



FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

TRABAJO DE SUFICIENCIA PROFESIONAL

**DISEÑO DE UN SISTEMA DE COBRO AUTOMATIZADO
EN EL PEAJE CHILLÓN, PARA MEJORAR EL TRÁFICO
VEHICULAR EN LA PANAMERICANA NORTE,
UTILIZANDO TECNOLOGÍA RFID**

PRESENTADO POR EL BACHILLER

ESPINOZA REYES, CLEVER ENRIQUE

**PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES**

LIMA – PERÚ

2017

DEDICATORIA

Dedico este proyecto a mi madre espiritual la señora Clara Rosario Salazar Gurmendi Vda. de Arévalo, quien es la persona que a lo largo de mi vida se convirtió en una gran amiga, consejera y mentora, ayudándome a tomar las mejores decisiones de mi vida. Así como por todo el apoyo y fortaleza que me viene dando día a día, enseñándome que “los estudios son la mejor herencia para un hijo”. Especialmente por contar siempre con ella, aún en los momentos más difíciles de mi vida, regalándome los valores que me han fortalecido como persona, permitiéndome así crecer, desarrollarme.

Igualmente, hago extensiva esta dedicatoria a mi hermana Ohtina Mishel Perez Reyes, porque en ella siempre encontrado el apoyo necesario para superar exitosamente situaciones difíciles, así como el consejo oportuno y las orientaciones necesarias que han ayudado a formarme como una mejor persona.

AGRADECIMIENTOS

Agradezco a la Universidad Alas Peruanas, mi alma mater, pues me ha formado profesionalmente brindándome los conocimientos necesarios que han ayudado al desarrollo de este proyecto. Al Ingeniero Cayo Vidal Sinche Jamjachi y al Señor José Orlando Ramos López, profesionales claves en el desarrollo de este proyecto. Por su ayuda en la realización del sistema de comunicación propuesto y de la estructuración del mismo, por su tiempo y dedicación desinteresada. Por último, quiero agradecer a mis compañeros de clase, con los que hemos compartido buenas y gratificantes experiencias a lo largo de los 5 años de estudios dentro de la facultad de Ingeniería y Arquitectura.

INTRODUCCIÓN

En los últimos años las autoridades en nuestro país han visto formas de como poder solucionar la congestión vehicular en Lima y sus alrededores; pero, sus acciones no han llegado a tener mayor relevancia. La construcción de nuevas vías de acceso y la realización de obras civiles de ampliación de vías por todos los alrededores de Lima están ocasionando una congestión mayor en nuestro capital, lo que hace que la ciudad pierda atractivo turístico y que los gases de efecto invernadero tengan una mayor concentración en esta congestión, ocasionando que nuestra ciudad pierda grandes sumas de dinero diariamente.

En el trabajo de investigación urbanística del Ingeniero de caminos Julio Pozueta Echavarrí, da a conocer que en algunos países de Europa, para limitar el ingreso de vehículos al centro de la ciudad, se han instalado peajes urbanos, los que han ayudado de manera importante en la reducción de vehículos, dando un mayor atractivo a sus principales monumentos y plazas, permitiendo que las personas se puedan desplazar con mayor libertad por estos lugares, consiguiendo así la reducción de los gases contaminantes que desechan los vehículos. Pero, el peaje no solamente lo usan para reducir la circulación vehicular, sino también para la recolección de dinero por el uso de la vía, donde una concesionaria se encargará de darle manteniendo y de realizar la construcción de nuevas vías en los alrededores del peaje, permitiendo que este sea autosustentable (Echavarrí, 2008, pág. 10).

En nuestro país, uno de los principales corredores viales es la Autopista Panamericana Norte. En el kilómetro 25 se encuentra el peaje Chillón ubicado en el distrito de Puente Piedra, que conecta los distritos de Ancón, Puente Piedra y los procedentes del norte de nuestro país como Chimbote, Trujillo, Piura con los distritos de Independencia, San Martín de Porres, el centro de Lima, sus alrededores y la salida al sur de nuestro país. En este corredor vial circulan a diario unos 75 mil vehículos, las horas puntas en esta vía son en las mañanas entre las 7.00 am hasta 9.00 am y en las tardes 5.30 pm hasta las 8.00 pm la menor afluencia de vehículos es en el transcurso de media mañana y hasta antes de las 5.00 pm de

la tarde de lunes a viernes y sábado al mediodía. Después de estas horas, por la tarde se forman largas colas en los módulos del peaje ubicados en el kilómetro 25 de la Autopista Panamericana Norte, causando la congestión vehicular y extensas colas de autos. Para solucionar este inconveniente, en otros países se viene implementado mejoras en los peajes, con una red de sensores que se basan en la tecnología inalámbrica zigbee del estándar IEEE 802.15.4, que propone mejorar el sistema de control de peaje vehicular, automatizando el proceso, evitando así que los vehículos tengan que detenerse, permitiendo ganar tiempo muerto que generan las personas al buscar dinero en efectivo y realizar dicho pago, aligerando así la congestión vehicular. Un sistema similar se propone para el peaje del kilómetro 25 de la Autopista Panamericana Norte.

Dentro del estándar IEEE 802.15.4, encontramos el sistema de RFID, más conocido como Identificación por Radiofrecuencia, el que puede estar formado por varios componentes como son las etiquetas, que almacenan información necesaria del propietario, lectores de etiquetas, estaciones de programación de etiquetas, lectores de circulación y equipamiento de ordenación. El propósito de un sistema RFID es permitir que se puedan transmitir datos mediante un dispositivo portátil, llamado etiqueta, que es leída por un lector y procesada según las necesidades de una aplicación determinada. Los datos transmitidos por la etiqueta pueden proporcionar información sobre la identificación del vehículo marcado con esta, datos como placa de rodaje del vehículo, datos personales del propietario, etc. (Wikipedia, 2017)

Esta tecnología apunta a convertirse en uno de los más importantes estándares de intercambio de comunicación. A pesar de que la tecnología RFID tiene más de 50 años, es hasta hoy que, gracias a los avances tecnológicos y a la disminución en su costo de adquisición, que se ha empezado a utilizar más en la automatización, control de acceso, paquetería, seguridad, para hacer más eficientes los procesos. Por eso, la aplicación que le daremos nos va permitir manejar la identificación de cada uno de los vehículos que utilicen el peaje (Atilano, 2007, pág. 2)

RESUMEN

El diseño del Control de Peaje Automatizado con la Tecnología de Identificación por Radiofrecuencia, consistirá en un sistema que tendrá como prioridad aligerar la carga vehicular en el peaje ubicado en el distrito de Puente Piedra, kilómetro 25 de la Autopista Panamericana Norte, en los carriles con dirección de norte a sur, donde gracias a la tecnología mejor conocida como RFID, los vehículos no tendrán que detenerse completamente a realizar el pago correspondiente por el uso de la vía.

Para que se pueda realizar este sistema, cada vehículo tendrá incorporado una etiqueta que tendrá la información del vehículo. Asimismo, contendrá también la información de crédito que tiene el vehículo para que pueda realizarse el débito por la utilización de la vía. Esta información estará registrada en una base de datos, que mediante un servidor podremos conectarnos de manera instantánea para obtener la información de los vehículos que han usado la vía. Para obtener datos de la etiqueta ubicada en el vehículo, necesitaremos de un lector el cual deberá tener la misma frecuencia de trabajo que las etiquetas. Será el encargado de energizar de manera inalámbrica las etiquetas y obtener la información ya mencionada del vehículo.

Para el correcto funcionamiento del sistema, se necesitará que el vehículo circule a una velocidad no mayor a 30 km/h, para que el lector ubicado en módulo de peaje pueda realizar el proceso anteriormente mencionado, sin que tenga que detener el vehículo completamente, para que los conductores respeten la velocidad máxima indicada, el módulo del peaje automatizado tendrá un sistema complementario de CCTV que se deberá integrar para el respectivo control, tomando fotografías del vehículo para realizar la sanción correspondiente en caso supere la velocidad establecida. Este sistema de CCTV no se encuentra incluido dentro de este proyecto.

ABSTRACT

The design of the automated toll control with radio frequency identification technology will consist on a system that will have as a priority to lighten up traffic on the toll placed on the district of Puente Piedra, 25 km from the Panamericana Norte, exactly at lanes which go from north to south, where thanks to the radio frequency identification technology, better known as RFID, vehicles will not have to stop completely to make the corresponding payment for the use of the track.

In order to be able to realize this system, each vehicle will have a tag incorporated, that will have the information of the vehicle as well as the credit information that the vehicle has, in order to use of the highway, this information will be registered in a Database, which by a server we can connect instantaneously to obtain the information of the vehicles that have used the way, to obtain data of the label located in the vehicle we will need a reader which must have the same frequency work that the Tags, will be in charge of wirelessly energizing the tags and obtaining the aforementioned information of the vehicle.

For the correct function of the system, it will be necessary that the vehicle does not run at a speed of less than 30 km/h so that the reader located in the toll module can perform the aforementioned process, without having to stop the vehicle completely, for this to be done, the automated toll module will have a complementary CCTV system, that must be integrated for the respective control, taking photographs of the vehicle to make the corresponding penalty in case of exceeding speed, this CCTV system is not included Within this project.

TABLA DE CONTENIDO

DEDICATORIA.....	ii
AGRADECIMIENTOS	iii
INTRODUCCIÓN	iv
RESUMEN	vi
ABSTRACT	vii
TABLA DE CONTENIDO.....	viii
CAPÍTULO I: GENERALIDADES DE LA EMPRESA OSC TELECOMS.....	1
1.1 Antecedentes de la empresa OSC.....	2
1.2 Perfil de la empresa OSC.....	4
1.3 Actividades de la empresa OSC.....	4
1.3.1 Misión	6
1.3.2 Visión.....	6
1.3.3 Objetivo.....	6
1.4 Organización actual de la empresa OSC	7
1.5 Descripción del entorno de la empresa OSC	7
CAPÍTULO II: REALIDAD PROBLEMÁTICA.....	9
2.1 Descripción de la realidad problemática.....	10
2.2 Análisis del problema	11
2.3 Objetivos del proyecto.....	15
2.3.1 Objetivo general.....	15
2.3.2 Objetivos específicos.....	15
CAPÍTULO III: DESARROLLO DEL PROYECTO.....	16
3.1 Sistema RFID	17
3.1.1 Lector RFID	18
3.1.2 Etiquetas RFID	19

3.2 Funcionamiento del sistema.....	22
3.3 Dispositivos para el proyecto.....	26
3.3.1 Módulo lector RFID.....	26
3.3.2 Antena externa RFID.....	27
3.3.3 Etiqueta RFID - Slu-f.....	28
3.3.4 Microcontrolador 18F2550.....	29
3.3.5 Rs-232 / Rs-485 (rts, tx, rx).....	30
3.4 Software utilizados.....	30
3.4.1 Mplab X.....	30
3.4.2 Python.....	31
3.4.3 PostgreSQL.....	32
3.4.4 Proteus.....	32
3.5 Desarrollo del proyecto.....	33
3.5.1 Configuración del lector RFID con reader 1000.....	33
3.5.2 Lectura y escritura de etiqueta RFID.....	36
3.5.3 Conexión de los lectores RFID con el microcontrolador.....	39
3.5.4 Construcción de la trama de comunicación.....	41
3.5.5 Construcción de la base datos.....	44
3.5.6 Comunicación entre el microcontrolador y PostgreSQL.....	49
3.5.7 Simulación del sistema.....	51
3.5.8 Construcción de los PCB para el Proyecto.....	58
3.5.9 Presupuesto.....	59
3.6 Conclusiones.....	60
3.7 Recomendaciones.....	61
CAPÍTULO IV: REFERENCIAS BIBLIOGRÁFICAS.....	62
CAPÍTULO V: GLOSARIO DE TÉRMINOS.....	66

CAPÍTULO VI: ÍNDICES	69
6.1 Índice general de figuras	70
6.2 Índice general de tablas	72
CAPÍTULO VII: ANEXOS	73
7.1 ANEXO 1: Programación del microcontrolador	74
7.2 ANEXO 2: Presentación de diapositivas del proyecto	86

CAPÍTULO I

GENERALIDADES DE LA EMPRESA OSC TELECOMS

1.1 ANTECEDENTES DE LA EMPRESA OSC

OSC es una multinacional de origen colombiano fundada en 1996, con más de dos décadas de experiencia en el sector de las telecomunicaciones, seguridad electrónica y automatización. (OSC Telecoms, 2011)

En los últimos años, OSC ha seguido creciendo en sus servicios de telecomunicaciones en toda América Latina, lo que genera un sentido de pertenencia a sus empleados que ven OSC como Optimo, Seguro y Confiable; OSC Telecoms & Security Solutions es una empresa multinacional con presencia en

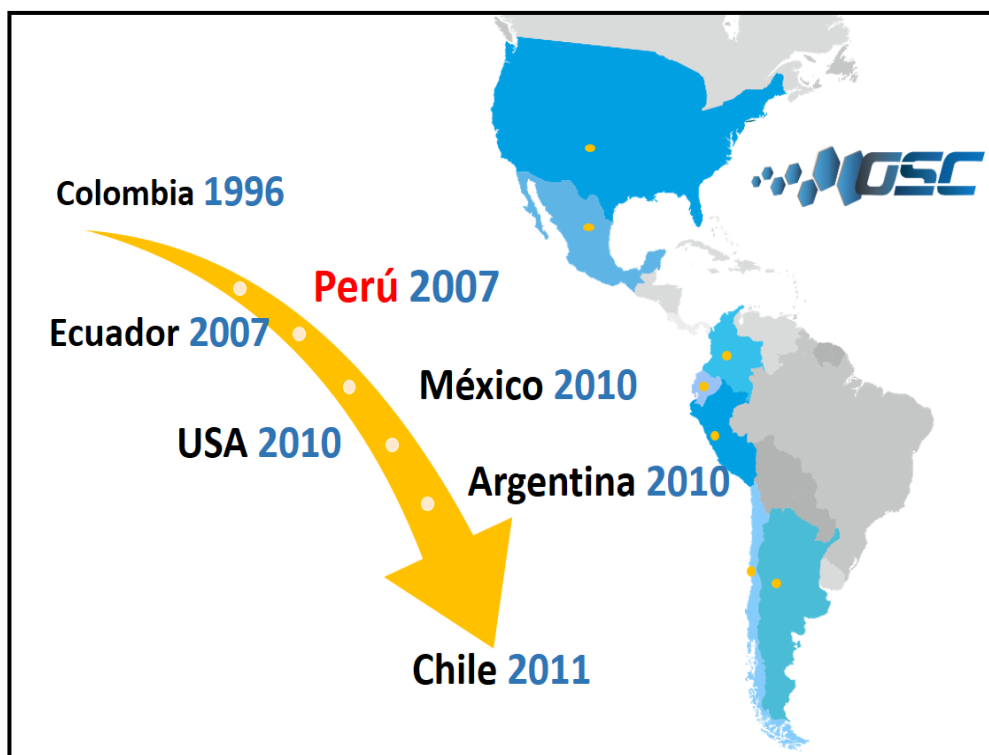
- ❖ Colombia
- ❖ Perú
- ❖ Ecuador
- ❖ Venezuela
- ❖ México

OSC comenzó a operar en los siguientes países en 2007

- ❖ Costa Rica
- ❖ Argentina
- ❖ Panamá
- ❖ Chile

En la figura 1, podemos observar el crecimiento de Osc Telecoms en Latino América y los años de inicio de sus actividades en los diferentes países, desde 1996 en Colombia y el 2011 en Chile. Osc se sigue proyectando para llevar sus servicios a los demás países hermanos de Colombia (Haro, 2012)

Figura 1 Expansión de Osc Telecoms en América



Fuente: (Benites, 2015)

En la figura 2 se puede apreciar el logo que representa a la empresa Osc Telecoms y su significado de las siglas OSC que son Óptimo, Seguro y Confiable.

Figura 2 Logo de la Empresa OSC Telecoms y su Significado



Fuente: (Benites, 2015)

1.2 PERFIL DE LA EMPRESA OSC

OSC Telecoms, es una empresa dedicada a brindar servicios de asesoría, consultoría y asistencia en materia de telecomunicaciones, está constituida al amparo de lo dispuesto en la ley de la Sociedad Anónima Cerrada. Es una empresa del rubro de telecomunicaciones dedicada a brindar soluciones que permitan satisfacer las necesidades técnicas y operativas de sus clientes, superando sus expectativas y generando valor agregado en la prestación de servicios de planeación, optimización, automatización y mantenimiento e instalación de redes de Telecomunicaciones fijas y móviles, seguridad electrónica, frequency hunting, soluciones Indoor y Outdoor de todas las tecnologías celulares. (Benites, 2015)

OSC considera que su activo máspreciado, son sus trabajadores, por ello, partiendo de los principios de Prevención, Capacitación y Protección, su principal objetivo es proporcionar ambientes de trabajo sanos y seguros para sus trabajadores en el desarrollo de sus actividades haciéndolo extensivo a sus clientes, proveedores, y visitantes. (Benites, 2015)

1.3 ACTIVIDADES DE LA EMPRESA OSC

OSC Telecoms & SEGURIDAD SAS es una empresa con 15 años de experiencia en el mercado de las telecomunicaciones y se dedica a soluciones de seguridad electrónica a través de consultoría, la venta y alquiler de equipos, mantenimiento y aplicación de la tecnología y el suministro de diferentes soluciones para sus clientes

En el área de Telecomunicaciones

OSC Telecoms & Security Solutions ofrece servicios de implementación de redes móviles y fijas. OSC tiene experiencia del diseño, Inspecciones del emplazamiento técnicos, aplicación de microondas de la EDH y SDH, RF, 2G, 3G y 4G, el núcleo y la fibra óptica. La logística y la experiencia de OSC en la

implementación de estos proyectos, aseguran su desarrollo correcto y adecuado. Así mismo cuenta con recursos humanos necesarios para (Osc Telecoms, 2011)

- ❖ Consultores de telecomunicaciones.
- ❖ Optimización de RF 2G-3G.
- ❖ Prueba de conducción para el ajuste inicial, la optimización y la evaluación comparativa.
- ❖ Post - procesamiento y análisis de red.
- ❖ Microondas.
- ❖ Planificación y Optimización.
- ❖ Aplicaciones de telecomunicaciones.
- ❖ Supervisión en Telecomunicaciones.
- ❖ Funcionamiento y mantenimiento.
- ❖ Especialista es Recursos Humanos.
- ❖ Alquiler de equipos de telecomunicaciones.

En el área de Seguridad y Automatización

OSC Telecoms ofrece servicios de seguridad y automatización, soluciones de diseño, suministro, implantación y mantenimiento de circuito cerrado de televisión (CCTV), control de acceso, detectores de metales y armas, explosivos; Los sistemas de detección de incendios y extintores, alarmas de intrusión, puertas de seguridad, y soluciones de seguridad basadas en redes y sistemas de integración de seguridad. Así mismo cuenta con recursos humanos necesarios para (Osc Telecoms, 2011)

- ❖ Servicios de CCTV.
- ❖ Servicio de Control de Activos.
- ❖ Servicio de Control de Acceso.
- ❖ Servicio de Control de presencia.
- ❖ Servicio de Automatización e Integración.

- ❖ Servicio de Aplicaciones de Alta Seguridad.
- ❖ Servicio de Control de detección de incendio y de intrusiones.
- ❖ Servicio de Mantenimiento y servicio Técnico.

1.3.1 MISIÓN

Garantizar la plena satisfacción de sus clientes, colaboradores, comunidad y demás partes interesadas, mediante la optimización de los procesos y la mejora continua en las actividades y acciones organizacionales. (Benites, 2015)

1.3.2 VISIÓN

Ser una de las primeras empresas de servicio en telecomunicaciones a nivel regional y el TOP of mind en nuestros clientes dentro del rubro. (Benites, 2015)

1.3.3 OBJETIVO

Como objetivo principal dedicarse a las actividades de ingeniería y actividades conexas de consultoría técnica, entre otras actividades. (Osc Telecoms SAC, 2015)

1.4 ORGANIZACIÓN ACTUAL DE LA EMPRESA OSC

La empresa Osc Telecoms se encuentra constituida por el gerente general, el señor Leonardo Escobar y él se encarga de verificar el correcto funcionamiento de Osc en Perú, como se aprecia en la figura 3.

Figura 3 Organigrama de OSC Telecoms SAC



Fuente: (Benites, 2015)

1.5 DESCRIPCIÓN DEL ENTORNO DE LA EMPRESA OSC

En OSC Telecoms & Security Solutions, busca satisfacer las necesidades de sus clientes que van más allá de sus expectativas y la generación de valor agregado a través del servicio, la entrega, la planificación, optimización, mantenimiento e instalación de redes de telecomunicaciones fijas y móviles y la seguridad Electrónica y automatización. Para lograr este objetivo OSC

Telecoms & Security Solutions trabajan para suministrar sus servicios en las mejores formas, más seguros y confiables, asegurando (Haro, 2012)

- ❖ El cumplimiento de los requisitos legales y/o los reglamentos necesarios para el desarrollo e implementación de sus servicios, mitigando el impacto ambiental y socio-ambiental durante los proyectos que se lleven a cabo. También la promoción de la responsabilidad social de sus proveedores y contratistas.
- ❖ El personal es altamente capacitado, conscientes de las normas de seguridad industrial con el fin de prevenir lesiones, enfermedades y daños a la propiedad.
- ❖ La adquisición y el mantenimiento de equipos y software de última generación, actualizada periódicamente y en excelentes condiciones de trabajo.
- ❖ Los recursos para mantener y mejorar constantemente sus procesos integrales del sistema de gestión.

CAPÍTULO II
REALIDAD PROBLEMÁTICA

2.1 DESCRIPCIÓN DE LA REALIDAD PROBLEMÁTICA

En nuestro país, una de las principales problemáticas, después de la inseguridad ciudadana, es la extensa cantidad de vehículos, que en los últimos años se ha incrementado de manera incontrolada; pero, esto no sería un problema si el sistema vial de nuestro país estuviese preparado para esta demanda de vehículos, en donde principalmente apreciamos esta problemática son en los peajes, donde las colas de autos no tienen donde acabar ocasionado una gran congestión vehicular, debido a que se utiliza un sistema manual de cobro, por el cual la persona al volante debe detener su vehículo para realizar el pago por el uso de la vía, pudiéndose apreciar mejor la congestión en las horas de mayor afluencia; pero, no sería así, si es que hubiese una mayor cantidad de módulos de cobro, eso significaría realizar obras civiles, ampliando la cantidad de módulos, pero esto no sería una solución rentable.

Sería rentable si hacemos uso de la identificación electrónica, para lo cual se propone la tecnología RFID, Identificación por Radiofrecuencia, evitando así que el auto tenga que detenerse. Para ello se desarrollará un módulo de comunicación que interactúe con un lector de uso comercial, donde se pueden mostrar los datos leídos de una etiqueta adherida al vehículo. Estos datos pueden ser almacenados en una base de datos, la cual tiene en su contenido la información de vehículos, propietarios y etiquetas. La información obtenida del lector es comparada con la que se encuentra en la base de datos y mostrada al usuario.

Para resolver esto, debemos considerar que el módulo de comunicación sea capaz de brindar seguridad al usuario en general, mostrando la información alojada en la base de datos y haciendo una comparación con los datos leídos a través del lector de etiquetas RFID. (Atilano, 2007, pág. 12)

2.2 ANÁLISIS DEL PROBLEMA

En nuestro país, diariamente crece nuestro parque automotor y para este año, según la publicación del diario El Comercio en su apartado de economía publicado en setiembre del 2003, proyectaba un crecimiento de 300 mil vehículos a fines de este año 2017 y, esta proyección se está cumpliendo, pues acceder a la compra de un automóvil cada vez se hace más accesible, por nuevos sistema de evaluación crediticia de los bancos, influyendo así al crecimiento del parque automotor en el Perú y su congestiónamiento diario, ya que las personas buscan una forma más directa y cómoda de poder trasladarse dentro de la ciudad, mejorando su calidad de vida y estatus social, pero esto no se cumple debido a que no contamos con infraestructura vial capaz de soportar la demanda sin causar congestión vehicular. La congestión vehicular cada vez se ve más en aumento debido a las obras de ampliaciones, remodelaciones que se dan en nuestra ciudad.

Para tratar de darle una solución, se deben realizar nuevos proyectos de ordenamiento o limitación de circulación por la vía, a cargo del Ministerio de Transportes y Comunicaciones, con las autoridades municipales. Donde se puede encontrar una mayor concentración de vehículos es en el centro de lima y avenidas como Javier Prado, Av. Alfonso Ugarte, Av. Arequipa; y donde se repite el escenario es en los peajes que tenemos en el norte y sur de Lima, en los que se aprecian notablemente las colas interminables de vehículos, como se muestra en la figura 4.

Figura 4 Peaje Chillón y su Congestión Vehicular

Fuente: Elaboración Propia

El problema principal no es el uso del peaje, sino el proceso manual de cobro con el que funciona actualmente, generado por el déficit que se tiene para poder atender a la gran cantidad de vehículos que circulan por el kilómetro 25 Autopista Panamericana Norte en los sentidos de norte a sur, produciendo pérdida de tiempo innecesaria al igual de un elevado gasto de combustibles. Realizando una sumatoria de este combustible perdido, se transforma en una cantidad importante de dinero, sin mencionar que este combustible consumido por un automóvil es desechado como gases contaminantes para nuestro medio ambiente, todos estos factores generan una mala perspectiva de lo que representa un peaje, que es visto como pérdida de tiempo innecesario, como se aprecia en la figura 5.

