UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

HERRAMIENTAS DE RADIO DEFINIDO POR SOFTWARE PARA LA EMULACIÓN DE PROTOCOLO DE RFID

POR
ING. MARIPAZ MORENO DIAZ

EN OPCIÓN AL GRADO DE MAESTRÍA EN CIENCIAS DE LA INGENIERÍA ELÉCTRICA

ENERO, 2019

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO

HERRAMIENTAS DE RADIO DEFINIDO POR SOFTWARE PARA LA EMULACIÓN DE PROTOCOLO DE RFID

POR

ING. MARIPAZ MORENO DIAZ

EN OPCIÓN AL GRADO DE MAESTRÍA EN CIENCIAS DE LA INGENIERÍA ELÉCTRICA

SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN, MÉXICO **ENERO, 2019**

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO

Los miembros del Comité de Tesis recomendamos que la Tesis Herramientas de radio definido por software para la emulación de protocolo de RFID] realizada por el alumno(a) Ing. Maripaz Moreno Diaz ,con número de matrícula 1501004 , sea aceptada para su defensa como opción al grado de Maestria en Ciencias de la Ingenieria Eléctrica

El Comité de Tesis

Dr. José Ramón Rodríguez Cruz
Director

Dr. José Antonio de la O
Revisor

Dr. Neale R. Smith
Revisor

Vo. Bo.

Dr. Simón Martínez Martínez
Subdirector de Estudios de Posgrado

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO
FIME

San Nicolás de los Garza, Nuevo León, 28 de enero de 2019

Índice General

Índice de cuadros

Índice de figuras

Capítulo 1

Introducción

Los sistemas RFID son empleados desde hace algunos años en aplicaciones que varían desde inventarios en almacenes, hasta sistemas para el cuidado de la salud.

También se han propuesto sistemas RFID en aplicaciones que presentan nuevos retos, por ejemplo en aplicaciones que involucran etiquetas en movimiento [1–3].

El protocolo ISO 18000-6C da las pautas para la comunicación entre la lectora y las etiquetas, y brinda opciones para hacer frente a las dificultades del sistema de comunicaciones inalámbrico. Sin embargo, en las lectoras comerciales no se encuentran disponibles todas estas opciones, y los investigadores deben de conformarse con el desempeño del sistema ante las limitadas configuraciones de fábrica [4]. Además, en las lectores comerciales disponibles no es posible realizar cambios en la capa física y MAC lo que dificulta la implementación de nuevas propuestas [5].

Ante esta situación, en este trabajo se desarrolla una herramienta para la emulación del protocolo ISO 18000-6C basada en tecnología de radio definido por software, y que puede ser usada con equipo de *hardware* de *National Instruments* como USRP y PXI. El programa con la lógica de esta herramienta fue desarrollado con el lenguaje de programación LabView. Ya que LabView es un lenguaje de programación gráfico, la modificación y adaptación de nuevas características a la lectora puede ser un trabajo menos complejo para el usuario/investigador.

Esta herramienta puede ser útil para el estudio y mejora de sistemas RFID actuales, así como para la creación y validación de nuevas propuestas.

A continuación se da una breve descripción de los sistemas RFID, después se presenta la definición del problema y los objetivos de este trabajo. Por último, se detalla la organización de este documento.

1.1. Identificación por radio frecuencia

RFID por sus siglas en inglés *Radio Frequency Identification* es una tecnología de identificación automática por medio de ondas de radio. Los sistemas de RFID tienen los siguientes componentes:

- **Etiquetas:** Se adhieren a los objetos/personas a identificar. Almacenan códigos de identificación(ID).

- **Lectora:** La lectora interroga a las etiquetas que se encuentran dentro de su área de cobertura para obtener su ID. Las lectoras también pueden escribir/editar información de las etiquetas.

Además de estos componentes, un sistema de RFID también puede contar con una computadora *host*, donde se ejecuta el software de aplicación. El software de aplicación se comunica con la lectora para obtener la información de las etiquetas, procesa esta información y la almacena.

El enlace de comunicación de la lectora a la etiqueta se conoce como enlace de bajada, y la comunicación entre etiqueta y lectora se conoce como enlace de subida. En la figura 1.1 se muestra los enlaces de comunicación y los componentes de un sistema RFID.

Figura 1.1: Componentes de un sistema RFID

RFID generalmente trabaja en las bandas de frecuencia de LF (*Low Frequency*), HF(*High Frequency*), UHF (Ultra High Frequency) y SHF (*Super High Frequency*). En la figura 1.2 se muestran las bandas usadas para aplicaciones de RFID.

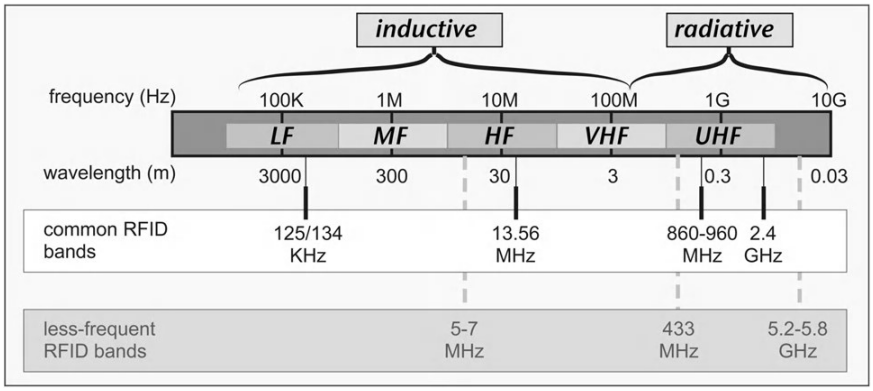


Figura 1.2: Bandas de frecuencia usadas en RFID [6]

Las Lectoras RFID se componen generalmente de de una interfaz de radiofrecuencia (transmisor y receptor) y una sección de control, en el capítulo tres de esta tesis se dan mas detalles de la arquitectura de las lectoras de RFID.

Las etiquetas de RFID se componen generalmente de una antena y un circuito integrado. Las etiquetas se pueden clasificar por la manera de alimentar su circuito y sus capacidades de transmisión de la siguiente manera:

- **Pasivas:** Las etiquetas pasivas no tienen una fuente de alimentación independiente. Obtienen energía de la señal que envía la lectora. Las etiquetas pasivas no tienen transmisor de radio, la lectora provee una señal no modulada y las etiquetas modulan las características eléctricas de la señal por medio de técnicas de *Load Modulation*. Para el caso de sistemas HF y LF se utiliza acoplamineto magnético, para sistemas RFID UHF y SHF se realiza por medio de retrodispersión (también conocido como *back scattering*) [7].

- **Semi-pasivas:** También conocidas como pasivas asistidas por batería. Estas etiquetas utilizan baterias para energizar su circuito, pero tampoco cuentan con un transmisor de radio, para el enlace de subida utilizan la señal que envía la lectora.

- **Activas:** Las etiquetas activas utilizan baterías para alimentar su circuito, y cuentan también con un transmisor de radio convencional.

Como en cualquier sistema de comunicación, para que las etiquetas y la lectora puedan comunicarse, es necesario establecer "reglas" sobre la manera en que se realizará el intercambio de información, lo que se conoce como protocolo de comunicación. El protocolo específica lo siguiente:

- Las interacciones físicas. Define datos técnicos como técnicas de modulación, frecuencia de operación, etc.

- Control de acceso al medio (MAC del inglés *Media Access Control*)

- Procesos de operación, formatos de los mensajes, comandos, etc.

Estos protocolos son aceptados por distintas entidades, creándose estándares de comunicación. Algunas entidades de estandarización para RFID son la ISO (*Organization for Standardization*) y EPCGlobal Inc. Los estándares permiten que lectoras y etiquetas de diferentes marcas sean compatibles e interoperables. En el cuadro 1.1.1 se muestra un resumen de algunos estándares usados en RFID.

Tipo de etiqueta	Frecuencias					
	125/134 KHz	5/7 MHz	13.56 MHz	303/433 MHz	860/960 MHz	2.45 GHz
Pasiva	ISO 11784/5, 14223 ISO18000-2	ISO10536	ISO14443 ISO15693 ISO18000-3		ISO18000-6A,B,C EPC class 0 EPC class 1	ISO18000-4
Semi-pasiva						ISO18000-4
Activa				ISO18000-7		ISO18000-4

Cuadro 1.1.1: Estándares de RFID [6]

El protocolo ISO 18000-6C especifica los requisitos físicos y lógicos para sistemas RFID pasivos que funcionan en el rango de frecuencias de 860 MHz a 960 MHz [8]. En el cuadro 1.1.2 se muestra un resumen de algunas caracterísitcas del protocolo ISO 18000-6C. Como se puede observar, la tasa de transmisión de datos es variable, lo que da flexibilidad al usuario a adaptarse a diferentes condiciones de operación del sistema. El mecanismo anti-colisión utilizado está basado en el *Slotted Aloha* (protocolo Q), el cual puede adaptarse a diferentes tamaños de

poblaciones de etiquetas [6]. El protocolo ISO 18000-6C señala el uso de CRC-16 y CRC-5 para la detección de errores, sin embargo no especifica nada para la corrección de errores. Otra característica del protocolo es que ofrece control espectral de las transmisiones de la lectora y etiquetas para minimizar la interferencia, imponiendo limitaciones en el ancho de banda en términos de máscaras espectrales.

Protocolo ISO 18000-6C	
Codificación del enlace de bajada	Codificación por intervalo de pulso PIE por sus siglas en inglés (Pulse Interval Encoding)
Modulación (Etiquetas)	ASK y/o PSK
Tasa de transmisión de datos (Etiquetas)	de 40 Kbps a 640 Kbps
Modulación (Lectora)	DSB-ASK, SSB-ASK, o PR-ASK
Tasa de transmisión de datos (Lectora)	de 26.7 kbit /s a 128 kbit /s (asumiendo datos equiprobables)
Codificación del enlace de subida	FM0, Miller subcarrier
Mecanismo anticolisión	Q- protocol : variación del Aloha ranurado (en inglés Slotted Aloha).
Identificación unica de etiqueta	Variable: mínimo 16 bits, máximo 496 bits
Detección de errores: enlace de bajada	CRC de 16 bits, excepto un CRC de 5 bits para el comando *(*Query)
Detección de errores: enlace de subida	CRC de 16 bits, excepto sin verificación de errores para RN16

Cuadro 1.1.2: Protocolo ISO 18000-6C

1.2. Definición del problema

Hoy en día la tecnología RFID es utilizada en diversas aplicaciones, gracias a las ventajas que tiene RFID en comparación con otras tecnologías de identificación (por ejemplo código de barras). Entre los mercados más sobresalientes del uso de tecnologías RFID se encuentran las cadenas de suministro, logística, industria de la salud, transporte aéreo [9] entre otras.

Se han propuesto sistemas RFID en aplicaciones que presentan nuevos retos para la tecnología, por ejemplo en aplicaciones que involucran etiquetas en movimiento [1–3], localización e inventario de activos físicos [10], uso de tecnologías RFID en sistemas de *smart-retail* [11] por mencionar algunos. También existen muchas líneas de investigación con propuestas para mejorar los sistemas RFID como son problemas relacionados a la seguridad [12], métodos de selección e identificación de etiquetas RFID [13], sistemas de localización [2], técnicas de corrección de errores [14], entre otros.

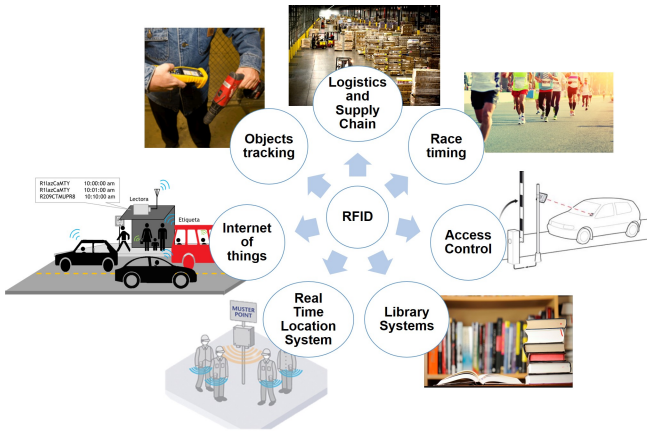


Figura 1.3: Aplicaciones en las que se usa exitosamente RFID

Muchas veces las nuevas propuestas son validadas mediante simulaciones, lectoras comerciales o equipo para prueba de RFID. Sin embargo, los investigadores se enfrentan con algunas dificultades: en el caso de utilizar lectoras comerciales disponibles, los investigadores deben usar las configuraciones estándar incluidas en las lectoras, las cuales son limitadas y no incluyen todas las opciones del protocolo de comunicación.

Entonces los investigadores deben conformarse con el desempeño del sistema ante las limitadas configuraciones de fábrica [4]. Además, en las lectores comerciales disponibles no es posible realizar cambios en la capa física y MAC lo que dificulta la implementación de nuevas propuestas [5]. Por otro lado, existen equipos especializados para prueba de tecnología RFID [15, 16]. Algunos equipos de prueba pueden ser costosos, los usuarios no tienen acceso a todos los parámetros de interés en los equipos de medición, y en otros se necesitan licencias para los códigos y librerías de pruebas RFID .

Ante esta situación, este proyecto propone el uso de herramientas de SDR (por sus siglas en inglés *Software Defined Radio*) para el estudio y mejora de sistemas RFID. Este trabajo se enfoca particularmente en el desarrollo de una herramienta para sistemas RFID UHF que trabajan bajo el protocolo ISO 18000-6C.

Se han realizado otros trabajos en los que se utiliza SDR para RFID, basados en FPGA o algunos desarrollados en leguanjes de programación como C++, python o VHDL [4, 17, 18], lenguajes que pueden resultar complejos para investigadores que no están familiarizados en el área de programación, pero que están interesados en la tecnología RFID, son expertos en otras disciplinas y desean implementar sus propuestas.

Este trabajo se desarrolló con equipo de SDR de *National Instruments* y con el entorno de programación de *LabView*. *LabView* es un entorno de programación gráfico que ayuda a visualizar cada aspecto de la aplicación, incluyendo configuración de hardware, datos, depuración, etc. Esta visualización hace que sea más fácil integrar hardware, representar lógica de la aplicación en diagramas y diseñar interfaces de usuario personalizadas [19]. Tiene una gran variedad de librerías que puede facilitar la modificación de las características de la herramienta de RFID desarrollada y también puede facilitar la implementación de nuevas propuestas.

Esta tesis resuelve retos técnicos que requieren de investigación tales como desarrollo de

herramientas para la emulación de protocolos de RFID, enseñanza de protocolos RFID, configuraciones customizadas de parámetros de lectora RFID, por lo que se considera que tiene un aporte modesto al estado del arte de la tecnología RFID, pero puede ser de gran interés para investigadores que busquen herramientas para probar sus propuestas, o personas interesadas en conocer más acerca de la tecnología RFID y el estudio/enseñanza del protocolo ISO 18000-6C.

1.3. Objetivos

El objetivo de esta tesis es desarrollar una herramienta, basada en la tecnología de radio definido por software y el entorno de desarrollo LabView, que facilite el estudio y experimentación de sistemas de RFID. La herramienta es la implementación de las funciones de una lectora RFID (comunicación con etiquetas comerciales) que opera en la banda de frecuencia de UHF bajo el estándar ISO 18000-6C. El desarrollo de esta herramienta con equipo de radio definido por software da la flexibilidad de realizar configuraciones del transmisor/receptor de radio por medio de software, siendo posible la configuración de distintos parámetros del enlace (acceso completo a los parámetros establecidos por el estándar ISO 18000-6C).

Se desarrolla una interfaz gráfica para que la configuración de los parámetros del enlace del sistema de RFID sea un proceso sencillo para el usuario.

