el Algoritmo, este algoritmo fue la base para la programación en Visual Basic 8, en esta fase lo único que se hace es realizar pruebas y corregir errores en base a lo establecido en el Algoritmo.

## f.- RESULTADOS

El diseño y construcción del prototipo de parqueadero inteligente se lo realizó de manera progresiva, es decir que fue elaborado etapa por etapa hasta finalizar con la construcción del mismo. Los resultados esperados una vez ensamblado el prototipo se describen a continuación.

### f.1 RESULTADOS CON EL SISTEMA DE VISUALIZACIÓN.

En la figura 34 se muestra una imagen del sistema de visualización de plazas disponibles ensamblado en una caja de acrílico.



*Figura 34.* Sistema de visualización finalizado

De este sistema se esperan los siguientes resultados:

- ✓ Generarle al usuario ahorro de combustible y reducir el desgaste del vehículo al momento de buscar un espacio disponible.
- ✓ Minimizar las emisiones contaminantes.
- ✓ Que el tráfico sea más suave y generar un ahorro de tiempo.



- ✓ Mejorar la experiencia del usuario (generando un humor agradable) con la facilidad para estacionarse, provocándole una reducción en el nivel de stress.

## f.2 RESULTADOS CON EL DETECTOR DE VEHÍCULOS.

En la figura 35 se muestra una imagen del Loop de piso y el detector de variación de frecuencia se muestra en la figura 5, estos componentes realizan la función de detectar metales voluminosos. Los loop de piso están contruidos con un cable calibre 14 (1.628mm de diámetro) y tiene una longitud 12 metros.



*Figura 35.* Enrollamiento de alambre (Loop)

Del detector de vehículos se espera lo siguiente:

- ✓ Definir un mecanismo sencillo para la detección de vehículos, en un entorno como el presentado en nuestra ciudad.
- ✓ La detección automática disminuirá el costo de control de vehículos estacionados.

### f.3 RESULTADOS CON EL DISPOSITIVO DE CONTROL

En la figura 36 se muestra el dispositivo de control finalizado, este está ensamblado en un chasis elaborado en acrílico, el dispositivo se asemeja a un cajero electrónico y se eligió este modelo debido a que la gente está familiarizada con este tipo de dispositivos y por ende se les hará más fácil su uso.

Dentro de este chasis se encuentra el Arduino Mega, el módulo RFID, el módulo Zigbee y el reloj en tiempo real (DS1307).

En la parte externa de este dispositivo se puede visualizar la pantalla LCD, los pulsadores, la ranura para que el usuario introduzca su tarjeta RFID y la antena del módulo Zigbee.



*Figura 36.* Dispositivo de control finalizado

Del dispositivo de control se esperan los siguientes resultados.

- ✓ No será necesario que el usuario llene una tarjeta de estacionamiento, solo deberá acercarse al dispositivo de control y seguir el procedimiento.
- ✓ La identificación del usuario se realizará de manera simple, rápida y segura. El dispositivo no se tardará más que unos pocos segundos en hacer la identificación.

- ✓ El uso del dispositivo de control será simple y el diseño se lo realizó de manera que el usuario pueda familiarizarse rápidamente con él.
- ✓ Que la transmisión de datos inalámbrica sea eficaz.
- ✓ El pago por hacer uso del SIMERT se realizará de forma automática.
- ✓ Ya no será necesario que un agente de tránsito este revisando los vehículos uno por uno para controlar que el usuario haya dejado su tarjeta de SIMERT con su correspondiente registro de tiempo que estará estacionado.

#### **f.4 RESULTADOS CON EL CENTRO DE CONTROL.**

En la figura 37 se presenta la interfaz, desarrollada en Visual Basic, que se usará para cumplir con el rol de controlar todos los eventos y acciones que vengan desde el dispositivo de control.

Los resultados que espero del centro de control son:

- ✓ El sistema debe avisar cuando un cliente ha cumplido con su tiempo de estacionamiento y se le ha acabado su tiempo de gracia, para que un agente se dirija hacia el vehículo y le imponga la sanción.
- ✓ También debe avisar cuando el cliente aparcó su vehículo y no realizó el registro del mismo.
- ✓ Ya se necesitará de un agente de tránsito para que este revisando vehículo por vehículo si es que se está cumpliendo con el tiempo de estacionamiento.

# SIMERT

FECHA - HORA:

### PLAZAS OCUPADAS SIN REGISTRAR

UBICACION	LUGAR

### USUARIOS REGISTRADOS

UBICACION	PLACA	LUGAR	TIEMPO	HORA

### USUARIOS

PLACA	SALDO
LCB-2501	S
PFH-0875	S
MGH-1256	S
LSD-9623	S
TSA-4305	S
LBA-1208	N
LHK0469	N
LCG-7421	N
LKA-9431	N
LZY-0203	N

### USUARIOS SIN SALDO

UBICACION	PLACA	LUGAR	TIEMPO

*Figura 37.* Interfaz del Centro de Control  
[Fuente: Autor]

## **g.- DISCUSIÓN**

Este trabajo empezó con la necesidad de contar con un mecanismo que sea rápido y de fácil visualización para encontrar espacios disponibles de estacionamiento (movilidad).

El problema de la movilidad se solucionó con la creación de un sistema de visualización de plazas disponibles. Hecho que resulta muy beneficioso para el conductor, los peatones y para el medio ambiente.

Para realizar la detección vehicular de forma automática se usa un sensor de lazo inductivo.

Cuando el usuario ha ocupado una plaza disponible ya no es necesario que llene la tarjeta del SIMERT, solo debe acercarse con su tarjeta RFID a los dispositivos de control y ponerla en la ranura para que pueda ser leída. Después de esto solo queda seguir las instrucciones que se muestran en la pantalla LCD.

Gracias a este dispositivo ya no existe la necesidad de que un agente de tránsito o empleado del Municipio este recorriendo todas las cuadras para constatar vehículo por vehículo si tienen la tarjeta del SIMERT y cuentan con tiempo de estacionamiento o si este ya se caducó.

El centro de control, donde son procesados los datos que llegan desde el dispositivo de control, recibe y transmite datos inalámbricamente a través del protocolo IEEE 802.15.4 (Zigbee).

El centro de control es el encargado de gestionar todo en cuanto a plazas ocupadas o plazas libres, que se cumplan los tiempos de estacionamiento y avisa si el usuario tiene saldo o no.

De manera que con la automatización de todos estos procesos el personal que realiza estas acciones en el SIMERT puede reducirse en gran medida, optimizando recursos financieros en el Ilustre Municipio de Loja.

## h.- CONCLUSIONES

- ✓ El diseño de este prototipo será un referente para impulsar el adelanto tecnológico de la ciudad de Loja, con miras a convertirla en una Smart City.
- ✓ Se logró diseñar un prototipo basado en el actual funcionamiento de SIMERT, optimizándolo de manera que sea más moderno gracias a la aplicación de nuevas tecnologías.
- ✓ Si este prototipo es implementado por el Municipio, se pueden reducir gastos económicos disminuyendo el personal que opera actualmente en el SIMERT.
- ✓ El protocolo que se va a usar para la transmisión de datos es ZigBee.
- ✓ Gracias a los letreros LED que estarán ubicados en las esquinas de cada cuadra, el usuario del SIMERT podrá encontrar espacios donde estacionarse de manera rápida, ahorrando de esta manera tiempo, recursos del vehículo y se reducen la emisiones de elementos contaminantes.
- ✓ Trabajar con Arduino resultó muy conveniente ya que es fácil usarlo, permite combinar periféricos de todo tipo y no resulta muy costoso.
- ✓ Este trabajo puede servir como referente teórico y práctico para ser mejorado, ya que se pueden implementar más módulos y funciones.
- ✓ Para que la gestión sea eficiente y hayan más facilidades al momento de procesar todos los eventos relacionados con el SIMERT hace falta la creación de una **base de datos**. En esta base se debería guardar todos los datos de los usuarios del SIMERT, el saldo con el que cuentan, las infracciones que han cometido, etc.