Figura 5 Sistema Actual de Cobro Manual



Fuente: Elaboración Propia

En la figura 5, se puede apreciar el sistema actual utilizado en el peaje Chillón en la autopista Panamericana Norte kilómetro 25, ubicado en el distrito de Puente Piedra, donde el cobro es de manera manual lo que genera tiempos muertos entre la atención de un vehículo al vehículo siguiente, generando colas interminables en las horas de mayor afluencia. El proceso de cobro del peaje cuenta con diferentes etapas estas se describen a continuación:

- ❖ **El conductor** del vehículo entrega el monto de s/. 5.00 nuevos soles en caso de vehículo liviano particular (el pago varía en cuanto al tipo de vehículo y la cantidad de sus ejes)
- ❖ **El operario** recibe el dinero, imprime el boleto y activa la elevación de la barrera de acceso a la vía, y por último entrega el comprobante de pago.
- ❖ **El conductor** recibe el comprobante y procede a continuar su paso hacia la autopista.

Esta es una descripción simple, debido a que el conductor paga con la cantidad de dinero específico y no solicita factura, solo un comprobante con el monto impreso, el tiempo promedio es 1.5 minutos, teniendo solo 01 vehículo anterior al que atiende. La situación cambia al solicitar factura ya que se deberá de indicar los 11 dígitos del RUC y la razón social para verificar la correspondencia. Otro agravante es que el conductor pague con un billete de S/. 20.00; S/. 50.00; S/. 100.00 ó S/.200.00 nuevos soles; debido al que el operario deberá buscar y contabilizar el vuelto y a su vez el conductor deberá contabilizar el dinero entregado por el operario para evitar errores; teniendo en consideración estas situaciones, el tiempo promedio se incrementa y se inicia la formación de colas de vehículos y si a este escenario le añadimos que son las 6.00 pm hora en la que existe mayor afluencia de vehículos, la cola se hace interminable rápidamente.

Habiendo encontrando causante de la congestión del peaje Chillón, de la Autopista Panamericana Norte kilómetro 25, se plantea mejorar el tiempo de atención del cliente, permitiendo que este pase por la cabina del operador con una velocidad no mayor a 30 km/h, y sin requerir detener su vehículo para realizar el pago por el uso de la vía. Esta propuesta se puede encuentra ampliamente detallada en el tercer capítulo del proyecto.

2.3 OBJETIVOS DEL PROYECTO

2.3.1 OBJETIVO GENERAL

El objetivo del proyecto es mejorar y optimizar el sistema de cobro del peaje Chillón, de la autopista panamericana norte kilómetro 25, mediante la automatización de este proceso, haciendo que el cobro se realice de manera virtual, evitándose así formar largas colas en el peaje y mejorando la fluidez de los vehículos.

2.3.2 OBJETIVOS ESPECÍFICOS

- ❖ Mejorar la calidad y cantidad de atención de los módulos en el peaje Chillón, al reducir los tiempos muertos donde el usuario busca el dinero para pagar o el trabajador demora en buscar el vuelto.
- ❖ Mejorar las condiciones de trabajo y la seguridad del personal, en el peaje Chillón de la autopista panamericana norte kilómetro 25, encargado de realizar el cobro por el uso de la vía,

CAPÍTULO III

DESARROLLO DEL PROYECTO

En este capítulo se detallará la información necesaria para comprender el funcionamiento de la tecnología RFID, y describir cómo se utilizará en el proyecto. Asimismo, se detallarán los componentes usados y la interacción que existirán entre ellos para el desarrollo de la comunicación; tomando los datos obtenidos por el lector para realizar el proceso de construcción del registro del uso del peaje, dentro de la base de datos en PostgreSQL. Se utiliza este software ya que es un sistema de gestión de bases de datos potente y sobre todo que su código fuente está disponible libremente, significando un ahorro en los gastos del proyecto así mismo se hace uso de la herramienta de desarrollo llamada Python, con su módulo de comunicación serial para la interacción con el PIC 18F2550 y la base de datos, la programación del PIC se desarrolló en la plataforma de MPLABX, en lenguaje C, esto se demostrará en el desarrollo del proyecto.

3.1 SISTEMA RFID

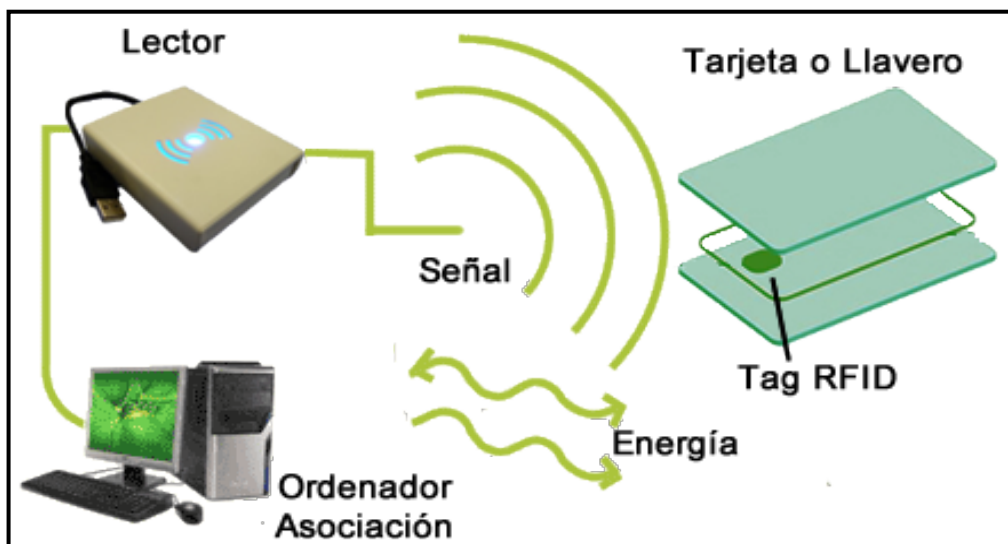
Es una tecnología que se remonta a la segunda guerra mundial donde era utilizada por Gran Bretaña para la identificación de sus aviones mediante el uso de los radares que leían el código implantado en los aviones, evitando confundir aviones propios con aviones enemigos, y que podían ser derribados por equivocación (Sotelo, 2006)

El RFID es una tecnología de comunicación inalámbrica por radiofrecuencia, que se da entre un lector y la etiqueta, esta última puede estar instalada en el cualquier dispositivo, ya sea en el parabrisas de un auto o en la oreja de cualquier animal.

Estos dispositivos deberán estar a la misma frecuencia intermedia de operación para que la etiqueta pueda comunicarse con el lector, debido a que la etiqueta es un elemento pasivo, esta se encuentra inactiva hasta estar dentro del alcance de lectura, una vez que esté al alcance del lector mediante las ondas de radio se energizará dicha etiqueta y responderá enviando la información almacenada dentro de sus bancos de memoria. Dicha información

varía según la aplicación que se requiera, para poder visualizar e interpretar el funcionamiento del sistema. Ver la figura 6, donde se aprecia el funcionamiento del sistema RFID. (Herrera & Casanovas).

Figura 6 Funcionamiento del Sistema RFID



Fuente: (Gestión Verde, 2015)

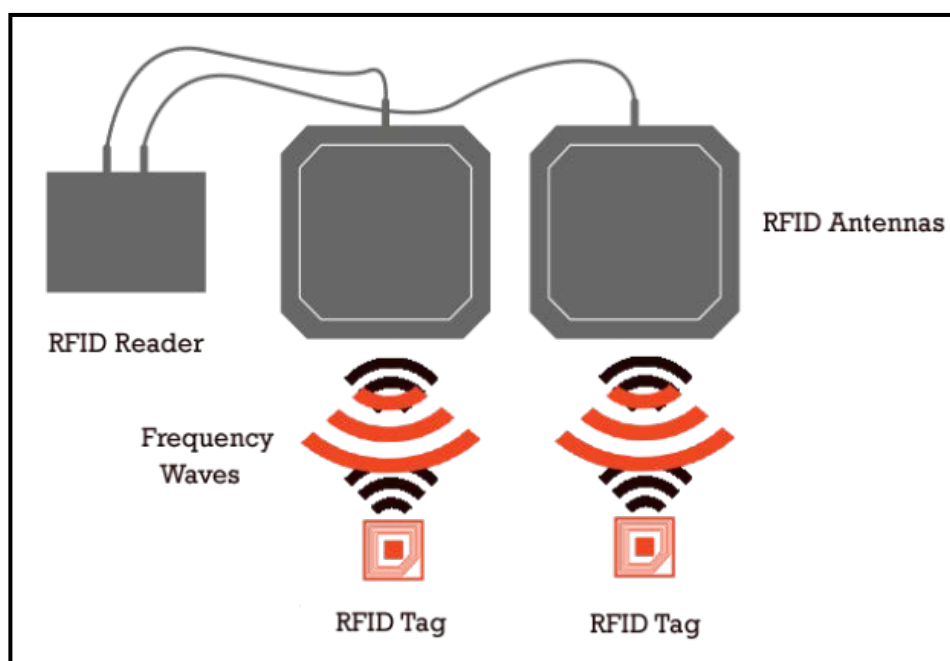
Ahora describiré los componentes del sistema de identificación por radio frecuencia que son 2 elementos, los cuales son el Lector y las Etiquetas; y son los que se describirán a continuación

3.1.1 Lector RFID

Es un dispositivo activo, que se encarga de interrogar y solicitar la información a la etiqueta e iniciar la comunicación ya que este emite ondas de radiofrecuencia a una frecuencia intermedia, como todo dispositivo inalámbrico, el alcance de lectura dependerá de la potencia de transmisión y de la ganancia que tenga la antena.

Cuando el lector obtiene una respuesta de la etiqueta RFID, convierte la señal de radio frecuencia en un código numérico que puede ser transmitido a otros dispositivos o sistemas. Los lectores RFID pueden variar en tamaño, funcionalidad, según la frecuencia de operación del alcance de lectura y la cantidad de lectura que puede hacer al mismo tiempo. Ahora se puede encontrar lectores RFID adaptables de manera fácil a un dispositivo móvil, utilizando el sistema de desarrollo dentro de la plataforma de Android (Sotelo, 2006, pág. 5). En la figura 7, se muestra un lector y la interacción con las etiquetas.

Figura 7 Lector RFID e Interacción con las etiquetas RFID



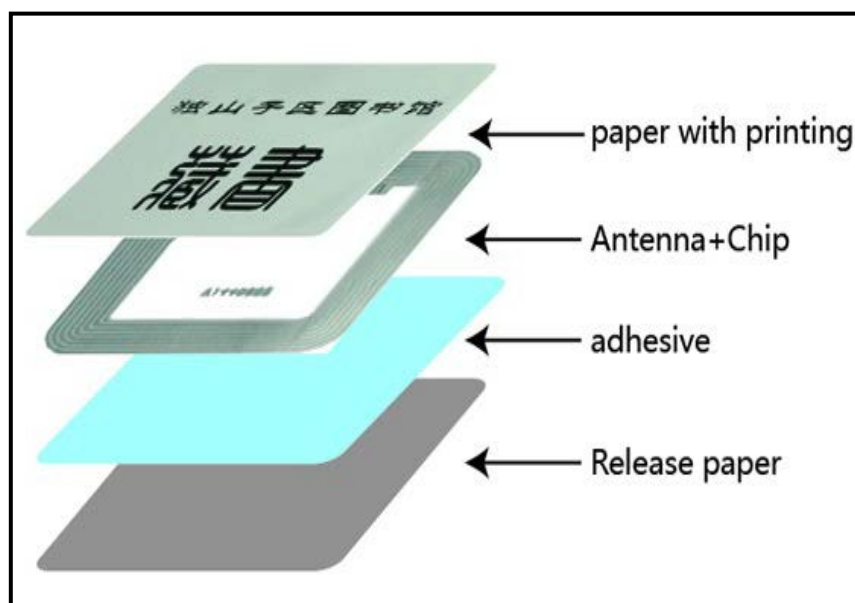
Fuente: (Smiley, 2014)

3.1.2 Etiquetas RFID

Dentro del sistema RFID es el elemento más pequeño, pero de gran complejidad en su elaboración, ya que contiene un microchip donde almacena datos que se procesan para la comunicación. La antena es por donde captura las ondas de radio. En muchos casos se

encuentran impresas en papel adhesivo y son casi del mismo tamaño de una moneda, y delgadas como una hoja de papel, lo que permite que sea adherida en autos, equipos electrónicos y materiales industriales. En la figura 8, se puede visualizar la arquitectura de una etiqueta RFID y posición de cada uno de los componentes que la conforman.

Figura 8 Arquitectura de la Etiqueta RFID



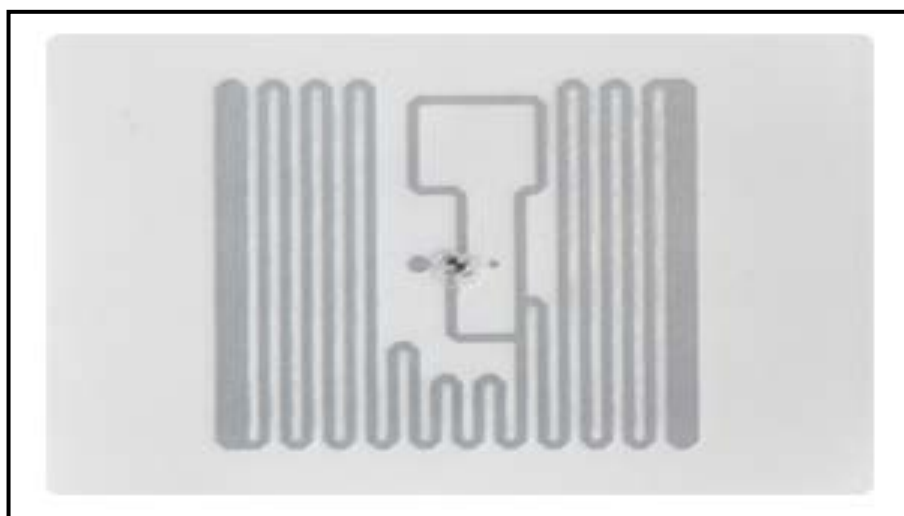
Fuente: (alibaba.com, 2016)

La etiqueta es elemental para poder realizar la comunicación, debido a que contiene el código de identificación que es capturado por el lector y desde ahí se realizan los diversos procesos. Estas etiquetas pueden ser pasiva y activas. Las activas se diferencian al tener una fuente de alimentación (pila), permitiendo que esta etiqueta emita su código de identificación de manera continua lo que permite que el lector no sea, el que directamente inicie la comunicación haciendo que sea más óptimo el enlace, ya que cuenta con una mayor distancia de lectura, alcanzando distancia de 20 metros entre otras, dependiendo de la frecuencia y del lector encargado del escaneado; pero, al ser más

efectiva, es también algo inaccesible económicamente hablando para un proyecto de pequeña escala debido a su elevado costo.

Cada etiqueta activa oscila su costo entre \$25.00 a \$40.00 dólares. En la figura 9, se muestra la etiqueta pasiva que se usara en este proyecto la cual es de bajo costo, alrededor de \$0.10 centavos de dólar comprando una gran cantidad aproximadamente 10 mil unidades a más, fue seleccionada para el proyecto por su bajo costo, por la frecuencia de operación y por sus dimensiones 30x40x0.2(mm), debido a que proviene del mismo fabricante del lector Rfid, nos permite tener mayor compatibilidad con el lector y sobre todo mayor confianza.

Figura 9 Etiqueta Pasiva SLU-F Series Strong Link



Fuente: (Strong Link, 2017)

Hasta ahora se han descrito los componentes que conforman el sistema de comunicación inalámbrica RFID; pero, es necesario conocer el comportamiento que este sistema tiene para poder determinar los requerimientos necesarios para nuestro proyecto; y esto dependerá de

nuestra aplicación, la cual consiste en automatizar el sistema de cobro actual del peaje.

A continuación, se verifican los parámetros como frecuencias de operación, la velocidad de transmisión, potencia irradiada. En la tabla 1 se describen estos parámetros.

Tabla 1 Frecuencia de Operación en RFID

Banda	LF – Baja Frecuencia	HF – Alta Frecuencia	UHF – Ultra Alta Frecuencia	Microondas
Frec.	30 – 300 kHz	3 – 30 MHz	300MHz – 3 GHz	2 – 30 GHz
Frec. típicas en RFID	125 – 134kHz	13.56 MHz	433 MHz o 865 – 956 MHz 2.45 GHz	2.45 GHz
Rango de lectura	Menor a 0.5 metros	Hasta 1.5 metros	433 MHz - Hasta 100 metros 865 – 956 MHz hasta 10 metros	Más de 10 metros
Tasa de transf. de datos	Menos de 1 kilobit por segundo (kbit/s)	Aprox. 25 kbit/s	433 – 956 - 30 kbit/s 2.45 - 100 kbit/s	Hasta 100 kbit/s

Fuente: (Delgadillo & Ortiz, 2011)

3.2 FUNCIONAMIENTO DEL SISTEMA

Después de este preámbulo de la tecnología RFID, se procede a realizar una descripción de la situación actual del peaje Chillón, ubicado en el kilómetro 25 de la Autopista Panamericana Norte en el distrito de Puente Piedra, por este peaje diariamente circulan alrededor de 75 mil vehículos con sentido de norte a sur, ya que es una vía que conecta el norte con el centro de Lima, pero el flujo de los vehículos, se desplaza con gran dificultad debido a que no se tiene un sistema óptimo, como se detalló en el segundo capítulo, el cobro actualmente se da de manera lo que genera una gran cantidad de intervalos de tiempo muertos entre la atención de una serie de vehículos.

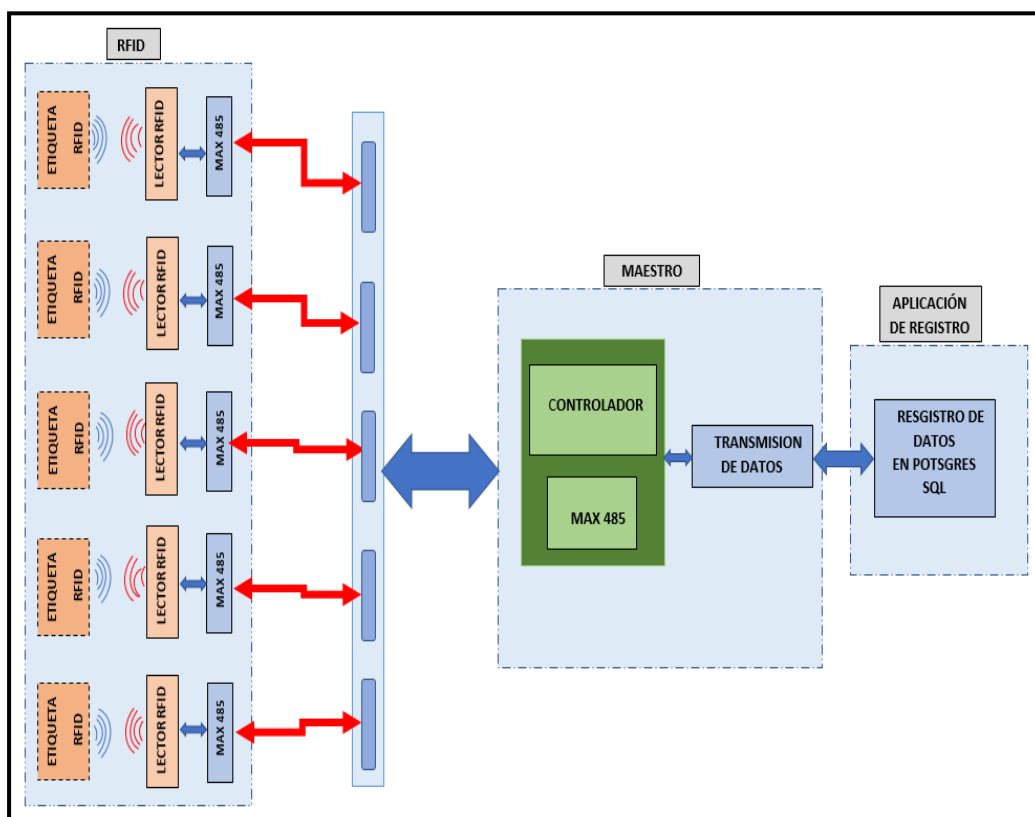
Con la finalidad de mejorar esta situación y de generar una satisfacción en los conductores por el uso del peaje, se propone un sistema libre de cobro en el sentido que se haga de manera virtual, y esto se puede dar gracias a la tecnología de Identificación de Radiofrecuencia, el cual optimizaría el sistema de cobro, descongestionando el peaje haciendo más fluida la Autopista Panamericana Norte.

El proyecto evitara la congestión de los vehículos en el peaje Chillón. Esta mejora consistirá en instalar en cada módulo de cobro del peaje la antena del Lector RFID a una distancia de 5.50 metros del suelo, lo que nos permite tener un área de lectura de 10 metros aproximadamente. El lector RFID SL130, está configurado de modo que cada 30 milisegundo envíe una señal en búsqueda de etiquetas, esto es lo que respecta al peaje y con respecto a los autos que deseen hacer uso del peaje deberán tener las etiquetas RFID SLU-F adheridas en el parabrisas del auto, estando dentro del área de lectura serán interrogadas por el Lector SL130 enviando la información almacenada en su memoria y realizar el registro en la base de datos de PostgreSQL.

La información obtenida por el Lector SL130, será enviada al módulo de comunicación que se encuentra conformado por un PIC 18F2550, que tendrá una comunicación RS-485, por donde interactúa con el lector y con la computadora mediante una adaptación RS-232, donde se encuentra el servidor. Asimismo, este módulo de comunicación es el encargado de enviar la información a nuestra base de datos desarrollada en PostgreSQL, donde serán almacenados los datos simultáneamente al ser detectada una etiqueta. Los datos a almacenar deberán ser la fecha y hora en la que el vehículo hizo uso del peaje con su respectivo código de etiqueta y el descuento según la cantidad de ejes del vehículo, para posteriormente realizar un conteo del uso del peaje para el débito correspondiente. En la figura 10, se puede apreciar la arquitectura del proyecto y el proceso de la adquisición de datos capturados por el Lector LS130, de igual manera muestra las diferentes etapas que componen nuestro sistema para automatizar el peaje, etapa de RFID, etapa comunicación con el microcontrolador 18f2550 y por último la etapa de servidor

y base de datos que se encontraran funcionando en una pc o laptop.

Figura 10 Arquitectura del Proyecto



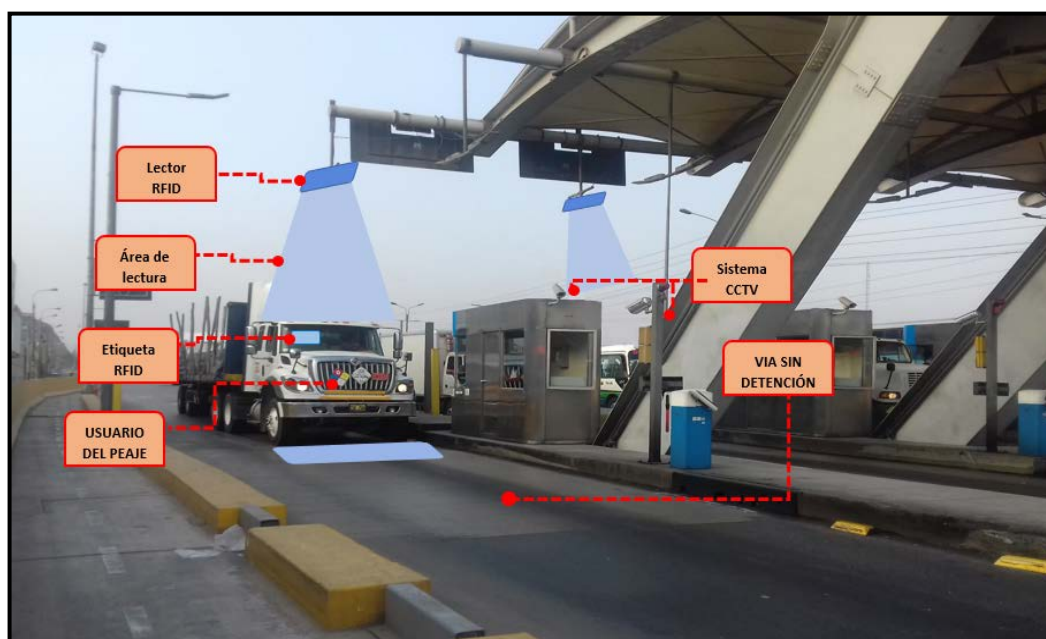
Fuente: Elaboración Propia

Con este sistema se propone reducir el tiempo del cruce por el peaje, a lo más mínimo aproximadamente en 10 segundos. Lo que, comparados con el tiempo anterior de 2 minutos con 30 segundos, mejorando el panorama, los vehículos circularían con mayor fluidez, y accesible sin colas de vehículos y sin obstáculos de detección por el uso de la vía, mostrándose un servicio óptimo para el usuario, dando a entender que no es simplemente cobrar por el uso de la vía, sino que se interesan por el usuario y evitar que el pierda su tiempo en colas en medio de los gases contaminantes que emiten los autos, otorgándole un grado de satisfacción a los usuarios del peaje. Como ya se mencionó con anterioridad, se conservará la estructura actual del peaje Chillón, tan solo se realizarán adecuaciones, ya que la misma se encuentra en buen estado a primera vista. Con las adecuaciones realizadas para el nuevo sistema

automatizado. El peaje quedaría como se muestra en la figura 11, que, como se mencionó al inicio este proyecto debe estar complementado por un sistema cerrado de televisión – CCTV, ya que en nuestro país aún no se tiene muy bien definido los valores, en base a esto el sistema será el ente regulador y sancionador para que se respete el límite máximo de velocidad permitido de 30 km/h, registrando a los infractores y para su futura sanción monetaria.