También se pretende que este trabajo sea herramienta útil para la validación y experimentación de nuevas propuestas: ya que LabView es un lenguaje de programación gráfico, la modificación y adaptación de nuevas características a la herramienta puede ser un trabajo menos complejo para el usuario/investigador.

1.4. Organización del documento

Este documento se organiza de la siguiente manera: En el primer capítulo se da una pequeña introducción a la tecnología RFID, tocando temas como clasificación de sistemas de RFID, bandas de frecuencia de operación, y estándares para RFID. También se presenta la definición del problema y los objetivos del trabajo.

El segundo capítulo trata aspectos relevantes del protocolo ISO-18000-6C, mismos que se tomaron en cuenta para el desarrollo del emulador de RFID. Algunos de estos aspectos son parámetros del enlace de comunicación y comandos del proceso de inventario. También se menciona los métodos de detección de errores y la estricta temporización del enlace (tiempos limites que tiene la lectora para responder a las etiquetas y poder completar el proceso de comunicación (obtención de RN16 y número de identificación (ID)).

El tercer capítulo da una breve descripción de la arquitectura de las lectoras de RFID que trabajan en la banda de UHF. La intención es mostrar los aspectos a considerarse al construir/diseñar lectoras RFID.

En el capítulo cuatro habla del desarrollo de la herramienta para la emulación del protocolo ISO 18000-6C, se muestran los componentes de la herramienta, el software de la aplicación y el *hardware* utilizado.

En el capítulo cinco se presentan los resultados obtenidos, también se comentan los retos y dificultades enfrentados en la comunicación con etiquetas comerciales. Por último, el capítulo

seis se presentan las conclusiones y se proponen temas de trabajo a futuro.

Capítulo 2

Estándar ISO 18000-6C

El protocolo ISO 18000-6C define la interfaz de aire para dispositivos de identificación por radio frecuencia (RFID), de sistemas pasivos, que se comunican por medio de técnica de retro-dispersión y que operan en la banda de 860 MHz a 960 MHz.

El protocolo ISO 18000-6C especifica:

- Interacciones físicas (la capa de señalización del enlace de comunicación) entre lectora y etiquetas.
- Procedimientos de operación y comandos de lectora y etiquetas.
- Mecanismo anticolisión empleado.

A continuación, se dará una explicación breve de algunas de las características del protocolo IS0 18000-6C.

2.1. Marcadores de "Inventario" y Marcadores "SL":

Una sesión es el proceso de inventario entre la lectora y un grupo de etiquetas. La lectora elige una de entre cuatro sesiones disponibles (S0, S1, S2, S3) y hace el inventario de las etiquetas asociadas a dicha sesión [20]. Las etiquetas tienen un marcador interno de "inventario"(en inglés, *inventoried flag*) independiente para cada sesión. Cada marcador tiene dos posibles estados: "A"(el valor por defecto) y "B". Cuando en una sesión una etiqueta ha sido leída o "inventariada",esta invierte automáticamente el estado de su marcador a "B". El tiempo que la etiqueta mantiene su marcador en el estado "B" antes de volver al estado "A" se conoce como tiempo de persistencia. El tiempo de persistencia se relaciona con el número de sesión como se muestra en el cuadro 2.1.1

Sesión	**Tiempo de persistencia Etiqueta energizada (seg)**	**Tiempo de persistencia etiqueta no energizada (seg)**
S0	Indefinido	0
S1	$0{,}5 < Tper < 5$	$0{,}5 < Tper < 5$
S2	Indefinido	$2 < Tper$
S3	Indefinido	$2 < Tper$

Cuadro 2.1.1: Tiempos de persistencia

La selección de la sesión depende del número de etiquetas que se encuentran en el sistema. Para sistemas con grandes poblaciones de etiquetas se recomienda emplear S3 y S4. Para poblaciones pequeñas se recomienda S0 y S1 [21].

Las etiquetas también cuentan con un marcador "SL"que puede ser usado en combinación con los marcadores de "inventario" para seleccionar a las etiquetas que participarán en la ronda de inventario.

2.2. Memoria de las Etiquetas RFID

Las etiquetas cuentan con cuatro bancos de memoria:

- **Memoria Reservada:** Contiene las siguientes contraseñas:
  - *Kill*: Se utiliza para desactivar la etiqueta
  - *Access*: Permite a la etiqueta pasar a un estado seguro para realizar transacciones de lectura/escritura en la memoria de la etiqueta
- **Memoria UII (del inglés, Unique Item Identifier):** Debe contener:
  - Código de control de errores (CRC-16), utilizado para comprobar la integridad de los mensajes transmitidos.
  - Control de Protocolo (del inglés *PC, protocol control*), describe la longitud del ID del objeto/persona, e información opcional acerca de la etiqueta.
  - El número de identificación del objeto/persona (ID/EPC) al que está adherido la etiqueta
- **Memoria de Identificación de etiqueta (TID)**: Contiene información para identificar la etiqueta y no al objeto al que está adherida, por ejemplo comandos especiales soportados por la etiqueta, información del fabricante de la etiqueta, etc)
- **Memoria de usuario**: Permite el almacenamiento de datos específicos del usuario.

2.3. Enlace de bajada ($L \rightarrow E$)

A continuación se expondrán algunos aspectos importantes del enlace de comunicación de Lectora a Etiqueta, también conocido como enlace de bajada.

2.3.1. Modulación

En los sistemas RFID las instrucciones de la lectora y las respuestas de las etiquetas están representadas por cadenas de datos binarios, los cuales serán transmitidas por un canal inalámbrico. Para transportar la información por el canal de comunicación, es necesario generar una señal que represente esta información y que tenga el formato adecuado para las características del canal. Esta señal se genera por medio de técnicas de modulación.

En RFID, las etiquetas son dispositivos sencillos, que no tienen mucha capacidad de procesamiento. Es por esto, que la comunicación entre lectora y etiqueta debe de ser por medio de técnicas de modulación simples. El estándar 18000-6C específica el uso de modulaciones binarias variantes de ASK (del inglés, *Amplitude Shift Keying*).

2.3.2. Codificación por Intervalo de Pulso.

Antes de ser modulados, los datos del enlace L→E son codificados con códigos de línea. Un código de línea es la representación de la señal que corresponde a los valores lógicos del bit (datos "0" y "1"). Dicha señal suele representarse mediante un número determinado de pulsos.

El estándar 18000-6C especifica el uso de codificación por intervalo de pulso (PIE, del inglés *Pulse Interval Encoding*). Esta codificación mantiene siempre una potencia de transmisión promedio que asegura que las etiquetas se encuentren activas durante todo el inventario aún cuando la lectora transmita cadenas largas de valores lógicos "0" [6].

Las señales que representan los valores lógicos "0" y "1" (símbolos) en la codificación PIE se muestran en la figura 2.1. Como se puede apreciar, la duración del símbolo "0" es menor a la del símbolo "1". La unidad de tiempo de referencia es el Tari, el cual puede tomar valores en el rango de 6.25 μs a 25μs.

El símbolo "0" tiene una duración de 1 Tari y consiste en dos pulsos: un pulso de encendido, seguido de un pulso de apagado. Ambos pulsos son de duración PW (del inglés, *Pulse Width*).

El símbolo "1" se representa por un pulso de apagado de duración PW al final de un pulso de encendido de mayor duración. La duración total del símbolo "1" debe estar en el rango de 1.5 Tari a 2 Tari.

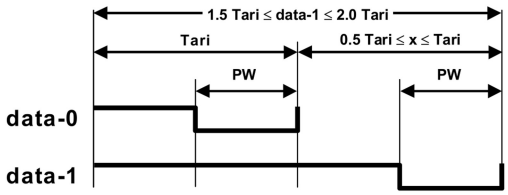


Figura 2.1: Símbolos codificación PIE. Fuente: [8]

2.3.3. Preámbulo y Frame-Sync

En los sistemas RFID Gen2 la transmisión de información se realiza por medio de paquetes. Cada paquete comienza transmitiendo símbolos especiales que facilitan que el receptor identifique el inicio del paquete. En el enlace de bajada se usan dos símbolos: el *Preámbulo* y el *Frame-Sync*:

Preámbulo: La representación del preámbulo se muestra en la figura 2.2. El preámbulo se usa siempre que la lectora inicia una ronda de inventario (comando *Query*). Por medio del preámbulo, la lectora establece los parámetros del enlace de subida y de bajada para la sesión de inventario. Se compone de las siguientes partes:

- **Delimitador**: Es un pulso de apagado que siempre tendrá una duración de 12,5μs. Se encuentra siempre al comienzo del símbolo.
- **Símbolo "0":** Después del delimitador, la lectora envía un "0" binario, con lo que define el valor del Tari utilizado para esa ronda de inventario.
- **RTcal:** Símbolo de calibración $L \rightarrow E$. La duración total del RTcal es la suma de la duración del símbolo "0" y del símbolo "1". El TRcal puede tener una duración en el rango de 2.5 Tari a 3 Tari. La etiqueta utiliza el RTcal para establecer la duración de los símbolos de la lectora y poder distinguir entre "0" y "1".
- **TRcal:** Símbolo de calibración $E \rightarrow L$. El TRcal tiene una duración en el rango de 1.1 RTcal a 3 RTcal. La etiqueta calcula la frecuencia de transmisión BLF (del inglés, *Backscatter Link Frequency*) con la duración de este símbolo dividido entre un parámetro conocido como *Divide Ratio* (DR) el cual es transmitido por la lectora al inicio del inventario.

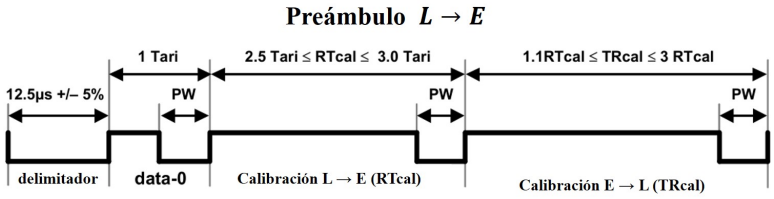


Figura 2.2: Secuencia de Preámbulo. Fuente: [8]

Frame-Sync: Este símbolo antecede a todos los comandos, a excepción del comando de inicio de ronda de inventario *Query*. El *Frame-Sync* contiene información de temporización de la lectora, que ayuda a las etiquetas a permanecer sincronizadas durante operaciones sucesivas. La estructura del *Frame-Sync* se muestra en la figura 2.3. Como se puede observar, se compone de las mismas partes que el preámbulo, excluyendo el símbolo TRcal.

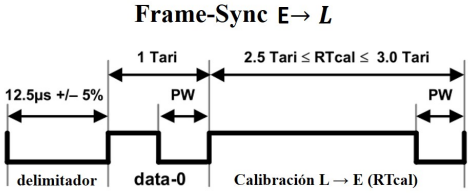


Figura 2.3: Secuencia de *Frame-Sync*. Fuente: [8]

2.4. Enlace de subida ($E \rightarrow L$)

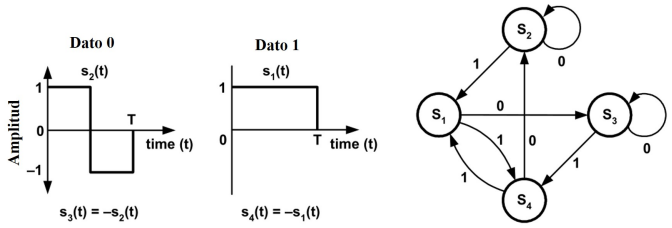
A continuación se expondrán algunos aspectos importantes del enlace de comunicación de Etiqueta a Lectora (se abreviará como ($E \rightarrow L$)), también conocido como enlace de subida.

2.4.1. Codificación de datos de la etiqueta

Los datos que envía la etiqueta a la lectora pueden ser codificados ya sea por codificación FM0 o por la técnica de sub-portadora modulada por Miller (MMS, del inglés Miller-modulated subcarrier). A continuación se da una breve exposición de codificación FM0.

Codificación FM0

El modo de operación por defecto es la codificación FM0. Esta codificación invierte la fase de la señal banda base al final de cada símbolo. La duración del símbolo, llamada T_{pri}, es la misma para los símbolos "1" y "0". La figura 2.4(a) muestra las funciones básicas para el "0" lógico y "1" lógico. Como se puede observar, el símbolo "0" tiene una inversión de fase adicional a la mitad del símbolo. La figura 2.4(b) muestra el diagrama de transición de estados para generar codificación FM0. La codificación FM0 tiene memoria, por lo que la selección de los símbolos dependen de las transmisiones anteriores.



(a) Funciones básicas FM0 (b) Diagrama de transición de estados FM0

Figura 2.4: Codificación FM0. Fuente: [8]

En la codificación FM0, todas las transmisiones deben terminar con un símbolo "1" adicional (*dummy*) y siempre deben terminar en el mismo estado de "apagado" o "bajo". Un ejemplo de codificación FM0 se muestra en la figura 2.5.

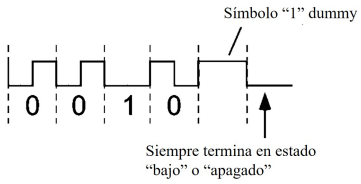


Figura 2.5: Ejemplo de transmisión FM0

Cuando la etiqueta transmite con codificación FM0 la tasa de transmisión de datos es el inverso del periodo del símbolo ($\frac{1}{T_{pri}}$) y es conocida como BLF (por sus siglas en inglés *Backscatter*

Link Frequency). La lectora especifica a la etiqueta la BLF al comienzo de la ronda de inventario, lo cual se detallará mas adelante.

2.4.2. Preámbulo FM0

Todas las transmisiones de etiqueta a lectora deben de comenzar con un preámbulo. El preámbulo sirve para marcar el inicio de transmisión $E \rightarrow L$. En la señalización FM0 se tienen dos posibles preámbulos, los cuales se muestran en la figura 2.6. Los preámbulos difieren por la presencia de una secuencia de 12 símbolos "0", conocidos como *pilot tone*. La lectora especifica el preámbulo que será empleado por medio del comando *Query*. La "v" que se muestra en la figura 2.6, indica una violación al código FM0 (un símbolo que no obedece las reglas de la codificación FM0).

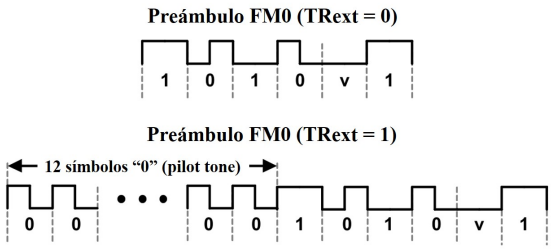


Figura 2.6: Preámbulo FM0. Fuente: [8]

2.5. Proceso de comunicación con etiquetas.

El protocolo ISO 18000-6C es un protocolo tipo ITF (del inglés, *Interrogator-talks-first*), dónde las etiquetas son energizadas por la señal de radiofrecuencia que envía la lectora, pero permanecen inactivas hasta que la lectora solicite que se identifiquen (hasta que la lectora comience una nueva ronda de inventario).

Las lectoras que trabajan bajo el estándar ISO 18000-6C gestionan a las etiquetas que estén dentro de su área de cobertura mediante tres operaciones básicas:

- **Selección:** Permite la selección del grupo de etiquetas que van a participar en la siguiente ronda de inventario o de acceso.
- **Inventario:** Proceso de identificación de etiquetas. La lectora comienza el proceso de identificación al emitir un comando *Query*, donde le especifica a las etiquetas los parámetros del enlace de subida. Las etiquetas que hayan recibido el comando *Query* y se encuentren dentro del grupo de etiquetas especificado por el comando *Select* responden por turnos a la lectora. La lectora detecta la respuesta de una sola etiqueta y le solicita su número de identificación (ID). Una ronda de inventario comienza con un comando *Query* y termina al emitirse otro comando *Query* (el cual comienza una nueva ronda de inventario) o al emitirse un comando *Select*.
- **Acceso:** Proceso por el cual la lectora lee o escribe información en la memoria de las etiquetas.