## **i.- RECOMENDACIONES**

- ✓ Debe evitarse la presencia de mallas metálicas debajo del área ocupada por el loop de piso, debido a que eso reduce la sensibilidad de la detección.
- ✓ Realizar una programación estructurada para después encontrar fácilmente las líneas de código, ya sea para corregir errores o para usar las mismas sentencias en caso de ser necesario.
- ✓ Para mejorar la visualización de los mensajes se puede usar una pantalla LCD más grande y que sea gráfica.
- ✓ Se pueden hacer muchas mejoras al centro de control, desarrollando un programa más complejo y sofisticado que reciba la información y la almacene en una base de datos.
- ✓ Con el fin de mejorar el prototipo y añadirle más funcionalidades se puede crear una aplicación en la red la cual indique al usuario dónde hay espacios disponibles para estacionarse, el lugar exacto en el cual encontrarlos, que le permita hacer recargas electrónicas, una aplicación que le permita ver con cuánto saldo cuenta, también que le avise o de una alarma cuando su tiempo de estacionamiento este por terminar.
- ✓ En caso de que este proyecto quiera ser implementado en las calles de nuestra ciudad, se recomienda que se implemente en una calle como un plan piloto y de esta manera las personas se puedan adiestrar en el uso de este sistema.
- ✓ Se podría implementar dispositivos para que el pago se pueda efectuar de otras maneras. Por ejemplo el pago también puede hacerse con monedas y para ello se debería implementar una máquina de monedas.
- ✓ Se puede desarrollar máquinas para realizar recargas automáticas, para la venta de tarjetas y para consulta de saldo.
- ✓ Otro plus que se puede añadir al prototipo es la facturación. Se recomienda que la emisión de facturas sea de manera electrónica para de esta manera contribuir con la conservación del medio ambiente.
- ✓ Para que el dispositivo de control se asemeje más a un cajero electrónico se puede añadir una voz pregrabada para que le indique al usuario por ejemplo: que inserte su tarjeta, que la retire, o que su transacción ha sido ya sea confirmada o rechazada, etc.

- ✓ Es muy necesaria la creación de una **base de datos** que sea robusta, eficaz y escalable.
- ✓ Para disminuir los gastos económicos del dispositivo de control se recomienda que los sensores para la detección del vehículo sean elaborados y no comprados.



## j.- BIBLIOGRAFÍA

- AMETIC. 2012.** *Informe de Smart Cities*. España : Gráficas Apel, 2012.
- ARDUINO. 2015.** ARDUINO. *ARDUINO*. [En línea] 2015. [Citado el: 10 de Julio de 2015.] <https://www.arduino.cc/en/Main/arduinoBoardMega2560>.
- Arévalo Barazas, Sergio. 2011.** *Programación de un nodo conmutador para la gestión remota de redes 802.12.4/ZigBee*. Mágala : Universidad Mágala, 2011.
- Barriga Vázquez, Walter Edwin y Zúñiga Calle, Fabian Paúl. 2006.** *Tecnologías inalámbricas de corto alcance: ZigBee y Bluetooth*. Cuenca-Ecuador : UPS, 2006.
- Blázquez del Toro, Luis Miguel. 2014.** *SISTEMAS DE IDENTIFICACIÓN POR RADIOFRECUENCIA*. Madrid : s.n., 2014. ISBN -10.8.
- CFA, Corporación Andina de Fomento. 2013.** *Emprendimientos en América Latina: Desde la subsistencia hacia la transformación productiva*. Bogotá-Colombia : CAF publicaciones, 2013.
- Concepto General de los Sensores*. **Cauchic Kú, José Mario. 2009.** Argentina : Alipso, 2009, Vol. 1.
- Diario La Hora. 2015.** SIMERT en el norte sigue sin convencer. *La Hora*. 1, 2015.
- Díaz, Yohana. 2014.** Alcaldía de Loja. *Alcaldía de Loja*. [En línea] 26 de Junio de 2014. [Citado el: 15 de Abril de 2015.] <https://www.loja.gob.ec/noticia/2014-06/personal-operativo-del-simert-recibe-capacitacion>.
- Dignani, Jorge Pablo. 2011.** *Análisis del Protocolo ZigBee*. La Plata-Argentina : UNLP, 2011.
- Diseño de un sensor de Flujo Vehicular basado en Lazo Inductivo*. **Alpizar Arteaga, Ernesto y Del Risco Sánchez, Arnaldo. 2012.** 1, La Habana : SciELO, 2012, Vol. 33. 1215-5928.
- Duarte, Andrés. 2014.** Interactivo. [En línea] 2014. [Citado el: 10 de Julio de 2015.] <http://www.andresduarte.com/arduino-y-xbee>.
- DynaGroup. 2014.** *Instructivo para construcción de Loop*. Buenos Aires : s.n., 2014.
- Globatel. 2007.** *Conceptos básicos de RFID: Conocimiento y uso de la identificación por radiofrecuencia*. Quito : Intermec, 2007.
- Herrador Muñoz, David. 2013.** *Sistema de aparcamiento inteligente aplicado a las Smart Cities*. Madrid : s.n., 2013.
- Integrated, Maxim. 2015.** Maxim Integrated. [En línea] 2015. [Citado el: 19 de Julio de 2015.] <http://www.maximintegrated.com/en/products/digital/real-time-clocks/DS1307.html>.
- Ionescu, Eugen. 2013.** *Maxim DS1307*. s.l. : Microchip Technology INC, 2013.

- Los Santos Aransay, Alberto. 2009.** *Aplicación de las Redes de Sensores en el Entorno Vehicular*. Galicia-España : UVIGO, 2009.
- MEDIALAB USAL. 2012.** *Básicos-Arduino*. s.l. : MEDIALAB USAL, 2012.
- Mujica Rojas, Cristóbal. 2013.** Geeky Theory. [En línea] 5 de Abril de 2013. [Citado el: 10 de Julio de 2015.]
- Municipio de Loja.** Alcaldía de Loja. *Alcaldía de Loja*. [En línea] Municipio de Loja. [Citado el: 15 de Abril de 2015.] <https://www.loja.gob.ec/contenido/simert>.
- Muñoz Quiñónez, Iván Auriol y Zárate García, Jaime. 2009.** *Sistema de adquisición de datos para la visualización de señales eléctricas de motores AC y DC*. Bucaramanga : UPB, 2009.
- Neira Ayuso, Álvaro. 2010.** *Tutorial para la configuración de XBee*. México : s.n., 2010.
- OECD, ECMT. 2006.** *Gestión de velocidad*. París : OECD Publishing, 2006. ISBN 92-821-0377-3.
- Pomares Baeza, Jorge. 2009.** *Manual de Arduino*. Alicante : GIEA- IEA, 2009.
- Sifuentes de la Hoya, Ernesto, y otros. 2013.** *Magnetómetro de bajo costo para detectar vehículos estáticos*. Ciudad Juárez : Culcyt, 2013. 49.
- Tecnologic, Cercle. 2012.** *Hoja de ruta para la Smart City*. Barcelona : CTecno, 2012.
- Thayer Ojeda, Luis. 2015.** XBee. [En línea] Enero de 2015. [Citado el: 01 de Julio de 2015.] <http://xbee.cl/tutorial-xbee/>.
- Traffic Detector Handbook*. **Federal Highway Administration Research and Technology. 2006.** 6, Estados Unidos : U.S. FHA, 2006, Vol. 1. FHWA-HRT.
- Valdés Castro, Edgar. 2014.** *Tecnologías de información que contribuyen con las prácticas Green IT*. Cali, Colombia : s.n., 2014.
- Yaguana Tapia, William y Valdivieso León, Carlos. 2014.** *Estudio para el monitoreo de espacios disponibles para vehículos usando una red de sensores para la ciudad de Loja*. Loja : UTPL, 2014.

## k. ANEXOS

### k.1 ANEXO 1. TABLA DE PRESUPUESTO DEL PROTOTIPO.

*Tabla 7.* Detalle del costo del dispositivo de control.