En el proyecto no se considera modificar la estructura actual del peaje Chillón, ya que los módulos son amplios y no obstaculizan el paso de los diferentes tipos de vehículos y sus dimensiones. Con la finalidad de no incrementar los costos de implementación y sobre todo para no crear una congestión vehicular, a la que se da actualmente en esta vía, simplemente la estructura se mantendrá y solo se realizarán las adecuaciones necesarias para la implantación del nuevo sistema, manteniendo así mismo el tipo de orden por donde fluyen los buses, camiones y tráiler que es por el carril derecho debido a sus dimensiones, para ver gráficamente la propuesta y se haga más fácil su entendimiento, ver la información descrita en la figura 11.

Figura 11 Esquema de Adecuación de los Equipos RFID en el Peaje Chillón



Fuente: Elaboración Propia

3.3 DISPOSITIVOS PARA EL PROYECTO

Para llevar a cabo la propuesta ya descrita, se utilizarán materiales distribuidos por la empresa StrongLink, que suministra cientos de miles de lectores RFID y módulos RFID OEM en todo el mundo. Los que se aplican en transporte público, control de acceso, máquinas de casino, sistemas de depósito etc. (Stronglink, 2017)

Así mismo cuenta un demo del software que controla al lector RFID, con el que se modelo la programación del código del circuito maestro para la obtención del código de la etiqueta. El Lector RFID se detalla en la tabla 2

Tabla 2 Datos de Operación del Módulo RFID – SL130

TIPO	DESCRIPCION	
Frecuencia	920MHz~925MHz	
Etiqueta apoyo	ISO18000-6B, ISO18000-6C, EPC Class1 Gen2 tags	
Interfaz	RS232 / RS485 / RJ45 / Wiegand	
Temperatura de funcionamiento	-10°C ~ +55°C	
Distancia de la operación	Leer distancia de> 8m, escribir distancia> 1 m (depende de las etiquetas)	
Potencia de salida	20~30dBm	
Velocidad de lectura	32bits/6ms	
Velocidad de escritura	32bits/50ms	
Peso	6.4kg	
Dimensión	600 × 480 × 110 mm	

Fuente: (Stronglink Technology Co., 2017)

3.3.1 Módulo lector RFID


El lector RFID seleccionado es el SL130 lector UHF integrado y de alto rendimiento, módulo de UHF RFID que se basa de un diseño de recorrido análogo avanzado y señal digital que procesa la tecnología especificada para RFID. Puede leer etiquetas de UHF o tarjetas compatibles con el protocolo de ISO18000-6B/C, EPC CLASE1 GEN2,

ofrece una distancia de operación mucho más larga, usada en control de procesos industriales, control de acceso de Personal, anti-falsificación electrónica, monitoreo logístico, sistema automático de gestión de producción. Por estos motivos se utilizará en este proyecto. (Stronglink Technology Co., 2017)

3.3.2 Antena externa RFID

Para complementar al lector RFID se utilizará una antena la cual es la encargada de entablar la comunicación con la etiqueta y obtener la información. La antena RFID también es comercializada en la misma empresa, el lector RFID al tener un puerto adicional para antena externa, permite buscar antenas con diferentes características como ganancia y valores de Roe, con la finalidad de mejorar la cobertura del lector; pero, con las características actuales tanto del lector y de la antena que se acoplan correctamente para la aplicación que se le quiere dar, dentro del proyecto, por lo tanto, se utilizara la antena que viene asociada al lector. Esta permite ser colocada en el exterior, por lo que está diseñada para estar expuesta a ambientes exteriores o ambientes industriales gracias a su sellado hermético y su protección anti golpes las características de operación se detallan en la tabla 3.

Tabla 3 Datos de Operación de la Antena RFID

TIPO	DESCRIPCIÓN	
Frecuencia	920~925MHz	
Frecuencia Central	922.5MHz	
Escribe	Antena de Cerámica	
V.S.W.R	1.5: 1	
Ancho de banda (V.S.W.R<1.9:1)	>5MHz	
Impedancia	50ohm	
Ganancia	2dBic	
Polarización	RHCP	
Longitud de cable RF	100mm	
Conectores RF	IPX-male	
Dimensión	50 × 50 × 7 mm	
Temperatura de Funcionamiento	-20 ~ +70°C	
Temperatura de Almacenamiento	-40 ~ +85°C	


Fuente: (Stronglink Technology Co, 2017)

Habiendo descrito la información de cada uno de los componentes del módulo de lectura RFID, del nuevo sistema Automatizado de control del peaje Chillón kilómetro 25 de la autopista Panamericana Norte; los dispositivos descritos se ubicaran en la estructura actual del peaje a una altura del suelo de 5.50 metros, para leer todo tipo de vehículo que use el peaje, esto es con respecto al peaje; pero, con respecto a los vehículos ellos deberán contar con un dispositivo pasivo de RFID llamando etiqueta, la cual deberá estar adherida en cualquier área del parabrisas como al conductor mejor le parezca, teniendo en cuenta que una vez pegada la etiqueta en el parabrisas, esta no se podrá ser retirar hasta su próximo reemplazo de la misma, ya que está elaborada de un material que no permite ser retirado para evitar cualquier tipo de fraude o adulteración.

3.3.3 Etiqueta RFID - Slu-f

La etiqueta seleccionada para este proyecto es SLU-F de UHF de StrongLink; que está desarrollada en base a tecnología UHF RFID, que apoya protocolo ISO18000-6C/EPC Gen2 y que puede ofrecer un rango de detección mucho más largo que la tarjeta de HF. Cuenta con carácter de anti-pisón y puede ser usado en el diseño anti-desmontaje y aplicación de verificación de autenticidad, los datos de operación y dimensiones se detallan en la tabla 4. (Strog Link, 2017)

Tabla 4 datos de Operación de la Etiqueta RFID - SLU-F

TIPO	DESCRIPCIÓN	
Frecuencia	840 ~ 960MHz	
Protocolo	ISO18000-6C, EPC Class1 Gen2	
Tamaño EEPROM	EPC 128bits, User 32-512bits	
Material	Material de sabotaje Frágil	
Temperatura	-25°C ~ +75°C	
Modo de Funcionamiento	Pasivo	
Distancia de la operación	3 ~ 10 (m)	
Dimensión	30×40×0.2(mm)	

Fuente: (Strong Link, 2017)

Dejando de lado el área de comunicación inalámbrica a cargo de la tecnología RFID, es necesario describir al microcontrolador encargado de la comunicación entre el módulo lector RFID con el servidor que almacena nuestra base de datos el microcontrolador utilizado es de la marca Microchip y de la familia 18.

3.3.4 Microcontrolador 18F2550

Es uno de los microcontroladores de alta gama, elemento principal en la tarjeta de comunicación. Fue seleccionado para el desarrollo de la comunicación entre el Lector RFID SL-130 con la computadora donde se almacena la base de datos. Este microcontrolador integra comunicación USB, con su gran capacidad de memoria EEPROM que nos permite registrar hasta 100 mil veces nuestra información, así mismo nos permite trabajar a alta frecuencia, gracias a su oscilador interno, permitiendo el desarrollo de aplicaciones que se basan en la capacidad de memoria RAM, para almacenamiento temporal y su memoria FLASH de programa mejoradas hacen que sea ideal para el control integrado y aplicaciones de monitoreo que requieren conexión periódica con un ordenador personal para la carga y descarga de datos. En la Tabla 5, se aprecia los datos de operación y la distribución de los pines Pic-18F2550. (Nextiafenix, 2017)

Tabla 5 Datos de Operación del Pic-18F2550

TIPO	DESCRIPCIÓN
Frecuencia Operación	DC - 48MHZ
Program Memory (Bytes)	32768
Program Memory (Instruct)	16384
Data Memory(Bytes)	2048
Data EEPROM Memory (Bytes)	256
Interrup Sources	19
I/O Ports	Ports A,B,C
Timers	4
Capture /compare / PWM Modules	2
Enhanced Capture /compare / PWM Modules	0
Universal Serial Bus (USB)	1
Streaming Parallel port (SPP)	NO
10-Bit analog-to-Digital modul	10 Input Channels
Comparators	2

Fuente: (SEDA, 2015)

3.3.5 Rs-232 / Rs-485 (rts, tx, rx)

Para poder concretar la comunicación entre el módulo lector RFID SL-130 y el controlador se tuvo que utilizar un adaptador de comunicación de intercambio de datos y así poder entablar la comunicación con la computadora y a su vez homologar los voltajes que son enviados por la computadora y los valores que son permitidos por los microcontroladores (TTL). Por ello se debió de recurrir a módulos rs-232, ya que nos permite comunicación con varios periféricos, con una velocidad mínima y con la limitación de 10 metros como máximo, ideal para la comunicación con el módulo RFID SL-130 ya que cuenta con comunicación serial para su interacción.

Para la adaptación de compatibilidad de comunicación con el microcontrolador y los demás lectores RFID ya que son 5 los que utilizaremos en nuestro proyecto recurriros al protocolo RS-485 ya que nos permite poder construir redes de actuadores (esclavos) y manejar una gran cantidad de información a largas distancias.

3.4 SOFTWARE UTILIZADOS

Para el desarrollo de este proyecto se utilizó diferentes softwares los cuales interactúan en las diferentes etapas que tiene el proyecto como la programación del microcontrolador, la construcción de la base de datos y las interfaces entre la simulación de circuitos y la base de datos.

3.4.1 Mplab X

Es el editor IDE utilizado para la configuración de nuestro microcontrolador 18F2550, ya que es plataforma gratuita diseñada para todo tipo de microcontroladores de la marca Microchip y otras. Permite la configuración, edición de texto, compilación, acceso simulación mediante Proteus Isis y se encuentra diseñada para los

distintos sistemas operativos Windows, Mac OS y Linux. (wikipedia, 2014)

Esta plataforma nos permite la programación de proyectos en lenguaje C, ya que cuenta con diferentes compiladores como el XC8, XC16 para microcontroladores de 8 bits y 16 bits respectivamente. Igualmente, también cuenta con el compilador MPASM, para el lenguaje Ensamblador, que aún es usado por ciertos programadores. Para nuestro proyecto se utilizó el compilador XC8, y una vez escritas y depuradas todas las directivas y sintaxis se procedió a compilarlo y grabarlo en el microcontrolador directamente desde esta plataforma desarrollada por Microchip y mediante el Pickit2.

3.4.2 Python

Es una plataforma con lenguaje de programación multiparadigma, ya que soporta orientación a objetos, por lo que no se requiere compiladores para el código fuente, programación imperativa y en menor medida, programación funcional para todo tipo de aplicación y programa para los diferentes sistemas operativos como Windows, Linux para comunicación o acceso de servidores, páginas web y diferentes aplicaciones con una alta velocidad de respuesta. (Wikipedia, 2017)

En nuestro proyecto, la herramienta Python fue utilizada para poder entablar la comunicación serial entre el microcontrolador y la computadora, a su vez enlazar los datos obtenidos por el microcontrolador con la base de datos desarrollada en PostgreSQL, y esto se pudo realizar gracias a dos módulos pyserial, psycopg2.

3.4.3 PostgreSQL

PostgreSQL es un Sistema de gestión de bases de datos relacional orientado a objetos. Es un gestor potente con grandes prestaciones graficas que permiten un fácil desenvolvimiento en la administración de los datos. Sobre todo, su código fuente está disponible libremente, este software se encuentra dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista, libre o apoyados por organizaciones comerciales, haciendo que sea igual de competente que otras MySQL, Oracle, por este motivo nuestra base de datos fue construida en esta plataforma. (Wikipedia, 2017)

3.4.4 Proteus

Proteus Design Suite es un software de automatización de diseño electrónico, desarrollado por Labcenter Electronics Ltd., que consta de dos programas principales, el primero es ISIS, software de sistema de enrutado de esquemas inteligente que permite el diseño de circuitos. En esta plataforma se desarrolló nuestro circuito de comunicación con el microcontrolador 18F2550, el segundo programa es ARES Software de edición y ruteo avanzado es la herramienta de enrutado, ubicación y edición de componentes, que se utilizó para la fabricación de nuestras placas de circuito impreso. (Wikipedia, 2017)

3.5 DESARROLLO DEL PROYECTO

Se describirán todas las partes correspondientes al proyecto de automatización de registro del uso del peaje para poder cumplir con nuestro objetivo, el optimizar el uso del peaje Chillón de la autopista panamericana norte, iniciando por la configuración y concluyendo con la simulación del envío de datos a la base de datos.

3.5.1 Configuración del lector RFID con reader 1000

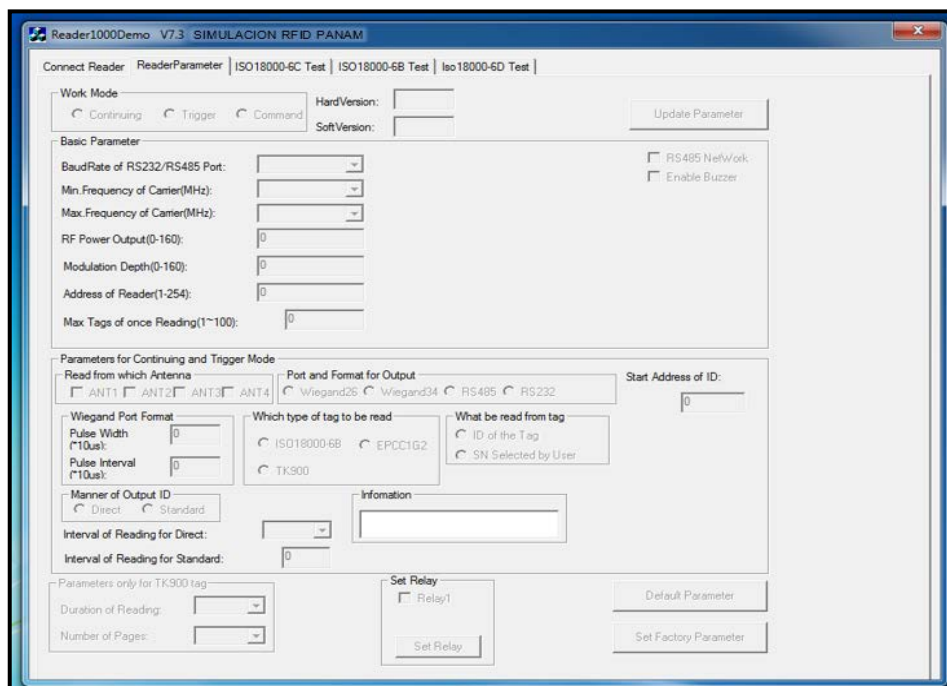
Para que las etiquetas ubicadas en el parabrisas de los autos interactúen en la comunicación del sistema RFID, procederemos a configurar el lector para que este pueda obtener los datos de la etiqueta, configurando la comunicación que sea continua y mantengan la misma frecuencia de operación y otros parámetros que se describen a continuación

- ❖ **Comunicación** seleccionaremos continúa debido a que leerá constante los vehículos que pase por los módulos del peaje.
- ❖ **Rango de Baudios** se configurará en 9600 bps.
- ❖ **Frecuencia de Operación** esta será la frecuencia media del lector 922.5 MHz, ya que las frecuencias laterales son 920~925MHz.
- ❖ **Potencia Tx** colocaremos 30 dBm, como potencia de salida, la cual nos dará el alcance del área de lectura.
- ❖ **Modulación** se asigna de manera automática - FSK
- ❖ **Dirección del lector** es un número del 0 – 254 que escogeremos mediante este número nuestro modulo maestro podrá comunicarse con el lector acceder para poder obtener la información leída.
- ❖ **Máximo de etiqueta leída** en nuestro caso colocaremos 1; ya que solo queremos que lea 1 etiqueta a la vez.
- ❖ **Duración de lectura** colocaremos 50 ms, por lectura ya que solo requerimos obtener el código de la etiqueta para adjuntarlo en la base de datos.

- ❖ **Comunicación** seleccionaremos RS485, ya que así el lector se podrá comunicar con el módulo maestro.
- ❖ **Tipos de etiquetas** seleccionaremos el ISO18000-6C, con este se obtendrá el código de la etiqueta con una numeración de 24 dígitos numéricos.
- ❖ **Información a obtener de la etiqueta**, se seleccionará el ID de la etiqueta, esta información será la que obtenga el lector y que se almacenará en la base de datos y estará ligada al usuario con el tipo de descuento asociado a su tipo de vehículo.

Por último, se guardará la configuración de los parámetros; y, así el lector estará listo para poder conectarse con nuestro maestro encargado de la comunicación, y estará preparado para instalarlo en la parte superior de la estructura del peaje. Para empezar con las lecturas, los parámetros mencionados se configurarán en este aplicativo que viene con el lector que hace el proceso de configuración mucho más flexible. El aplicativo se muestra en la figura 12.

Figura 12 Aplicativo de Configuración del Lector RFID- LS130



Fuente: Elaboración Propia

A continuación, se realiza una prueba en el demo que viene de manera libre en la página web de StrongLink desde el siguiente link: <http://stronglink-rfid.com/es/>; desde aquí se podrá ver el código que almacena las etiquetas RFID SLU-F, las cuales operan bajo el protocolo ISO18000-6C/EPC Gen2, las cuales entre sus parámetros de comunicación, nos permite en su memoria almacenar los códigos de las etiquetas las que nos permitirá diferenciar entre cada tipo de auto que pase por el peaje según la clasificación de la cantidad de los ejes, ya que el peaje Chillón, es utilizado por vehículos públicos y privados, buses, camiones y tráiler, por ello se construyó una tabla en la cual nos permite clasificarlos y asignarle un tipo de código con una longitud de 24 caracteres y este a su vez está asociado a un tipo de descuento, los montos usados en este cuadro son los montos usados actualmente por la concesionaria Rutas de Lima, encargada de los cobros y mantenimientos de vial del peaje Chillón y sus alrededores.

Tabla 6 Clasificación de los Vehículos y Asignación de Código RFID

ETIQUETA ASIGNADA	TIPO DE VEHÍCULO	SUBTOTAL	IGV	TOTAL
AA0027004934ACDE25000000	VEHÍCULOS LIVIANO PRIVADO	S/. 4,24	S/. 0,76	S/. 5,00
AB0027004934ABFD25000000	VEHÍCULOS LIVIANO PUBLICO	S/. 3,82	S/. 0,68	S/. 4,50
B005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 02 EJES	S/. 7,63	S/. 1,37	S/. 9,00
C005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 03 EJES	S/. 11,45	S/. 2,05	S/. 13,50
D005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 04 EJES	S/. 15,26	S/. 2,74	S/. 18,00
E005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 05 EJES	S/. 19,08	S/. 3,42	S/. 22,50
F005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 06 EJES	S/. 22,90	S/. 4,10	S/. 27,00
G005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 07 EJES	S/. 26,71	S/. 4,79	S/. 31,50
H005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 08 EJES	S/. 30,53	S/. 5,47	S/. 36,00
K005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 09 EJES	S/. 34,34	S/. 6,16	S/. 40,50
L005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 10 EJES	S/. 38,16	S/. 6,84	S/. 45,00

Fuente: Elaboración Propia

Ya con esta clasificación de los vehículos según la cantidad de sus ejes, se pudo crear un código para identificar cada tipo de vehículo y así llevar un correcto registro en la base de datos. Por ejemplo, si por el peaje pasa un vehículo liviano privado el lector obtendrá de la etiqueta un código AB0027004934ABFD25000000, y este código se encuentra asociado un valor de débito de S/5.00 nuevos soles, y lo registrará en la base de datos junto a la hora y fecha en la que el vehículo cruzo por el peaje para llevar un control. Lo mismo sucedería con un vehículo pesado de 4 ejes, el lector obtendrá de la etiqueta ubicada en su parabrisas, un código F005FB63AC1F3841EC880000 y asociado un valor de débito de S/27.00 nuevos soles, siendo cada código distinto en su primer caracter de acuerdo a la cantidad de ejes.

3.5.2 Lectura y escritura de Etiqueta RFID

Con la descripción anterior se procede a realizar una simulación del grabado de las etiquetas RFID SLU-F, para ello se deberá conectar el lector RFID a la computadora mediante el adaptador USB a RS232 y seleccionar el COM1, en el demo del lector RFID, se deberá seleccionar la comunicación RS232, y realizar la lectura de la etiqueta como se ve en la figura 13, una vez detectada la etiqueta que estamos utilizando, tendremos acceso a sus 4 bancos de memoria.

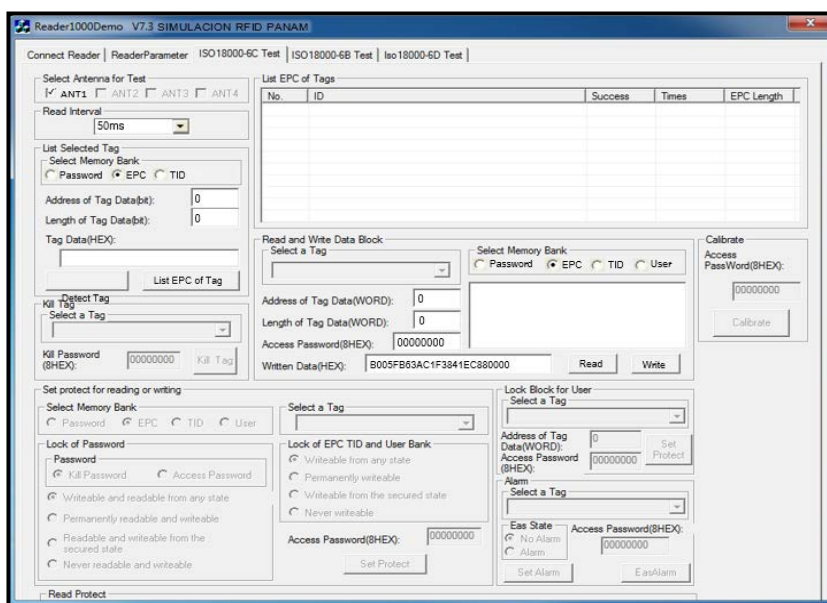
Los bancos de la etiqueta RFID SLU-F son el banco EPC con 200 bits, TID con 64 bits, Usuario con 96 bits y el banco de memoria cuenta con 32 bits, asimismo cada banco cuenta con la opción de protección para datos almacenados, para que estos no sean modificados, por motivos de seguridad.

Para añadir nuestro código en las etiquetas RFID, de los 4 bancos mencionados anteriormente utilizamos el banco de memoria EPC, donde se almacenará el código de clasificación de vehículos, explicado con anterioridad en la tabla 6, que cuenta con una longitud de 24

caracteres por ejemplo “B005FB63AC1F3841EC880000” (ver figura 14); y, en lo que respecta la protección del sistema, se activara el banco de memoria Contraseña, donde se usara la contraseña que viene por defecto que es “00000000”, esta clave será verificada al momento que se requiere modificar datos del banco EPC en otras palabras es un código de protección de lectura. El banco de memoria TID no se puede utilizar debido a que tiene una protección de fábrica que viene por defecto. El banco de memoria Usuario, por el momento no se utilizará quedando memoria para posteriores modificaciones y ampliando la fábrica que viene por defecto. El banco de memoria Usuario, por el momento no se utilizará quedando memoria para posteriores modificaciones y ampliando la funcionalidad del sistema, para no hacer extenso esta etapa del grabado y verificación del código Id de etiqueta, se realizará directamente una demostración.