2.6. Comandos ISO 18000-6C

A continuación, se exponen los comandos involucrados en los procesos de selección e inventario.

2.6.1. Comandos del proceso de Selección

En este proceso se emplea un único comando, *Select*, el cual contiene los siguientes parámetros:

- *Target* y *Action* indican como el comando *Select* modificará los marcadores de "SL" o los marcadores de "inventario" de las etiquetas y para que sesión.
- *Pointer*,*Length* y *Mask*: *Mask*, especifica una cadena de bits que la etiqueta compara con alguna de las memorias en el segmento especificado por *Pointer*. *Length* indica la longitud de *Mask*. Las etiquetas que en el segmento de memoria especificado tengan almacenada información que concuerde con *Mask* modificarán sus marcadores como lo indica *Action* en la sesión que especifica *Target*. El parámetro *Action* también proporciona indicaciones para aquellas etiquetas que no coinciden con *Mask*, como se muestra en el cuadro 2.7(b)
- *MemBank:* Específica el banco de memoria al cual será aplicado *Mask*, entre los que se encuentran UII, TID, o Memoria de Usuario.
- *Truncate*: Especifica si en su respuesta la etiqueta enviará el ID completo, o será truncado a los datos del ID después de *Mask*. Aplica solo si en el siguiente comando *Query* se especifican los parametros *Sel*=10 o *Sel*=11.

En la figura 2.7(a) se muestra la sintaxis del comando *Select*:

	Comando	Target	Action	MemBank	Pointer	Length	Mask	Truncate	CRC-16
# de bits	4	3	3	2	EBV	8	Variable	1	16
Descripción	1010	000: Inventoried (S0) 001: Inventoried (S1) 010: Inventoried (S2) 011: Inventoried (S3) 100: SL 101: RFU 110: RFU 111: RFU	Ver cuadro B	00: RFU 01: UII 10: TID 11: User		Longitud de Mask (bits)	Valor de máscara	0: deshabilitado 1: habilitado	

(a) Comando *Select*

Action	Etiquetas que coinciden con Mask	Etiquetas que no coinciden con Mask
000	Fijar SL o marcador de inventario → A	No fijar SL o marcador de inventario → B
001	Fijar SL o marcador de inventario → A	----------
010	----------	No fijar SL o marcador de inventario → B
011	Negar SL o (A → B, B → A)	----------
100	No fijar SL o marcador de inventario → B	Fijar SL o marcador de inventario → A
101	No fijar SL o marcador de inventario → B	----------
110	----------	Fijar SL o marcador de inventario → A
111	----------	Negar SL o (A → B, B → A)

(b) Respuesta de etiquetas al parámetro *Action*

Figura 2.7: Formato comando *Select*

2.6.2. Comandos del proceso de Inventario.

El tipo de comunicación entre lectora y etiquetas es half-duplex, lo que significa que la lectora transmite información, mientras las etiquetas escuchan y viceversa, pero no transmiten información ambas al mismo tiempo. El control de acceso al medio utilizado es un algoritmo aleatorio anti-colisión ranurado (en inglés, *random slotted collision-arbitration*). Se conoce como ranurado ya que el tiempo en el que la lectora escucha a las etiquetas se divide en $2^Q - 1$ ranuras o "slots" y cada etiqueta selecciona de manera aleatoria uno de estos "slots" para realizar su transmisión. Podría verse como si las etiquetas tomaran "turnos" para comunicarse con la lectora. Para realizar el proceso de inventario la lectora emplea los comandos *Query*, *QueryAdjust*, *QueryRep*, *ACK* y *NAK*. En la figura 2.8 se muestra un esquema del proceso de inventario.

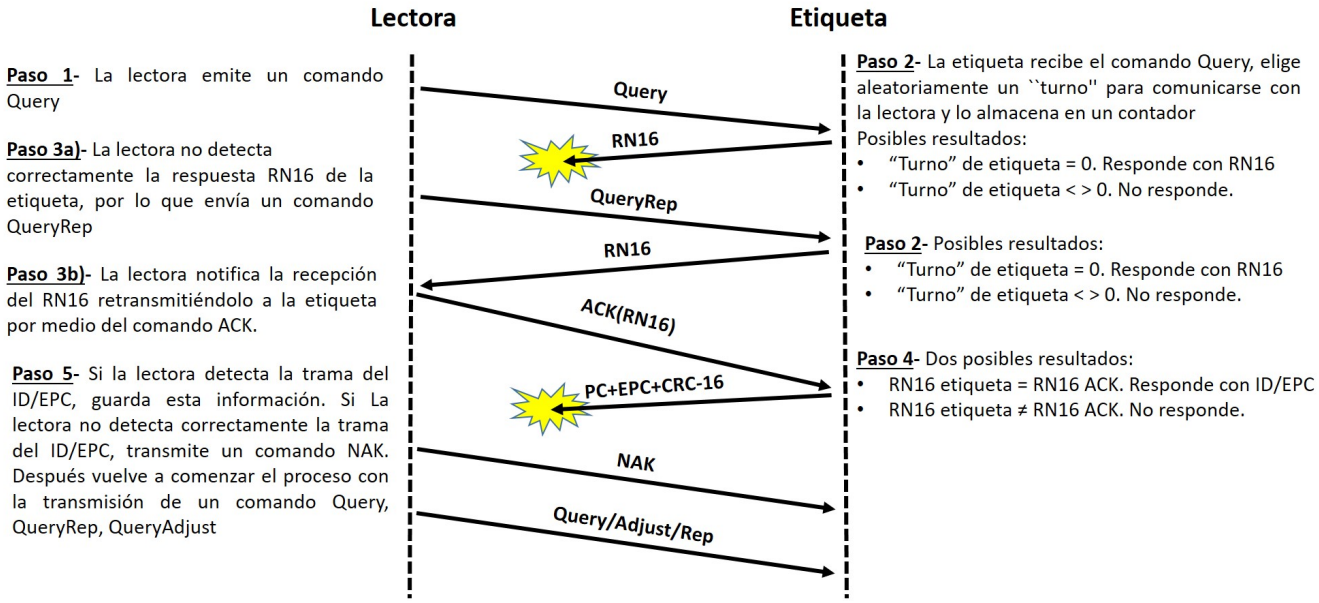


Figura 2.8: Esquema de proceso de inventario

A continuación se describe el formato de cada uno de los comandos del proceso de inventario.

Comando *Query*

Este comando inicia y específica una ronda de inventario. El comando tiene los siguientes parámetros:

- *DR*: El parámetro *DR* junto con el preámbulo establecen la tasa de transmisión $E \rightarrow L$. Las etiquetas deben medir la duración de TRcal, calcular la BLF mediante la ecuación 2.6.1 y ajustar su tasa de transmisión $E \rightarrow L$ para que sea igual a BLF.

$$BLF = \frac{DR}{TRcal} \tag{2.6.1}$$

- *M(ciclos por segundo)*: Especifica la tasa de transmisión $E \rightarrow L$ y el formato de modulación. El valor "1" establece la tasa de transmisión BLF, y codificación FM0.

- *TRext*: Indica si el preámbulo $E \rightarrow L$ esta precedido por un tono piloto (*pilot tone*).
- *Sel*: Selecciona que etiquetas responden al comando.
- *Session*: Selecciona la sesión en que se realizará la ronda de inventario
- *Target*: Selecciona las etiquetas que participarán en la ronda de inventario, dependiendo del estado en que se encuentre su marcador de "inventario".
- *Q*: Establece el número de "turnos" o "ranuras" disponibles en la ronda de inventario

En la figura 2.9 se muestra los parámetros del comando *Query*

	Comando	DR	M	TRext	Sel	Session	Target	Q	CRC-5
# de bits	4	1	2	1	2	2	1	4	5
Descripción	1000	0: DR=8 1: DR = 64/3	00: M = 1 01: M = 2 10: M = 4 11: M = 8	0: Sin tono piloto 1: Con tono piloto	00: Todas 01: Todas 10: ~SL 11: SL	00: S0 01: S1 10: S2 11: S3	0 : A 1 : B	0-15	

Figura 2.9: Comando *Query*

Comando *QueryAdjust*

Este comando sirve para ajustar el valor de Q, sin modificar ningún otro parámetro en la ronda de inventario. El comando *QueryAdjust* incluye los siguientes campos:

- *Session*: Indica la sesión en la que se realizará la ronda de inventario.
- *UpDn*: Especifica como se debe ajustar el parámetro Q.

El comando *QueryAdjust* se muestra en la figura 2.10.

	Comando	Session	UpDn
# de bits	4	2	3
Descripción	1001	00: S0 01: S1 10: S2 11: S3	110: Q = Q+1 000: Sin modificar Q 011: Q = Q-1

Figura 2.10: Comando *QueryAdjust*

Comando *QueryRep*

La lectora interroga por "turnos" a las etiquetas para poder identificarlas y emite un comando *QueryRep* cuando el "turno" de una etiqueta ha terminado (la lectora ha obtenido el ID de la etiqueta o se detectó un error en el enlace). Entonces las demás etiquetas disminuyen su contador de "turnos". La siguiente etiqueta en participar en el inventario es la que tenga el "turno" 0. El comando *QueryRep* se muestra en la figura 2.11.

	Comando	Session
# de bits	2	2
Descripción	00	00: S0 01: S1 10: S2 11: S3

Figura 2.11: Comando *QueryRep*

Comando *ACK*

La lectora notifica a la etiqueta que recibió un RN16 válido al retransmitirselo por medio del comando *ACK*. La etiqueta compara el RN16 recibido con su RN16. Si los dos RN16 coinciden la etiqueta responde a la lectora con su ID, en caso contrario la etiqueta no responde a la lectora. La figura 2.12 muestra el comando ACK

	Comando	RN
# de bits	2	16
Descripción	01	Retransmisión de RN16 que se recibió de la etiqueta

Figura 2.12: Comando *ACK*

Comando *NAK*

La lectora emite este comando para indicar a una etiqueta que su EPC no se recibió correctamente. Después de que una etiqueta ha recibido un comando *NAK*, debe de esperar a la siguiente ronda de inventario para intentar comunicarse con la lectora.

La figura 2.13 muestra el comando NAK

	Comando
# de bits	8
Descripción	11000000

Figura 2.13: Comando *nak*

2.7. Temporización del enlace

El protocolo ISO 18000-6C sigue una estricta temporización para el proceso de inventario, la cual se muestra en la figura 2.14. Como puede observarse, el estándar establece los tiempos necesarios entre comandos, así como el tiempo entre el último símbolo transmitido por la lectora y la respuesta de la etiqueta.

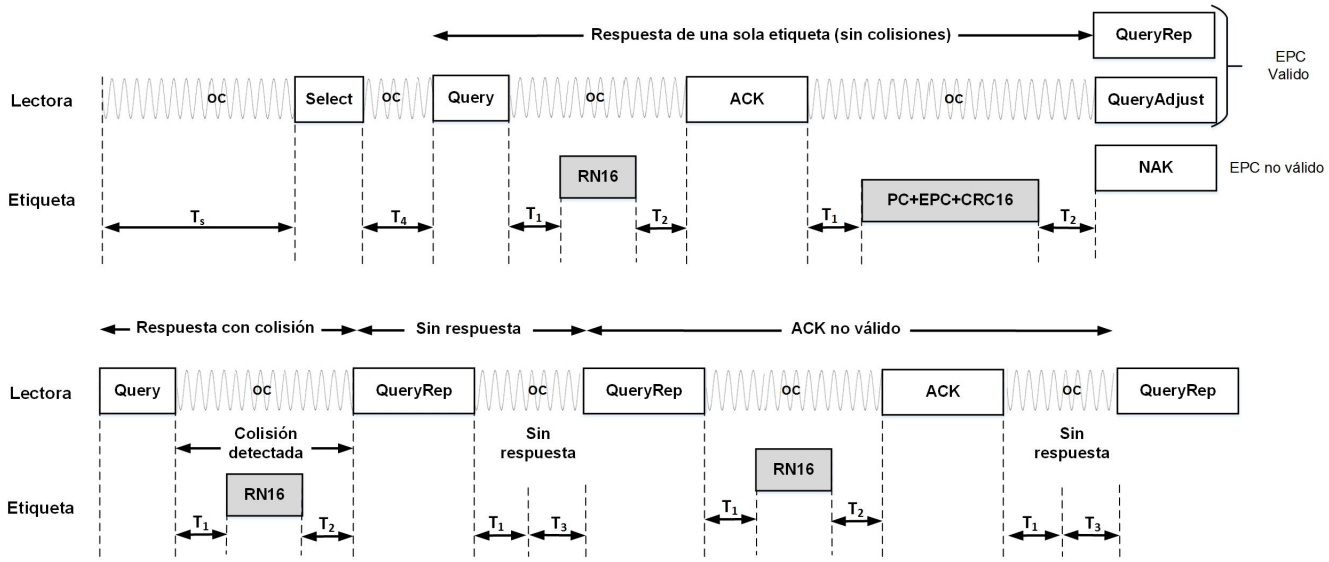


Figura 2.14: Temporización de enlace RFID según el estándar ISO 18000-6C [8]

Parámetro	Mínimo	Nominal	Máximo	Descripción
T_s			$1500\mu s$	Tiempo de estabilización (transmisión de onda continua)
T_1	$MAX(RTcal, 10T_{pri})$ $\times(1-FT)-2\mu s$	$MAX(RTcal, 10T_{pri})$	$MAX(RTcal, 10T_{pri})$ $\times(1+FT)+2\mu s$	Tiempo de espera para que la etiqueta responda a un comando de la lectora
T_2	$3{,}0T_{pri}$		$20T_{pri}$	Tiempo entre respuesta de etiqueta y siguiente comando
T_3	$0{,}0T_{pri}$			Tiempo de espera de la lectora antes de emitir otro comando, después de haber transcurrido T_1 seg.
T_4	$2{,}0RTcal$			Tiempo mínimo entre comandos.

Cuadro 2.7.2: Parámetros de temporización de enlace [8]

El tiempo de estabilización de la señal de la lectora después de encendido no debe ser mayor de $1500\mu s$, en ese tiempo se transmitirá una señal de onda continua. A continuación la lectora puede emitir cualquier comando, ya sea comando *Select* o iniciar un inventario mediante el comando *Query*. El tiempo entre comandos consecutivos T_4 debe ser mayor a dos veces RTcal. Si al emitir cualquier comando en el que se espera una respuesta de la etiqueta, esta última no responde en un tiempo menor o igual a T_1, la lectora vuelve a emitir un comando *Query*, *QueryRep*, *Query Adjust* dependiendo de la etapa del proceso en la que se encuentre. Hay que mencionar que en muchas aplicaciones, la información de interés es el número de identificación de la etiqueta, la cual se transmite en el paquete PC + EPC + CRC16 que se muestra en la figura 2.14. Para que la etiqueta envíe esta trama, debe de haber recibido un comando *ACK* en un tiempo $T_2(max)$, el cual la etiqueta comienza a calcular justo después de la transmisión del último bit del RN16.

La temporización del enlace varía con la tasa de transmisión tanto del enlace de subida, como del enlace de bajada.

2.8. Técnicas de control de errores

Las aplicaciones de uso de RFID es muy variada, y en ellas la información que se obtiene por medio del sistema RFID es usada para la toma de decisiones. Es por esto que los sistemas RFID deben garantizar la correcta identificación de personas/objetos. Sin embargo, en muchas ocasiones se tienen errores en la comunicación.