CANTIDAD	MATERIAL	PRECIO	TOTAL
2	2n3904	0,1	0,2
1	Arduino	31	31
7	borneras de 2 pines	0,35	2,45
1	Caja	50	50
1	capacitor 100 uf	0,1	0,1
1	capacitor 470 uf	0,1	0,1
1	capacitor 68 uf	0,1	0,1
4	conector AC-DC para chasis	3	12
2	conector molex 2 pines	0,2	0,4
1	conector molex de 10 pines	0,8	0,8
1	conector molex de 12 pines	0,85	0,85
1	conector molex de 5 pines	0,35	0,35
3	conector molex de 8 pines	0,5	1,5
1	crystal 8 MHz	0,3	0,3
2	disipador de calor to220	0,4	0,8
1	ds1307 reloj en tiempo real	3	3
1	fuentes 12V-3A	16	16
2	l7805	0,4	0,8
1	Lcd 20x4	14	14
3	Led smd	0,5	1,5
3	Loop detector	75	225
1	metro cable multicolor	5	5
2	metros cable rojo negro #14	0,45	0,9
2	Módulos Xbee	90	180
1	pic16f876a	5	5
1	pila cr2032	1	1
7	postes 20 mm	0,5	3,5
8	Pulsadores	3,5	28
2	pulsadores smd	0,3	0,6
3	regleta de male 40 pines	0,25	0,75
1	resistor 110 ohm	0,02	0,02
10	resistor 3.9 Kohm	0,02	0,2
3	resistor 360 ohm	0,02	0,06
1	resistor 470 ohm	0,02	0,02
1	Riel	0,5	0,5
8	sujeta cables	0,5	4
3	switch on/off	0,5	1,5

1	teclado 4x4	5	5
1	tip110	0,4	0,4
14	Tornillos	0,02	0,28
1	zócalo 28 pines	0,1	0,1
1	zócalo para lipa cr2032	0,5	0,5
1	zócalo para tarjetas	3	3
2	zócalos 8 pines	0,1	0,2
1	Soporte para módulo Xbee	30	30
	<b>TOTAL</b>		<b>631,78</b>

Fuente: Autor

**Tabla 8.** Detalle del costo del sistema de visualización.

CANTIDAD	MATERIAL	PRECIO	TOTAL
2	2n3904	0,1	0,2
3	baquelita fibra de vidrio 20x30	4	12
18	bd135	0,2	3,6
3	bornera 2 pin	0,4	1,2
1	Caja	40	40
1	conector AC-DC	3	3
3	conectores moles de 4 pin	0,35	1,05
1	fuentes 24V - 4,2A	28	28
1	L7805	0,45	0,45
580	leds verde alto brillo	0,1	58
1	metro de cable rojo negro #14	0,35	0,35
2	módulo Xbee	90	180
1	pic16f876a	5	5
1	pic16f877a	6	6
10	postes de 30mm	0,75	7,5
20	resistores 150 ohm	0,02	0,4
20	resistores 2,2 Kohm	0,02	0,4
2	tip42	0,4	0,8
22	tornillos	0,02	0,44
1	uln2803	0,4	0,4
	<b>TOTAL</b>		<b>348,79</b>

Fuente: Autor

## k.2 ANEXO 2. INSTRUCTIVO PARA LA CONSTRUCCIÓN DEL LOOP

La inductancia conseguida debe ser superior a los 100  $\mu\text{H}$ , lo que se instrumenta siguiendo la **tabla 8** en función del tamaño del loop elegido. Las formas geométricas son diversas, según el tipo de vehículos a detectar. Por ejemplo, un loop para camiones debe ser mayor, para evitar el corte de detección entre camión y acoplado.

Debe evitarse la presencia de mallas metálicas debajo del área ocupada por el loop, dado que eso disminuye la sensibilidad de la detección.

El cable a emplear será de 0,75 a 1  $\text{mm}^2$  de sección. Se evitará el roce del mismo contra ángulos filosos que puedan dañar su aislación. Es necesario no realizar ángulos agudos, para evitar la pérdida de aislación. La salida hacia el equipo se hará retorciendo los dos conductores resultantes entre sí, para disminuir capacidades parásitas.

**Tabla 9.** Construcción del Loop en función del tamaño

Dimensiones Loop (metros)	Cantidad mínima de vueltas	Modelo Estándar	Modelo Industrial	Modelo Extendido
1 x 1	8	✓	✓	-
2 x 0,5 *	6	✓	✓	-
2 x 1	6	✓	✓	-
2 x 2 *	5	✓	✓	-
3 x 1	5	✓	✓	-
4 x 1	5	✓	✓	-
5 x 1	5	-	-	✓
6 x 1	5	-	-	✓
7 x 1	5	-	-	✓
8 x 1	5	-	-	✓
9 x 1	5	-	-	✓
10 x 1	5	-	-	✓
* Loop con formato trapezoidal. A continuación se detallan las medidas precisas para su construcción.				

Fuente: (DynaGroup, 2014)

## CONSTRUCCIÓN

1) **PASO 1: Dibujo.**

Dibuje en el piso la geometría del loop elegido.

2) **PASO 2: Corte del piso**

Realizar el corte en el piso de 2,5 cm de profundidad y 3 a 4mm de ancho (ancho de sierra). Realice cortes diagonales en las 4 esquinas para evitar ángulos filosos que puedan dañar el cable.

3) **PASO 3: Extensión del corte.**

Dibuje el recorrido desde la esquina del loop más cercana hasta el lugar donde se colocará el Detector de masa metálica (DMM). Posteriormente, corte con la sierra.

4) **PASO 4: Limpieza.**

Limpie la zona con una escoba para eliminar restos de polvo. Luego ayúdese de una espátula u otro elemento para eliminar la suciedad dentro del corte.

5) **PASO 5: Colocación de espiras.**

Utilice cable de 0,75 a 1 mm<sup>2</sup>. Comience dejando un tramo extra de 1,5m al comienzo del loop. Coloque el cable dentro del corte ayudándose con una espátula. La cantidad de vueltas dependerá de la geometría elegida (ver tabla) siendo 5 el mínimo de vueltas.

6) **PASO 6: Sellado**

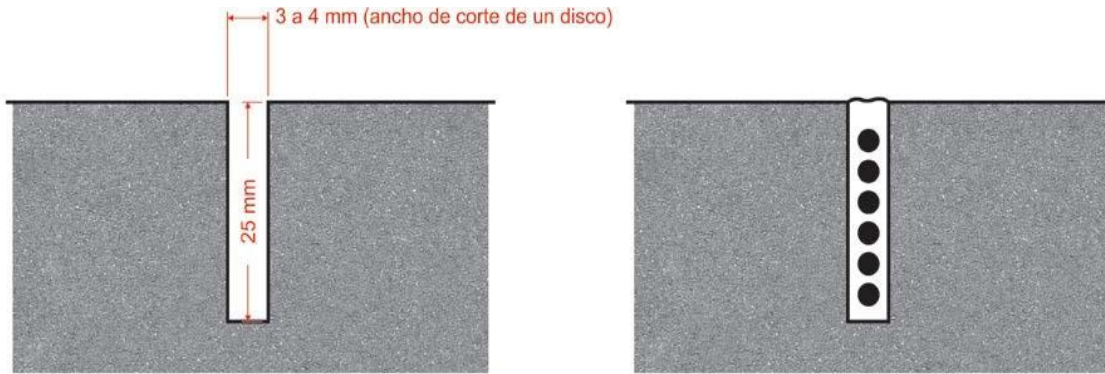
Rellene prolijamente el corte con sellador poliuretánico. Una alternativa es usar alquitrán.

7) **PASO 7: Nivelación del sellador**

Ayúdese de un trapo o una esponja para esparcir y nivelar el sellador.

8) **PASO 8: Trenzado.**

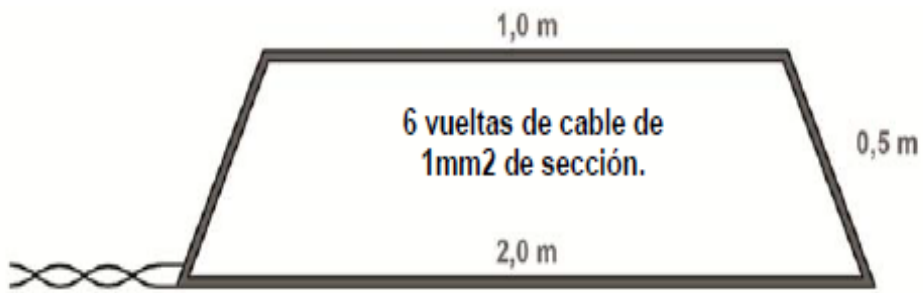
El último paso anterior a la conexión con el DMM es retorcer entre sí ambos extremos del cable dando aproximadamente 20 vueltas por metro de cable.