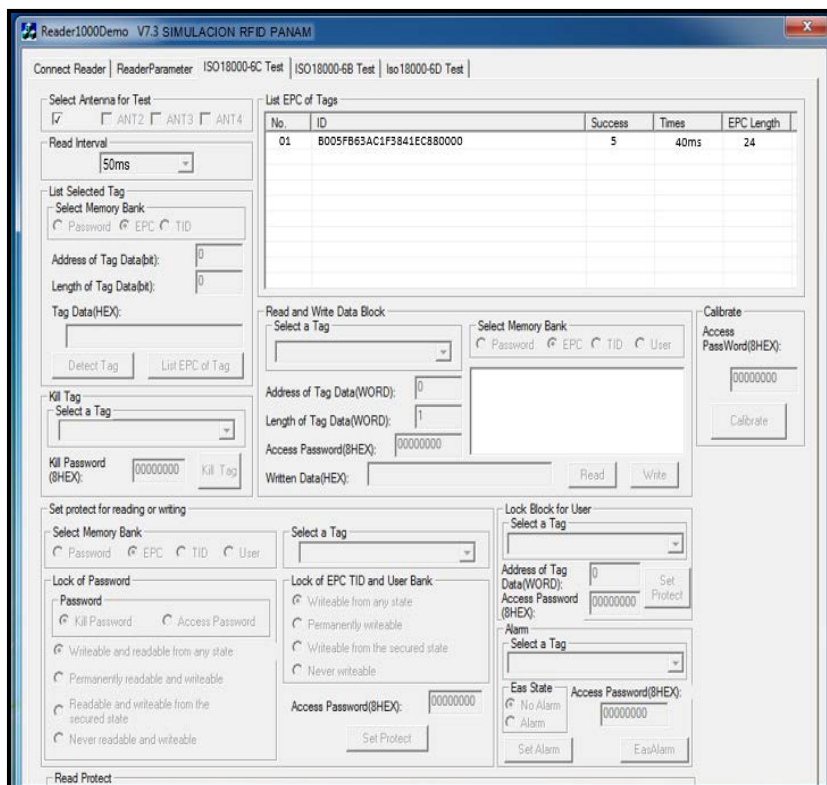
Si desea verificar las etapas de configuración e información del lector RFID-SL130, así como el grabado del código de etiqueta RFID, revise la página web http://www.stronglinkrfid.com/download/SL130_Software_Operation_Manual.pdf

Figura 13 Simulación de Lectura de la Etiqueta SLU-F



Fuente: Elaboración Propia

Figura 14 Verificación del Grabado del Código en la Etiqueta



Fuente: Elaboración Propia

El Lector RFID LS-130, cuenta con sus interfases desarrolladas por el distribuidor StrongLink, el cual ha generado un demo por el cual nosotros tan solo nos limitamos a configurar el modo de operación, frecuencia de operación y nos permite tener la facilidad de realizar la lectura y sobre escritura de las etiquetas RFID SLU-F con nuestro código que se hace referencia en la tabla 6, a partir de aquí todo lo que respecta al sistema RFID ya se encontraría listo para su funcionamiento y entablar comunicación con el Microcontrolador; pero, para ello debemos de configurar la trama de comunicación, para que todo el sistema sea homogéneo.

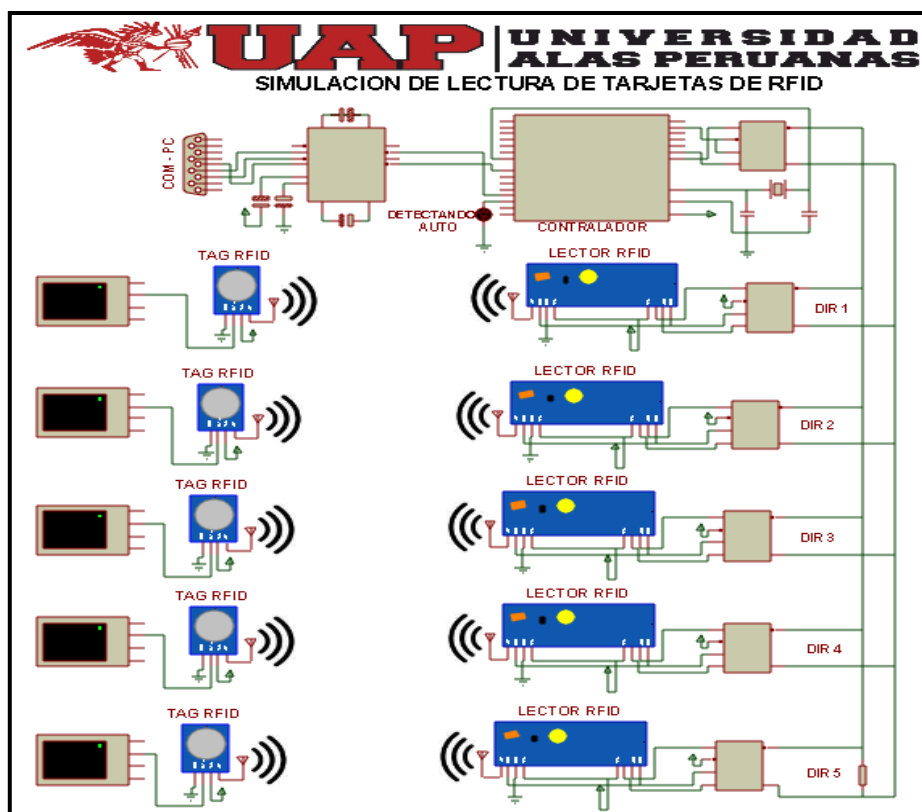
3.5.3 Conexión de los Lectores RFID con el Microcontrolador

Obteniendo la información de la etiqueta y registrada en el lector recién se estaría en la parte inicial, pues aún falta la comunicación con nuestro modulo maestro como ya se indicó, cada lector tiene una dirección del (1 – 254), pero nosotros solo utilizaremos 5 lectores por este motivo serán enumerado 1,2,3,4,5 y así estarán instalados en cada módulo de cobro, este sistema estaría funcionando como un maestro y un esclavo. Todo el sistema RFID representa al esclavo, y el maestro estará conformado por el microcontrolador 18F2550.

El circuito maestro será el encargado de solicitar y recepcionar los datos del Lector RFID (esclavo), como se observa en la figura 15. Para poder realizar esta comunicación y poder crear esta pequeña red sea tenido que recurrir al protocolo RS-485, permitiendo tener una pequeña red de los lectores RFID.

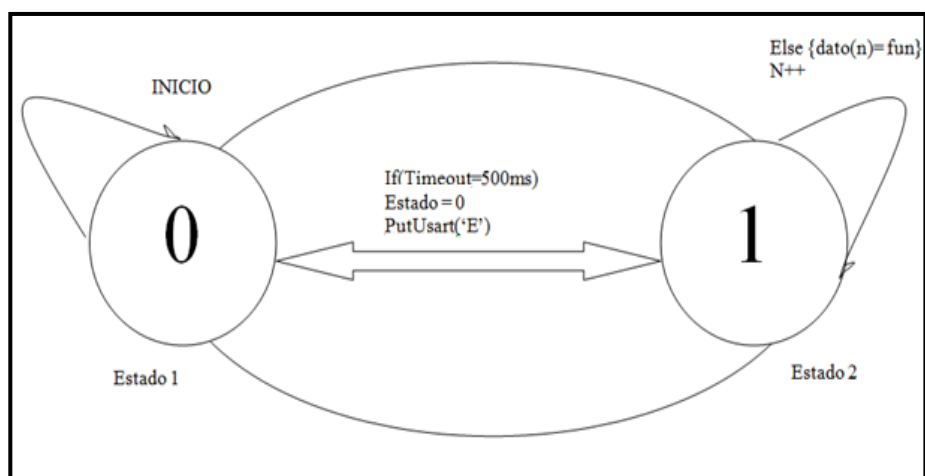
Como se observa en la figura 15, la conexión de los lectores RFID al microcontrolador se realiza mediante el max-485 lo que nos permite tener una larga distancia en la ubicación de los lectores dentro de los módulos del peaje, recordar que estos estarán a 5 metros de altura, la comunicación se realiza de la siguiente forma, los lectores están configurados para cada 30ms realizar lectura, en busca de etiquetas, al detectar una etiqueta, el Lector RFID-SL mandara la información al microcontrolador 18F2550, el cual está constantemente a la escucha, al momento que le llegue solicitud de comunicación de algunos de los 5 lectores RFID activa el canal para la Comunicación como se muestra en la figura 16 que representa la máquina de estado del sistema de comunicación.

Figura 15 Conexión Microcontrolador 18F2550 con
Lector RFID



Fuente: Elaboración Propia

Figura 16 Máquina de Estado de la comunicación del
Lector RFID y el Microcontrolador



Fuente: Elaboración Propia

3.5.4 Construcción de la trama de comunicación

Observamos la interacción de los estados, donde tenemos un inicio y una recepción, el microcontrolador maestro se encuentra en su estado inicial y se va a mantener ahí, hasta que se le mande un comando de solicitud de comunicación por parte del Lector RFID, una vez que se mande dicho comando el cual tendrá una cabecera que será la que inicie la transmisión, hemos configurado esta cabecera con la dirección ADDRES con el valor 50, luego se enviara la dirección del Lector RFID (esclavo) la cual puede ser 01 – 05, ya que se tienen 5 Lectores RFID, continuando con la trama tendremos nuestra información con un longitud de 24 caracteres para enviar datos, así mismo la transmisión no terminara hasta la llegada de la dirección FIN que tiene el valor 255, que indica el fin de la trama, esta trama no cuenta con un sistema de control de errores ya sea por paridad o por Crc.

Desde que el microcontrolador recepciona la cabecera de la trama, empezará a contar si un byte demora más de 500 milisegundos en llegar, el microcontrolador detendrá la comunicación y se volverá a su estado inicio. En caso de que cada byte llegue antes de los 500 ms el sistema mantendrá la comunicación con el esclavo hasta que llegue la dirección con el valor 255 (FIN), con lo cual sabremos que se ha realizado un envío de trama de manera correcta y satisfactoria la trama que se muestra en la figura 17 cada uno de ellos tiene una dirección y valor que ya se encuentra programado en la memoria del microcontrolador.

**Figura 17 Trama de Comunicación entre Lector RFID con
Microcontrolador 18F2550**

INICIO	DIRECCION	DATOS	FIN
2 bytes	2 bytes	24 Bytes.....	3 bytes
50	01....05	A005FB63AC1F3841EC880000	255

Fuente: Elaboración Propia

Con todo lo mencionado anteriormente ya podemos construir nuestro algoritmo, el cual hará lo mismo que hemos explicado con la máquina de estado, la cual estará constituida mediante casos. En la figura 18 observaremos la programación de nuestro algoritmo.

Figura 18 Construcción de Algoritmo de Comunicación

```

#ifndef PROTOCOLOER_H
#define PROTOCOLOER_H
# define ADDR_MASTER '10' // DIRECCION DE MAESTRO
# define ADDR_SLAVE1 '1' //DIRECCION DE 1° ESCLAVO
# define ADDR_SLAVE2 '2' //DIRECCION DE 2° ESCLAVO
# define ADDR_SLAVE3 '3' //DIRECCION DE 3° ESCLAVO
# define ADDR_SLAVE4 '4' //DIRECCION DE 4° ESCLAVO
# define ADDR_SLAVES5 '5' //DIRECCION DE 5° ESCLAVO
# define ADDR '50' //DIRECCION DE INICIO DE TRAMA
# define FIN '255' //DIRECCION DE FIN DE TRAMA
# define TOUT 500 //TIEMPO DE VERIFICACION DE TRAMA
# define PBRX 24 //DATOS DE LA TRAMA EN TRANSMISION
# define PBTX 24 //DATOS DE LA TRAMA EN RECEPCION
# define FIN 'f' //DIRECCION DE FIN DE TRAMA
# define TOUT 500 //TIEMPO DE VERIFICACION DE TRAMA
# define PBRX 16 //DATOS DE LA TRAMA EN TRANSMISION
# define PBTX 16 //DATOS DE LA TRAMA EN RECEPCION

char DATO_RX[PBRX];
char DATO_TX[PBTX];
int estado = 0 ; // ESTADO INICIAL
int timeout = 0; // TIEMPO DE CORTE EN VERIFICACION DE LA TRAMA
char fun; // FUNCION
int N = 0;
int bandera = 0;

```

Fuente: Elaboración Propia

Figura 19 Construcción de los Estados del sistema

```

void vprotocolo(char dato)
{
switch(estado)
{
case 0:
if( dato==ADDR_MASTER)
{estado= 1;
N=0;}
else
{estado= 0;
bandera= 0;}
break;
case 1:
if( dato== FIN & N> (PBRX -1 ))
{estado= 0;
N=0;
bandera = 1; }
else if (timeout >= TOUT){ estado = 0 ; }
else {DATO_RX[N]=dato;
N++;
vInittimer();
}
}
}

```

Fuente: Elaboración Propia

En la figura 18 y figura 19 se muestra la configuración que se realiza en el microcontrolador que se explicó en la máquina de estado, si se encuentra interesado y desea ver con más detalle, en lo que respecta a la configuración del microcontrolador puede encontrar el código fuente se ha adjunta en el Anexo 1.

La construcción de la trama como ya se mencionó, contara de 31 bytes los cuales le indicara al microcontrolador, que un vehículo ha sido detectado por el lector RFID al cruzar por el peaje, dentro de esta trama tiene la información del ID de la etiqueta que tiene una longitud de 24 caracteres, los que el microcontrolador detectara a finalizar la trama y los enviara a la computadora para su registro en la base de datos.

3.5.5 Construcción de la base datos

La base de datos se desarrolló en PostgreSQL. En ella se almacenará a los usuarios que hagan uso del peaje. La base de datos se encuentra conformada por 14 tablas donde se encuentra distribuidas de la siguiente forma

Tabla peaje

En esta tabla se almacena en tiempo real los datos del vehículo que hacen uso del peaje y está conformada por número de orden del vehículo que cruza por el peaje, almacena el código de su etiqueta RFID el cual consta de 24 caracteres y que obtiene desde lector RFID, la fecha y hora que se realizó el uso del peaje y por último según su código se encuentra asociado a un descuento según la clasificación del tipo de vehículo, así como se representa en la figura 20.

Figura 20 Contenido de la Tabla Peaje

	norde [PK] s	codpeaje character varying(40)	fecha date	hora time without	descuento money
1	556	AA0027004934ACDE250000C	2017-06-04	17:15:52	-4,50 €
2	557	B005FB63AC1F3841EC8800C	2017-06-04	17:15:54	-4,50 €
3	558	AA0027004934ACDE250000C	2017-06-04	17:15:59	-4,50 €
4	559	AA0027004934ACDE250000C	2017-06-04	17:16:00	-4,50 €
5	560	AA0027004934ACDE250000C	2017-06-04	17:16:02	-4,50 €
6	561	AA0027004934ACDE250000C	2017-06-04	17:16:04	-4,50 €

Fuente: Elaboración propia

Tabla Registro

Almacena la información referente a los usuarios que se van registrando en el sistema para adquirir la etiqueta RFID y hacer uso del peaje de manera automática y evitar hacer colas en su trayecto al cobro por la vía. Esta tabla almacena el ID de etiqueta RFID, los nombres del propietario del vehículo, DNI, la dirección del propietario, distrito, el tipo de vehículo (auto, camión), cantidad de ejes lo que permite clasificar al vehículo, la placa de rodaje, la fecha y hora que se realizó el registro y por último el crédito que tiene la etiqueta que obtiene desde lector RFID, así como se representa en la figura 21.

Figura 21 Contenido de la Tabla Registro

Id_Tag_asignado [PK] character varyin	Nombres_apellidos text	Dni_usuario numeric	Direccion_usuario text	Distrito text	Tipo_veh text	Cant_ejes text	Placa_vel character	Fecha_insc timestamp with time	Credito money
AA0027004934ACDE	MEDELLIN ACEVEDO	73020467	CA. GARCILASO DE	LIMA	AUTO	L.PRIVADO	7FM-800	2017-06-03 12:11	30,00
AA0027004934ACDE	MONTERIA AGUIRRE	73020468	AV. ALFREDO SALA	PUEBLO	AUTO	L.PRIVADO	7FM-801	2017-06-03 12:12	30,00
AA0027004934ACDE	BUCARAMANGA BETANC	73020470	AV. HIPOLITO UNA	PUEBLO	AUTO	L.PRIVADO	7FM802	2017-06-03 12:24	30,00
AA0027004934ACDE	BUCARAMANGA BETANC	73020471	URB. CIUDAD DEL	CALLAO	AUTO	L.PRIVADO	7FM803	2017-06-03 12:25	30,00
AA0027004934ACDE	MEDELLIN CIFUENTES	73020473	AV. EL DERBY -	SANTIA	AUTO	L.PRIVADO	7FM-804	2017-06-03 12:37	30,00
AA0027004934ACDE	MEDELLIN COCK IGNA	73020474	CA. CHINCHON - 5	JESUS	AUTO	L.PRIVADO	7FM-805	2017-06-03 12:38	30,00
AB0027004934ABFD	MEDELLIN ARANGO	70474677	AV. PORONGOCHÉ -	LINCE	AUTO	L.PUBLICO	EGG-625	2017-06-03 12:14	30,00
AB0027004934ABFD	BOGOTA BUSTOS DAVI	70474680	CA. CORONEL INCL	BRENA	AUTO	L.PUBLICO	EGG-626	2017-06-03 12:27	30,00
AB0027004934ABFD	PALMIRA DE SANTIS	70474683	JR. ELOY REATEGU	CHORRI	AUTO	L.PUBLICO	EGG-650	2017-06-03 12:40	30,00
B005FB63AC1F3841	MONTERIA ALZATE FR	44789418	AV. AREQUIPA - 7	JESUS	CAMION	V.2 EJES	NGA-333	2017-06-03 12:13	30,00
B005FB63AC1F3841	BOGOTA ARANGO AITC	44789418	AV. AVENTURA VEN	CHORRI	CAMION	V.2 EJES	NGA-334	2017-06-03 12:15	30,00
B005FB63AC1F3841	PALMIRA BURGOS INI	44789421	CA. LOS JAZMINE	PUEBLO	CAMION	V.2 EJES	NGA-335	2017-06-03 12:26	30,00
B005FB63AC1F3841	BOGOTA CANO ANDER	44789421	AV. TERUEL - 315	LINCE	CAMION	V.2 EJES	NGA-336	2017-06-03 12:28	30,00
B005FB63AC1F3841	BOGOTA DIAZ CARLC	44789424	AV. JAVIER PRADO	SAN IS	CAMION	V.2 EJES	NGA-380	2017-06-03 12:41	30,00
B005FB63AC1F3841	PALMIRA CORTES ROE	44789424	CALLE LOS HALCON	LINCE	CAMION	V.2 EJES	NGA-370	2017-06-03 12:39	30,00
C005FB63AC1F3841	MEDELLIN ARANGO JC	40792343	URB. MAGISTERIAL	BRENA	CAMION	V.3 EJES	RCA-631	2017-06-03 12:17	30,00
C005FB63AC1F3841	MONTERIA CARMONA	40792346	AV. ARICA -180	BRENA	CAMION	V.3 EJES	RCA-632	2017-06-03 12:30	30,00
C005FB63AC1F3841	MONTERIA DIAZ XAE	40792349	CA. GENOVA - 96	ANCON	CAMION	V.3 EJES	RCA-639	2017-06-03 12:43	30,00
D005FB63AC1F3841	BOGOTA ARISTIZABAL	72004950	CA. AGUAMARINA -	JESUS	CAMION	V.4 EJES	OGM-462	2017-06-03 12:20	30,00
D005FB63AC1F3841	MEDELLIN CARO JOSE	72004953	CA. ALCANFORES	CHORRI	CAMION	V.4 EJES	OGM-463	2017-06-03 12:33	30,00
D005FB63AC1F3841	MONTERIA AGUIRRE	72004956	AV. LA ENCALADA	SANTIA	CAMION	V.4 EJES	OGM-472	2017-06-03 12:46	30,00

Fuente: Elaboración Propia

Tabla Recarga

Almacena la información referente a los usuarios que quiere incrementar su crédito para el uso del peaje. Las recargas pueden

oscilar entre s/. 5.00 hasta s/. 100.00 nuevos soles; la información almacenada y la que se requiere para realizar la recarga es Id (numeración de recarga); para poder identificar al usuario de la recarga se requiere su DNI, placa de rodaje del vehículo, esta se encuentra asociada con el ID de la etiqueta, así mismo se requiere la fecha y hora de la operación junto con el monto de la recarga, como se aprecia en la figura 22.

Figura 22 Contenido de la Tabla Recarga

	Id [PK] char	Dni_usuario numeric	Placa_veh character varying(7)	Fecha_insc timestamp with time zone	Credito money
1	0001	73020467	7FM-800	2017-06-05 14:11:00-05	10,00 €
2	0002	73020468	7FM-801	2017-06-05 14:12:00-05	15,00 €
3	0003	44789418	NGA-333	2017-06-05 14:13:00-05	10,00 €
4	0004	70474677	EGG-625	2017-06-05 14:14:00-05	20,00 €
5	0005	44789418	NGA-334	2017-06-05 14:15:00-05	15,00 €
6	0006	63560272	RBT-361	2017-06-05 14:16:00-05	5,00 €
7	0007	40792343	RCA-631	2017-06-05 14:17:00-05	10,00 €
8	0008	73560159	FAV-451	2017-06-05 14:18:00-05	20,00 €
9	0009	73560170	FAV-452	2017-06-05 14:19:00-05	30,00 €
10	0010	72004950	OGM-462	2017-06-05 14:20:00-05	10,00 €
11	0011	42004923	AMR-692	2017-06-05 14:21:00-05	15,00 €
12	0012	60004983	MRV-322	2017-06-05 14:22:00-05	10,00 €
13	0013	70035599	ATV-815	2017-06-05 14:23:00-05	20,00 €
14	0014	73020470	7FM802	2017-06-05 14:24:00-05	15,00 €
15	0015	73020471	7FM803	2017-06-05 14:25:00-05	5,00 €
16	0016	44789421	NGA-335	2017-06-05 14:26:00-05	10,00 €
17	0017	70474680	EGG-626	2017-06-05 14:27:00-05	20,00 €
18	0018	44789421	NGA-336	2017-06-05 14:28:00-05	30,00 €

Fuente: Elaboración Propia

Tabla tipo de Vehículo según sus Ejes

Esta tabla sirve para clasificar a todos los usuarios que se encuentran dentro tabla de registro, clasificándolos por la cantidad de ejes (vehículos livianos públicos, privados) vehículos que tienen entre 2 ejes hasta 10 ejes, para cada uno de ellos se creado una tabla haciendo un total de 11 tablas. Estás se resumen en almacenar el ID de etiqueta, nombre y apellido del usuario, DNI, el tipo de vehículo

(indicando la cantidad de ejes) placa de rodaje del vehículo, fecha y hora en que el usuario se registró en el sistema, como se aprecia en la figura 23 que hace referencia a la tabla que almacena la información de vehículos de 2 ejes.

Figura 23 Tabla de Vehículos de 2 Ejes

	Id_tag [PK] character varying(30)	Usuario_tag text	Dni_usuario numeric	Tipo_auto character v	Placa_aut character	Fecha_insc timestamp with time zc	Credito money
1	B005FB63AC1F3841EC880000	ACEVEDO MEDELLÍN MELINA	44789323	V.2 EJES	NGA-238	2017-05-30 18:35:00	40,00 €
2	B005FB63AC1F3841EC880001	AGUIRRE MONTERIA BRENDA	44789324	V.2 EJES	NGA-239	2017-05-30 18:37:00	35,00 €
3	B005FB63AC1F3841EC880002	ALZATE MONTERIA ANGELA	44789325	V.2 EJES	NGA-240	2017-05-30 18:39:00	50,00 €
4	B005FB63AC1F3841EC880003	ARANGO MEDELLÍN MONICA	44789326	V.2 EJES	NGA-241	2017-05-30 18:41:00	20,00 €
5	B005FB63AC1F3841EC880004	ARANGO BOGOTÁ MAURICIO	44789327	V.2 EJES	NGA-242	2017-05-30 18:43:00	30,00 €
6	B005FB63AC1F3841EC880005	ARANGO MEDELLÍN FELIPE	44789328	V.2 EJES	NGA-243	2017-05-30 18:45:00	40,00 €
7	B005FB63AC1F3841EC880006	ARANGO MEDELLÍN JUAN	44789329	V.2 EJES	NGA-244	2017-05-30 18:47:00	10,00 €
8	B005FB63AC1F3841EC880007	ARANGO MEDELLÍN TATIANA	44789330	V.2 EJES	NGA-245	2017-05-30 18:49:00	5,00 €
9	B005FB63AC1F3841EC880008	ARIAS PALMIRA FRANCISCO	44789331	V.2 EJES	NGA-246	2017-05-30 18:51:00	10,00 €
10	B005FB63AC1F3841EC880009	ARISTIZABAL BOGOTÁ NATALI	44789332	V.2 EJES	NGA-247	2017-05-30 18:53:00	25,00 €

Fuente: Elaboración Propia

La administración de esta base de datos se da mediante los usuarios. Nuestra base de datos cuenta con dos tipos de usuarios que son:

Operador

Este tipo de usuario se ha creado con la finalidad de que manipule o visualice los datos del uso del peaje, que pueda verificar en tiempo real a los vehículos que están haciendo uso del peaje y que pueda gestionar el registro para nuevos usuarios y la opción de realizar recargas de crédito para el uso de la vía, pero no tiene la opción de poder modificar datos, para ello debe tener acceso de superusuario, privilegios que solo tiene el Administrador. Para poder ingresar al sistema este usuario cuenta con Id: *Operador* y su clave de acceso: *123456*

Administrador

Este tipo de usuario mantiene los mismos accesos que el usuario Operador, pero también cuenta con todos los privilegios de un superusuario, acceso a los diferentes módulos dentro de PostgreSQL, tiene la opción de realizar cualquier tipo modificación en la información almacenada, así como la opción de poder verificar valores y corrección en el sistema y acceso remoto, para poder ingresar al sistema este usuario cuenta con Id: Administrador y su clave de acceso: admin; en la figura 24 muestra la composición de nuestra base de datos y en la figura 25 se aprecian los usuarios de la base de datos.