Es por esto que, para asegurar la integridad de los mensajes en un sistema de comunicación se emplean técnicas de detección de errores, cuyo objetivo es dotar de información al receptor para que sea capaz de reconocer si el mensaje que recibió fue corrompido durante su transmisión. Una de las técnicas de detección de errores empleada por el protocolo ISO 18000-6C es la verificación por redundancia cíclica (CRC , por sus siglas en inglés *cyclic redundancy check*), en la que el transmisor calcula una suma de comprobación (en inglés *checksum*), que se obtiene del residuo de una división entre un polinomio generador y el contenido del mensaje, este residuo es anexado al mensaje antes de la transmisión.

El CRC es una técnica ampliamente utilizada y existen varios estándares. El protocolo ISO 18000-6C emplea dos tipos de CRC: CRC-16 y CRC-5. Para mayor información de CRC empleado, consultar el protocolo [8]. Es importante mencionar que el protocolo ISO 18000-6C no toma acciones para corregir los errores detectados, y los mensajes que contienen error, son desechados.

En el cuadro 2.8.3 se muestran los comandos de la lectora, mensajes de la etiqueta y la técnica de control de errores empleado, todo esto correspondiente únicamente a los procesos de selección e inventario. Cómo se puede observar, los comandos *QueryRep*, *ACK*, *Query*, *QueryAdjust* y *NAK* tienen una longitud única (ningún otro comando tiene la misma longitud). Si una etiqueta recibe uno de estos comandos con una longitud incorrecta o detecta un error con el CRC en comandos que lo utilizan, la etiqueta ignorará la instrucción recibida. De igual manera, si la lectora recibe el PC+EPC+CRC16 de la etiqueta, y detecta un error con el CRC-16, la lectora descartará toda la trama recibida, y enviará un comando de NAK a la etiqueta para notificarle que no ha sido identificada y que debe esperar la siguiente ronda de inventario para ser identificada. El recibir mensajes con errores resultará en intentos repetitivos hasta obtener correctamente la información, lo que a su vez ocasionará incrementos en el tiempo de identificación de etiquetas, degradando el desempeño del estándar. Esta degradación puede ser mas significativa cuando ocurre un error de bit de la trama de PC+EPC+CRC16 de la etiqueta, ya que se ha consumido tiempo para llegar hasta ese punto del proceso y además se debe esperar a que un comando NAK sea transmitido antes de continuar con el proceso de identificación [6].

COMANDOS DE LECTORA			
Comando	**Código**	**Longitud**	**Protección**
QueryRep	00	4	Longitud única
ACK	01	18	Longitud única
Query	1000	22	Longitud única y CRC-5
QueryAdjust	1001	9	Longitud única
Select	1010	>44	CRC-16
NAK	11000000	8	Longitud única
RESPUESTA DE ETIQUETAS			
Comando de lectora	**Respuesta**	**Longitud**	**Protección**
ACK	PC+UII+CRC-16	De 21 a 528	CRC-16

Cuadro 2.8.3: Protección a comandos y mensajes.

Capítulo 3

Lectoras RFID UHF

Para obtener la información de las etiquetas, se necesita de una lectora de RFID. En el caso de un sistema RFID pasivo la lectora de RFID tiene las siguientes funciones:

- Proveer de energía a las etiquetas
- Guiar la comunicación con las etiquetas
- Transferencia de información entre la aplicación de software (la cual se encuentra en una computadora huesped) y las etiquetas

En este capítulo se dará una breve descripción de la arquitectura de las lectoras de RFID que trabajan en la banda de UHF. La intención es mostrar los aspectos a considerarse al construir/diseñar lectoras RFID, algunos son típicos de los sistemas de radio, y otros son propios de la tecnología RFID.

3.1. Principio de maestro-esclavo en sistemas RFID

La información que se obtiene de las etiquetas es utilizada de diferentes maneras dependiendo de la aplicación en que se encuentre el sistema. Normalmente el análisis y almacenamiento de esta información se realiza en una computadora huésped. Para obtener la información de las etiquetas, el software de aplicación que corre en la computadora huésped debe comunicarse con la lectora de RFID. Para la lectura y escritura de datos en las etiquetas, la comunicación entre el software de aplicación y la lectora se basa en el principio del maestro-esclavo. El software de aplicación toma el rol de maestro y es quien comienza la comunicación con la lectora y envía comandos para indicar qué operación se va a realizar. Para poder ejecutar una instrucción del software de aplicación, la lectora debe comunicarse con las etiquetas. La comunicación entre etiquetas y lectora siempre es guiada por la lectora, al enviar comandos a las etiquetas para comenzar rondas de inventario, o para escribir datos en ellas. Entonces la lectora toma el papel de maestro y las etiquetas son esclavas, ya que esperan a las instrucciones de la lectora para comenzar con la transferencia de información. En la figura 3.1 se muestra un diagrama de la relación de maestro-esclavo entre software de aplicación, lectora y etiquetas.

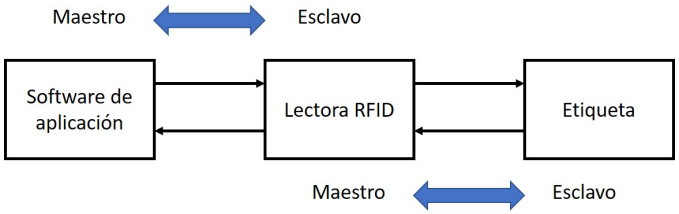


Figura 3.1: Relación de maestro-esclavo entre software de aplicación, lectora y etiquetas [22]

3.2. Componentes de una Lectora de RFID

En la figura 3.2 se muestra un diagrama de bloques de una lectora RFID. Según [22], una lectora de RFID se puede dividir en los siguientes bloques:

- **Interfaz de Radio Frecuencia:** Se encarga de la comunicación con las etiquetas por medio de ondas de radio. Está conformado por un transmisor y un receptor.
- **Sección de Control:** Está encargado de la comunicación con la computadora *host* y la interacción con la interfaz de radio frecuencia. También realiza procesamiento digital y procedimientos a la señal recibida por la interfaz de radio

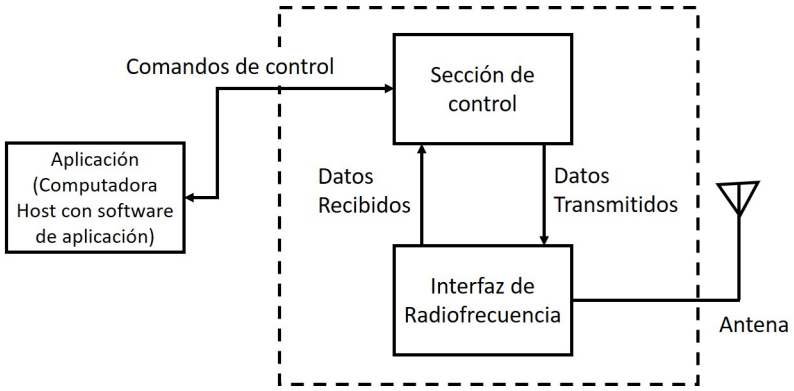


Figura 3.2: Diagrama de bloques de lectora de RFID, con sus dos componenes: Interface de RF y Sistema de Control [22]

A continuación se da una breve descripción de cada una de las partes que componen a una lectora RFID.

3.2.1. Interfaz de Radio Frecuencia

Este componente es un transceptor de radio compuesto por un transmisor y un receptor, que trabajan juntos para comunicarse con las etiquetas. Los transmisores y receptores de las lectoras de RFID pasivas se construyen generalmente usando arquitectura de conversión directa, también conocida como homodino.

La interfaz de radio frecuencia tiene las siguientes funciones:

- El transmisor debe generar la señal de alta frecuencia con la que las etiquetas son activadas. Las etiquetas se alimentan con esta señal y también usan esta señal para enviar información a la lectora
- El transmisor debe transmitir una señal modulada para enviar información a las etiquetas.
- Recepción y demodulación de las señales enviadas por las etiquetas.

El protocolo ISO 18000-6C especifica que la transferencia de datos entre etiquetas y lectoras es *Half-Duplex*, es decir, la lectora transmite (habla) mientras que las etiquetas reciben (escuchan) y viceversa. Sin embargo, la lectora debe alimentar a las etiquetas en todo momento, de manera simultánea a la transferencia de datos.

Figura 3.3: Comunicación *Half Duplex* y modo de transferencia de energía simultánea [7]

En el enlace de bajada esto es posible gracias al tipo de codificación empleada (PIE). Para el enlace de subida, la lectora debe energizar a las etiquetas y también debe proveerlas de un medio físico(en la forma de una señal no modulada) para que puedan responder a los comandos de la lectora [7].

La señal de retrodispersión de la etiqueta al llegar al receptor de la lectora está en el orden de los nW y está presente al mismo tiempo y en la misma frecuencia que la señal que transmite la lectora (la cual puede llegar hasta 4W(PIRE) dependiendo de las regulaciones de cada país). Esto impone un desafío para el receptor de la lectora, ya que debe ser capaz de obtener información de la débil señal de retrodispersión, aún en presencia de la potente señal de interferencia transmitida por la lectora. Debido a que la señal transmitida por la lectora está en la misma frecuencia que la señal de retrodispersión, esta no puede ser filtrada en el receptor. Con la configuración de antena monoestática y biestática se intenta minimizar esta señal de interferencia.

Configuración Monoestática y Biestática.

Hay dos configuraciones de antena para la lectora RFID, que tienen como objetivo mantener la señal recibida proveniente de la etiqueta separada de la señal enviada por el transmisor de la lectora. Estas configuraciones se muestran en la figura 3.4.

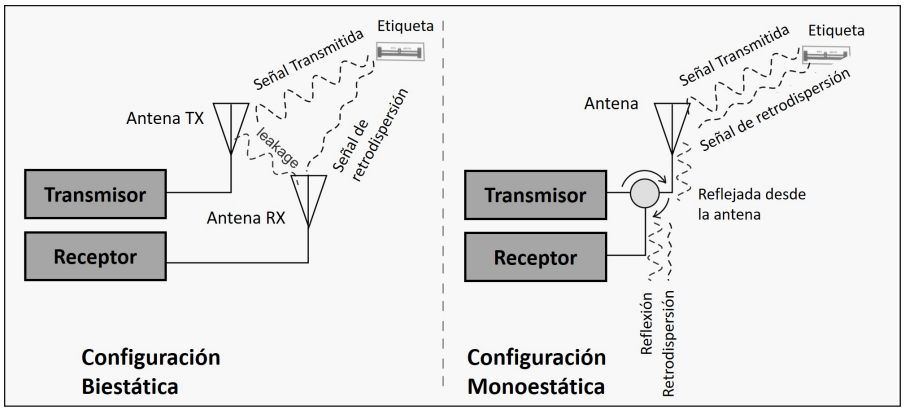


Figura 3.4: Configuración de antena biestática y monostática [6].

- **Configuración Monoestática:** En la configuración monoestática se utiliza la misma antena para la transmisión y para la recepción. Dicho sistema está diseñado para evitar que la energía presente en el *path* de transmisión hacia la antena se desvíe al receptor. Esto se logra utilizando componentes como circuladores o acopladores direccionales. A pesar de esta configuración es posible que parte de la señal transmitida sea re-inyectada a la sección de recepción.

- **Configuración Biestática:** En esta configuración se emplean dos antenas, una para la transmisión y otra para la recepción. Esta configuración puede prevenir que la señal que es transmitida por la lectora llegue al receptor, si se logra un arreglo adecuado con las antenas.

A continuación se dará una breve descripción de la arquitectura del transmisor y el receptor que componen a la interfaz de radiofrecuencia.

Transmisor

La arquitectura de transmisor mas sencilla se muestra en la figura 3.5. Para transmitir la señal OOK (por sus siglas en inglés *on-off signal*), este transmisor cuenta con un sintetizador que provee la señal portadora, un amplificador (que provee suficiente potencia a la señal de salida) y un interruptor (*switch*) que prende y apaga la señal generada por el sintetizador.

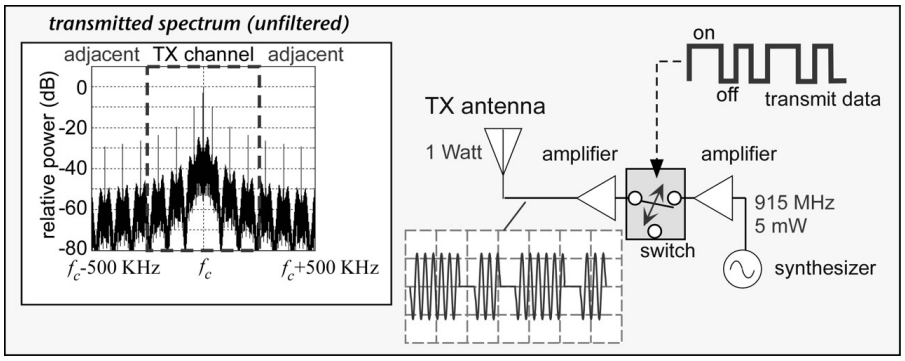


Figura 3.5: Arquitectura de transmisor simple, también se muestra el espectro de la señal transmitida (Modulación PIE sin filtrado) [6].

En este diseño la señal es prendida y apagada abruptamente (representando símbolos 1 y 0 con prendido y apagado). Símbolos con cambios abruptos normalmente usan más ancho de banda de lo necesario (que podría causar interferencia con otras lectoras que estén funcionando cerca). Un mejor diseño se muestra en la figura 3.6, en el que se emplea un atenuador variable en vez del interruptor para que los cambios de la señal entre encendido y apagado sean más suaves. Otra opción de diseño para modular la señal podría ser ajustar el voltaje o la corriente disponible para el amplificador de salida [6], en lugar de usar el atenuador variable.

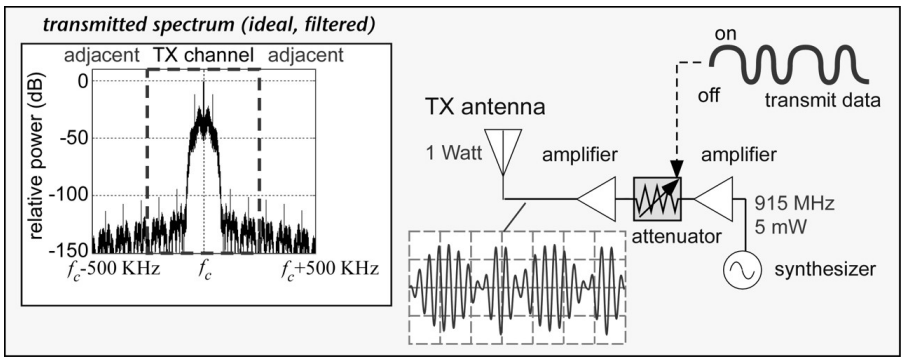


Figura 3.6: Arquitectura de transmisor con atenuador variable y símbolos filtrados (suavizados), también se muestra el espectro de la señal transmitida (Modulación PIE y filtrado) [6].

Una arquitectura más compleja es el modulador I/Q que se muestra en la figura 3.7. Esta arquitectura cuenta con dos mezcladores, uno para para la señal en fase y otro para la señal en cuadratura. Con este modulador además de generar señales ASK, se pueden generar otro tipo de modulaciones más sofisticadas (PSK, QAM,...), entre ellas modulación de banda lateral única, abreviado como *SSB* por sus siglas en inglés *Single Side Band*, con la cual se puede reducir el ancho espectral utilizado en la transmisión de datos, sin comprometer la complejidad de recepción y demodulación (ya que la etiqueta pasiva cuenta con un receptor sencillo que debe ser capaz de demodular la señal transmitida por la lectora).