*Figura 38.* Perforación en el piso para el Loop

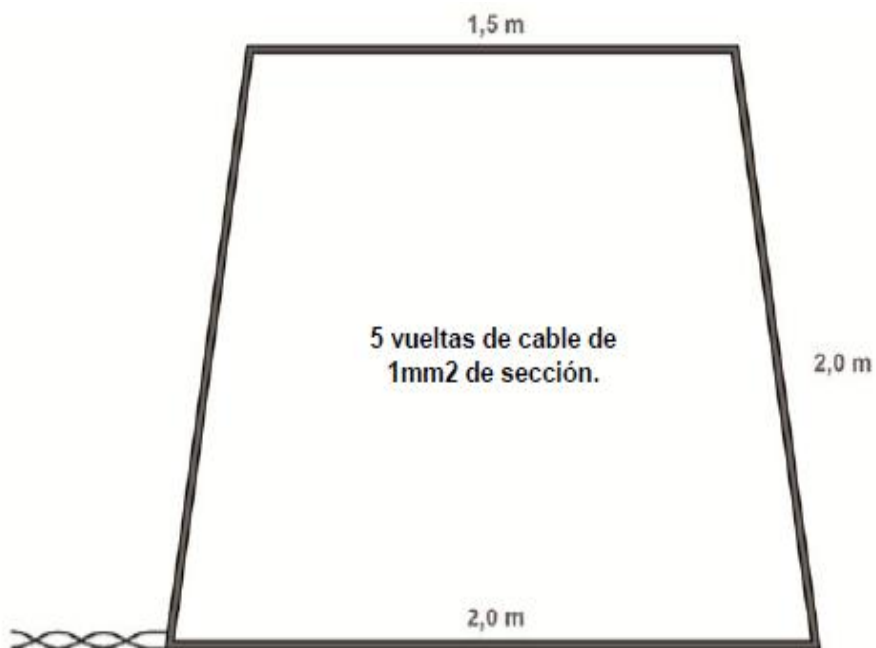
## GEOMETRÍAS TRAPEZOIDALES

### LOOP PARA VEHÍCULOS LIVIANOS SOLAMENTE



*Figura 39.* Loop para vehículos livianos.

### LOOP PARA ACOPLADO U ÓMNIBUSES



*Figura 40.* Loop para vehículos pesados.

## k.3 ANEXO 3. DESCRIPCIÓN DEL MÓDULO XBEE PRO 900



**XBee-PRO® 900HP**  
900 MHz RF Module

Best-in-class range RF module features software-complete mesh or multipoint topologies with advanced sleep modes.

### Overview

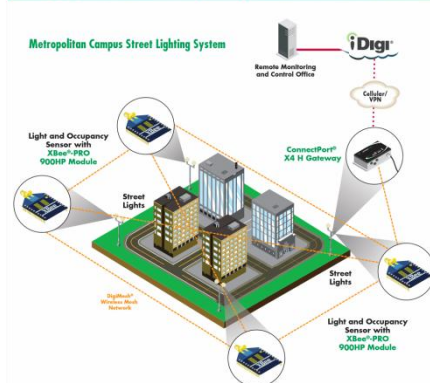
XBee-PRO 900HP embedded modules provide best-in-class range wireless connectivity to devices. They take advantage of the DigiMesh® networking protocol, featuring dense network operation and support for sleeping routers, and are also available in a proprietary point-to-multipoint configuration. Supporting RF line-of-sight ranges up to 28 miles (with high-gain antennas), and data rates of up to 200 Kbps, these modules are ideal for extended-range applications requiring increased data throughput.

The XBee-PRO 900HP requires no programming and can be configured easily using Digi's free X-CTU software or via a simplified AT command set. XBee modules are pre-certified for use in multiple countries, further reducing development costs and time to market.

XBee® modules are easy to use, share a common hardware footprint and are available in a variety of different protocols, enabling users to substitute one XBee for another with minimal development time and risk.

### Application Highlight

**Metropolitan Campus Street Lighting System**



The diagram illustrates a mesh network of XBee-PRO 900HP modules installed on streetlights. These modules communicate with a central gateway (ConnectPort® X4 H Gateway) which is connected to a Remote Monitoring and Control Office. The office is also connected to a Cellular/VN network. The system is used for monitoring and controlling streetlights in a metropolitan campus.

### Features/Benefits

- Superior outdoor LOS range of up to 28 miles
- Simplified AT command set or advanced XBee API accelerates time to market
- Over-the-air firmware updates via other XBee-PRO 900HP modules or iDigi® Device Cloud™
- Software-selectable channel mask for interference immunity
- Advanced sleep modes include sleeping routers, pin sleep, cyclic sleep
- RF data rate up to 200 Kbps
- Patent-pending channel selectivity algorithm allows for more dense networks

### Related Products



Gateways      Development Kits      ZigBee Modules



www.digi.com



## XBee-PRO® 900HP

Hardware	
Processor	ADF7023 transceiver, Cortex-M3 EFM32G230 @ 28 MHz
Frequency Band	902 to 928 MHz, software selectable channel mask for interference immunity
Antenna Options	Wire, U.FL and RPSMSA
Performance	
RF Data Rate	10 Kbps or 200 Kbps
Indoor/Urban Range	up to 2000 ft (610 m)
Outdoor/ Line-Of-Sight Range	Up to 9 miles (14 km) w/ dipole antenna; Up to 28 miles (45 km) w/ high-gain antenna
Transmit Power	Up to 24 dBm (250 mW) software selectable
Receiver Sensitivity	-101 dBm @ 200 Kbps, -110 dBm @ 10 Kbps
Features	
Data Interface	UART (3V), SPI
GPIO	Up to 15 Digital I/O, 4 10-bit ADC inputs, 2 PWM outputs
Networking Topologies	DigiMesh, Repeater, Point-to-Point, Point-to-Multipoint, Peer-to-Peer
Spread Spectrum	FHSS (Software Selectable Channels)
Power	
Supply Voltage	2.1 to 3.6 VDC
Transmit Current	215 mA
Receive Current	29 mA
Sleep Current	2.5 uA
Regulatory Approvals	
FCC (USA)	MCQ-XB900HP
IC (Canada)	1846A-XB900HP
C-Tick (Australia)	Yes
Anatel (Brazil)	Pending

#### k.4 ANEXO 4. ALGORITMO DEL ARDUINO

```
#include <LiquidCrystal.h>
#include <Wire.h>
#include "RTCLib.h"

RTC_DS1307 rtc;

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 9, 8, 7, 6);
//declaracion de parametros del dispositivo
String codigo="COLONPAR01#";
String comand1="OCUPADO#";
String comand2="TIME#";
String comand3="DESOCUPADO#";
String Time;
char Lugares[10];
char pos;

//definición de pines a utilizar
char contadorlugar=0;
char comandtime=0;
char tag=2;
char zona=3;
char parking1=45;
char parking2=43;
char parking3=41;
char parking4=39;
char parking5=37;
char parking6=35;
char parking7=33;
char parking8=31;
```

```
char parking9=29;
char parking10=27;

char pulsador1=48;//
char pulsador2=51;//
char pulsador3=53;
char pulsador4=52;//
char pulsador5=46;//
char pulsador6=47;//
char pulsaSi=49;
char pulsaNo=50;
char redled=44;
char greenled=42;
char orangeled=40;
char buzer=38;
char lcdPower =36;
```

```
//declaración de variables a utilizar
char ocupado1=0;
char ocupado2=0;
char ocupado3=0;
char ocupado4=0;
char ocupado5=0;
char ocupado6=0;
char SensorActivado=0;
char SensorDesactivado=0;
boolean ConectServer=0;
char cntVehiculo=0;
char tagString[25];
char readRfid=0;
char tagok=0;
char ocupadook=0;
```

```

int index = 0;
char readByte;
char ServerData[25];
char cedula[11];
char placa[10];
int indexServer=0;
char readServer;
char seekAc=0;
char cont;
char b=0;
char b1=0;
char b2=0;
char b3=0;
char b4=0;
char b5=0;
char b6=0;
char b7=0;
char Indn1;
char Indn2;
char Indn3;
char contn;
char ocupado=0;
char tiempo=0;
char hora;
char minu;
char teclado=0;
char lugar;
char lugar2;
char cntTeclado=0;
char PulsaActivado=0;
char teclaActivado=0;
char PSi;