Figura 24 Base de Datos para el Peaje Chillón

The screenshot displays the PostgreSQL Enterprise Console interface for the 'RFID' database. The left pane shows the object explorer with the following structure:

- Bases de Datos (2)
 - RFID (highlighted with a red box and label 'Nombre de base de datos')
- Catálogos (2)
- Eventos disparadores
- Extensiones (1)
- Esquemas (1)
 - public
 - Clasificados (0)
 - Dominios (0)
 - Configuraciones FTS (0)
 - Diccionarios FTS (0)
 - Analizadores FTS (0)
 - Plantillas FTS (0)
 - Funciones (0)
 - Secuencias (1)
 - Tablas (14)
 - peaje
 - recargas
 - registorfid
 - veh_10_ejes
 - veh_2_ejes
 - veh_3_ejes
 - veh_4_ejes
 - veh_5_ejes
 - veh_6_ejes
 - veh_7_ejes
 - veh_8_ejes
 - veh_9_ejes
 - veh_privados
 - veh_publicos
 - Funciones Disparador
 - Vistas (0)
 - Réplica Slony (0)
 - postgres
 - Tablespaces (2)
 - Roles de Grupos (1)
 - Roles de Login (3)
 - Administrador
 - postgres
 - Operador

The right pane shows the 'Propiedades' (Properties) tab for the 'RFID' database:

Propiedad	Valor
Nombre	RFID
OID	16384
Propietario	postgres
ACL	
Tablespace	pg_default
Espacio de tabla por defecto	pg_default
Codificado	UTF8
Colación	Spanish_Spain.1252
Tipo caracter	Spanish_Spain.1252
Schema por defecto	
Tabla ACL por defecto	
Secuencia ACL por defecto	
Función ACL por defecto	
Tipo ACL por defecto	
Permitir conexiones?	Si
¿Conectado?	Si
Límite de conexión	-1
¿Base de datos del sistema?	No
Comentario	

The bottom pane shows the 'Panel SQL' with the following command:

```
-- Database: "RFID"
-- DROP DATABASE "RFID";

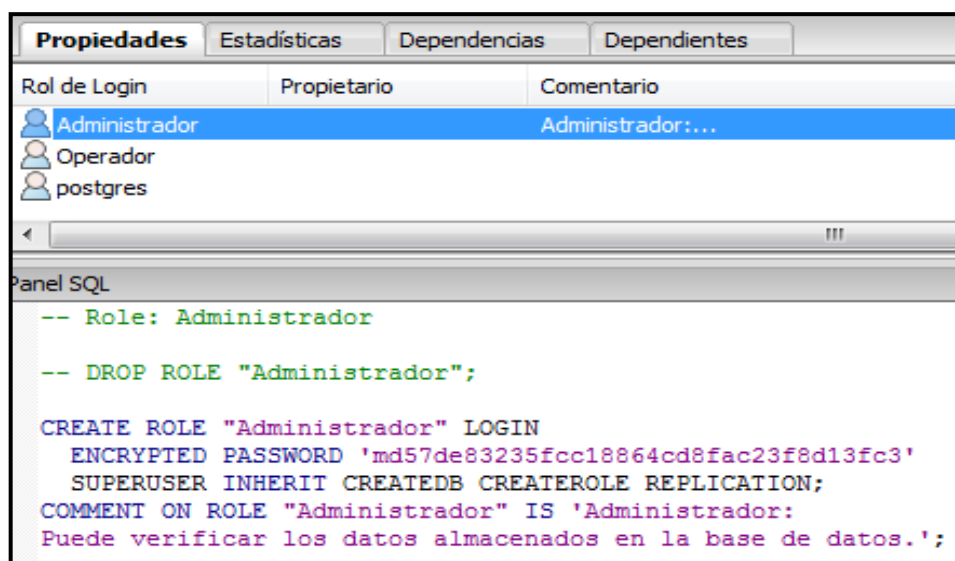
CREATE DATABASE "RFID"
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
LC_COLLATE = 'Spanish_Spain.1252'
LC_CTYPE = 'Spanish_Spain.1252'
CONNECTION LIMIT = -1;
```

Red annotations in the image point to these specific elements:

- 'Nombre de base de datos' points to the 'RFID' database name in the tree.
- 'Información de la base de dato' points to the properties table.
- 'Tablas Creadas para el Sistema RFID' points to the 'Tablas' folder.
- 'Panel de Comandos SQL' points to the SQL editor.
- 'Usuarios de la base de datos' points to the 'Roles de Login' folder.

Fuente: Elaboración Propia

Figura 25 Usuarios de Administración de la Base Datos



Fuente: Elaboración Propia

3.5.6 Comunicación entre el microcontrolador y PostgreSQL

Para entablar esta comunicación entre el microcontrolador 18F2550 y la base de datos RFID, la cual ha sido creada en PostgreSQL, se tuvo que recurrir al módulo de adaptación psycopg2, que es una herramienta que permite que los datos de Python puedan ser interpretados por PostgreSQL.

En el proyecto cumple una función importante ya que el lector RFID al detectar que un vehículo está haciendo uso del peaje, tomará el ID de la Etiqueta asociada al vehículo, mandará los datos al microcontrolador 18F2550 y este mediante la comunicación serial enviará los datos a la base de datos mediante la configuración creada en Python, para ello se deberá de configurar la comunicación serial COM3, para enlazarlo con el microcontrolador.

Los datos que recibe Python deberán ser añadidos a nuestra tabla en PostgreSQL, que tiene como nombre Peaje, que almacena ID

del vehículo, fecha y hora del uso del peaje y el valor del débito a realizar, para que suceda esto debemos de crear el acceso a la base de datos y se hace escribiendo la ubicación de la base de datos en nuestro caso es localhost, el nombre de la base de datos RFID, junto a la contraseña que es 123456789 e indicar el nombre de la tabla que ha sido creada previamente con el nombre de peaje, en la figura 26 se aprecia la construcción del redireccionamiento de los datos a la base de datos.

Figura 26 Redireccionamiento de Datos la PostgreSQL

```

*RFID5.py - C:\Python27\SIMULACION RFID\RFID5.py (2.7.13)*
File Edit Format Run Options Window Help
Python 2.7.13 (v2.7.13:a06454b1afaf1, Dec 17 2016, 20:42:59)
[MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.

# Puerta de enlace GWRfid y Base de Datos PGSQL
import serial
import psycopg2
import time

con = None

try:
    gwrfid=serial.Serial('COM20')
    con = psycopg2.connect(host='localhost', database='RFID',
                          user='postgres', password='123456789')
    cur = con.cursor()
    cur.execute("SELECT version()")
    ver = cur.fetchone()
    print ver

    while(True):
        try:
            datos=gwrfid.readline()
            idrfid=datos[0:26]
            fecha=time.strftime("%Y/%m/%d")
            hora=time.strftime("%H:%M:%S")
            print datos[0:17], fecha , hora
            #print "insert into peaje(codpeaje,fecha,hora,descuento)
            values"+"+"+idrfid+"+"+"+"+fecha+"+"+"+"+hora+"+"+"+"-
            cur.execute("insert into peaje(codpeaje,fecha,hora,descuento)
            values"+"+"+idrfid+"+"+"+"+fecha+"+"+"+"+hora+"+"+"+"-
            con.commit()

        except (Exception, psycopg2.DatabaseError) as error:
            print(error)

        except: #Excepcion por interrupciones Teclado (CTRL+C)
            print "Servicio Detenido"
            break

    gwrfid.close()
    con.close()

```

Ln: 41 Col: 15

Fuente: Elaboración Propia

3.5.7 Simulación del sistema

Habiendo descrito todas las etapas que interactúan en el sistema y habiendo concluido todo lo referente a la configuración del sistema, se procederá a realizar la comprobación del funcionamiento del sistema y para ello se utilizará el simulador de circuitos electrónico Proteus Isis.

En lo referente a la etapa del sistema RFID (lector y etiqueta), teniendo en cuenta su alto costo, se procederá a simularlo con un módulo de comunicación inalámbrica un transmisor y receptor dentro de la plataforma de Proteus Isis, para poder comprobar las demás etapas del sistema, las que se irán detallando paso a paso. En la figura 27, se muestra el circuito de simulación del proyecto, donde se muestra el microcontrolador y la comunicación con la computadora y la adquisición de datos del sistema RFID, existe un virtual terminal llamado "Tag", por el cual mandaremos los datos al microcontrolador el cual va recepcionar y los mandara a la computadora mediante el comunicación serial por este motivo se utilizara otro virtual terminal llamado "Datos", para verificar que los datos recepcionados por el microcontrolador sean los mismo que se registraran en la computadora.

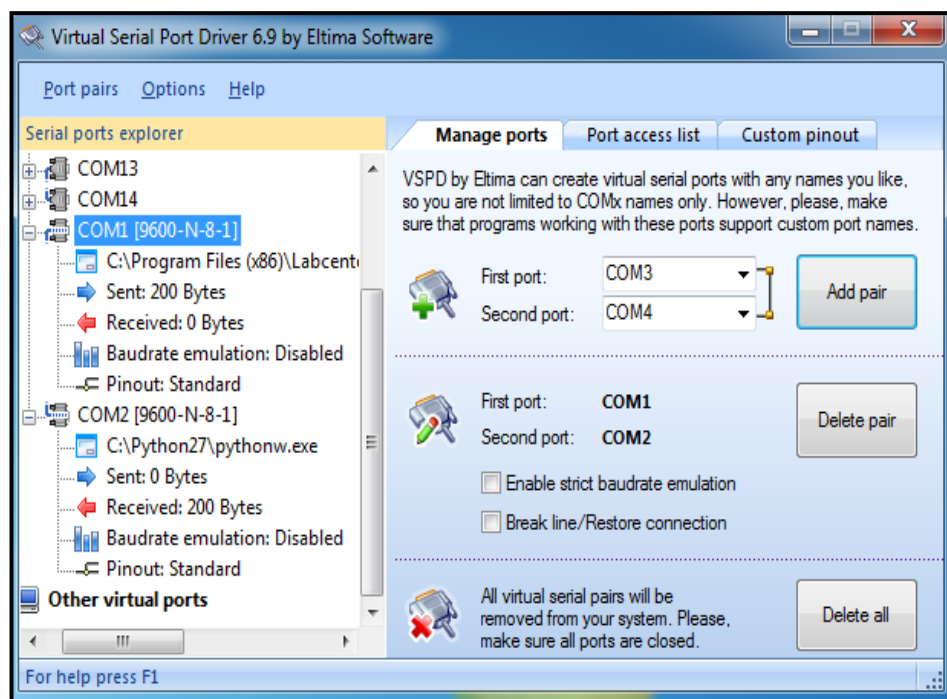
Figura 27 Circuito de Simulación del Proyecto



Fuente: Elaboración Propia

Como se visualiza en la figura 27, el microcontrolador ha recepcionado el código de la etiqueta AB0027004934ABFD25000010, que se muestra en el virtual terminal llamado "Datos", que fue enviado con ayuda del virtual terminal llamado "TAG", este código de etiqueta se encuentra asociado a un tipo de vehículo, en este caso hace referencia vehículo liviano público con un descuento por el uso de la vía de ^{s/} 5.00 nuevos soles, por último se adjuntara la fecha y hora del evento, para ser almacenados a nuestra base de datos que se almacena en la computadora, para entablar la comunicación recurrimos a software Virtual Serial Port, que permite la conexión serial de manera virtual, se asignará el COM1 para la comunicación con el microcontrolador y el COM2 estará asociado a Python la para interface con PostgreSQL, como se muestra en la figura 28.

Figura 28 Conexión Serial Virtual



Fuente: Elaboración Propia

Una vez establecida la conexión de los puertos seriales y verificando el envío y recepción de datos como se muestra en la figura

28, se procede a verificar en la plataforma Python la recepción del ID de la etiqueta RFID, que han sido enviados por el microcontrolador 18F2550, el ID es “AB0027004934ABFD25000010”, que gracias a esta herramienta el dato será redirigido mediante unos comandos que podrán ser interpretado por la plataforma de PostgreSQL y estos datos se irán añadiendo a tabla llamada peaje, ver figura 29 que muestra la adquisición del código de la Etiqueta RFID.

Figura 29 Recepción de Datos por Python

The image shows two side-by-side windows from a Python 2.7.13 Shell. The left window displays the output of a script, showing a series of RFID IDs and timestamps. The right window shows the source code of the script, which uses the serial module to read data from a COM2 port and the psycopg2 module to connect to a PostgreSQL database and insert the data into a table named 'peaje'.

```

Python 2.7.13 (v2.7.13:a06454b1af1, Dec 17
(Intel) on win32
Type "copyright", "credits" or "license()"
>>>
===== RESTART: C:\Python27\SIMULACI
('PostgreSQL 9.6.2, compiled by Visual C++
AB0027004934ABFD25000010
2017/06/23 23:15:22
AB0027004934ABFD25000013
2017/06/24 00:01:29
AB0027004934ABFD25000015
2017/06/24 00:01:31
AB0027004934ABFD25000016
2017/06/24 00:01:32
AB0027004934ABFD25000017
2017/06/24 00:01:36
AB0027004934ABFD25000021
2017/06/24 00:01:40
AB0027004934ABFD25000022
2017/06/24 00:01:42

# Puerta de enlace GWR RFID y Base de Datos PostgreSQL
import serial
import psycopg2
import time

con = None

try:
    gwrifid=serial.Serial('COM2')
    con = psycopg2.connect(host='localhost', database='RFID', user='postgres',
        password='123456789')

    cur = con.cursor()
    cur.execute("SELECT version()")
    ver = cur.fetchone()
    print ver

    while(True):
        try:
            datos=gwrifid.readline()
            idrfid=datos[0:26]
            fecha=time.strftime("%Y/%m/%d")
            hora=time.strftime("%H:%M:%S")
            print datos[0:26], fecha , hora
            #insert into peaje (codpeaje, fecha, hora, descuento)
            values="("+"+"+idrfid+"+"+"+"+fecha+"+"+"+"+hora+"+"+"+"
            cur.execute("insert into peaje (codpeaje, fecha, hora, descuento)
            values"+"("+"+"+idrfid+"+"+"+"+fecha+"+"+"+"+hora+"
            con.commit()

        except (Exception, psycopg2.DatabaseError) as error:
            print(error)

        except: #Excepcion por interrupciones Teclado (CTRL+C)
            print "Servicio Detenido"
            break
    gwrifid.close()
    con.close()

except: #Excepcion por interrupciones Puerto USB
    print "Puerto Inactivo"

```

Fuente: Elaboración Propia

En la figura 30 se muestra la recepción del código de la Etiqueta RFID junto con la hora y el descuento asociado que se almacenara en la tabla peaje.

**Figura 30 Verificación del Registro de Código
en la Base de Datos**

The screenshot shows the pgAdmin III interface. On the left, the 'Explorador de Objetos' (Object Explorer) displays the database structure for 'POSTGRES-WAPP (localhost:5432)', including 'Bases de Datos (2)', 'RFID', 'Catálogos (2)', 'Eventos disparadores (0)', 'Extensiones (1)', 'Esquemas (1)', 'public', 'Clasificados (0)', 'Dominios (0)', 'Configuraciones FTS', 'Diccionarios FTS (0)', 'Analizadores FTS (0)', 'Plantillas FTS (0)', 'Funciones (0)', 'Secuencias (1)', 'Funciones Disparadas', 'Vistas (14)', 'Replica Slony (0)', 'postgres', 'Tablespaces (2)', 'Roles de Grupos (1)', and 'Roles de Login (3)'. The 'Propiedades' (Properties) pane shows the 'Tabla' (Table) 'peaje' owned by 'postgres'. The 'Panel SQL' (SQL Panel) contains the following code:

```
-- Table: public.peaje
-- DROP TABLE public.peaje
CREATE TABLE public.peaje
```

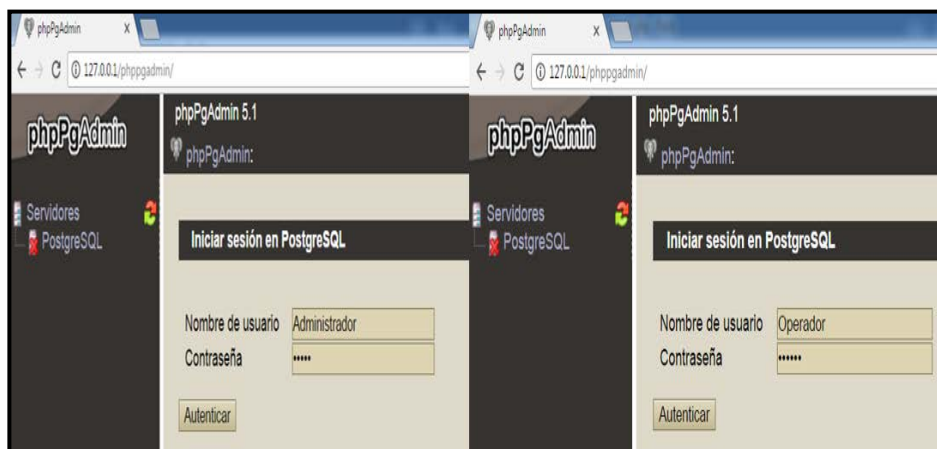
On the right, the 'Editar Datos - POSTGRES-WAPP (localhost:5432) - RFID - public.peaje' window displays a table with the following data:

	norden	codpeaje	fecha	hora	descuento
	[PK]	se character varying(40)	date	time without	money
1	576	AB0027004934ABFD25000012	2017-06-22	02:44:56	-5,00 €
2	577	AB0027004934ABFD25000012	2017-06-22	02:45:58	-5,00 €
3	578	AB0027004934ABFD25000013	2017-06-22	02:46:44	-5,00 €
4	579	AB0027004934ABFD25000015	2017-06-22	02:46:47	-5,00 €
5	580	AB0027004934ABFD25000016	2017-06-22	02:46:49	-5,00 €
6	581	AB0027004934ABFD25000017	2017-06-22	02:46:52	-5,00 €
7	582	AB0027004934ABFD25000021	2017-06-22	02:46:57	-5,00 €
8	583	AB0027004934ABFD25000010	2017-06-23	23:15:22	-5,00 €
9	584	AB0027004934ABFD25000013	2017-06-24	00:01:29	-5,00 €
10	585	AB0027004934ABFD25000015	2017-06-24	00:01:31	-5,00 €
11	586	AB0027004934ABFD25000016	2017-06-24	00:01:32	-5,00 €
12	587	AB0027004934ABFD25000017	2017-06-24	00:01:36	-5,00 €
13	588	AB0027004934ABFD25000021	2017-06-24	00:01:40	-5,00 €
14	589	AB0027004934ABFD25000022	2017-06-24	00:01:42	-5,00 €
*					

Fuente: Elaboración Propia

Como se aprecia en la figura 30, los datos enviados por el lector han seguido todos los procesos anteriormente mencionados antes de llegar a la base de datos haciendo uso de diferentes tipos de softwares, para poder llevar a cabo la propuesta inicial. Por último, para realizar el monitoreo y conteo de los vehículos que utilicen el peaje se han creado dos usuarios uno llamado operador y el otro administrador, los cuales estarán facultados y limitados en la administración de la base de datos como se explicó con anterioridad. Tendrán acceso a estos datos y lo podrán hacer en una plataforma más amigable que desde la base de datos y se logra hacer con la ayuda pgAdmin PHP en la cual introducir los datos de usuario y clave nos permitirá ver de manera remota y en tiempo real el flujo del uso del peaje como se muestra en la figura 31 y figura 32 respectivamente.

Figura 31 Inicio de Sección al Administrador de base de datos



Fuente: Elaboración Propia

Figura 32 Monitoreo en Tiempo Real el Uso del Chillón

Acciones	norden	codpeaje	fecha	hora	descuento
Editar Eliminar	576	AB0027004934ABFD25000012	2017-06-22	02:44:56	-5,00 €
Editar Eliminar	577	AB0027004934ABFD25000012	2017-06-22	02:45:58	-5,00 €
Editar Eliminar	578	AB0027004934ABFD25000013	2017-06-22	02:46:44	-5,00 €
Editar Eliminar	579	AB0027004934ABFD25000015	2017-06-22	02:46:47	-5,00 €
Editar Eliminar	580	AB0027004934ABFD25000016	2017-06-22	02:46:49	-5,00 €
Editar Eliminar	581	AB0027004934ABFD25000017	2017-06-22	02:46:52	-5,00 €
Editar Eliminar	582	AB0027004934ABFD25000021	2017-06-22	02:46:57	-5,00 €
Editar Eliminar	583	AB0027004934ABFD25000010	2017-06-23	23:15:22	-5,00 €
Editar Eliminar	584	AB0027004934ABFD25000013	2017-06-24	00:01:29	-5,00 €
Editar Eliminar	585	AB0027004934ABFD25000015	2017-06-24	00:01:31	-5,00 €
Editar Eliminar	586	AB0027004934ABFD25000016	2017-06-24	00:01:32	-5,00 €
Editar Eliminar	587	AB0027004934ABFD25000017	2017-06-24	00:01:36	-5,00 €
Editar Eliminar	588	AB0027004934ABFD25000021	2017-06-24	00:01:40	-5,00 €
Editar Eliminar	589	AB0027004934ABFD25000022	2017-06-24	00:01:42	-5,00 €

Fuente: Elaboración Propia

La figura 33, que fue obtenida de la página web oficial de la Empresa Rutas de Lima, concesionaria encargada de la administración del peaje Chillón que se ubica en el distrito de Puente Piedra, kilómetro 25 de la Panamericana Norte, muestra la cruda realidad que sufren miles de conductores que diariamente hacen uso de esta vía, extensas colas, módulos sin atención y sobre todo gran cantidad de congestión vehicular y la contaminación que los vehículos producen, esta imagen es raíz de nuestra propuesta de mejorar el sistema actual del peaje como se muestra en la figura 34, el peaje Chillón sin congestión y sin contaminación.

Figura 33 Situación Actual del Peaje Chillón



Fuente: (Rutas de Lima, 2017)

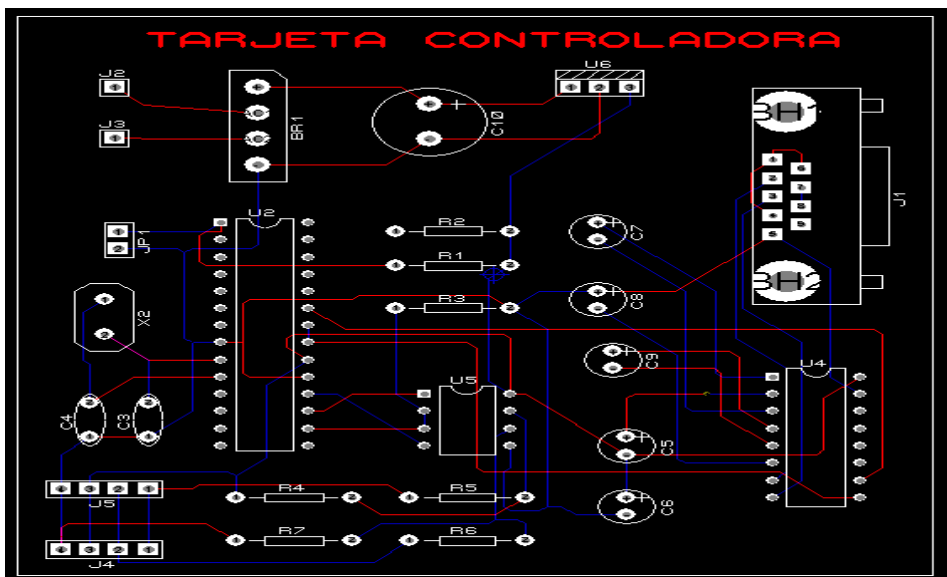
**Figura 34 Peaje Chillón Automatizado sin
Congestión vehicular**



Fuente: Elaboración Propia

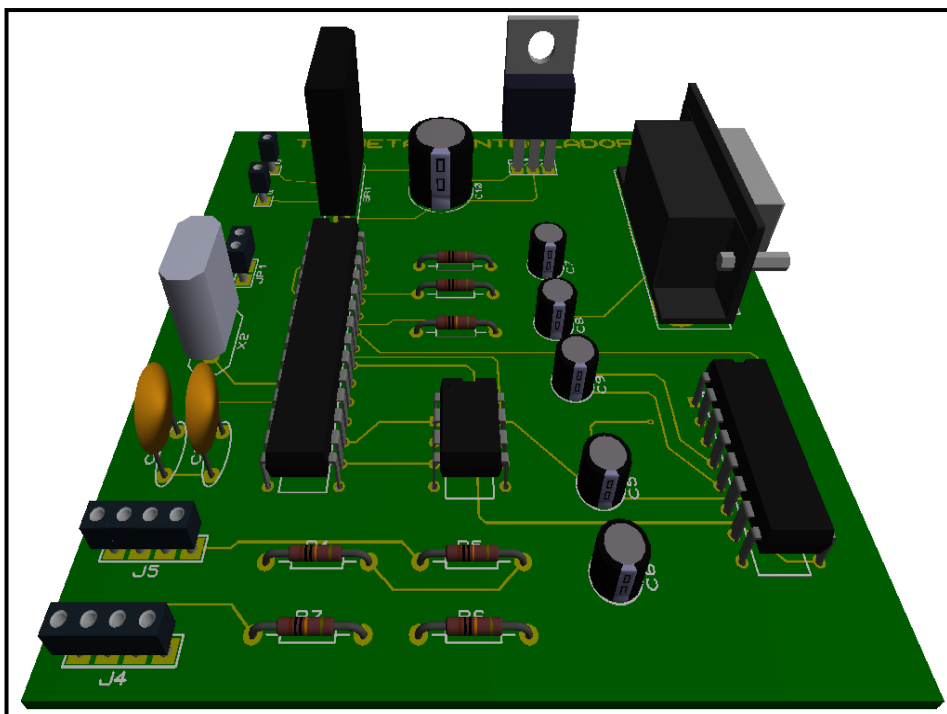
3.5.8 Construcción de los PCB para el Proyecto

Figura 35 Construcción de la Tarjeta Controladora



Fuente: Elaboración Propia

Figura 36 Simulación en 3D de la Tarjeta Controladora



Fuente: Elaboración Propia

3.5.9 Presupuesto

Para la elaboración de este proyecto se están tomando los precios de la página web Stronglink.com en lo que respecta al sistema RFID. Asimismo los otros materiales han sido cotizados en diferentes tiendas de electrónica, para poner en marcha el sistema se debe contar con un computadora o laptop que se usara de servidor para el almacenamiento de nuestra base de datos y la ejecución del sistema, en lo que respecta a la lectura de los vehículos que transitan por el peaje, para ello se requiere 5 lectores RFID-SL130 una para cada módulo, las etiquetas RFID se vende en grandes cantidades y para poder iniciar con el proyecto se está cotizando en la adquisición de 5 mil unidades; los otros dispositivos y su costo se detallan en la tabla 7.