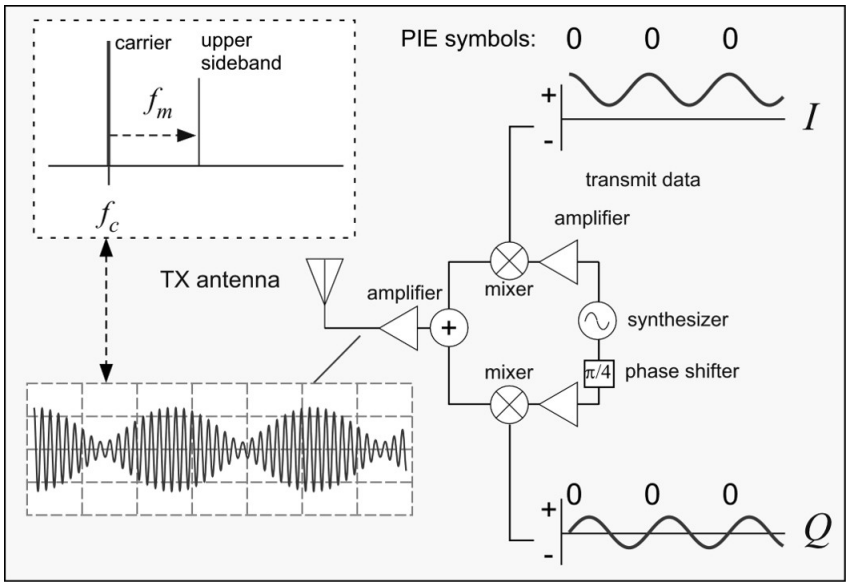


Figura 3.7: Arquitectura de modulador I/Q para *SSB* con señal de salida y espectro [6].

Receptor

La arquitectura básica del receptor de lectora RFID se muestra en la figura 3.8. Es un demodulador I/Q de conversión directa. La señal recibida se divide en dos ramas: en una rama la señal recibida es mezclada con la señal generada por el oscilador local (*LO* por sus siglas en inglés *Local Oscillator*), en la otra rama, la señal recibida es mezclada con la señal del *LO* desfasada 90°. La señal resultante se filtra para eliminar la portadora y los armónicos, dejando una señal de baja frecuencia que contiene la respuesta de la etiqueta [6].

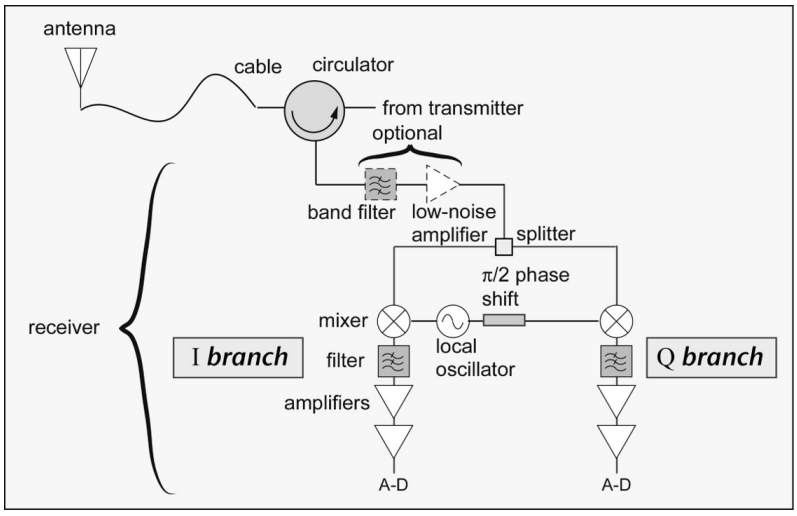


Figura 3.8: Receptor I/Q [6].

3.2.2. Sección de Control

La sección de control de la lectora de RFID realiza procesamiento de las señales obtenidas por la interfaz de RF y controla que todas las partes de la lectora trabajen juntas. Algunas de las funciones de la sección de control son las siguientes:

- Comunicación con la aplicación de software de la computadora huesped y ejecución de comandos
- Control de la interfaz de radio frecuencia
- Comunicación en tiempo real con las etiquetas
- Codificación y decodificación de señales

En algunos sistemas, la sección de control también ejecuta algoritmos anticolisión, encriptación/desencriptación de datos y autenticación entre la lectora y las etiquetas [22].

La sección de control generalmente se compone de un microprocesador, bloque de memoria, convertidores analógico-digital/digital-analógico, y un bloque para la comunicación con el software de aplicación (interfaz de comunicación) [23]. Algunas de las tecnologías empleadas para la interfaz de comunicación con el software de aplicación son Ethernet, RS232, RS485, USB, entre otros. Para el procesamiento de señal han sido usadas tecnologías como ASIC, FPGA, DSP, etc. La figura 3.9 muestra un diagrama de bloques de la sección de control.

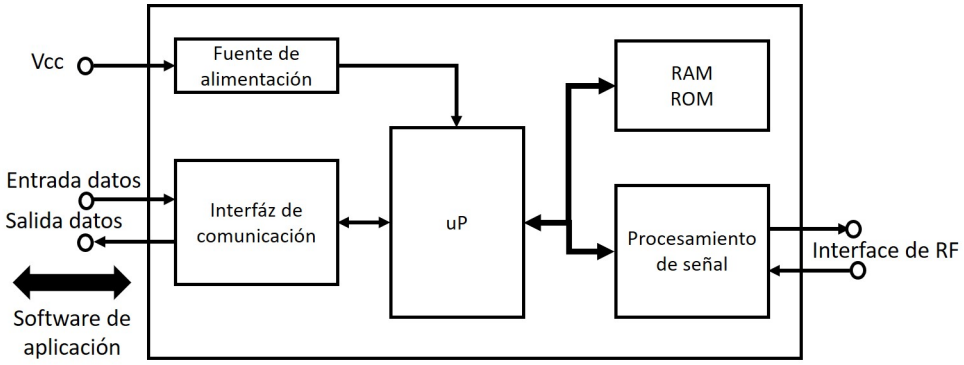


Figura 3.9: Diagrama de bloques de sección de control [23]

Capítulo 4

Emulación de Lectora RFID ISO 18000-6C

En este capítulo se presentará los principios básicos de la herramienta para la emulación de lectora RFID ISO 18000-6C. Para la emulación de la lectora, se aprovechan las ventajas que ofrecen los sistemas de radio definido por software (SDR por sus sus siglas en inglés: Software Defined Radio) al permitir configurar componentes del transceptor por medio de software y las herramientas que proporciona el software LabView para la adquisición y procesamiento de señales.

La radio definida por software es un sistema de transmisión de radio donde ciertas funciones son realizadas por *hardware* dedicado, controlable por *software*, y donde otras funciones, como el procesamiento digital de señales son implementadas por medio de *software* [24]. En la figura 4.1 se muestra un diagrama de bloques de un transceptor SDR. El emulador RFID de este trabajo fue construido según esa arquitectura. Para la sección de RF y la sección de conversión se utilizó equipo de *hardware* para SDR de la marca National Instruments. Se utilizó una computadora personal y el entorno de desarrollo de software LabView para la sección de procesamiento y también para controlar y configurar el equipo de *hardware*.

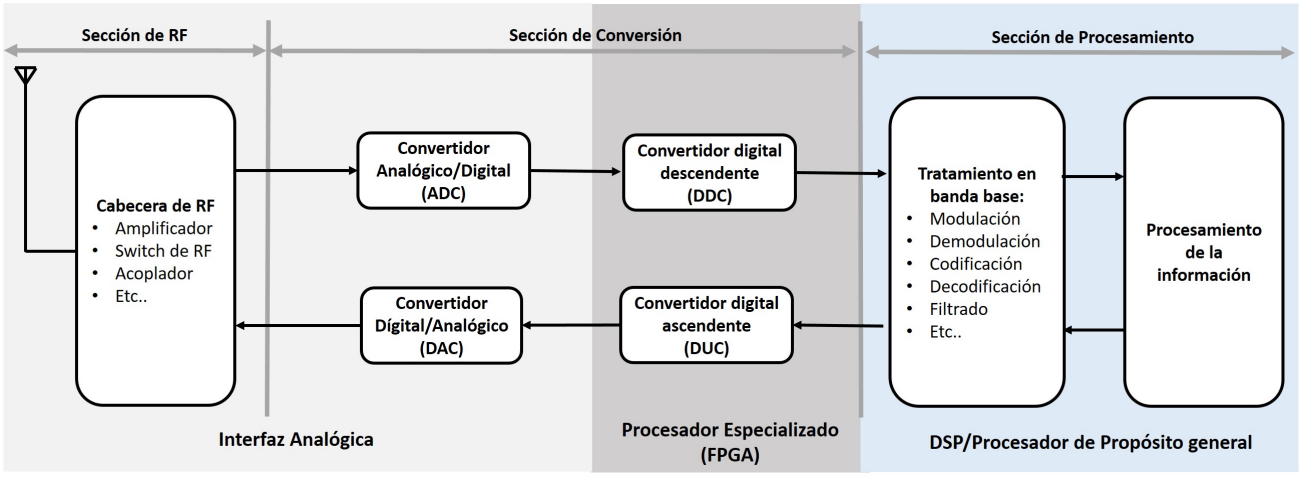


Figura 4.1: Diagrama de bloques de un transceptor SDR [25]

La figura 4.2 muestra un diagrama de bloques del *software* para la sección de procesamiento

y configuración/control del *hardware* del emulador RFID. Este se compone de un transmisor y un receptor. El transmisor es el encargado de generar y enviar los comandos que guían la comunicación con las etiquetas. El receptor debe identificar la señales transmitidas por las etiquetas y obtener información de estas. A continuación se exponen cada una de las partes del emulador.

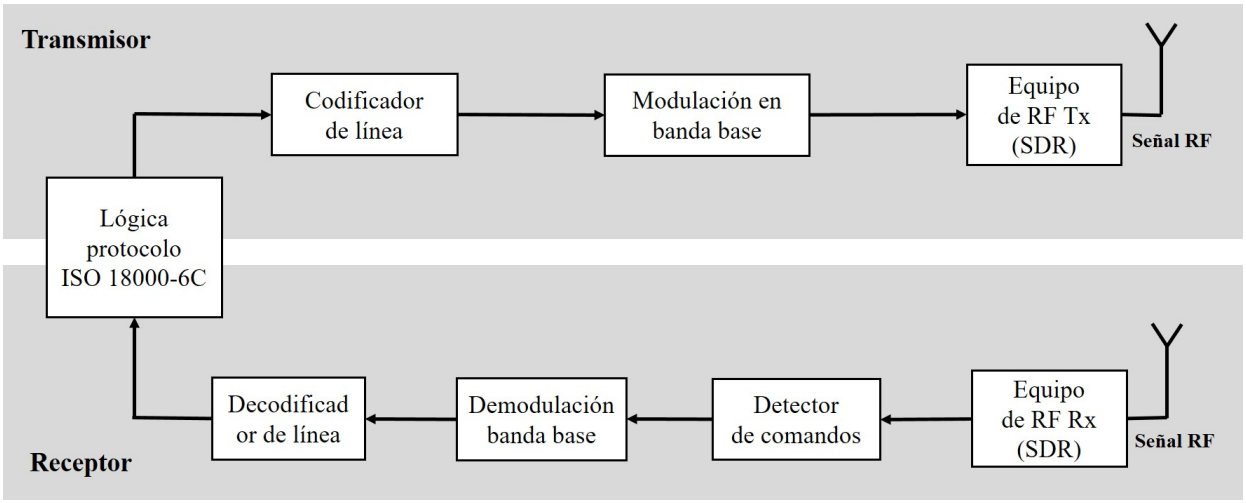


Figura 4.2: Componentes del emulador de lectora RFID

4.1. Transmisor

4.1.1. Lógica de Protocolo ISO 18000-6C

En este trabajo, el emulador se concentrará únicamente en los procesos de *Inventory* y *Select*, ya que son los responsables del desempeño del protocolo. El bloque de lógica del protocolo forma parte tanto transmisor como receptor. En el transmisor, se generan los comandos con los cuales la lectora guía la comunicación con las etiquetas, para que en bloques siguientes, sean codificados, modulados e insertados a una cola que alimenta al equipo de *hardware* SDR para ser transmitidos. En el receptor, se recibe la información emitida por las etiquetas, y decide que comando se transmitirá a continuación.

En la figura 4.3 se muestra el diagrama de flujo del bloque de lógica de protocolo elaborado para el emulador. El programa comienza con la generación de una señal de onda continua $T_c > 1500\mu$s, según lo indicado por el protocolo (tiempo de estabilización T_s).

Siguiendo el diagrama, se comienza el proceso de selección al generar el comando *Select* seguido de una señal de onda continua de duración T_4. Cabe recalcar que se trata de un sistema RFID pasivo, por lo que la lectora debe en todo momento transmitir una señal que brinde a las etiquetas la suficiente potencia para mantenerlas en funcionamiento y para responder a las peticiones de la lectora por medio de señales de retrodispersión. Después de la transmisión de la señal de T_4 comienza el proceso de inventario con la generación del comando *Query*, seguido de la señal de onda continua de duración $T_{RN16} = T_1 + RN16 + T_2$, tiempo en el que se espera que la etiqueta responda con el RN16. El receptor es el encargado de obtener el RN16 de la señal de retrodispersión de la etiqueta, que ha recibido por medio del equipo de *hardware* SDR. Si el

receptor logró obtener correctamente el RN16 transmitido por la etiqueta en un tiempo menor o igual a T_e, que en teoría debería ser igual a T_{RN16}, entonces se genera el comando ACK y a continuación la señal de onda continua de duración $T_{EPC} = T_1 + PC + EPC + CRC16 + T_2$, intervalo de tiempo en que se espera que la etiqueta envíe su número de identificación, el cual será almacenado. Si el receptor no logra obtener el RN16 en el tiempo indicado, se genera nuevamente un comando *Query*, comenzando un nuevo proceso de inventario.

Según el protocolo, en caso de que la lectora no logré obtener el RN16 de la etiqueta en el tiempo indicado, la lectora debe descartar esta información, pero para fines de pruebas y experimentación, el emulador no descarta este RN16, en vez de esto , genera comandos *Query* hasta que obtiene el RN16 en el que había estado trabajando, y forma el comando *ACK* con esa información.

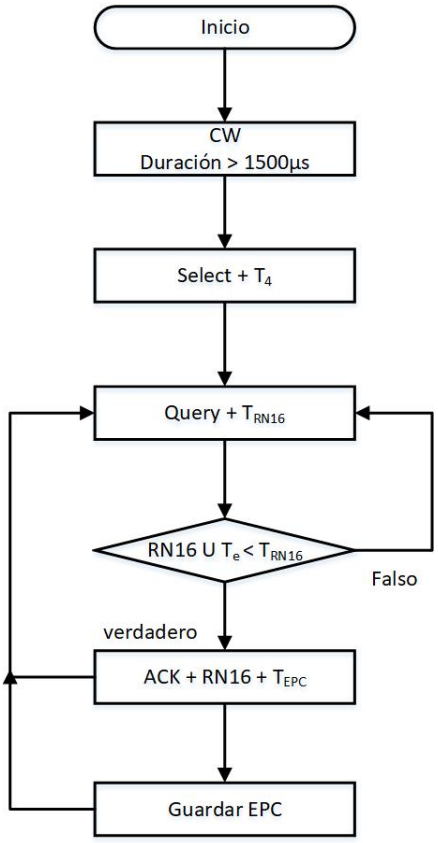


Figura 4.3: Diagrama de flujo bloque de lógica de protocolo ISO 18000-6C

El bloque de lógica del protocolo fue construido en *LabView* con una arquitectura de máquina de estados en colas (en inglés *Queued State Machine*), el diagrama de bloques de *Labview* se muestra en la figura 4.4.