```

```

char PNo;
char P30;
char P1h;
char P1hm;
char P2h;
char P2hm;
char P3h;
char conexion=2;
char buscartag;
char zonaAB=0;
char comando=0;
int contador=0;
char lugarok=0;

```

**//rfid**

```

char tag1[7] = {0xFF,0x06,0x82,0x02,0x85,0x4C,0x00};
char tagleido=0;
char sb;
char indice=0;
char b0=0;
char errortag=0;
char cnterror=0;
char nombre1[16];
char apellido[16];

```

**//declaración de mensajes**

```

String msj1="BIENVENIDO AL SIMERT";
String msj13="HORA: H MIN";
String msj14="RED NO DISPONIBLE";
String msj15="LLENAR TARJETA";
String msj22="PLACA #:";
String msj24="ACEPTAR";

```

```

String msj25="TRASACCION TERMINADA";
String msj26="POR EL USUARIO ";
String msj27="TRASACCION CANCELADA";
String msj31="INGRESAR LUGAR DONDE";
String msj32="ESTA UBICADO SU";
String msj33="VEHICULO";
String msj42="ELIJA EL TIEMPO DE";
String msj43="ESTACIONAMIENTO";
String msj51="EL TIEMPO QUE USTED";
String msj52="SELECCIONO ES:";
String msj62="TRANSACCION EXITOSA";
String msj63="GRACIAS";
String msj71="SU TRANSACCION NO SE";
String msj72="REALIZO POR SALDO";
String msj73="INSUFICIENTE";
String msj74="REALIZAR UNA RECARGA";
String msj81="RETIRE SU TARJETA";
String msj82="POR FAVOR";
String msj91="TARJETA INVALIDA";
String msj92="ERROR EN LECTURA";
String msj93="DE TARJERA";

//declaración de funciones
void detectar_vehiculo(void);
void seek1(void);
void leerRFID(void);
void verificar(void);
void leerPulsador(void);
void antirevotote(void);
void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(20, 4);

```

```

#ifdef AVR
  Wire.begin();
#else
  Wire1.begin(); // Shield I2C pins connect to alt I2C bus on
Arduino Due
#endif
  rtc.begin();
  if (! rtc.isrunning() ) {
    Serial.println("RTC is NOT running!");
    // following line sets the RTC to the date & time this sketch was
compiled
    // rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    // This line sets the RTC with an explicit date & time, for example
to set
    // January 21, 2014 at 3am you would call:
    //rtc.adjust(DateTime(2015,10,10,23,10,0));
  }
//inicializar como entrada los pines 35 37 39 41 43 45 donde están
los sensores de vehículos
pinMode(parking1, INPUT);
pinMode(parking2, INPUT);
pinMode(parking3, INPUT);
pinMode(parking4, INPUT);
pinMode(parking5, INPUT);
pinMode(parking6, INPUT);

//inicializar como entrada los pines donde están los pulsadores
pinMode(pulsador1, INPUT);
pinMode(pulsador2, INPUT);
pinMode(pulsador3, INPUT);
pinMode(pulsador4, INPUT);
pinMode(pulsador5, INPUT);

```

```
pinMode(pulsador6, INPUT);
pinMode(pulsaSi, INPUT);
pinMode(pulsaNo, INPUT);
pinMode(tag, INPUT);
pinMode(zona, INPUT);
```

**//inicialización de pines de salida**

```
pinMode(redled, OUTPUT);
pinMode(greenled, OUTPUT);
pinMode(orangeled, OUTPUT);
pinMode(buzer, OUTPUT);
pinMode(lcdPower, OUTPUT);
digitalWrite(lcdPower, LOW);
```

**// Mostrar mensajes en la LCD.**

```
lcd.print(" SIMERT ");
Serial.begin(9600);
Serial.println("Start");
Serial1.begin(9600);
Serial2.begin(9600);
Serial3.begin(115200);
delay(1000);
SensorActivado=0;
digitalWrite(lcdPower,LOW);
```

```
if(digitalRead(zona)==1){
  zonaAB=1;
}
if(digitalRead(zona)==0){
  zonaAB=0;
}
}
```

**//CICLO REPETITIVO**

```
void loop() {
  DateTime now = rtc.now();
  hora=now.hour();
  minu=now.minute();

  if(tagleido==0){
    // Serial.println(conexion+48);
    if(conexion==2){
      lcd.setCursor(0, 0);
      lcd.print(msj1);//BIENVENIDO AL SIMERT
    }
    if(conexion==1){
      lcd.setCursor(0, 0);
      lcd.print(msj1);//BIENVENIDO AL SIMERT
    }
    if(conexion==0){
      lcd.setCursor(0, 0);
      lcd.print(msj14);//RED NO DISPONIBLE
      lcd.setCursor(3, 1);
      lcd.print(msj15);//LLENAR TARJETA
    }
    lcd.setCursor(3, 2);
    lcd.print(msj13);//HORA: H MIN
    lcd.setCursor(8, 2);
    lcd.print(hora,DEC);
    lcd.setCursor(11, 2);
    lcd.print(minu,DEC);
  }
  leerServer();
  verificarComando();
  if(comando==1){
```

```

contador++;
Serial.println("sins");
if(contador>10){
  Serial.println("tiempo");
  contador=0;
  conexion=0;
  lcd.clear();
}
}
if(tagok==1){
  cntVehiculo--;
  lcd.clear();
  lcd.setCursor(0, 1);
  lcd.print(msj62);//TRANSACCION EXITOSA
  lcd.setCursor(7, 3);
  lcd.print(msj63);//GRACIAS
  tagleido=0;
  delay(4000);
  retirartarjeta();
  tagok=0;
}
if(tagok==2){
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(msj71);//SU TRANSACCION NO SE
  lcd.setCursor(1, 1);
  lcd.print(msj72);//REALIZO POR SALDO
  lcd.setCursor(4, 2);
  lcd.print(msj73);//INSUFICIENTE
  lcd.setCursor(0, 3);
  lcd.print(msj74);//REALIZAR UNA RECARGA
  tagleido=0;

```

```

  delay(4000);
  retirartarjeta();
  tagok=0;
}
detectar_vehiculo();
if(SensorActivado==1){
  cntVehiculo++;
  enviarServer(1);
  SensorActivado=0;
  // digitalWrite(lcdPower, HIGH);
}
if(SensorDesactivado==1){
  SensorDesactivado=0;
  enviarServer(3);
}
if(cntVehiculo>0){buscartag=1;}
if(cntVehiculo==0){buscartag=0;}
if(buscartag==1){
  detectartag();
}
if(tagleido==1){
  interactuar();
}
if(errortag==1){
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(msj91);//TARJETA INVALIDA
  delay(3000);
  lcd.clear();
  errortag=0;
  tagleido=0;
  retirartarjeta();

```

```

}
if(errortag==2){
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(msj91);//TARJETA INVALIDA
  delay(3000);
  lcd.clear();
  errortag=0;
  tagleido=0;
  retirartarjeta();
}
if(errortag==3){
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(msj92);//ERROR EN LECTURA
  lcd.setCursor(0, 1);
  lcd.print(msj93);//DE TARJERA
  delay(3000);
  lcd.clear();
  errortag=0;
  tagleido=0;
  retirartarjeta();
}
}
void beep(){
digitalWrite(buzer, HIGH);
delay(50);
digitalWrite(buzer, LOW);
}
void retirartarjeta(){
  lcd.clear();
  lcd.setCursor(2, 0);

```

```

  lcd.print(msj81);//RETIRE SU TARJETA
  lcd.setCursor(5, 2);
  lcd.print(msj82);//POR FAVOR
  delay(3000);
  while(b0==2){
    detectartag();
    delay(50);
    beep();
  }
  lcd.clear();
}
void interactuar(){
  tagleido=2;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(nombre1);
  lcd.setCursor(9, 0);
  lcd.print(apellido);
  lcd.setCursor(0, 2);
  lcd.print(msj22);//PLACA #:
  lcd.setCursor(9, 2);
  lcd.print(placa);
  lcd.setCursor(7, 3);
  lcd.print(msj24);//ACEPTAR
  PulsaActivado=0;
  while(PulsaActivado==0){
    leerPulsadorSN();
  }
  if(PulsaActivado==2){PNo=1;}
  if(PNo==1){
    PNo=0;
    lcd.clear();