Tabla 7 Presupuesto para el Proyecto

DISPOSITIVO	CANT	PRECIO * 1	SUBTOTAL	IGV	TOTAL
Laptop Toshiba i3	1	S/. 1.800,00	S/. 1.476,00	S/. 324,00	S/. 1.800,00
Ckt de Comunicación	1	S/. 1.000,00	S/. 820,00	S/. 180,00	S/. 1.000,00
Sist. Base de Datos	1	S/. 800,00	S/. 656,00	S/. 144,00	S/. 800,00
Lector RFID	5	S/. 1.900,00	S/. 7.790,00	S/. 1.710,00	S/. 9.500,00
Etiquetas RFID	5000	S/. 0,30	S/. 1.230,00	S/. 270,00	S/. 1.500,00
Instalacion	1	S/. 2.500,00	S/. 2.050,00	S/. 450,00	S/. 2.500,00
TOTAL			S/. 14.022,00	S/. 3.078,00	S/. 17.100,00

Fuente: Elaboración Propia

3.6 CONCLUSIONES

El sistema de comunicación que se elaboró en este proyecto, funcionó acorde a las expectativas, ya que, al coger información obtenido por el lector ubicada en el vehículo, y adjuntarla a un archivo de registro en la base de datos y con ella generar el cobro virtual por el uso de la vía, nos permite la automatización del proceso de cobro, mediante el uso de diferentes softwares de desarrollo. Se generó el acceso por internet y, monitorear en tiempo real, los diferentes sucesos evitándose la congestión vehicular

El desarrollo de este proyecto permitirá mejorar la cantidad de vehículos atendidos por minuto, incluso en las horas de mayor afluencia, generándose la satisfacción en el usuario, reduciendo el tiempo de respuesta ante el cliente o usuario de 90 segundos a tan solo 10 segundos. Gracias a la integración del sistema de comunicación que permite obtener, de manera instantánea, la información del vehículo.

La utilización de la tecnología RFID, permitirá que nuestro sistema actual de cobro del peaje sea mucho más eficiente, y tenga un mayor control del cobro, permitiendo mejorar las condiciones de trabajo y la seguridad evitándose así la manipulación de dinero en efectivo, debido a que el sistema de cobro se realizara de manera virtual.

3.7 RECOMENDACIONES

El proyecto presentado se encuentra dentro de sus primeras etapas con respecto a la recopilación de datos, anexándola directamente en un registro de información que se puede enriquecer con la ayuda de compañeros de ingeniería de Sistemas los cuales dominan con mayor facilidad la transformación, recopilación de los datos y del manejo de grandes desarrolladores de bases de datos con ello se puede generar diferentes aplicaciones para la mejorar este proyecto e inclusive otros, realizando un diseño más productivo.

Respecto a la tecnología RFID (identificación por radio frecuencia) hoy en día en nuestro país este sistema no se encuentra ampliamente desarrollado. Se hace la recomendación a la Universidad Alas Peruanas, de gestionar comunicación con ingenieros de diferentes universidades dentro o fuera de nuestro país con la finalidad de firmar convenios de capacitación a los alumnos y profesores con la única finalidad de que el alumno domine ampliamente las nuevas tecnologías que son el inicio de la modernización de nuestro país, ya que este proyecto, con la ayuda necesaria se puede ampliar para poder generar una red de peaje haciendo que nuestros sistemas sean más óptimos y productivos y estar al alcance de comparar nuestros avances con los de otros países hermanos en el ámbito regional como en el caso de Chile y Argentina donde se está aplicando esta y otras tecnologías.

CAPÍTULO IV
REFERENCIAS BIBLIOGRÁFICAS

4.1 LIBROS

1. Atilano, I. R. (2007). *Identificación de Vehículos empleando Radio Frecuencia(Rfid-Epc)*. Mexico.
2. Benites, S. (2015). *Bienvenidos a Osc Telecoms SAC*. Descripción de la empresa Osc, Lima.
3. Delgadillo, R. S., & Ortiz, C. J. (2011). *Diseño de un sistema de control de acceso mediante tecnología rfid con implementación de un servidor web embebido en un pic*". TESIS, Zacatecas - Mexico. Recuperado el 08 de 02 de 2017.
4. Echavarrí, J. P. (2008). *Cuaderno de Investigación Urbanística Nº 58*. Madrid: Instituto Juan Herrera.
5. Osc Telecoms SAC. (20 de Abril de 2015). Introducción. *Reglamento Interno de Trabajo Osc Telecoms SAC*. Lima, Lima, Perú.

4.2 PAGINAS WEB

1. Stronglink. (2017). *Stronglink Technology Co*. Recuperado el 07 de 01 de 2017, de <http://www.stronglink-rfid.com/es/>
2. alibaba.com. (febrero de 2016). *alibaba.com*. Recuperado el 04 de 2017, de <https://spanish.alibaba.com/product-detail/stable-reading-rfid-monza-4-library-sticker-60578963849.html>
3. Gestión Verde. (19 de Diciembre de 2015). Recuperado el 04 de 15 de 2017, de <http://www.gestionclubcannabis.es/blog/tarjetas-contacto-rfid-identificacion-socio>
4. Haro, A. D. (Junio de 2012). *OSC Telecoms Security Solutions*. Recuperado el 15 de 03 de 2017, de <http://www.osctelecoms.com/novedades.php>
5. Herrera, J. M., & Casanovas, E. S. (s.f.). *Estudio, Diseño y Simulación de un sistema de RFID basado EPC*. Trabajo de Investigación . Recuperado el 10 de 04 de 2017, de <chrome-extension://oemmndcbldboiebfnladdacbfdmada>

dm/https://upcommons.upc.edu/bitstream/handle/2099.1/3552/40883-2.pdf

6. Nextiafenix. (04 de 2017). *Electronica Tecnologia y Más Nextia Fenix*. Recuperado el 03 de 04 de 2017, de <http://www.nextiafenix.com/producto/pic18f4550/>
7. Osc Telecoms. (2011). *Osc Telecoms & security Solutions*. Recuperado el 12 de 03 de 2017, de <http://www.osctelecoms.com/telecomunicaciones.php>
8. Osc Telecoms. (2011). *Osc Telecoms & Security Solutions*. Obtenido de <http://www.osctelecoms.com/seguridadAutomatizacion.php>
9. OSC Telecoms. (2011). *Osc Telecoms & Security Solutions*. Recuperado el 10 de 03 de 2017, de <http://www.osctelecoms.com/quienesSomos.php>.
10. Rutas de Lima. (2017). *Rutas de Lima*. Recuperado el 12 de 04 de 2017, de <http://rutasdelima.pe/servicios/tarifa>
11. SEDA. (2015). *Seda*. Recuperado el 05 de 02 de 2017, de http://190.24.150.73/seda/seda_wp_00/?page_id=1216
12. Smiley, S. (18 de Julio de 2014). *Rfid Insider Tracking the Rfid Industry*. Recuperado el 08 de 04 de 2017, de <http://blog.atlasrfidstore.com/monostatic-vs-bistatic-rfid>
13. Sotelo, T. G. (2006). *Emulacion de RFID activo mediante la plataforma MICAz*. Barcelona - España: Universitat Politecnica de Catalunya.
14. Strog Link. (2017). *Strog Link*. Recuperado el 04 de 03 de 2017, de <http://www.stronglink-rfid.com/es/rfid-labels/SLU-F-Series.html>
15. Stroling link. (2017). *Lector UHF de Audio Jack SL120*. Recuperado el 10 de 04 de 2017, de Lector UHF de Audio Jack SL120: <http://www.stronglink-rfid.com/es/rfid-readers/sl120.html>
16. Strong Link. (2017). *SLU-F Series*. Recuperado el 10 de 04 de 2017, de SLU-F Series: <http://www.stronglink-rfid.com/es/rfid-labels/SLU-F-Series.html>
17. Stronglink Technology Co. (2017). *Módulo RFID UHF RT400*. Recuperado el 17 de 03 de 2017, de <http://www.stronglink-rfid.com/es/rfid-modules/rt400.html>
18. Stronglink Technology Co. (2017). *Stronglink*. Recuperado el 06 de 01 de 2017, de <http://www.stronglink-rfid.com/es/rfid-readers/sl130.html>

19. wikipedia. (20 de 03 de 2014). *Wikipedia la Enciclopedia Libre*. Recuperado el 06 de 02 de 2017, de <https://es.wikipedia.org/wiki/MPLAB>
20. Wikipedia. (05 de 06 de 2017). *wikipedia la Enciclopedia Libre*. Recuperado el 10 de 03 de 2017, de <https://es.wikipedia.org/wiki/Python>
21. Wikipedia. (16 de Abril de 2017). *Wikipedia la Enciclopedia Libre*. Recuperado el 05 de 04 de 2017, de <https://es.wikipedia.org/wiki/RFID>
22. Wikipedia. (14 de 03 de 2017). *Wikipedia la Enciclopedia Libre*. Recuperado el 05 de 01 de 2017, de https://es.wikipedia.org/wiki/Proteus_Design_Suite
23. Wikipedia. (13 de 05 de 2017). *Wikipedia la Enciclopedia Libre*. Recuperado el 04 de 03 de 2017, de <https://es.wikipedia.org/wiki/PostgreSQL>

CAPÍTULO V
GLOSARIO DE TÉRMINOS

- 2G** Se conoce como telefonía móvil 2G a la segunda generación de telefonía móvil. La telefonía móvil 2G no es un estándar ni un tipo de protocolo, sino que es una forma de marcar el cambio de protocolos de telefonía móvil analógica a digital.
- 3G** Es la abreviación de tercera generación de transmisión de voz y datos a través de la telefonía móvil mediante UMTS que es el servicio Universal de telecomunicaciones móviles.
- 4G** En telecomunicaciones, 4G es la sigla utilizada para referirse a la 4^{TA} generación de tecnologías de telefonía móvil. Es la sucesora de las tecnologías 2G y 3G, y precede a la próxima generación, la 5G.
- 802.15.4** Es un estándar que define el nivel físico y el control de acceso al medio de redes inalámbricas de área personal con tasas bajas de transmisión de datos
- ANDROID** Android es un sistema operativo basado en el núcleo Linux el que fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tabletas y también para relojes Smart televisores y automóviles.
- CCTV** Circuito cerrado de televisión es una tecnología de video vigilancia diseñada para supervisar una diversidad de ambientes y actividades.
- EPC** Utiliza una cadena de números para identificar fabricante, producto y un número de serie exclusivo para cada unidad de artículo. Esta cadena de números se graba en el chip de la etiqueta RFID.
- JACK** El conector de audio (audio Jack en inglés) de señales analógicas se utiliza para conectar micrófonos, auriculares y otros sistemas de señal analógica a dispositivos electrónicos, aunque sobre todo audio.

- OSC** Es la abreviación Optimo, Seguro y Confiable frase que representa el trabajo de la empresa OSC Telecoms & SEGURIDAD SAS es una empresa con 15 años de experiencia en el mercado de la seguridad electrónica y se dedica a soluciones telecomunicaciones.
- PEAJE** Se denomina peaje al pago que se efectúa como derecho para poder circular por un camino. En la antigüedad, se llamaba portazgo a la suma que debía pagarse para cruzar cierto límite (puerta) entre dos zonas territoriales o por cruzar un puente
- RF** Es la abreviación de Radiofrecuencia, también denominado espectro de radiofrecuencia, es un término que se aplica a la porción menos energética del espectro electromagnético, situada entre el rango de los 3 kilohercios (KHz) y 300 gigahercios (GHz).
- RFID** Tecnología de identificación por radio frecuencia, que utiliza ondas de radio para codificar y almacenar datos en transponders.
- TAG** Las etiquetas RFID son la forma más común y habitual son delgadas y autoadhesivas, con la capacidad de poder ser impresas con códigos legible y con las capacidades de memoria que dependerán del circuito integrado que lleve incorporado.
- ZIGBEE** Es el nombre de la especificación de un conjunto de protocolos de alto nivel de comunicación inalámbrica para su utilización con radio_ _difusión digital de bajo consumo, basada en el estándar de redes inalámbricas de área personal en el estándar IEEE 802.15.4

CAPÍTULO VI

ÍNDICES

6.1 ÍNDICE GENERAL DE FIGURAS

Figura 1 Expansión de Osc Telecoms en América	3
Figura 2 Logo de la Empresa OSC Telecoms y su Significado.....	3
Figura 3 Organigrama de OSC Telecoms SAC	7
Figura 4 Peaje Chillón y su Congestión Vehicular	12
Figura 5 Sistema Actual de Cobro Manual.....	13
Figura 6 Funcionamiento del Sistema RFID	18
Figura 7 Lector RFID e Interacción con las etiquetas RFID.....	19
Figura 8 Arquitectura de la Etiqueta RFID	20
Figura 9 Etiqueta Pasiva SLU-F Series Strong Link	21
Figura 10 Arquitectura del Proyecto.....	24
Figura 11 Esquema de Adecuación de los Equipos RFID en el Peaje....	25
Figura 12 Aplicativo de Configuración del Lector RFID- LS130.....	34
Figura 13 Simulación de Lectura de la Etiqueta SLU-F	37
Figura 14 Verificación del Grabado del Código en la Etiqueta.....	38
Figura 15 Conexión Microcontrolador 18F2550 con Lector RFID.....	40
Figura 16 Maquina de Estado de la comunicación..	40
Figura 17 Trama de Comunicación entre Lector RFID con Microcontrolador.	42
Figura 18 Construcción de Algoritmo de Comunicación	42
Figura 19 Construcción de los Estados del sistema	43
Figura 20 Contenido de la Tabla Peaje.....	44
Figura 21 Contenido de la Tabla Registro.....	45
Figura 22 Contenido de la Tabla Recarga	46
Figura 23 tabla de Vehículos de 2 Ejes	47
Figura 24 Base de Datos para el Peaje Chillón	48
Figura 25 Usuarios de Administración de la Base Datos	49
Figura 26 Redireccionamiento de Datos la PostgreSQL.....	50
Figura 27 Circuito de Simulación del Proyecto	51
Figura 28 Conexión Serial Virtual	52
Figura 29 Recepción de Datos por Python	53
Figura 30 Verificación del Registro de Código en la Base de Datos.....	54
Figura 31 Inicio de Sección al Administrador de base de datos.....	55
Figura 32 Monitoreo en Tiempo Real el Uso del Chillón.....	55

Figura 33 Situación Actual del Peaje Chillón	56
Figura 34 Peaje Chillón Automatizado - sin Congestión vehicular.....	57
Figura 35 Construcción de la Tarjeta Controladora	58
Figura 36 Simulación en 3D de la Tarjeta Controladora	58

6.2 ÍNDICE GENERAL DE TABLAS

Tabla 1 Frecuencia de Operación en RFID.....	22
Tabla 2 Datos de Operación del Módulo RFID – SL130	26
Tabla 3 Datos de Operación de la Antena RFID.....	27
Tabla 4 datos de Operación de la Etiqueta RFID - SLU-F	28
Tabla 5 Datos de Operación del Pic-18F2550	29
Tabla 6 Clasificación de los Vehículos y Asignación de Código RFID.....	35
Tabla 7 Presupuesto para el Proyecto.....	59

CAPÍTULO VII

ANEXOS

7.1 ANEXO 1: PROGRAMACIÓN DEL MICROCONTROLADOR

```

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

/*
 * CONFIGURACION DEL MICROCONTROLADOR
 */
#include <xc.h>
#define _XTAL_FREQ 20000000
#define delay_ms(x)    __delay_ms(x)
#define delay_us(x)    __delay_us(x)
#define TRUE  1
#define FALSE 0
typedef union
{ unsigned int reg16;
  struct{
    unsigned char reg8l;
    unsigned char reg8h; }; }
reg16to8;
typedef union
{ unsigned char reg8;
  struct{ unsigned reg4l:4;
    unsigned reg4h:4; }; }
reg8to4;
typedef union
{ unsigned char reg8;
  struct{ unsigned reg6l:6;
    unsigned reg2h:2; }; }
reg8to2_6;
/* Configuracion de los parametros de operacion del MCU */
#pragma config PLLDIV = 1
#pragma config CPUDIV = OSC1_PLL2
#pragma config USBDIV = 1
#pragma config FOSC = HS
#pragma config FCMEN = OFF
#pragma config IESO = OFF
#pragma config PWRT = ON
#pragma config BOR = OFF
#pragma config BORV = 3
#pragma config VREGEN = OFF
#pragma config WDT = OFF
#pragma config WDTPS = 32768
#pragma config CCP2MX = OFF
#pragma config PBADEN = OFF
#pragma config LPT1OSC = OFF
#pragma config MCLRE = ON
#pragma config STVREN = OFF
#pragma config LVP = OFF

```

```

#pragma config XINST = OFF
#pragma config CP0 = OFF
#pragma config CP1 = OFF
#pragma config CP2 = OFF
#pragma config CP3 = OFF
#pragma config CPB = OFF
#pragma config CPD = OFF
#pragma config WRT0 = OFF
#pragma config WRT1 = OFF
#pragma config WRT2 = OFF
#pragma config WRT3 = OFF
#pragma config WRTC = OFF
#pragma config WRTB = OFF
#pragma config WRTD = OFF
#pragma config EBTR0 = OFF
#pragma config EBTR1 = OFF
#pragma config EBTR2 = OFF
#pragma config EBTR3 = OFF
#pragma config EBTRB = OFF
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
/*
 * TIMER.H
 */
#include "global.h"
void vlnitTIMER0 ()
{ T0CS = 0;
  T0SE = 1;
  PSA = 0;
  PS0 = 1;
  PS1 = 1;
  PS2 = 1; }
void delay_10ms ()
{ T0IF = 0;
  TMR0 = 60;
  while (!T0IF); }
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
/*
 * RS485.H
 */
#include <xc.h>
#include "global.h"
#define RT TRISBbits.RB3
#define ENABLE PORTBbits.RB3
reg16to8 SPBRGX;
void vlnit_RS485(unsigned int BAUD)
{ RT = 0; // Enable Pin
  ENABLE = 0;

```

```

TRISCBits.TRISC6=0; /* Pin TX */
TRISCBits.TRISC7=1; /* Pin RX */
SPBRGX.reg16 = ((_XTAL_FREQ/4)/BAUD) - 1;
SPBRG = SPBRGX.reg8l;
SPBRGH = SPBRGX.reg8h;
/*Registro TXSTA*/
CSRC=0;
TX9=0;
TXEN=0;
TXEN=1;
SYNC=0;
SENDB=0; /*Parametro en pruebas*/
BRGH=1;
/*Registro RCSTA*/
SPEN=1;
RX9=0;
CREN=1;
/*Registro BAUDCON*/
RXDTP=0; /*Parametro en pruebas 0 <> invertido y 1 <> no invertido*/
TXCKP=0; /*Parametro en pruebas 0 <> invertido y 1 <> no invertido*/
BRG16=1;
WUE=0;
ABDEN=0;
RCIE=1;
TXIE=0;      }
void vPutc_RS485(unsigned char Byte) // Writes a character to the serial port
{ ENABLE=1;
  delay_us(25);
  while(!TXIF); // wait for previous transmission to finish
  TXREG = Byte;
  delay_us(200);
  ENABLE=0;    }
unsigned char uGetc_RS485(void) // Reads a character from the serial port
{ if(OERR) // If over run error, then reset the receiver
  { CREN = 0;
    CREN = 1;  }
  while(!RCIF); // Wait for transmission to receive
  return RCREG;    }
void vPuts_RS485(const unsigned char* st)
{ while(*st){ vPutc_RS485(*st++);    } }
void vDisable_ISR()
{ GIE = 0; // Deshabilita Interrupciones
  PEIE = 0;
  IPEN = 0; }
void vEnable_ISR()
{ GIE = 1; // Habilitacion Interrupciones
  PEIE = 1;
  IPEN = 1;
  RCIP = 0; // Interrupcion de Baja Prioridad    }

```

```

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
/*
 * Protocolo Maestro Esclavo - pme
 */
#include <xc.h>
#include "global.h"
#define ADDR_MASTER 'a' //Direccion del Dispositivo Maestro
#define ADDR_SLAVE 'b' //Direccion del Dispositivo Esclavo
#define FIN 'c' //Fin de la Transmision
#define PRX 24 //Longitud del registro de RX
#define PTX 24 //Longitud del registro de TX
#define LDP 14
#define CR 0x80
#define SR 0x40
unsigned char DATOSTX[PTX];
unsigned char DATOSRX[PRX];
unsigned char DATOSP[LDP];
unsigned char estado=0, N=0, bandera=0;
reg8to4 DIM;
reg8to2_6 DIRF;
void vReadPacket_PME(unsigned char dato)
{ switch(estado)
  { case 0: //Estado de Espera
    if(dato==ADDR_SLAVE)
    { estado=1; N=0; }
    else
    { estado=0; }
    break;
    case 1: //Estado de Recepcion
    if(dato==FIN & N>(PRX - 1))
    { estado=0; N=0;
      bandera=1; }
    else
    { DATOSRX[N]=dato;
      N++; }
    break;
    default: break; } }
//DirL : Direccion Logica (puerto)
//Long : Longitud de los Datos
//Mues : Numeros de muestras del Dato
//dts: Datos 24 Bytes
void vSendPacket_PME(unsigned char DirL, char Long, char Mues, char *dts)
{ int i;
  DIRF.reg8=(SR+DirL);
  DIM.reg4l=Mues;
  DIM.reg4h=Long;
  DATOSTX[0]=DIRF.reg8;
  DATOSTX[1]=DIM.reg8;

```



```

    UART_RX_PIN = 1;           // Input
    UART_TX_PIN = 0;          // Output }
unsigned char uGetc_SW_RS485(void)
{
    unsigned char DataValue = 0;
    //wait for start bit
    while(UART_RX==1);
    delay_us(OneBitDelay);
    delay_us(OneBitDelay/2); // Take sample value in the mid of bit duration
    for ( unsigned char i = 0; i < DataBitCount; i++ )
    {
        if ( UART_RX == 1 ) //if received bit is high
        {
            DataValue += (1<<i);
        }
        delay_us(OneBitDelay);
    }
    // Check for stop bit
    if ( UART_RX == 1 ) //Stop bit should be high
    {
        delay_us(OneBitDelay/2);
        return DataValue;
    }
    else //some error occurred !
    {
        delay_us(OneBitDelay/2);
        return 0x00;
    }
}
void vPutc_SW_RS485(const char DataValue)
{
    ENABLE=1;
    delay_us(50);
    // Send Start Bit
    UART_TX = 0;
    delay_us(OneBitDelay);
    for ( unsigned char i = 0; i < DataBitCount; i++ )
    {
        //Set Data pin according to the DataValue
        if( ((DataValue>>i)&0x1) == 0x1 ) //if Bit is high
        {
            UART_TX = 1;
        }
        else //if Bit is low
        {
            UART_TX = 0;
        }
        delay_us(OneBitDelay);
    }
    //Send Stop Bit
    UART_TX = 1;
    delay_us(OneBitDelay);
    delay_us(200);
    ENABLE=0;
}
void vPuts_SW_RS485(const unsigned char* st)
{
    while(*st){ vPutc_SW_RS485(*st++); }
}
void vEnable_ISR_SW()
{
    //nRBPU=1;
    INTEDG0=0; // ISR RB0 flanco de Bajada
    GIE=1; // ISR Global
    INTOIE=1; // ISR RB0
}
void vDisable_ISR_SW()
{
    INTOIE=0;
    //INTE=0; // ISR RB0
}