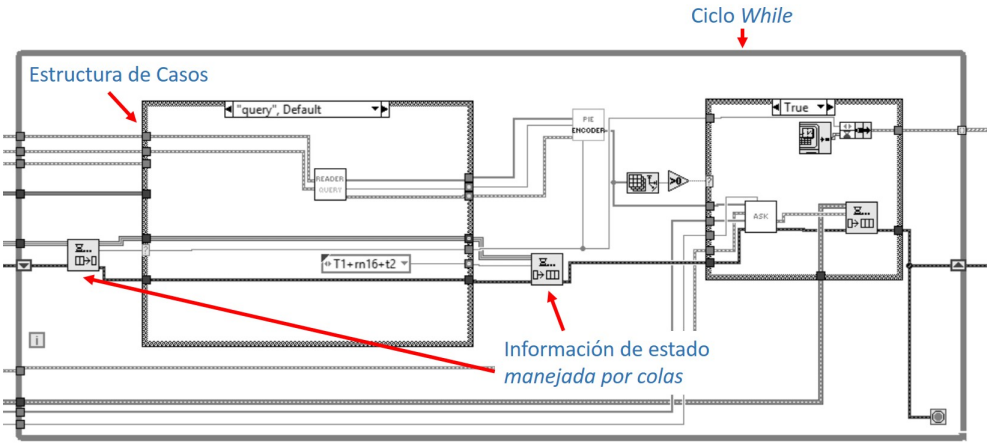


Figura 4.4: Diagrama de bloques *LabView* de máquina de estados de protocolo ISO 18000-6C

La máquina de estados tiene los siguientes componentes:

- **Ciclo *While*:** En cada iteración se ejecuta un estado.
- **Estructura de casos:** Ejecuta las acciones para cada estado de la lectora. Cada caso/subdiagrama representa un comando del protocolo ISO 18000-6C.
- **Colas:** Determina el caso/subdiagrama que se va a ejecutar.

Generación de Comandos de la lectora

Se tiene un subVI para cada comando del protocolo ISO 18000-6C. Estos programas se encargan de formar el arreglo binario que representa al comando. Por ejemplo, en la figura 4.5 se muestra el panel frontal del VI del comando *Query*. Los programas realizados en LabView referentes a la generación de comandos de la lectora y de los símbolos de la lectora están basados en el ejemplo Demo-RFID públicado en la página de National Instruments [26].

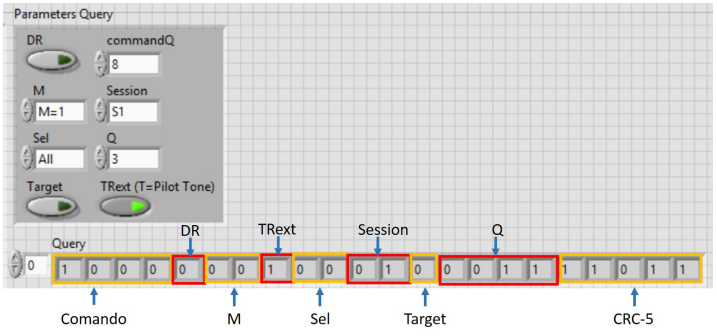


Figura 4.5: Panel frontal comando *Query*

Generación de símbolos de la lectora

Para generar los comandos, se genera un arreglo de datos binarios. La generación del arreglo se hace de acuerdo al formato especificado por el protocolo ISO 18000-6C. La longitud del arreglo dictará la duración de la señal, y se calcula con la siguiente operación:

$$T_{Comando} = L \times R_{Lectora}. \tag{4.1.1}$$

Donde L es la longitud del arreglo de datos binarios (ya codificados por PIE) y $R_{Lectora}$ es la tasa de transmisión de símbolos. La generación de las señales de onda continua como tiempo entre comandos T_4, tiempo de respuesta de etiqueta T_{RN16} y T_{EPC} se generan como arreglos de datos binarios "1" y de longitud $L = \frac{T}{R_{Lectora}}$. Estos arreglos pasarán directamente al bloque de codificación ASK teniendo como resultado la transmisión de señales de onda continua.

4.1.2. Codificador de Línea

El emulador realiza codificación PIE. La manera en que se realiza la codificación es al generar arreglos de datos binarios con el código que representa el valor "0' y "1". Debido a que la codificación PIE especifica un pulso de encendido y uno de apagado para la señal que representa el valor "0", el arreglo de datos binarios estará dado por $smbolo0 = [1, 0]$. El código que representa el valor lógico "1" se forma con un arreglo de datos binario de tamaño $L_{S1} = L_{S0} * Factor$, donde L_{S0} es el tamaño del arreglo $smbolo0$ y $Factor$ es un factor de escalamiento que puede ser 1.5 o 2, lo cual está dentro de las especificaciones del protocolo. El arreglo $smbolo1$ entonces tendrá $L_{S1} - 1$ elementos con valor 1 y su último elemento de valor 0.

Cada comando formado por el programa de Lógica del Protocolo se codifica al sustituir los elementos con valor "1" por el arreglo $smbolo1$ y los elementos con valor "0" por el $smbolo0$. Además en este bloque se agrega a los comandos ya codificados el encabezado del paquete, es decir, los símbolos de *Frame-sync* y *Preamble*. En la figura 4.6 se muestra un diagrama del proceso de codificación, teniendo que $Factor = 1{,}5$.

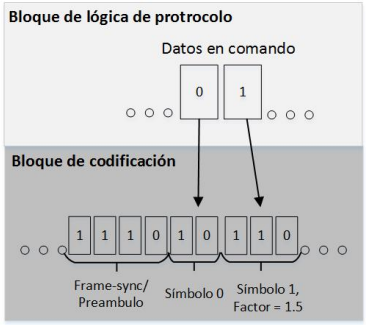


Figura 4.6: Bloque de codificación PIE

4.1.3. Modulación

Señal pasa-baja y Señal pasa-banda

En una modulación, una señal de información (señal moduladora o señal banda base) modificará algún parámetro de una señal portadora como la amplitud, frecuencia o fase. La señal modulada (también conocida como señal pasa-banda) puede representarse de la siguiente manera:

$$s(t) = s_I(t)\cos(2\pi f_c t + \phi_0) + s_Q(t)\sin(2\pi f_c t + \phi_0) \tag{4.1.2}$$

Dónde $S_I(t)$ es la componente en fase, y $S_Q(t)$ es la componente en cuadratura.

También se puede expresar de la siguiente manera:

$$s(t) = \Re\{u(t)e^{j(2\pi f_c t+\phi)}\} \tag{4.1.3}$$

Donde $u(t) = S_I(t) + jS_Q(t)$ es conocida como la señal banda base o pasa baja.
La envolvente de la señal $s(t)$ se define como $r_s(t) = \sqrt{S_I^2(t) + S_Q^2(t)}$ y su fase como $\theta_s(t) = \arctan\frac{S_Q}{S_I}$ [27]

Espacio de señales o representación vectorial de señales

La señal modulada $s(t) = s_I(t)\cos(2\pi f_c t + \phi_0) + s_Q(t)\sin(2\pi f_c t + \phi_0)$ puede expresarse como se indica en la ecuación 4.1.4

$$s(t) = s_{i1}\phi_1(t) + s_{i2}\phi_2(t) \tag{4.1.4}$$

donde $\phi_1(t) = g(t)\cos(2\pi f_c t + \phi_0)$ y $\phi_2(t) = -g(t)\sin(2\pi f_c t + \phi_0)$ son las funciones base, $g(t)$ es un función conformadora de pulso y $S_I = s_{i1}g(t)$, $S_Q = s_{i2}g(t)$. La representación vectorial de la señal $s(t)$ está dada por:

$$s_i = [s_{i1}, s_{i2}] \tag{4.1.5}$$

Modulación en RFID

Para la emulación del protocolo se utilizará únicamente modulación ASK binaria, dejando como trabajo a futuro la introducción de otras variantes ASK. En BASK (del inglés,*Binary Amplitude Shift Keying*), la amplitud de la señal portadora puede tomar dos niveles, por lo general apagado (amplitud 0) o encendido. La condición de encendido típicamente representa un "1" binario, y la condición de apagado un "0" binario.
La señal BASK puede ser definida por la siguiente ecuación:

$$s_i(t) = A_i g(t)cos(2\pi f_c t) \qquad , \qquad 0 \leq t \leq T \tag{4.1.6}$$

donde $A_i = 0, 1$, $g(t)$ es la señal conformadora de pulso, f_c es la frecuencia de la portadora, y T es la duración del bit. La función base para la señal ASK está dada por 4.1.7

$$\phi(t) = \sqrt{\frac{2}{\varepsilon_g}}g(t)\cos(2\pi f_c t) \tag{4.1.7}$$

Usando 4.1.7 se tiene que

$$s(t)_i = A\sqrt{\frac{\varepsilon_g}{2}}\phi(t) \tag{4.1.8}$$

El diagrama de bloques del modulador para el emulador RFID se muestra en la figura 4.7.La modulación se realiza en banda base por medio de software. Para modular los datos binarios se utilizó el kit de herramientas de modulación de Labview (*Modulation Toolkit*). En el cuadro 4.1.1 se muestran algunos de los VIs del kit de modulación utilizados, y se da una breve descripción de la tarea que realizan.

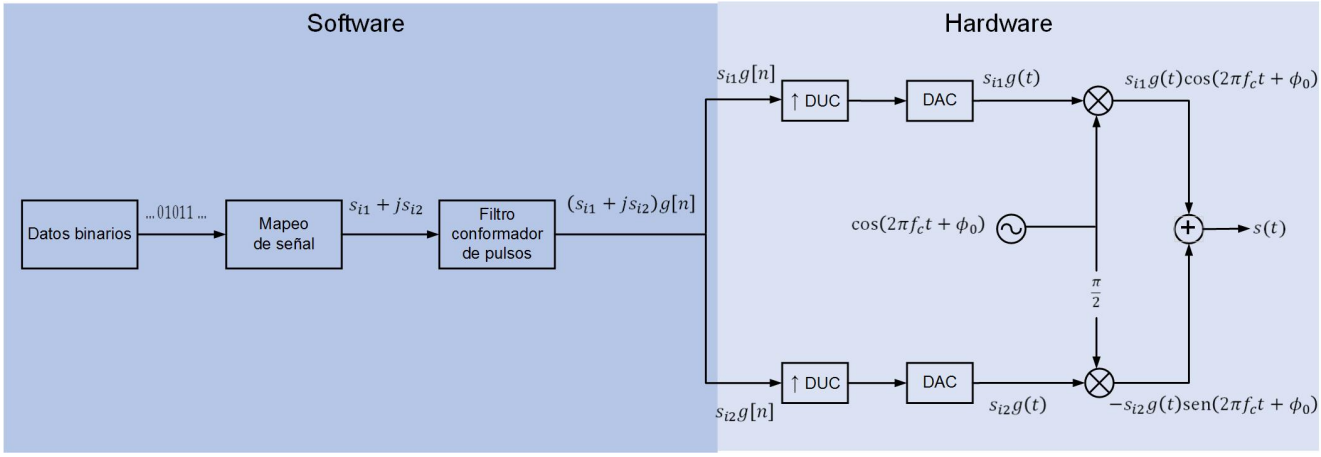


Figura 4.7: Diagrama de bloques del proceso de modulación.

VI	Descripción
MT Generate System Parameters	Calcula los parámetros para usar con el VI de modulación\demodulación. La salida de este VI es un cluster que contiene información del número de muestras por símbolo y un arreglo ordenado que asigna cada símbolo booleano a un nivel de amplitud deseado.
MT Generate Filter Coefficients	Calcula los coeficientes para filtros de conformación de impulsos y filtro adaptado (matched filter) para usar con VI de modulación digital y VI de demodulación.
MT Modulate ASK	Recibe una secuencia de datos binarios, realiza modulación ASK. A la salida de este VI se obtiene la señal pasa banda compleja modulada. Este VI mapea una secuencia de datos binarios a símbolos de ASK, interpola por zero-insertion, y aplica el filtro conformador de pulsos a la señal I/Q interpolada.

Cuadro 4.1.1: VI de *Modulation Kit* utilizados para construir el Emulador RFID [28]

4.1.4. Configuración de la temporización del enlace

Al ejecutar el código, lo primero que aparece es el panel frontal del VI *SetReaderParam* con el que se configuran todos los parámetros para la temporización del enlace. Los cálculos se realizan

siguiendo el estándar ISO-180006C (vease temporización del enlace).

Como se observa en la figura 4.8, los parámetros a configurar son:

- Enlace de subida:
  - Factor del símbolo 1 (en relación con el símbolo 0).
  - Duración del Tari en segundos. Es posible configurar valores estándar de Tari de 6.25μ, 12.5μ y 25μ.
  - TRcal puede seleccionarse valores discretos dentro del rango de 1.1 a 3 RTcal.
  - Los parámetros RTcal, PW y tasa de símbolo son calculados por el emulador.
- Enlace de bajada
  - Se puede seleccionar distintas configuraciónes para el comando de Query, hay que recordar que algunos parámetros del Query afectan la tasa de transmisión del enlace de subida, como son el DR y el parámetro M (véase sección del protocolo). Hasta ahora el emulador solo es capáz de procesar señales de la etiqueta con modulación FM0.
  - Los parámetros TRcal y BLF son calculados por el emulador.

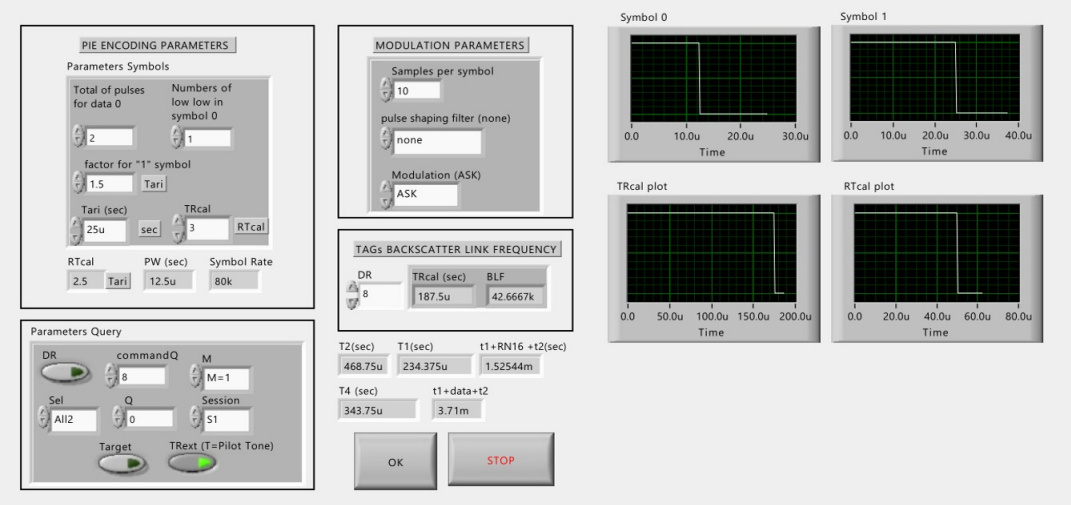


Figura 4.8: Panel frontal VI SetReaderParam

La tasa de muestreo de transmisión de la señal banda base (IQ rate) es calculada por medio de la ecuación 4.1.9 y es proporcionada al VI de configuración del equipo de RF.

$$\begin{aligned} IQ &= R_L \times N_{samples} \\ R_{LTransmisin} &= \frac{1}{0{,}5Tari} = \frac{1}{PW} \\ R_{LRecepcin} &= BLF \times M \times 2 \end{aligned} \tag{4.1.9}$$

Donde $R_{LTransimisn}$, $R_{LRecepcin}$ es igual a la tasa de transmisión de símbolos de la lectora, PW es el ancho de pulso (duración de símbolo de la lectora), M es el parámetro del comando $Query$ de ciclos por símbolo, $N_{samples}$ es el número de muestras por símbolo de la lectora.

4.2. Equipo de RF

El código del emulador se realizó con la idea de poder utilizar los bloques de código con diferente *hardware*, o incluso pudiera ser re-utilizado para simulación de sistemas RFID con LabView.