```

```

lcd.setCursor(0, 0);
lcd.print(msj25);//TRASACCION TERMINADA
lcd.setCursor(3, 1);
lcd.print(msj26);//POR EL USUARIO
lcd.setCursor(6, 2);
lcd.print(msj63);//GRACIAS
delay(4000);
retirartarjeta();
lcd.clear();
tagleido=0;
}
if(PSi==1){
  PSi=0;
  elegirlugar();
  if(lugarok==0){
    lcd.clear();
    lcd.setCursor(7, 0);
    lcd.print("ERROR");
    lcd.setCursor(2, 2);
    lcd.print("VERIFICAR LUGAR");
    lcd.setCursor(1, 3);
    lcd.print("DE ESTACIONAMIENTO");
    delay(3000);
    PNo=1;
    PSi=0;
  }
  if(lugarok==2){
    lcd.clear();
    lcd.setCursor(2, 0);
    lcd.print("LUGAR YA ASIGNADO");
    lcd.setCursor(2, 2);
    lcd.print("VERIFICAR LUGAR");
  }
}

```

```

lcd.setCursor(1, 3);
lcd.print("DE ESTACIONAMIENTO");
delay(3000);
PNo=1;
PSi=0;
}
if (PNo==1){
  PNo=0;
  elegirlugar();
  if(lugarok==0){
    lcd.clear();
    lcd.setCursor(7, 0);
    lcd.print("ERROR");
    lcd.setCursor(2, 2);
    lcd.print("VERIFICAR LUGAR");
    lcd.setCursor(1, 3);
    lcd.print("DE ESTACIONAMIENTO");
    delay(3000);
    PNo=1;
    PSi=0;
  }
  if(lugarok==2){
    lcd.clear();
    lcd.setCursor(2, 0);
    lcd.print("LUGAR YA ASIGNADO");
    lcd.setCursor(2, 2);
    lcd.print("VERIFICAR LUGAR");
    lcd.setCursor(1, 3);
    lcd.print("DE ESTACIONAMIENTO");
    delay(3000);
    PNo=1;
    PSi=0;
  }
}

```



```

    }
}
if (PNo==1){
    PNo=0;
    elegirlugar();
    if(lugarok==0){
        lcd.clear();
        lcd.setCursor(7, 0);
        lcd.print("ERROR");
        lcd.setCursor(2, 2);
        lcd.print("VERIFICAR LUGAR");
        lcd.setCursor(1, 3);
        lcd.print("DE ESTACIONAMIENTO");
        delay(3000);
        PNo=1;
        PSi=0;
    }
    if(lugarok==2){
        lcd.clear();
        lcd.setCursor(2, 0);
        lcd.print("LUGAR YA ASIGNADO");
        lcd.setCursor(2, 2);
        lcd.print("VERIFICAR LUGAR");
        lcd.setCursor(1, 3);
        lcd.print("DE ESTACIONAMIENTO");
        delay(3000);
        PNo=1;
        PSi=0;
    }
}
if (teclaActivado==2){tagleido=0;lcd.clear();PNo=1;}
if (PNo==1){

```

```

    PNo=0;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(msj27);//TRASACCION CANCELADA
    lcd.setCursor(6, 2);
    lcd.print(msj63);//GRACIAS
    delay(4000);
    retirartarjeta();
    lcd.clear();
    tagleido=0;
}
if(PSi==1){
    PSi=0;
    elegirTime();
    if(PSi==1){
        PSi=0;
        enviarServer(2);
        lcd.clear();
        lcd.setCursor(2, 0);
        lcd.print("ESPERE POR FAVOR");
    }
}
if(PNo==1){
    PNo=0;
    elegirTime();
}
if(PSi==1){
    PSi=0;
    enviarServer(2);
}
if(PNo==1){
    PNo=0;
    elegirTime();
}

```

```

}
if(Psi==1){
  Psi=0;
  enviarServer(2);
}
if(PulsaActivado==2){PNo=1;}
if(PNo==1){
  PNo=0;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(msj27);//TRASACCION CANCELADA
  lcd.setCursor(6, 2);
  lcd.print(msj63);//GRACIAS
  delay(4000);
  retirartarjeta();
  lcd.clear();
  tagleido=0;
}
}
}
}
//función para elegir lugar de estacionamiento
void elegirlugar(){
  lugarok=0;
  cntTeclado=0;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(msj31);//INGRESAR LUGAR DONDE
  lcd.setCursor(2, 1);
  lcd.print(msj32);//ESTA UBICADO SU
  lcd.setCursor(6, 2);

```

```

lcd.print(msj33);//VEHICULO
teclaActivado=0;
while(teclaActivado==0){
  leerteclado();
  if(digitalRead(tag)==0){b0=0;teclaActivado=2;}
}
if(teclaActivado==1){
  lcd.setCursor(5, 3);
  lcd.print(teclado);
  lcd.setCursor(8, 3);
  lcd.print(msj24);//"ACEPTAR
  teclaActivado=0;
  PulsaActivado=0;
  delay(50);
  while(PulsaActivado==0){
    leerteclado();
    leerPulsadorSN();
    if(teclaActivado==1){PulsaActivado=1;}
    if(digitalRead(tag)==0){b0=0;teclaActivado=2;}
  }
  if (teclaActivado==1){
    lcd.setCursor(6, 3);
    lcd.print(teclado);
    PulsaActivado=0;
    while(PulsaActivado==0){
      leerPulsadorSN();
    }
  }
}
//Serial.println(lugar);
if (lugar>48){
  // Serial.println("lugar1");

```



```

beep();
teclado = Serial1.read();
//Serial.write(teclado);
teclaActivado=1;
cntTeclado++;
if(cntTeclado==1){
  lugar=teclado;
  Serial.println("contadorteclado1");
}
if(cntTeclado==2){
  lugar2=teclado;
  Serial.println("contadorteclado2");
  cntTeclado=0;
}
}
}
//función para detectar que un vehículo se estaciono
void detectar_vehiculo(){
  digitalWrite(redled, HIGH);
  delay(100);
  digitalWrite(redled, LOW);
  delay(100);
  if((digitalRead(parking1)==1)&&(b1==0)){b1=1;}
  if((digitalRead(parking1)==0)&&(b1==1)){
    delay(25);
  if(digitalRead(parking1)==0){b1=2;ocupado1=1;SensorActivado=1;}
  }
  if((digitalRead(parking1)==1)&&(b1==2)){
    delay(25);
  if(digitalRead(parking1)==1){b1=0;ocupado1=2;SensorDesactivado=1;}
  }
}

```

```

}
if((digitalRead(parking2)==1)&&(b2==0)){b2=1;}
if((digitalRead(parking2)==0)&&(b2==1)){
  delay(25);
  if(digitalRead(parking2)==0){b2=2;ocupado2=1;SensorActivado=1;}
}
if((digitalRead(parking2)==1)&&(b2==2)){
  delay(25);
  if(digitalRead(parking2)==1){b2=0;ocupado2=2;SensorDesactivado=1;}
}
if((digitalRead(parking3)==1)&&(b3==0)){b3=1;}
if((digitalRead(parking3)==0)&&(b3==1)){
  delay(25);
  if(digitalRead(parking3)==0){b3=2;ocupado3=1;SensorActivado=1;}
}
if((digitalRead(parking3)==1)&&(b3==2)){
  delay(25);
  if(digitalRead(parking3)==1){b3=0;ocupado3=2;SensorDesactivado=1;}
}
if((digitalRead(parking4)==1)&&(b4==0)){b4=1;}
if((digitalRead(parking4)==0)&&(b4==1)){
  delay(25);
  if(digitalRead(parking4)==0){b4=2;ocupado4=1;SensorActivado=1;}
}
if((digitalRead(parking4)==1)&&(b4==2)){
  delay(25);
}
}
}