```

```

////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////
#include <xc.h>
#include "global.h"
#define Baudrate      19200 // bps
#define DataBitCount  8    // no parity, no flow control
#define UART_RX      PORTBbits.RB0 // UART RX pin
#define UART_TX      PORTBbits.RB3 // UART TX pin
#define UART_RX_PIN  TRISBbits.TRISB0 // UART RX pin direction
register
#define UART_TX_PIN  TRISBbits.TRISB3 // UART TX pin direction
register
/* Duracion del Bit USART en us*/
#if Baudrate <= 4800
#define OneBitDelay    (1000000/Baudrate)
#elif Baudrate == 9600
#define OneBitDelay    95 // [F=20Mhz,100]; [F=10Mhz,90]
#elif Baudrate == 19200
#define OneBitDelay    46 // [F=20Mhz,45]; [F=10Mhz,x]
#elif Baudrate == 38400
#define OneBitDelay    18 // [F=20Mhz,18]; [F=10Mhz,x]
#else
#define OneBitDelay    101
#endif
void vInitSW_UART(void) // Initialize UART pins to proper values
{
    UART_TX = 1; // TX pin is high in idle state
    UART_RX_PIN = 1; // Input
    UART_TX_PIN = 0; // Output }
unsigned char uGetc_SW(void)
{
    unsigned char DataValue = 0;
    //wait for start bit
    while(UART_RX==1);
    delay_us(OneBitDelay);
    delay_us(OneBitDelay/2); // Take sample value in the mid of bit duration
    for ( unsigned char i = 0; i < DataBitCount; i++)
    {
        if ( UART_RX == 1 ) //if received bit is high
        {
            DataValue += (1<<i);
            delay_us(OneBitDelay);
        }
    }
    // Check for stop bit
    if ( UART_RX == 1 ) //Stop bit should be high
    {
        delay_us(OneBitDelay/2);
        return DataValue;
    }
    else //some error occurred !
    {
        delay_us(OneBitDelay/2);
        return 0x00;
    }
}
void vPutc_SW(const char DataValue)
{
    // Send Start Bit
    UART_TX = 0;
    delay_us(OneBitDelay);
}

```

```

for ( unsigned char i = 0; i < DataBitCount; i++ )
{
    //Set Data pin according to the DataValue
    if( ((DataValue>>i)&0x1) == 0x1 ) //if Bit is high
    {   UART_TX = 1;   }
    else //if Bit is low
    {   UART_TX = 0;   }
    delay_us(OneBitDelay); }
//Send Stop Bit
UART_TX = 1;
delay_us(OneBitDelay); }
void vPuts_SW(const unsigned char* st)
{ while(*st){ vPutc_SW(*st++); } }
void vEnable_ISR_SW()
{ //nRBPU=1;
  INTEDG0=0; // ISR RB0 flanco de Bajada
  GIE=1; // ISR Global
  INTOIE=1; // ISR RB0 }
void vDisable_ISR_SW()
{ INTOIE=0;
  //INTE=0; // ISR RB0
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
/*
* CONSTRUCCION DE LA TRAMA Y ESTADOS
*/
#ifndef PROTOCOLOER_H
#define PROTOCOLOER_H
# define ADDR_MASTER 'a' //DIRECCION DE MAESTRO
# define ADDR_SLAVE 'b' //DIRECCION DE ESCLAVO
# define FIN 'f' //DIRECCION DE FIN DE TRAMA
# define TOUT 500 //TIEMPO DE VERIFICACION DE TRAMA
# define PBRX 24 //DATOS DE LA TRAMA EN TRANSMISION
# define PBTX 24 //DATOS DE LA TRAMA EN RECEPCION
char DATO_RX[PBRX];
char DATO_TX[PBTX];
int estado = 0 ; //ESTADO INICIAL
int timeout = 0; //TIEMPO CORTE EN VERIFICACION DE TRAMA
char fun; //
int N = 0; //
int bandera = 0;
void vprotocolo(char dato)
{ switch(estado)
  {   case 0:
      if( dato==ADDR_MASTER)
      {estado= 1;
       N=0;}
      else
      {estado= 0;
       bandera= 0;}
  }
}

```



```

        break;
        case 1:
        if( dato== FIN & N> (PBRX -1 ))
        {estado= 0;
        N=0;
        bandera = 1; }
        else if (timeout >= TOUT){ estado = 0 ; }
        else {DATO_RX[N]=dato;
        N++;
        vlnittimer(); } } }
void trama ( unsigned char addr , unsigned char fin )
{ putcUSART(addr);
  putcUSART (DATO_TX[0]);
  putcUSART (DATO_TX[1]);
  putcUSART (DATO_TX[2]);
  putcUSART (DATO_TX[3]);
  putcUSART (DATO_TX[4]);
  putcUSART (DATO_TX[5]);
  putcUSART (DATO_TX[6]);
  putcUSART (DATO_TX[7]);
  putcUSART (DATO_TX[8]);
  putcUSART (DATO_TX[9]);
  putcUSART (DATO_TX[10]);
  putcUSART (DATO_TX[11]);
  putcUSART (DATO_TX[12]);
  putcUSART (DATO_TX[13]);
  putcUSART (DATO_TX[14]);
  putcUSART (DATO_TX[15]);
  putcUSART (DATO_TX[16]);
  putcUSART (DATO_TX[17]);
  putcUSART (DATO_TX[18]);
  putcUSART (DATO_TX[19]);
  putcUSART (DATO_TX[20]);
  putcUSART (DATO_TX[21]);
  putcUSART (DATO_TX[22]);
  putcUSART (DATO_TX[23]);
  putcUSART (DATO_TX[24]);
  putcUSART (fin); }
void vSetdatos (unsigned char cmd, unsigned char dato1, unsigned char
dato2, unsigned char dato3, )
{ DATO_TX[0]= cmd;
  DATO_TX[1]= dato1;
  DATO_TX[2]= dato2;
  DATO_TX[3]= dato3;
  DATO_TX[4]= dato4;
  DATO_TX[5]= dato5;
  DATO_TX[6]= dato6;
  DATO_TX[7]= dato7;
  DATO_TX[8]= dato8;

```



```

if(bandera==1)
{ DIRF.reg8=DATOSRX[0];
  DIM.reg8=DATOSRX[1];
  if(DIRF.reg2h==1) // Solicitud sin Confirmacion == 1
  { switch(DIRF.reg6l) // Funciones 1 - 63
    { case 1:
      vApp_01();
      break;
      case 2:
      PORTBbits.RB1=~PORTBbits.RB1;
      break;
      case 3:
      vApp_01();
      break;
      default: break; } }
  else if(DIRF.reg2h==2) // Solicitud con Confirmacion == 2
  { switch(DIRF.reg6l) // Funciones 1 - 63
    { case 1:
      vApp_01();
      break;
      case 2:
      PORTBbits.RB1=~PORTBbits.RB1;
      break;
      case 3:
      vApp_01();
      break;
      default: break;
    } }
  bandera=0;
} } }

void interrupt vISR_HIGH(void)
{ if(INT0IF) // Interrupcion RX UART
  { fun=uGetc_SW_RS485();
    vPutc_SW_RS485(fun);
    vReadPacket_PME(fun);
    INT0IF = 0; } }

void interrupt low_priority vISR_LOW(void)
{ if(PIR1bits.RCIF==1)
  { cmd = uGetc();
    vPutc(cmd);
    PIR1bits.RCIF=0; } }
if(PIR1bits.RCIF) // Si Ocurre una interrupcion UART RX
{ if(OERR)
  { CREN = 0;
    CREN = 1; } }
cmd=uGetc();
vPutc(cmd);
vReadPacket_PME(cmd);
PIR1bits.RCIF==0; }

```

V7.2 ANEXO 2: PRESENTACIÓN DE DIAPOSITIVAS DEL PROYECTO



UAP | **UNIVERSIDAD
ALAS PERUANAS**

FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

TRABAJO DE SUFICIENCIA PROFESIONAL

DISEÑO DE UN SISTEMA DE COBRO AUTOMATIZADO EN EL PEAJE CHILLÓN, PARA
MEJORAR EL TRÁFICO VEHICULAR EN LA PANAMERICANA NORTE, UTILIZANDO
TECNOLOGÍA RFID

PRESENTADO POR EL BACHILLER
ESPINOZA REYES, CLEVER ENRIQUE

PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO ELECTRÓNICO Y
TELECOMUNICACIONES

LIMA - PERÚ - 2017

CONTENIDO

- OSC TELECOMS SAC
- ACTIVIDADES DE OSC TELECOMS SAC
- OSC TELECOMS (MISION, VISION, OBJETIVO)
- ORGANIGRAMA DE OSC TELECOMS SAC
- INTRODUCCIÓN
- DESCRIPCIÓN DE LA REALIDAD PROBLEMÁTICA
- ANÁLISIS DEL PROBLEMA
- OBJETIVOS DEL PROYECTO
- PROPUESTA DEL PROYECTO
- ARQUITECTURA DE LA PROPUESTA DISPOSITIVOS PARA EL PROYECTO
- CONFIGURACIÓN DEL LECTOR RFID
- CONTRUCCIÓN DEL CÓDIGO ID PARA LA ETIQUETA RFID
- LECTURA DE ETIQUETA RFID
- FUNCIONAMIENTO DE LA COMUNICACIÓN ENTRE LECTOR RFID – PIC 18F2550
- CONSTRUCCIÓN DEL ALGORITMO DE LA MAQUINA DE ESTADO
- CONSTRUCCIÓN DE LA BASE DE DATOS
- ENLACE ENTRE PYTHON Y POSTGRESQL
- SIMULACIÓN DEL PROYECTO
- SITUACIÓN DEL PEAJE CHILLÓN
- PRESUPUESTO DEL PROYECTO
- CONCLUSIONES
- RECOMENDACIONES

ACTIVIDADES DE OSC TELECOMS

Es una empresa con 20 años de experiencia en el mercado de las telecomunicaciones y se dedica a soluciones de seguridad electrónica, ofrece servicios de implementación de redes móviles y fijas.

EN EL ÁREA DE TELECOMUNICACIONES

- Consultores de telecomunicaciones.
- Optimización de RF 2G-3G.
- Enlaces microondas.
- Alquiler de equipos de telecomunicaciones.



EN EL ÁREA DE SEGURIDAD Y AUTOMATIZACIÓN

- Servicios de CCTV.
- Servicio de Control de Activos.
- Servicio de Control de Acceso.



ACTIVIDADES DE OSC TELECOMS

Es una empresa con 15 años de experiencia en el mercado de las telecomunicaciones y se dedica a soluciones de seguridad electrónica, ofrece servicios de implementación de redes móviles y fijas.

EN EL ÁREA DE TELECOMUNICACIONES

- ▶ Consultores de telecomunicaciones.
- ▶ Optimización de RF 2G-3G.
- ▶ Enlaces microondas.
- ▶ Alquiler de equipos de telecomunicaciones.



EN EL ÁREA DE SEGURIDAD Y AUTOMATIZACIÓN

- ▶ Servicios de CCTV.
- ▶ Servicio de Control de Activos.
- ▶ Servicio de Control de Acceso.





OSC TELECOMS S.A.C

- ▶ 1.3.1 MISIÓN

Garantizar la plena satisfacción de sus clientes, colaboradores, comunidad y demás partes interesadas, mediante la optimización de los procesos y la mejora continua en las actividades y acciones organizacionales.

- ▶ 1.3.2 VISIÓN

Ser una de las primeras empresas de servicio en telecomunicaciones a nivel regional y el TOP of mind en nuestros clientes dentro del rubro.

- ▶ 1.3.3 OBJETIVO

Como objetivo principal dedicarse a las actividades de ingeniería y actividades conexas de consultoría técnica, entre otras actividades.

ORGANIGRAMA DE OSC TELECOMS SAC



INTRODUCCIÓN

- En los últimos años las autoridades en nuestro país han visto formas de como poder solucionar la congestión vehicular en Lima y sus alrededores; pero, sus acciones no han llegado a tener mayor relevancia. La construcción de nuevas vías de acceso y la realización de obras civiles de ampliación de vías por todos los alrededores de Lima están ocasionando una congestión mayor en nuestro capital, lo que hace que la ciudad pierda atractivo turístico y que los gases de efecto invernadero tengan una mayor concentración en esta congestión, ocasionando que nuestra ciudad pierda grandes sumas de dinero diariamente.
- En nuestro país, uno de los principales corredores viales es la Autopista Panamericana Norte. En el kilómetro 25 se encuentra el peaje Chillón ubicado en el distrito de Puente Piedra, que conecta los distritos de Ancón, Puente Piedra y los procedentes del norte de nuestro país con los distritos de Independencia, San Martín de Porres, el centro de Lima, sus alrededores y la salida al sur de nuestro país. En este corredor vial circulan a diario unos 75 mil vehículos.

INTRODUCCION

- En los módulos del peaje Chillón, se forman largas colas, causando la congestión vehicular y extensas colas de autos. Para solucionar este inconveniente, en otros países se viene implementado mejoras en los peajes, con una red de sensores que se basan en la tecnología inalámbrica zigbee, que propone mejorar el sistema de control de peaje vehicular, automatizando el proceso, evitando así que los vehículos tengan que detenerse, aligerando así la congestión vehicular. Un sistema similar se propone para el peaje Chillón del kilómetro 25 de la Autopista Panamericana Norte.



DESCRIPCIÓN DE LA REALIDAD PROBLEMÁTICA

- ▶ En nuestro país, una de las principales problemáticas, después de la inseguridad ciudadana, es la extensa cantidad de vehículos, que en los últimos años se ha incrementado de manera incontrolada; pero, esto no sería un problema si el sistema vial de nuestro país estuviese preparado para esta demanda de vehículos.
- ▶ Donde principalmente apreciamos esta problemática son en los peajes, donde las colas de autos no tienen donde acabar ocasionado una gran congestión vehicular, debido a que se utiliza un sistema manual de cobro, por el cual la persona al volante debe detener su vehículo para realizar el pago por el uso de la vía, pudiéndose apreciar mejor la congestión en las horas de mayor afluencia; pero, no sería así, si es que hubiese una mayor cantidad de módulos de cobro, eso significaría realizar obras civiles, ampliando la cantidad de módulos, pero esto no sería una solución rentable.



ANÁLISIS DEL PROBLEMA

- ▶ En nuestro país, diariamente crece nuestro parque automotor pues acceder a la compra de un automóvil cada vez se hace más accesible, por nuevos sistema de evaluación crediticia de los bancos, influyendo así al congestionamiento diario en nuestro país, ya que las personas buscan una forma más directa y cómoda de poder trasladarse dentro de la ciudad, mejorando su calidad de vida y estatus social, pero esto no se cumple debido a que no contamos con infraestructura vial capaz de soportar la demanda sin causar congestión vehicular, que cada vez se ve más en aumento debido a las obras de ampliaciones, remodelaciones que se dan en nuestra ciudad.

- El problema principal no es el uso del peaje, sino el proceso manual de cobro con el que funciona actualmente, generado por el déficit que se tiene para poder atender



PROCESO NORMAL

AGRAVANTES



OBJETIVOS DEL PROYECTO

OBJETIVO GENERAL

- ▶ El objetivo del proyecto es mejorar y optimizar el sistema de cobro del peaje Chillón, de la autopista panamericana norte kilómetro 25, mediante la automatización de este proceso, haciendo que el cobro se realice de manera virtual, evitándose así formar largas colas en el peaje y mejorando la fluidez de los vehículos.

OBJETIVOS ESPECÍFICOS

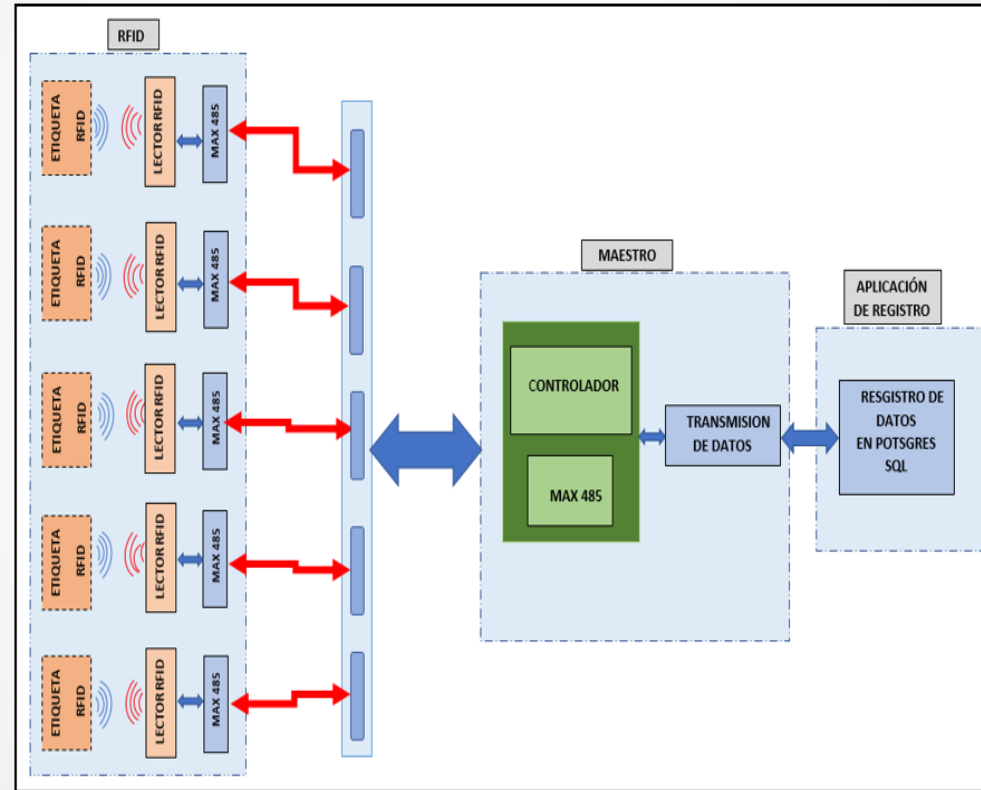
- ▶ Mejorar la calidad y cantidad de atención de los módulos en el peaje Chillón, al reducir los tiempos muertos donde el usuario busca el dinero para pagar o el trabajador demora en buscar el vuelto.
- ▶ Mejorar las condiciones de trabajo y la seguridad del personal, en el peaje Chillón de la autopista panamericana norte kilómetro 25, encargado de realizar el cobro por el uso de la vía,

PROPUESTA DEL PROYECTO

- Diseñar un Control de Peaje Automatizado con la Tecnología de Identificación por Radiofrecuencia, que consistirá en un sistema que tendrá como prioridad aligerar la carga vehicular en el peaje ubicado en el distrito de Puente Piedra, kilómetro 25 de la Autopista Panamericana Norte, en los carriles con dirección de norte a sur.



ARQUITECTURA DE LA PROPUESTA



DISPOSITIVOS PARA EL PROYECTO

TIPO	DESCRIPCION
Frecuencia	920MHz-925MHz
Etiqueta apoyo	ISO18000-6B, ISO18000-6C, EPC Class1 Gen2 tags
Interfaz	RS232 / RS485 / RJ45 / Wiegand
Temperatura de funcionamiento	-10°C ~ +55°C
Distancia de la operación	Leer distancia de> 8m, escribir distancia> 1 m (depende de las etiquetas)
Potencia de salida	20-30dbm
Velocidad de lectura	32bits/6ms
Velocidad de escritura	32bits/50ms
Peso	6.4kg
Dimensión	600 x 480 x 110 mm



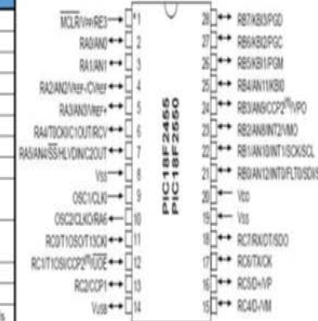
TIPO	DESCRIPCION
Frecuencia	920-925MHz
Frecuencia Central	922.5MHz
Escribe	Antena de Cerámica
V S W R	1.5: 1
Ancho de banda (V.S.W.R<1.9:1)	>5MHz
Impedancia	50ohm
Ganancia	2dBic
Polarización	RHCP
Longitud de cable RF	100mm
Conectores RF	IPX-male
Dimensión	50 x 50 x 7 mm
Temperatura de Funcionamiento	-20 ~ +70°C
Temperatura de Almacenamiento	-40 ~ +85°C



TIPO	DESCRIPCION
Frecuencia	840 ~ 960MHz
Protocolo	ISO18000-6C, EPC Class1 Gen2
Tamaño EEPROM	EPC 128bits, User 32-512bits
Material	Material de sabotaje Frágil
Temperatura	-25°C ~ +75°C
Modo de Funcionamiento	Pasivo
Distancia de la operación	3 ~ 10 (m)
Dimensión	30x40x0.2(mm)



TIPO	DESCRIPCION
Frecuencia Operación	DC - 48MHZ
Program Memory (Bytes)	32768
Program Memory (Instruct)	16384
Data Memory(Bytes)	2048
Data EEPROM Memory (Bytes)	256
Interrupt Sources	19
I/O Ports	Ports A,B,C
Timers	4
Capture /compare / PWM Modules	2
Enhanced Capture /compare / PWM Modules	0
Universal Serial Bus (USB)	1
Streaming Parallel port (SPP)	NO
10-Bit analog-to-Digital module	10 Input Channels
Comparators	2



CONFIGURACIÓN DEL LECTOR RFID

The screenshot displays the 'Reader1000Demo V7.3' software interface. The window title bar includes 'Connect Reader', 'ReaderParameter', and three test options: 'ISO18000-6C Test', 'ISO18000-6B Test', and 'Iso18000-6D Test'. The interface is organized into several sections:

- Work Mode:** Includes radio buttons for 'Continuing', 'Trigger', and 'Command'. It also features 'Hard Version' and 'Soft Version' input fields and an 'Update Parameter' button.
- Basic Parameter:** Contains dropdown menus for 'BaudRate of RS232/RS485 Port', 'Min. Frequency of Carrier(MHz)', and 'Max. Frequency of Carrier(MHz)'. It also has input fields for 'RF Power Output(0-160)', 'Modulation Depth(0-160)', 'Address of Reader(1-254)', and 'Max. Tags of once Reading(1~100)'. Checkboxes for 'RS485 Network' and 'Enable Buzzer' are also present.
- Parameters for Continuing and Trigger Mode:** This section includes 'Read from which Antenna' (ANT1-ANT4), 'Port and Format for Output' (Wiegand26, Wiegand34, RS485, RS232), and 'Start Address of ID:'. It also has 'Wiegand Port Format' (Pulse Width and Pulse Interval) and 'Manner of Output ID' (Direct, Standard).
- Which type of tag to be read:** Radio buttons for 'ISO18000-6B', 'EPCC1G2', and 'TK300'. A 'What be read from tag' section offers 'ID of the Tag' and 'SN Selected by User'.
- Information:** A text input field for additional data.
- Parameters only for TK300 tag:** Includes 'Duration of Reading' and 'Number of Pages' dropdowns.
- Set Relay:** A checkbox for 'Relay?' and a 'Set Relay' button.
- Buttons:** 'Default Parameter' and 'Set Factory Parameter' buttons are located at the bottom right.