Para este trabajo, tanto para la transmisión como para la recepción de señales se utilizó el equipo NI-USRP 2920, antena de transmisión VERT 400 y una computadora portátil que se comunica con el equipo por medio de conexión gigabit ethernet.

Figura 4.9: USRP 2920 de National Instruments

El NI-USRP 2920 es un transceptor de radio definido por *software* basado en FPGA. En la figura 4.10 se muestra la arquitectura del USRP 2920. El receptor comienza con un *front-end* altamente sensible capaz de recibir señales muy pequeñas y digitalizarlas usando conversión descendente directa a señales en banda base (en fase (I) y cuadratura (Q)). La conversión descendente es seguida por una conversión analógico-digital y un DDC (del inglés *Digital Down Converter*) es el encargado de reducir la taza de muestreo y empaquetar las señales I y Q para su transmisión a la computadora *host* usando una conexión Gigabit Ethernet. Para la transmisión, la señal en banda base se genera en la computadora *Host* y se transfiere al equipo USRP por medio de la conexión Ethernet. El DUC (del inglés *Digital Up Converter*) prepara la señal para la conversión analógico-dígital. La señal analógica resultante es entonces mezclada para producir la señal de RF deseada, la cual es amplificada y transmitida [29].

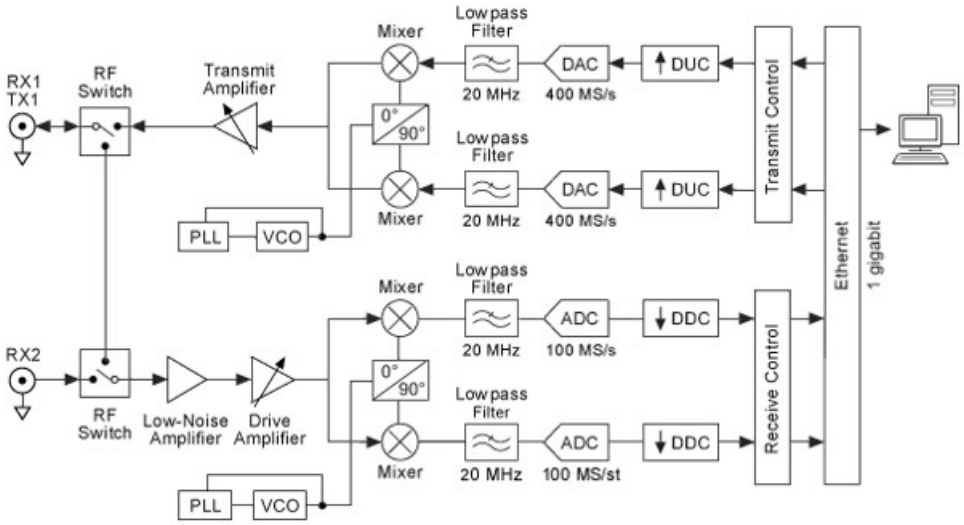


Figura 4.10: Diagrama de bloques del USRP 2920 de National Instruments [29]

En la figura 4.11 se muestra el panel frontal del VI de configuración del equipo de RF. Para el emulador se consideró usar dos equipos USRP, uno para la transmisión y uno para la recepción. Tanto para la configuración del *hardware* como para la transmisión de datos I/Q en banda base se utilizó el *driver* de National Instruments NI-USRP. En el emulador es posible configurar por medio del software los siguientes parámetros:

- IP del equipo USRP transmisor
- Frecuencia de portadora
- Ganancia
- Tiempo de adquisición

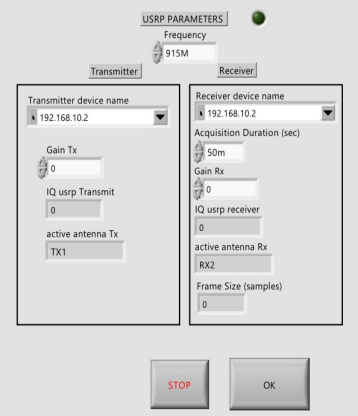


Figura 4.11: Panel frontal del VI para la configuración del equipo de RF

La configuración de la antena de transmisión no puede ser modificada por medio del panel frontal, y se tiene como configuración por defecto TX1 y RX2. La tasa de muestreo de transmisión de la señal banda base (IQ rate) es calculada por medio de la ecuación 4.1.9.

4.3. Receptor

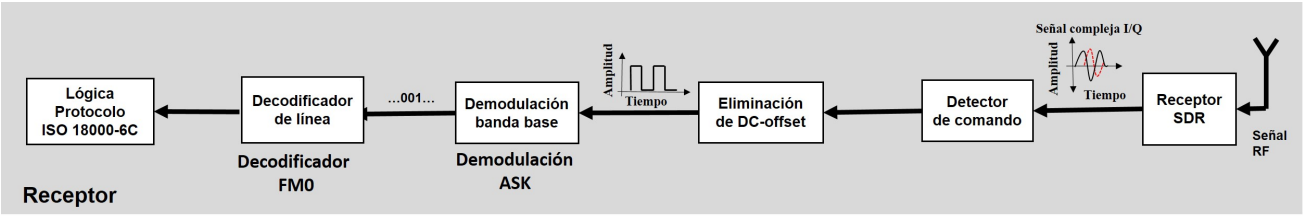


Figura 4.12: Diagrama de bloques del receptor del Emulador RFID

Teniendo las muestras de la señal recibida en su representación pasa baja con componentes I y Q se debe de realizar los siguientes procedimientos para obtener información de las etiquetas:

- Las etiquetas envían información a la lectora únicamente durante ciertos instantes de tiempo, sin embargo, la configuración utilizada en el equipo de RF es la de recepción continua. Se seleccionó esta configuración ya que al detener y reactivar la adquisición del USRP se produce un retraso que podría ocasionar la pérdida de información. Debido a esto, se tendrá en el receptor tanto la señal que transmitió la lectora, como la respuesta de las etiquetas. El emulador debe ser capaz de distinguir el intervalo de tiempo en el que responde la etiqueta.

- La etiqueta transmite información a la lectora por medio de técnicas de retrodispersión, por lo que la señal que transporta la información útil tendrá un off- set, el cual debe ser removido.

- El calculo de este offset se dificulta al variar la potencia de la señal recibida, ya sea por etiquetas móviles o debido a que el USRP no es un equipo calibrado por lo que no siempre mantiene una potencia de transmisión constante.

- Después de detectar la señal de la respuesta de la etiqueta, debe ser demodulada y decodificada para obtener información de ella.

Cabe mencionar que la tarea de recuperación de la información de las etiquetas se dificulta al imponer la estricta temporización del enlace dictada por el protocolo ISO 18000-6C, ya que todo este proceso debe ser realizado en un tiempo menor a T_2. En la figura 4.13 se muestra un diagrama de bloques de las etapas para la demodulación de la señal de las etiquetas y se detallan cada una de ellas a continuación.

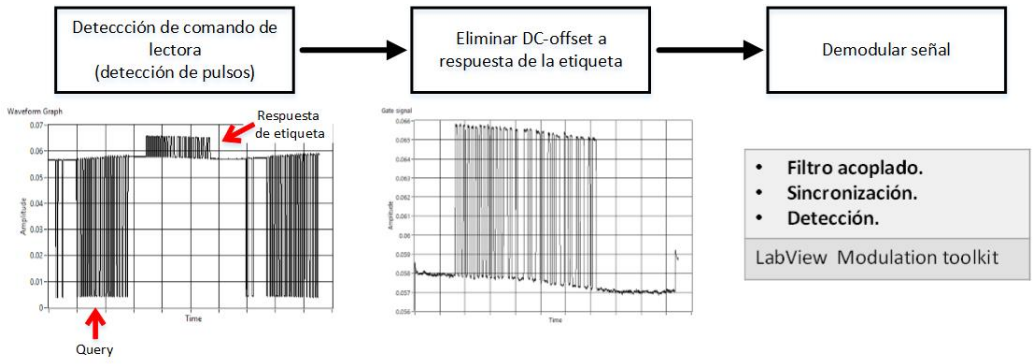


Figura 4.13: Proceso para la demodulación de la señal de etiquetas.

4.3.1. Detección de comando de lectora:

El proceso de detección de comandos se realiza en tres pasos, los cuales se detallan a continuación y también se ilustran en la figura 4.14.

1. Detector de pulsos: Todas las señales con las que se trabajará están moduladas en amplitud, por lo que la información transmitida se encontrará en la envolvente de la señal recibida. La envolvente de la señal presentará pulsos "altos" y pulsos "bajos", con los cuales se representan los símbolos de la lectora. Para identificar estos pulsos, se establece un

valor umbral, si el valor de la siguiente muestra está por arriba de ese umbral entonces, se tiene un pulso alto. Si el valor de la muestra está por debajo de ese umbral, se tendrá un pulso bajo. El umbral se obtiene al realizar un promedio móvil de las muestras de la señal recibida.

2. Detección de símbolo de lectora: La identificación de los símbolos de la lectora se realiza al obtener el número de muestras que forman a los pulsos identificados (tamaño del pulso en muestras), ya que cada símbolo de la lectora tiene un tamaño específico (PW, pulso alto de $RTcal$, $TRcal$, símbolo 1). El tamaño del símbolo en muestras se calcula mediante la expresión 4.3.1

$$S_{simbolo} = T \times IQ_{Receptor} \tag{4.3.1}$$

Dónde T es la duración del símbolo en segundos, y $IQ_{Receptor}$ es la tasa de muestreo de la señal pasa bajo.

3. Identificación de comandos: Cada comando comienza con símbolo *preambulo* o *Frame-sync*, seguido de un código de comando. De esta manera se puede identificar el inicio del comando que se transmitió y se puede calcular el fin del mismo al contabilizar el número de símbolos que continúan al código del comando.

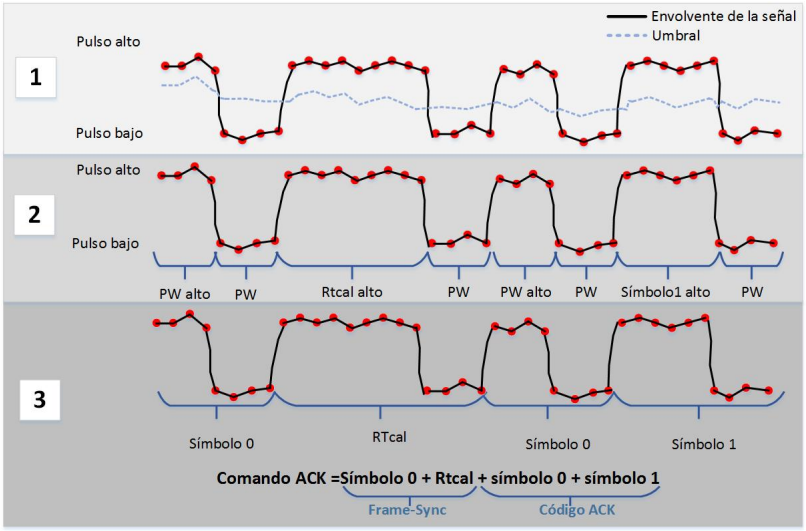


Figura 4.14: Esquema del proceso de detección de pulsos

Hay que recordar que la finalidad de este proceso es identificar la porción de la señal en la que la etiqueta transmite información, entonces teniendo detectado el fin de los comandos, se puede separar la porción de tiempo en que la etiqueta responde a la lectora, y es únicamente esta porción de la señal la que pasa al siguiente bloque para su procesamiento.

4.3.2. Eliminación de DC-offset

Antes de pasar al bloque de demodulación, la señal de retrodispersión de la etiqueta se pasa por un bloque de eliminación de DC-offset. Cómo se mencionó en el capítulo anterior,

el protocolo especifica la temporización del enlace, en la que se establece los tiempos que se muestran en la figura 4.15. Después de que se ha emitido un comando *Query* o *ACK*, la etiqueta debe de esperar un tiempo T_1 para responder a la lectora. El DC-offset se calcula al obtener la media de la envolvente de la señal durante el tiempo T_1.

$$dc_{offset} = \frac{1}{n}\sum_{i=0}^{n-1} r_s[i] \tag{4.3.2}$$

donde: $r_s[i] = \sqrt{S_I^2[i] + S_Q^2[i]}$ es la envolvente de la señal recibida $n = T1 * IQ_{Receptor}$ es el tiempo T_1 en muestras.

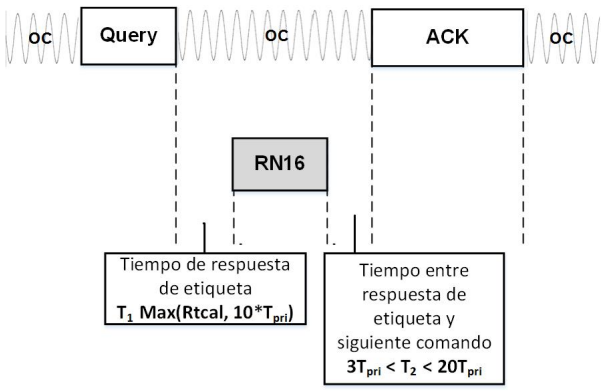


Figura 4.15: Temporización del enlace. La etiqueta debe de responder al comando de la lectora después de transcurrido un tiempo T_1

4.3.3. Demodulación

El diagrama de bloques simplificado del demodulador se muestra en la figura 4.16.La demodulación se realiza en banda base por medio de software. Para este proceso se utilizó el kit de herramientas de modulación de Labview (*Modulation Toolkit*). En la cuadro 4.3.2 se muestran algunos de los VIs del kit de modulación utilizados, y se da una breve descripción de la tarea que realizan.

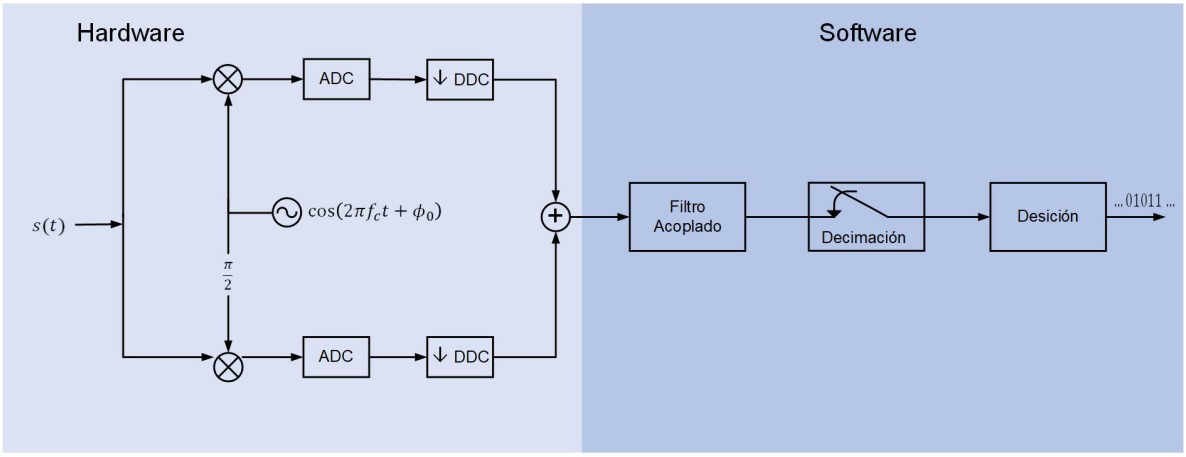


Figura 4.16: Diagrama de bloques del proceso de demodulación.

VI	Descripción
MT Resample	A la entrada de este VI se tiene una forma de onda compleja. A la salida de este VI se tiene la señal remuestreada y/o realineada en función de los parámetros de entrada especificados
MT Detect ASK	Demodula la señal compleja modulada ASK y devuelve los bits demodulados. A este VI se le proporciona el preambulo de FM0 como secuencia para la sincronización del flujo de bits.