```

```

if(digitalRead(parking4)==1){b4=0;ocupado4=2;SensorDesactiva
do=1;}
}
if((digitalRead(parking5)==1)&&(b5==0)){b5=1;}
if((digitalRead(parking5)==0)&&(b5==1)){
  delay(25);
}
if(digitalRead(parking5)==0){b5=2;ocupado5=1;SensorActivado
=1;}
}
if((digitalRead(parking5)==1)&&(b5==2)){
  delay(25);
}
if(digitalRead(parking5)==1){b5=0;ocupado5=2;SensorDesactiva
do=1;}
}
if((digitalRead(parking6)==1)&&(b6==0)){b6=1;}
if((digitalRead(parking6)==0)&&(b6==1)){
  delay(25);
}
if(digitalRead(parking6)==0){b6=2;ocupado6=1;SensorActivado
=1;}
}
if((digitalRead(parking6)==1)&&(b6==2)){
  delay(25);
}
if(digitalRead(parking6)==1){b6=0;ocupado6=2;SensorDesactiva
do=1;}
}
}
//función para detectar una tarjeta en la ranura
void detectartag(){
  if((digitalRead(tag)==0)&&(b0==0)){b0=1;}
  if((digitalRead(tag)==1)&&(b0==1)){
    delay(25);
    if(digitalRead(tag)==1){b0=2;ReadTagRfid();}
  }
}

```

```

}
if((digitalRead(tag)==0)&&(b0==2)){
  delay(25);
  if(digitalRead(tag)==0){b0=0;}
}
}
//función que resetea el módulo RFID
void halt(){
  //Serial.println("halt");
  //search for RFID tag
  Serial3.write(255);
  Serial3.write(0);
  Serial3.write(1);
  Serial3.write(147);
  Serial3.write(148);
  delay(2);
  Serial3.write(255);
  Serial3.write(0);
  Serial3.write(2);
  Serial3.write(147);
  Serial3.write(76);
  Serial3.write(225);
  delay(2);
}
//función que activa el lector de tarjetas rfid para buscar una tarjeta
void seek1(){
  seekAc=1;
  // Serial.println("buscando rfid");
  //search for RFID tag
  Serial3.write(255);
  Serial3.write(0);
  Serial3.write(1);
}

```

```

Serial3.write(130);
Serial3.write(131);
delay(2);
}
//función que envía el comando para autenticar un tag rfid
void autentica(char block){
// Serial.println("autentica");
//search for RFID tag
Serial3.write(255);
Serial3.write(0);
Serial3.write(3);
Serial3.write(133);
if(block==1){
  Serial3.write(1);
  Serial3.write(255);
  Serial3.write(136);
}
if(block==2){
  Serial3.write(2);
  Serial3.write(255);
  Serial3.write(137);
}
if(block==4){
  Serial3.write(4);
  Serial3.write(255);
  Serial3.write(139);
}
if(block==5){
  Serial3.write(5);
  Serial3.write(255);
  Serial3.write(140);
}
}

```

```

delay(10);
}
//función que envía el comando para autenticar un tag rfid con la
key guardada en el modulo
void autentica2(char block){
//Serial.println("autentica");
//search for RFID tag
Serial3.write(255);
Serial3.write(0);
Serial3.write(3);
Serial3.write(133);
if(block==1){
  Serial3.write(1);
  Serial3.write(16);
  Serial3.write(153);
}
if(block==2){
  Serial3.write(2);
  Serial3.write(16);
  Serial3.write(154);
}
}
delay(10);
}
//función que envía el comando para leer un tag rfid
void readComand(char block){
//Serial.println("read comand");
//search for RFID tag
Serial3.write(255);
Serial3.write(0);
Serial3.write(2);
Serial3.write(134);
if (block==1){

```

```

Serial3.write(1);
Serial3.write(137);
}
if (block==2){
Serial3.write(2);
Serial3.write(138);
}
if (block==4){
Serial3.write(4);
Serial3.write(140);
}
if (block==5){
Serial3.write(5);
Serial3.write(141);
}
delay(10);
}
// función leer el lector de tarjetas rfid que está en el puerto serie 3
void leerRFID(){
index=0;
readRfid=0;
//Serial.println("leer rfid");
if(Serial3.available() {
//Serial.print("reading Serial String: "); //optional
confirmation
delay(200);
while (Serial3.available()){
readByte = Serial3.read(); //read next available byte
tagString[index] = readByte;
//Serial.write(tagString[index]);
if(readByte==tag1[0]){
indice=index;

```

```

readRfid=1;
//Serial.println("ok");
}
index++;
}
}
}
void identificarRfid(){
//Serial.println("verificando");
//verifica la lectura del codigo del tag
if(tagString[indice+2]==tag1[1]){
if(tagString[indice+3]==tag1[2]){
if(tagString[indice+4]==tag1[3]){
seekAc=2;
//Serial.println("lectura de codigo ok");
}
}
}
}
//verifica que la autentificacion es correcta
if(tagString[indice+2]==tag1[3]){
if(tagString[indice+3]==tag1[4]){
if(tagString[indice+4]==tag1[5]){
seekAc=3;
//Serial.println("autentica b1 ok");
}
}
}
}
}
void ReadTagRfid(){
errortag=0;
tagleido=0;
halt();

```

```

seek1();
leerRFID();
if(readRfid==1){identificarRfid();readRfid=0;}
if (seekAc!=2){errortag=1;}
if(errortag==1){Serial.println("error");}
if(errortag==0){
  if(seekAc==2){autentica(1);seekAc=0;}
  leerRFID();
  if(readRfid==1){identificarRfid();readRfid=0;}
  if (seekAc!=3){errortag=2;}
  if(errortag==2){Serial.println("error2");}
  if(errortag==0){
    if(seekAc==3){readComand(1);seekAc=0;}
    leerRFID();
    indice=indice+5;
    for(int x=0;x<10;x++){
      if(tagString[indice+x]==tag1[6]){
        cnterror++;
      }
    }
    if(cnterror>7){errortag=3;}
    if(errortag==3){Serial.println("error3");}
    for(int x=0;x<10;x++){
      cedula[x]=tagString[indice+x];
    }
    halt();
    seek1();
    leerRFID();
    if(readRfid==1){identificarRfid();readRfid=0;}
    if(seekAc==2){autentica(2);seekAc=0;}
    leerRFID();
    if(readRfid==1){identificarRfid();readRfid=0;}

```

```

if(seekAc==3){readComand(2);seekAc=0;}
leerRFID();
indice=indice+5;
for(int x=0;x<8;x++){
  placa[x]=tagString[indice+x];
}
halt();
seek1();
leerRFID();
if(readRfid==1){identificarRfid();readRfid=0;}
if(seekAc==2){autentica(4);seekAc=0;}
leerRFID();
if(readRfid==1){identificarRfid();readRfid=0;}
if(seekAc==3){readComand(4);seekAc=0;}
leerRFID();
indice=indice+5;
for(int x=0;x<15;x++){
  nombre1[x]=tagString[indice+x];
}
halt();
seek1();
leerRFID();
if(readRfid==1){identificarRfid();readRfid=0;}
if(seekAc==2){autentica(5);seekAc=0;}
leerRFID();
if(readRfid==1){identificarRfid();readRfid=0;}
if(seekAc==3){readComand(5);seekAc=0;}
leerRFID();
indice=indice+5;
for(int x=0;x<15;x++){
  apellido[x]=tagString[indice+x];
}

```



```

tagleido=1;
Serial.print("nombre: ");
Serial.print(nombre1);
Serial.print(" ");
Serial.println(apellido);
Serial.print("cedula: ");
Serial.println(cedula);
Serial.print("placa: ");
Serial.println(placa);
}
}
}
void leerServer(){
  indexServer=0;
  contn=0;
  if(Serial2.available()) {
    //Serial.print("reading Serial String: "); //optional
    confirmation
    delay(1000);
    while (Serial2.available()){
      readServer = Serial2.read(); //read next available byte
      Serial.print(readServer);
      ServerData[indexServer] = readServer;
      indexServer++;
      if(readServer==35){
        contn++;
      }
      if(contn==1){Indn1=indexServer;conexion=1;comando=0;lcd.cle
ar();Serial.print(" n1 ");}
      if(contn==2){Indn2=indexServer;Serial.print(" n2 ");}
      if(contn==3){Indn3=indexServer;Serial.print(" n3 ");}
    }
  }
}