CONSTRUCCIÓN DEL CÓDIGO ID PARA LA ETIQUETA RFID

ETIQUETA ASIGNADA	TIPO DE VEHÍCULO	SUBTOTAL	IGV	TOTAL
AA0027004934ACDE25000000	VEHÍCULOS LIVIANO PRIVADO	S/. 4,24	S/. 0,76	S/. 5,00
AB0027004934ABFD25000000	VEHÍCULOS LIVIANO PUBLICO	S/. 3,82	S/. 0,68	S/. 4,50
B005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 02 EJES	S/. 7,63	S/. 1,37	S/. 9,00
C005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 03 EJES	S/. 11,45	S/. 2,05	S/. 13,50
D005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 04 EJES	S/. 15,26	S/. 2,74	S/. 18,00
E005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 05 EJES	S/. 19,08	S/. 3,42	S/. 22,50
F005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 06 EJES	S/. 22,90	S/. 4,10	S/. 27,00
G005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 07 EJES	S/. 26,71	S/. 4,79	S/. 31,50
H005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 08 EJES	S/. 30,53	S/. 5,47	S/. 36,00
K005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 09 EJES	S/. 34,34	S/. 6,16	S/. 40,50
L005FB63AC1F3841EC880000	VEHÍCULOS PESADOS - 10 EJES	S/. 38,16	S/. 6,84	S/. 45,00

LECTURA DE ETIQUETA RFID

Reader1000Demo V7.3 SIMULACION RFID PANAM

Connect Reader | ReaderParameter ISO18000-6C Test | ISO18000-6B Test | ISO18000-6D Test

Select Antenna for Test
 ANT1 ANT2 ANT3 ANT4

Read Interval

List Selected Tag
 Select Memory Bank
 Password EPC TID

Address of Tag Data(bit):
 Length of Tag Data(bit):
 Tag Data(HEX):

Kill Tag
 Select a Tag:
 Kill Password (BHEX):

Set protect for reading or writing
 Select Memory Bank
 Password EPC TID User

Lock of Password
 Password
 Kill Password Access Password
 Writeable and readable from any state
 Permanently readable and writeable
 Readable and writeable from the secured state
 Never readable and writeable

Lock of EPC TID and User Bank
 Writeable from any state
 Permanently writeable
 Writeable from the secured state
 Never writeable
 Access Password(BHEX):

Read and Write Data Block
 Select a Tag:
 Address of Tag Data(WORD):
 Length of Tag Data(WORD):
 Access Password(BHEX):
 Written Data(HEX):

Lock Block for User
 Select a Tag:
 Address of Tag Data(WORD):
 Access Password (BHEX):

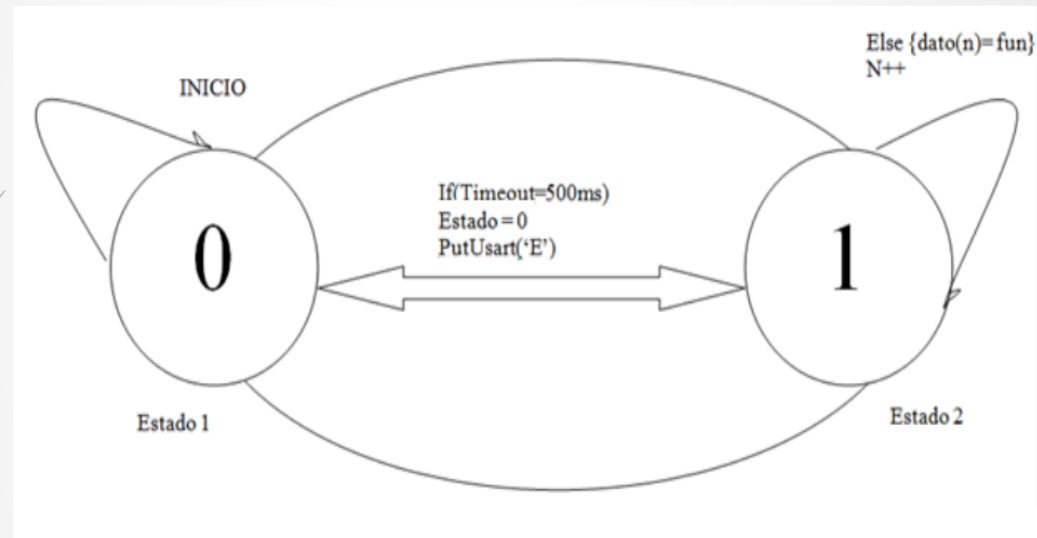
Alarm
 Select a Tag:
 Alarm State
 No Alarm Alarm
 Access Password(BHEX):

Calibrate
 Access Password(BHEX):

No.	ID	Success	Times	EPC Length
01	B005FB65AC1F3841EC8B0000		5	40ms

Read Protect

FUNCIONAMIENTO DE LA COMUNICACIÓN ENTRE LECTOR RFID – PIC 18F2550



INICIO	DIRECCION	DATOS	FIN
2 bytes	2 bytes	n bytes.....	3 bytes
50	01.....05	A005FB63AC1F3841EC880000	255

CONSTRUCCIÓN DEL ALGORITMO DE LA MAQUINA DE ESTADO

```

#ifndef PROTOCOLOER_H
#define PROTOCOLOER_H
#define ADDR_MASTER '10' // DIRECCION DE MAESTRO
#define ADDR_SLAVE1 '1' //DIRECCION DE 1º ESCLAVO
#define ADDR_SLAVE2 '2' //DIRECCION DE 2º ESCLAVO
#define ADDR_SLAVE3 '3' //DIRECCION DE 3º ESCLAVO
#define ADDR_SLAVE4 '4' //DIRECCION DE 4º ESCLAVO
#define ADDR_SLAVES '5' //DIRECCION DE 5º ESCLAVO
#define ADDR '50' //DIRECCION DE INICIO DE TRAMA
#define FIN '255' //DIRECCION DE FIN DE TRAMA
#define TOUT 500 //TIEMPO DE VERIFICACION DE TRAMA
#define PBRX 24 //DATOS DE LA TRAMA EN TRANSMISION
#define PBTX 24 //DATOS DE LA TRAMA EN RECEPCION
#define FIN '#' //DIRECCION DE FIN DE TRAMA
#define TOUT 500 //TIEMPO DE VERIFICACION DE TRAMA
#define PBRX 16 //DATOS DE LA TRAMA EN TRANSMISION
#define PBTX 16 //DATOS DE LA TRAMA EN RECEPCION

char DATO_RX[PBRX];
char DATO_TX[PBTX];
int estado = 0; // ESTADO INICIAL
int timeout = 0; // TIEMPO DE CORTE EN VERIFICACION DE LA TRAMA
char fun; // FUNCION
int N = 0;
int bandera = 0;

void vprotocolo(char dato)
{
    switch(estado)
    {
        case 0:
            if( dato==ADDR_MASTER)
            {estado= 1;
              N=0;}
            else
            {estado= 0;
              bandera= 0;}
            break;
        case 1:
            if( dato== FIN & N> (PBRX -1 ))
            {estado= 0;
              N=0;
              bandera = 1; }
            else if (timeout >= TOUT){ estado = 0 ; }
            else {DATO_RX[N]=dato;
              N++;
              vinittimer();
            }
    }
}

```


CONSTRUCCIÓN DE LA BASE DE DATOS

Nombre de base de datos

Información de la base de datos

Tablas Creadas para el Sistema RFID

Panel de Comandos SQL

Usuarios de la base de datos

Propiedad	Valor
Nombre	RFID
OID	16384
Propietario	postgres
ACL	
Tablespace	pg_default
Espacio de tabla por defecto	pg_default
Codificado	UTF8
Colación	Spanish_Spain.1252
Tipo caracter	Spanish_Spain.1252
Schema por defecto	
Tabla ACL por defecto	
Secuencia ACL por defecto	
Función ACL por defecto	
Tipo ACL por defecto	
Permitir conexiones?	Si
¿Conectado?	Si
Límite de conexión	-1
¿Base de datos del sistema?	No
Comentario	

```

-- Database: "RFID"
-- DROP DATABASE "RFID";

CREATE DATABASE "RFID"
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
LC_COLLATE = 'Spanish_Spain.1252'
LC_CTYPE = 'Spanish_Spain.1252'
CONNECTION LIMII = -1;

```

Recuperando detalles en base de datos RFID... Hecho.

ENLACE ENTRE PYTHON Y POSTGRESQL

```
*RFID5.py - C:\Python27\SIMULACION RFID\RFID5.py (2.7.13)*
File Edit Format Run Options Window Help
Python 2.7.13 (v2.7.13:a06454blafal, Dec 17 2016, 20:42:59)
[MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.

# Puerta de enlace GWRfid y Base de Datos PGSQL
import serial
import psycopg2
import time

con = None

try:
    gwrfid=serial.Serial('COM20')
    con = psycopg2.connect(host='localhost', database='RFID',
                          user='postgres', password='123456789')
    cur = con.cursor()
    cur.execute("SELECT version()")
    ver = cur.fetchone()
    print ver

    while(True):
        try:
            datos=gwrfid.readline()
            idrfid=datos[0:26]
            fecha=time.strftime("%Y/%m/%d")
            hora=time.strftime("%H:%M:%S")
            print datos[0:17], fecha , hora
            #print "insert into peaje (codpeaje, fecha, hora, descuento)
            values"+"("+"'+idrfid+'"+", "+"'+fecha+'"+", "+"'+hora+'"+", "+"-
            cur.execute("insert into peaje (codpeaje, fecha, hora, descuento)
            values"+"("+"'+idrfid+'"+", "+"'+fecha+'"+", "+"'+hora+'"+", "+"-
            con.commit()

        except (Exception, psycopg2.DatabaseError) as error:
            print(error)

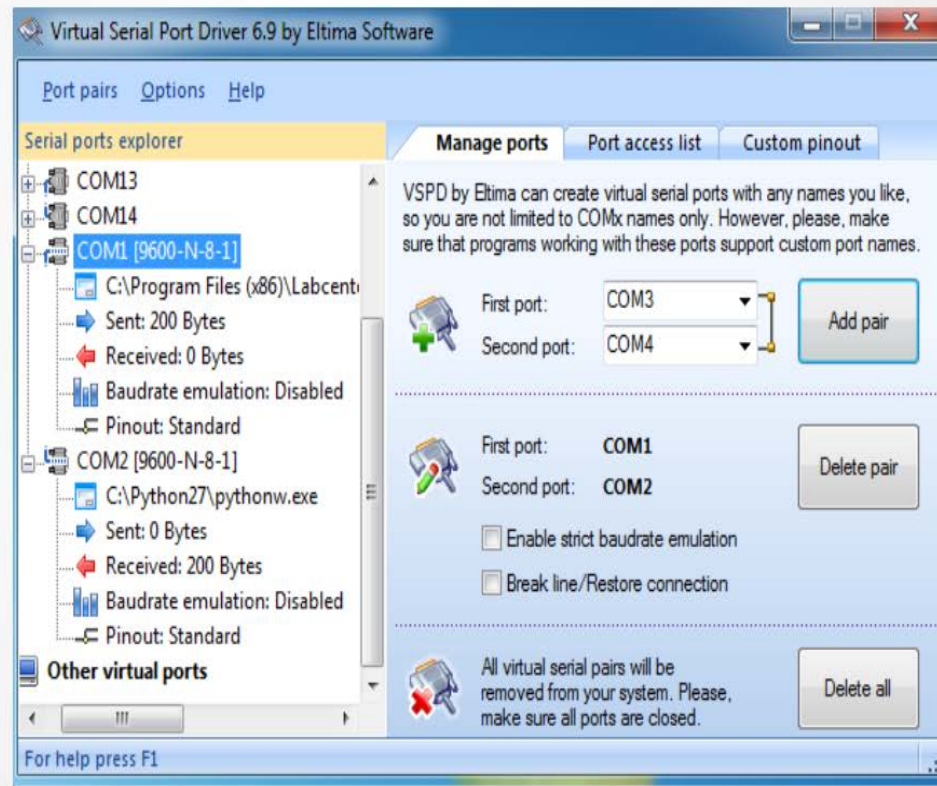
        except: #Excepcion por interrupciones Teclado (CTRL+C)
            print "Servicio Detenido"
            break
    gwrfid.close()
except:
    pass
```

Ln: 41 Col: 15

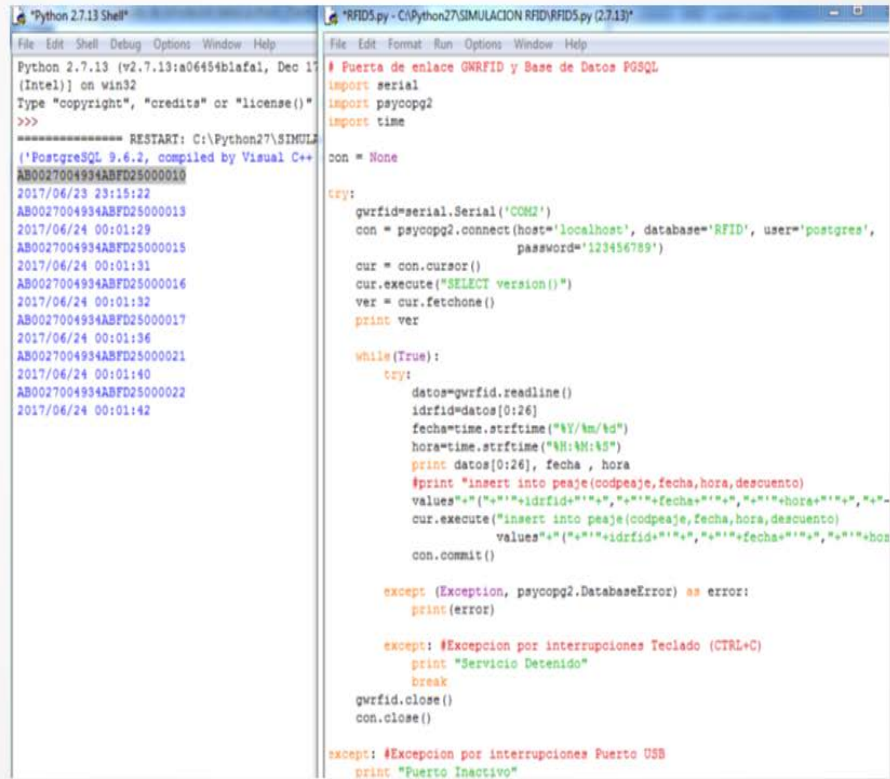
SIMULACIÓN DEL PROYECTO



SIMULACIÓN DEL PROYECTO



SIMULACIÓN DEL PROYECTO



```

Python 2.7.13 Shell
Python 2.7.13 [v2.7.13:a06454b1af1, Dec 17 2014] on win32
Type "copyright", "credits" or "license()"
>>>
===== RESTART: C:\Python27\SIMULACION RFID\RFID5.py
('PostgreSQL 9.6.2, compiled by Visual C++
AB0027004934ABFD25000010
2017/06/23 23:15:22
AB0027004934ABFD25000013
2017/06/24 00:01:29
AB0027004934ABFD25000015
2017/06/24 00:01:31
AB0027004934ABFD25000016
2017/06/24 00:01:32
AB0027004934ABFD25000017
2017/06/24 00:01:36
AB0027004934ABFD25000021
2017/06/24 00:01:40
AB0027004934ABFD25000022
2017/06/24 00:01:42

RFID5.py - C:\Python27\SIMULACION RFID\RFID5.py (2.7.13)
# Puerta de enlace GWRfid y Base de Datos PGSQL
import serial
import psycopg2
import time

con = None

try:
    gwrfid=serial.Serial('COM2')
    con = psycopg2.connect(host='localhost', database='RFID', user='postgres',
        password='123456789')
    cur = con.cursor()
    cur.execute("SELECT version()")
    ver = cur.fetchone()
    print ver

    while(True):
        try:
            datos=gwrfid.readline()
            idrfid=datos[0:26]
            fecha=time.strftime("%Y/%m/%d")
            hora=time.strftime("%H:%M:%S")
            print datos[0:26], fecha , hora
            #print "insert into peaje (codpeaje, fecha, hora, descuento)
            values"+"("+"*"+idrfid+"*"+", "+"*"+fecha+"*"+", "+"*"+hora+"*"+", "+"*"+
            cur.execute("insert into peaje (codpeaje, fecha, hora, descuento)
            values"+"("+"*"+idrfid+"*"+", "+"*"+fecha+"*"+", "+"*"+hora
            con.commit()

        except (Exception, psycopg2.DatabaseError) as error:
            print(error)

        except: #Excepcion por interrupciones Teclado (CTRL+C)
            print "Servicio Detenido"
            break

    gwrfid.close()
    con.close()

except: #Excepcion por interrupciones Puerto USB
    print "Puerto Inactivo"
  
```

SIMULACIÓN DEL PROYECTO

The screenshot displays the pgAdmin III interface. On the left, the 'Explorador de Objetos' (Object Explorer) shows the database structure for 'POSTGRES-WAPP (localhost:5432)'. The 'Bases de Datos (2)' folder is expanded to show 'RFID', which contains 'Catálogos (2)', 'Eventos disparadores (0)', 'Extensiones (1)', and 'Esquemas (1)'. The 'public' schema is expanded to show various objects including 'Clasificados (0)', 'Dominios (0)', 'Configuraciones FTS', 'Diccionarios FTS (0)', 'Analizadores FTS (0)', 'Plantillas FTS (0)', 'Funciones (0)', 'Secuencias (1)', 'Tablas (14)', 'Funciones Disparador', 'Vistas (0)', and 'Répica Stony (0)'. The 'postgres' user is also visible under 'Roles de Login (3)'. The 'Propiedades' (Properties) pane shows the 'Tabla' (Table) and 'Propietario' (Owner) for the 'peaje' table, which is owned by 'postgres'. The 'Estadísticas' (Statistics) pane is also visible.

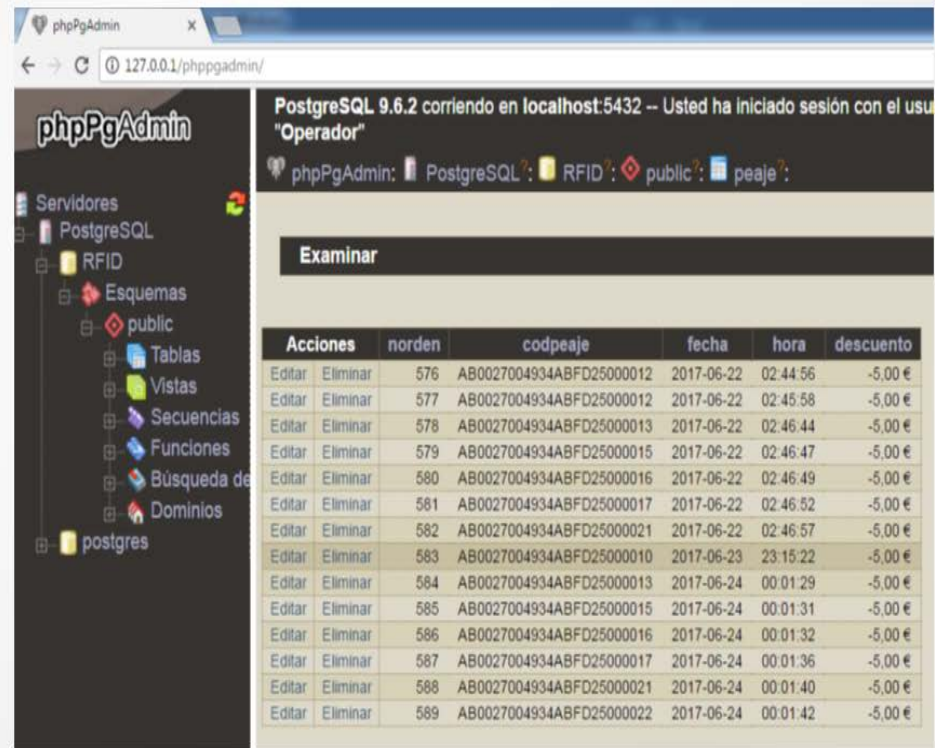
On the right, the 'Editar Datos - POSTGRES-WAPP (localhost:5432) - RFID - public.peaje' window shows a data table with the following columns: 'norden [PK] se', 'codpeaje character varying(40)', 'fecha date', 'hora time without', and 'descuento money'. The table contains 14 rows of data, with the 8th row highlighted.

	norden [PK] se	codpeaje character varying(40)	fecha date	hora time without	descuento money
1	576	AB0027004934ABFD25000012	2017-06-22	02:44:56	-5,00 €
2	577	AB0027004934ABFD25000012	2017-06-22	02:45:58	-5,00 €
3	578	AB0027004934ABFD25000013	2017-06-22	02:46:44	-5,00 €
4	579	AB0027004934ABFD25000015	2017-06-22	02:46:47	-5,00 €
5	580	AB0027004934ABFD25000016	2017-06-22	02:46:49	-5,00 €
6	581	AB0027004934ABFD25000017	2017-06-22	02:46:52	-5,00 €
7	582	AB0027004934ABFD25000021	2017-06-22	02:46:57	-5,00 €
8	583	AB0027004934ABFD25000010	2017-06-23	23:15:22	-5,00 €
9	584	AB0027004934ABFD25000013	2017-06-24	00:01:29	-5,00 €
10	585	AB0027004934ABFD25000015	2017-06-24	00:01:31	-5,00 €
11	586	AB0027004934ABFD25000016	2017-06-24	00:01:32	-5,00 €
12	587	AB0027004934ABFD25000017	2017-06-24	00:01:36	-5,00 €
13	588	AB0027004934ABFD25000021	2017-06-24	00:01:40	-5,00 €
14	589	AB0027004934ABFD25000022	2017-06-24	00:01:42	-5,00 €
*					

The 'Panel SQL' window shows the following SQL commands:

```
-- Table: public.peaje
-- DROP TABLE public.peaje
CREATE TABLE public.peaje
```

SIMULACIÓN DEL PROYECTO



The screenshot shows the phpPgAdmin web interface. The browser address bar displays "127.0.0.1/phpPgAdmin/". The interface title is "PostgreSQL 9.6.2 corriendo en localhost:5432 -- Usted ha iniciado sesión con el usuario 'Operador'". The left sidebar shows a tree view of the database structure: Servidores > PostgreSQL > RFID > Esquemas > public > Tablas, Vistas, Secuencias, Funciones, Búsqueda de..., Dominios. The main content area is titled "Examinar" and displays a table of data.

Acciones	norden	codpeaje	fecha	hora	descuento
Editar Eliminar	576	AB0027004934ABFD25000012	2017-06-22	02:44:56	-5,00 €
Editar Eliminar	577	AB0027004934ABFD25000012	2017-06-22	02:45:58	-5,00 €
Editar Eliminar	578	AB0027004934ABFD25000013	2017-06-22	02:46:44	-5,00 €
Editar Eliminar	579	AB0027004934ABFD25000015	2017-06-22	02:46:47	-5,00 €
Editar Eliminar	580	AB0027004934ABFD25000016	2017-06-22	02:46:49	-5,00 €
Editar Eliminar	581	AB0027004934ABFD25000017	2017-06-22	02:46:52	-5,00 €
Editar Eliminar	582	AB0027004934ABFD25000021	2017-06-22	02:46:57	-5,00 €
Editar Eliminar	583	AB0027004934ABFD25000010	2017-06-23	23:15:22	-5,00 €
Editar Eliminar	584	AB0027004934ABFD25000013	2017-06-24	00:01:29	-5,00 €
Editar Eliminar	585	AB0027004934ABFD25000015	2017-06-24	00:01:31	-5,00 €
Editar Eliminar	586	AB0027004934ABFD25000016	2017-06-24	00:01:32	-5,00 €
Editar Eliminar	587	AB0027004934ABFD25000017	2017-06-24	00:01:36	-5,00 €
Editar Eliminar	588	AB0027004934ABFD25000021	2017-06-24	00:01:40	-5,00 €
Editar Eliminar	589	AB0027004934ABFD25000022	2017-06-24	00:01:42	-5,00 €

SITUACIÓN DEL PEAJE CHILLÓN

AHORA



PROPUESTA



PRESUPUESTO DEL PROYECTO

- Para la elaboración de este proyecto se están tomando los precios de la página web Stronglink.com en lo que respecta al sistema RFID. Asimismo los otros materiales han sido cotizados en diferentes tiendas de electrónica, para poner en marcha el sistema se debe contar con un computadora o laptop que se usara de servidor para el almacenamiento de nuestra base de datos y la ejecución del sistema, en lo que respecta a la lectura de los vehículos que transitan por el peaje, para ello se requiere 5 lectores RFID-SL130 una para cada módulo, las etiquetas RFID se vende en grandes cantidades y para poder iniciar con el proyecto se está cotizando en la adquisición de 5 mil unidades.

DISPOSITIVO	CANT	PRECIO * 1	SUBTOTAL	IGV	TOTAL
Laptop Toshiba i3	1	S/. 1.800,00	S/. 1.476,00	S/. 324,00	S/. 1.800,00
Ckt de Comunicación	1	S/. 1.000,00	S/. 820,00	S/. 180,00	S/. 1.000,00
Sist. Base de Datos	1	S/. 800,00	S/. 656,00	S/. 144,00	S/. 800,00
Lector RFID	5	S/. 1.900,00	S/. 7.790,00	S/. 1.710,00	S/. 9.500,00
Etiquetas RFID	5000	S/. 0,30	S/. 1.230,00	S/. 270,00	S/. 1.500,00
Instalacion	1	S/. 2.500,00	S/. 2.050,00	S/. 450,00	S/. 2.500,00
TOTAL			S/. 14.022,00	S/. 3.078,00	S/. 17.100,00

CONCLUSIONES

- El sistema de comunicación que se elaboró en este proyecto, funcionó acorde a las expectativas, ya que, al coger información obtenido por el lector ubicada en el vehículo, y adjuntarla a un archivo de registro en la base de datos y con ella generar el cobro virtual por el uso de la vía, nos permite la automatización del proceso de cobro, mediante el uso de diferentes softwares de desarrollo. Se generó el acceso por internet y, monitorear en tiempo real, los diferentes sucesos evitándose la congestión vehicular
- El desarrollo de este proyecto permitirá mejorar la cantidad de vehículos atendidos por minuto, incluso en las horas de mayor afluencia, generándose la satisfacción en el usuario, reduciendo el tiempo de respuesta ante el cliente o usuario de 90 segundos a tan solo 10 segundos. Gracias a la integración del sistema de comunicación que permite obtener, de manera instantánea, la información del vehículo.
- La utilización de la tecnología RFID, permitirá que nuestro sistema actual de cobro del peaje sea mucho más eficiente, y tenga un mayor control del cobro, permitiendo mejorar las condiciones de trabajo y la seguridad evitándose así la manipulación de dinero en efectivo, debido a que el sistema de cobro se realizara de manera virtual.

RECOMENDACIONES

- El proyecto presentado se encuentra dentro de sus primeras etapas con respecto a la recopilación de datos, anexándola directamente en un registro de información que se puede enriquecer con la ayuda de compañeros de ingeniería de Sistemas los cuales dominan con mayor facilidad la transformación, recopilación de los datos y del manejo de grandes desarrolladores de bases de datos con ello se puede generar diferentes aplicaciones para la mejorar este proyecto e inclusive otros, realizando un diseño más productivo.
- Respecto a la tecnología RFID (identificación por radio frecuencia) hoy en día en nuestro país este sistema no se encuentra ampliamente desarrollado. Se hace la recomendación a la Universidad Alas Peruanas, de gestionar comunicación con ingenieros de diferentes universidades dentro o fuera de nuestro país con la finalidad de firmar convenios de capacitación a los alumnos y profesores con la única finalidad de que el alumno domine ampliamente las nuevas tecnologías que son el inicio de la modernización de nuestro país, ya que este proyecto, con la ayuda necesaria se puede ampliar para poder generar una red de peaje haciendo que nuestros sistemas sean más óptimos y productivos y estar al alcance de comparar nuestros avances con los de otros países hermanos en el ámbito regional como en el caso de Chile y Argentina donde se está aplicando esta y otras tecnologías.