```

```

}
}
void enviarServer(char s){
  comando=1;
  if(s==1){
    ocupado=1;

    if(ocupado1==1){Serial2.println("COLONPAR01#OCUPADO#01+"
);ocupado1=3;}
    if(ocupado2==1){Serial2.println("COLONPAR01#OCUPADO#02+"
);ocupado2=3;}
    if(ocupado3==1){Serial2.println("COLONPAR01#OCUPADO#03+"
);ocupado3=3;}
    if(ocupado4==1){Serial2.println("COLONPAR01#OCUPADO#04+"
);ocupado4=3;}
    if(ocupado5==1){Serial2.println("COLONPAR01#OCUPADO#05+"
);ocupado5=3;}
    if(ocupado6==1){Serial2.println("COLONPAR01#OCUPADO#06+"
);ocupado6=3;}

    Serial.println("detectado");
  }
  if(s==2){
    String DatoOcupado2;
    comandtime=1;

    if(cntTeclado==1){
      DatoOcupado2=codigo+comand2+placa+"#0"+lugar+"#" +Time
+" ";
      cntTeclado=0;
    }
  }
}

```

```

if(cntTeclado==1){
DatoOcupado2=codigo+comand2+placa+"#0"+lugar+lugar2+"#
"+Time+" ";
cntTeclado=0;
}
if (lugar==49){ocupado1=4;}
if (lugar==50){ocupado2=4;}
if (lugar==51){ocupado3=4;}
if (lugar==52){ocupado4=4;}
if (lugar==53){ocupado5=4;}
if (lugar==54){ocupado6=4;}

Serial2.println(DatoOcupado2);
Serial.println("s2");
Serial.println(DatoOcupado2);
}
if(s==3){
String DatoOcupado3;
DatoOcupado3=codigo+comand3;

if(ocupado1==2){DatoOcupado3+="01+";ocupado1=0;pos=49;}
if(ocupado2==2){DatoOcupado3+="02+";ocupado2=0;pos=50;}
if(ocupado3==2){DatoOcupado3+="03+";ocupado3=0;pos=51;}
if(ocupado4==2){DatoOcupado3+="04+";ocupado4=0;pos=52;}
if(ocupado5==2){DatoOcupado3+="05+";ocupado5=0;pos=53;}
if(ocupado6==2){DatoOcupado3+="06+";ocupado6=0;pos=54;}

Serial2.println(DatoOcupado3);
Serial.println("s1");
Serial.println(DatoOcupado3);
for(int z=0;z>10;z++){

```

```

if(Lugares[z]==pos){
Lugares[z]=0;
z=12;
}
}
}
void verificarComando(){
if(ServerData[Indn1]==84){
ServerData[Indn1]=0;
Serial.println(" tiempo ");
if(ServerData[Indn3]==49){
ServerData[Indn3]=0;
tagok=1;
Serial.println(" tagOK1 ");
}
if(ServerData[Indn3]==48){
ServerData[Indn3]=0;
tagok=2;
}
}
if(ServerData[Indn1]==79){
ServerData[Indn1]=0;
Serial.println(" ocupado ");
if(ServerData[Indn2]==49){
ServerData[Indn2]=0;
ocupadook=1;
Serial.println(" tagOK2 ");
}
if(ServerData[Indn2]==48){
ServerData[Indn2]=0;
ocupadook=2;
}
}
}
}

```

```

    }
  }
}
void antirevote(){
  while (digitalRead(pulsaSi)==LOW){PSi=1;}
  while (digitalRead(pulsaNo)==LOW){PNo=1;}
  while
(digitalRead(pulsador1)==LOW){P2hm=1;Time="2H30";}
  while
(digitalRead(pulsador2)==LOW){P1h=1;Time="1H00";}
  while
(digitalRead(pulsador3)==LOW){P30=1;Time="0H30";}
  while
(digitalRead(pulsador4)==LOW){P3h=1;Time="3H00";}
  while (digitalRead(pulsador5)==LOW){P2h=1;Time="2H00";}
  while
(digitalRead(pulsador6)==LOW){P1hm=1;Time="1H30";}
  A
  beep();
  delay(25);
}
void leerPulsadorSN(){
  if(digitalRead(pulsaSi)==LOW){antirevote();PulsaActivado=1;}

if(digitalRead(pulsaNo)==LOW){antirevote();PulsaActivado=1;}

  if(digitalRead(tag)==0){b0=0;PulsaActivado=2;}
}
void leerPulsador(){

if(digitalRead(pulsador2)==LOW){antirevote();PulsaActivado=1;}
}

```

```

if(digitalRead(pulsador3)==LOW){antirevote();PulsaActivado=1;}
}

if(digitalRead(pulsador6)==LOW){antirevote();PulsaActivado=1;}
}
  if(zonaAB==1){

if(digitalRead(pulsador1)==LOW){antirevote();PulsaActivado=1;}
}

if(digitalRead(pulsador4)==LOW){antirevote();PulsaActivado=1;}
}

if(digitalRead(pulsador5)==LOW){antirevote();PulsaActivado=1;}
}
}
  if(digitalRead(tag)==0){b0=0;PulsaActivado=2;}
}

```

## K.5 ANEXO 5. ALGORITMO PARA CONTROL DEL TECLADO MATRICIAL

```
char uart_rd;
char num;
anti(){
    delay_ms(25);
    while(PORTB.F0==1){}
    while(PORTB.F1==1){}
    while(PORTB.F2==1){}
    while(PORTB.F3==1){}
    delay_ms(25);
}
void main() {
PORTB=0;
TRISB=0X0F;

UART1_Init(9600);    // Initialize UART module at 9600 bps
Delay_ms(2000);     // Wait for UART module to stabilize

//UART1_Write_Text("Start");
// UART1_Write(10);
// UART1_Write(13);

while (1) {          // Endless loop
// if (UART1_Data_Ready()) { // If data is received,
//  uart_rd = UART1_Read(); // read the received data,
//  UART1_Write(uart_rd); // and send data via UART
// }

PORTB.F4=1;
if(PORTB.F0==1){num=1;anti();UART1_Write(49);}
```

```
if(PORTB.F1==1){num=2;anti();UART1_Write(52);}
if(PORTB.F2==1){num=3;anti();UART1_Write(55);}
if(PORTB.F3==1){num=10;anti();UART1_Write(69);}
delay_ms(20);
```

```
PORTB.F4=0;
PORTB.F5=1;
if(PORTB.F0==1){num=4;anti();UART1_Write(50);}
if(PORTB.F1==1){num=5;anti();UART1_Write(53);}
if(PORTB.F2==1){num=6;anti();UART1_Write(56);}
if(PORTB.F3==1){num=11;anti();UART1_Write(48);}
delay_ms(20);
```

```
PORTB.F5=0;
PORTB.F6=1;
if(PORTB.F0==1){num=7;anti();UART1_Write(51);}
if(PORTB.F1==1){num=8;anti();UART1_Write(54);}
if(PORTB.F2==1){num=9;anti();UART1_Write(57);}
if(PORTB.F3==1){num=12;anti();UART1_Write(70);}
delay_ms(20);
```

```
PORTB.F6=0;
PORTB.F7=1;
if(PORTB.F0==1){num=14;anti();UART1_Write(65);}
if(PORTB.F1==1){num=15;anti();UART1_Write(66);}
if(PORTB.F2==1){num=16;anti();UART1_Write(67);}
if(PORTB.F3==1){num=13;anti();UART1_Write(68);}
delay_ms(20);
PORTB.F7=0;
```

```
}
}
```

**k.6 ANEXO 6. UBICACIÓN DE LOS ELEMENTOS QUE COMPONEN EL PROTOTIPO DE PARQUEADERO INTELIGENTE.**



Más práctica por minuto... menor tiempo de aprendizaje.....

### CERTIFICACION

#### WASHINGTON ENGLISH INSTITUTE.

#### CERTIFIES:

Que el presente documento " DESARROLLO DE UN PROTOTIPO DE PARQUEADERO INTELIGENTE PARA LA AUTOMATIZACION DEL SISTEMA DE APARCAMIENTO SIMERT EN LA CIUDAD DE LOJA " está correctamente traducido del idioma Español al Idioma Inglés, sin haber cambiado, aumentado o disminuido su sentido en ninguna línea o párrafo del mismo.

#### LO CERTIFICO:

Loja, 24 de Noviembre de 2015