Cuadro 4.3.2: VI de *Modulation Kit* utilizados para construir el Emulador RFID [28]

4.3.4. Decodificador de Línea

Cómo se muestra en la figura 4.17, a la entrada del decodificador se tienen un arreglo de datos binarios, los cuales fueron obtenidos de la señal recibida por el USRP y el proceso de demodulación. Estos datos están códificados con FM0. A la salida del decodificador, se tiene un arreglo de datos binarios decodificados.

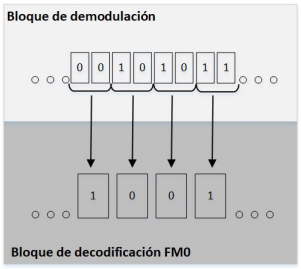


Figura 4.17: Decodificación FM0

Para realizar la decodificación y enviar la información de la etiqueta al VI de lógica de protocolo se escribieron dos VI, que se describen a continuación:

- **FM0 Decode:** A la entrada de este VI se tiene un arreglo de datos binarios obtenido por el proceso de demodulación. Para realizar la decodificación, el VI FM0 Decode realiza dos procesos:

    1. Se obtiene una cadena de datos enteros como se muestra en la figura 4.18, donde cada elemento representa un estado y puede tomar los valores de 0, 1, 2 y 3. El 0 y el 1 representan los valores binarios decodificados. El 2 representa que se está esperando el siguiente bit para poder identificar el código FM0 y el número 3 indica que se ha producido una violación (un cambio de fase debió ocurrir, pero no sucedió). Para obtener este resultado, se utiliza una máquina de estados formada por un ciclo *While*, una estructura de casos, y registros de desplazamiento para guardar el estado anterior

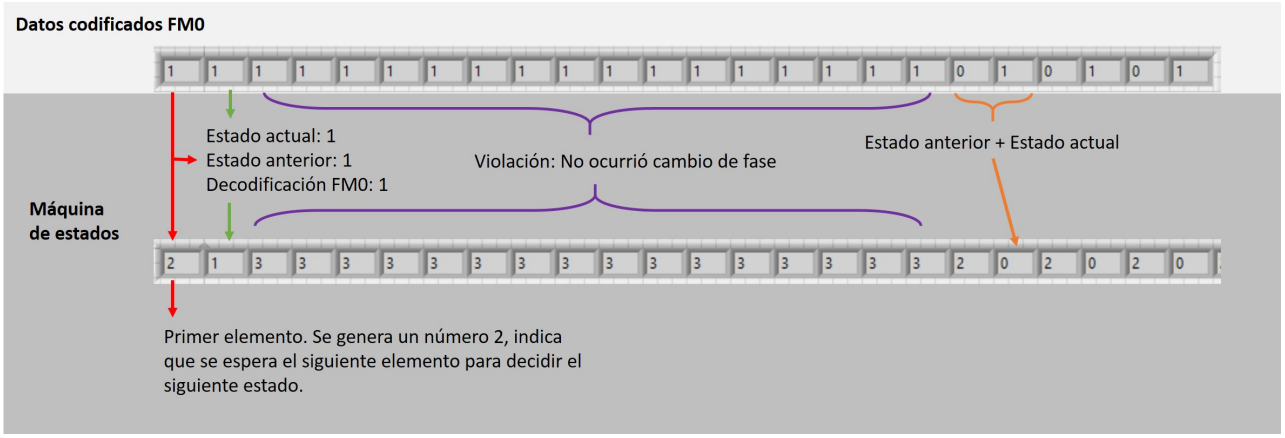


Figura 4.18: Cadena de datos obtenida por la máquina de estados de decodificación FM0

    2. Elimina los valores "2" y detecta "3" del arreglo obtenido por la máquina de estados

- **Tag Data:** Extrae la información de la etiqueta y la pone en cola para ser enviada a la lógica del protocolo.

4.3.5. Breve explicación del código

Como se mencionó anteriormente, el Emulador RFID consta de varios componentes. En Labview cada componente se dividió en estructuras de ejecución (*while loops*, o en español ciclos *while*) como se muestra en la figura 4.19. Esto es con la finalidad de mejorar el rendimiento de transmisión y recepción de datos, siguiendo las recomendaciones de National Instruments para el código en LabView [30]. La manera en que se comparten los datos entre cada *loop* es por medio de colas (queue).

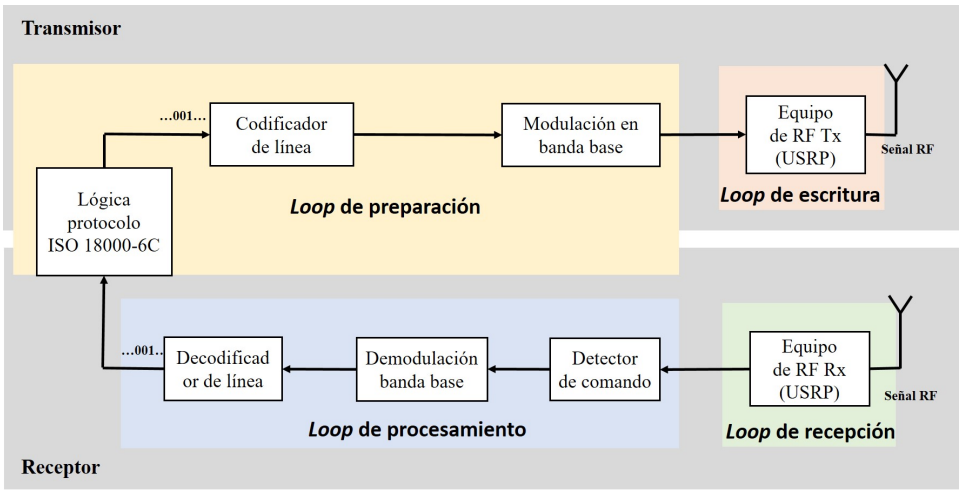


Figura 4.19: División de tareas del emulador en 4 ciclos *while*

A continuación se describen las tareas realizadas en cada *loop*

1. Loop de preparación: Máquina de estados con la que se programó el protocolo de comunicación (ISO-180006C), creación de comandos de la lectora, codificación de línea (PIE) y modulación ASK.

2. Loop de escritura de datos: Escribe los datos al USRP transmisor.

3. Loop de recepción: Obtiene los datos del USRP receptor.

4. Loop de procesamiento: Se encarga de detectar los comandos de la lectora y respuestas de etiquetas, demodular, y decodificar la información recibida. Envía la información de $RN16$ recibido al *loop* de preparación.

Capítulo 5

Resultados

5.1. Emulador + USRP

Para emular la lectora de RFID y lograr una comunicación con etiquetas comerciales, la señal de RF que el USRP transmite debe energizar a las etiquetas. Esto presenta un desafío en la implementación del transmisor lectora RFID, ya que los equipos USRP 2920 no son dispositivos calibrados, por lo que no es posible configurar una potencia de transmisión especifica. La potencia transmitida varía de equipo en equipo y depende tanto de la banda de frecuencia de operación como de la configuración que se haga al parámetro de ganancia del amplificador del USRP, como se muestra en el cuadro 5.1.1

Potencia de transmisión máx	50 MHz -1.2 GHz.....50 mW to 100 mW (17 dBm to 20 dBm) 1.2 GHz-2.2 GHz.....30 mW to 70 mW (15 dBm to 18 dBm)
Rango de Ganancia	0 dB a 31 dB

Cuadro 5.1.1: Potencia de transmisión USRP-2920 [31]

Con ayuda de un equipo PXI-5646R de National Instruments se caracterizó la potencia de transmisión del equipo USRP que será usado como transmisor de la lectora RFID. El PXI-5646R es un transceptor vectorial de señales (VST por sus siglas en inglés) que combina un analizador y generador vectorial de señales de RF y banda base con FPGA programable por el usuario e interfaces digitales paralelas y seriales de alta velocidad para procesamiento y control de las señales en tiempo real [32].

Para realizar la caracterización del USRP, se conectó el puerto TX1 del USRP al puerto RFIN del equipo PXI 5646R por medio de un cable SMA-SMA. En el PXI se utilizó el programa de Labview NI-RFSA Soft Front Panel para realizar la medición.

Se configuraron distintos valores de ganancia de amplificador del USRP, para la frecuencia de operación de 915 MHz. Se seleccionó esta frecuencia debido a que es la frecuencia de operación más común para sistemas RFID. En la imagen 5.1 se muestra el espectro de potencia de la transmisión de una onda continua generado por el USRP-2920, con diferentes configuraciones de ganancia.

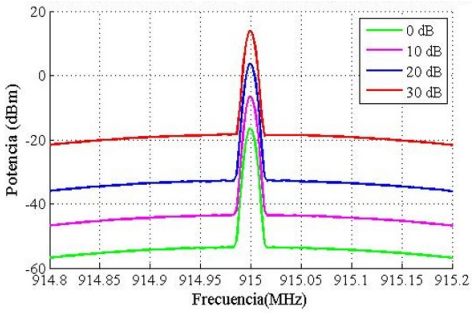


Figura 5.1: Espectro de potencia onda continua generada por USRP-2920 a 915MHz.

Se obtuvo que la potencia de transmisión para el USRP 2920 para una onda continua con frecuencia de 915 MHz, con una ganancia del amplificador del USRP de 30 dB es de 13.77 dBm. Se utilizó este dato para realizar el presupuesto del enlace. La potencia de la señal que recibe la etiqueta se calculó con la ecuación 5.1.1, donde P_{TxL} es la potencia de transmisión del equipo USRP, G_{TxL} y G_{RxE} son las ganancias de la antena de transmisión del USRP (VERT 400) y ganancia de recepción de antena de etiquetas respectivamente, λ es la longitud de onda y r es la distancias de separación entre emulador y etiqueta.

$$P_{RxE} = P_{TxL} G_{TxL} G_{RxE} \left(\frac{\lambda}{4\pi r} \right)^2 \tag{5.1.1}$$

En la gráfica se muestra la potencia recibida teórica obtenida al variar el valor de la distancia en la ecuación 5.1.1. También se muestra el umbral de recepción reportado en la hoja de datos de etiquetas pasivas Alien 9654. Para realizar el cálculo, los valores de las ganancias de antenas se obtuvieron de sus hojas de especificaciones.

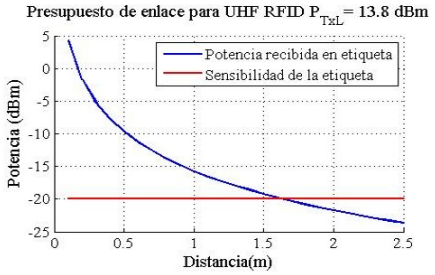


Figura 5.2: Presupuesto de enlace para emulador ISO-18000 6C

Se realizó también el presupuesto del enlace de bajada mediante la ecuación 5.1.2, donde P_{RxL} es la potencia recibida en el USRP(lectora), T_b es la pérdida de transmisión de retrodispersión [6].

$$P_{RxL} = P_{TxL} T_b G_{TxL}^2 G_{RxE}^2 \left(\frac{\lambda}{4\pi r} \right)^4 \tag{5.1.2}$$

La potencia de la señal de retrodispersión en el receptor (USRP) se muestra en la figura 5.3.

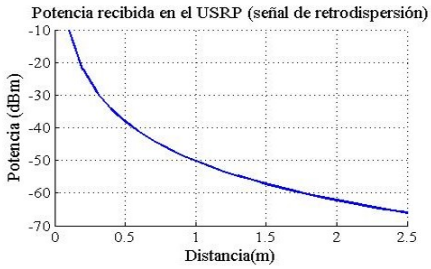


Figura 5.3: Presupuesto de enlace para la señal de retrodisperción de etiqueta

Con esto se obtuvo la distancia teórica entre el emulador RFID y la etiqueta. Para el enlace de subida, el sistema está limitado por el umbral de recepción de la etiqueta, y para el enlace de bajada, el sistema está limitado por el umbral de recepción del equipo USRP.

Para el emulador RFID, se consideraron dos equipos USRP, uno para la transmisión y otro para la recepción, en la figura 5.4 se muestra un dibujo del arreglo de los USRP, los cuales están conectados por un cable MIMO con la finalidad de que los dos USRP compartan relojes y la conexión Ethernet con la que se comunican con la computadora *host*.

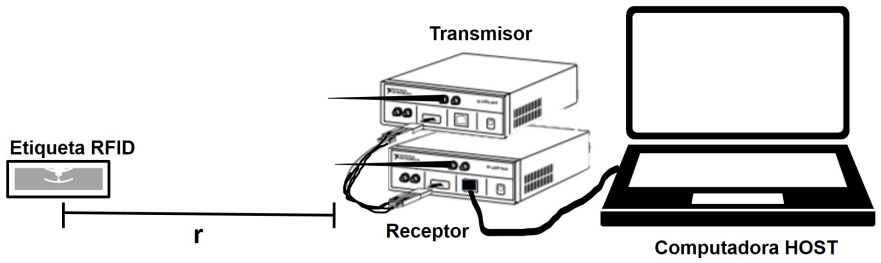


Figura 5.4: Equipo para emulador de lectora RFID

Cómo se mencionó anteriormente, la configuración de la ganancia para el equipo transmisor se estableció en $30dB \approx 13{,}77dBm$ para asegurar que las etiquetas sean energizadas. Los USRP reportan en sus hojas de especificaciones una potencia máxima a la entrada de su receptor de 0 dBm. Ya que la distancia que separa los dos USRP es mínima, para proteger al equipo de recepción fue necesario agregar a la entrada de la antena del USRP un atenuador de 30 dB. Esto impone una limitación en el sistema, ya que la señal de retrodispersión en el receptor se verá atenuada aún mas, dificultando la detección de la misma. Por esta razón, en todos los experimentos siguientes se tomaron distancias de separación entre USRP y etiqueta menores a 50 cm. Se propone como trabajo a futuro mejorar el rango de lectura del emulador con el uso de antenas direccionales.

5.1.1. Restricciones de la temporización del enlace

Para poder completar la ronda de inventario, el protocolo dice que después de recibir el último bit del RN16 de la etiqueta, la lectora debe ser capaz de responder con un comando de *Acknowledge* (ACK) en un tiempo T_2 menor a $20T_{pri}$, donde T_{pri} es el período de enlace de subida (de etiqueta => emulador).

Para el Emulador RFID, se consideró que el tiempo total T_{total} que tiene para identificar la señal de retrodispersión, demodularla, decodificar, obtener el RN16 y responder a la etiqueta completando la ronda de inventario está dado por la ecuación 5.1.3. En la figura 5.5 se muestra de manera gráfica los componentes de T_{total} y en el cuadro 5.1.2 se muestran los valores de T_1 y T_2 según el protocolo ISO-18000-6C [8].

$$T_{total} = T_1 + T_{TagRN16} + T_2 \tag{5.1.3}$$

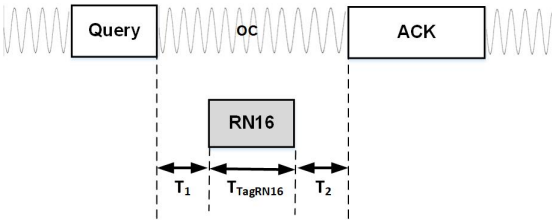


Figura 5.5: Temporización del enlace según ISO-180006C

Parámetro	Minimo	Nominal	Máximo
T_1	máx$(RT_{cal}, 10T_{pri})\times$ $(1 - \|FT\| + 2\mu s$	máx$(RT_{cal}, 10T_{pri})$	máx$(RT_{cal}, 10T_{pri})\times$ $(1 - \|FT\| + 2\mu$
T_2	$3{,}0T_{pri}$		$20{,}0T_{pri}$

Cuadro 5.1.2: Temporización del enlace segun ISO-160006C [8]

Para las pruebas realizadas, se especificó mediante el comando *Query* que la respuesta de la etiqueta estuviera acompañada del *pilot tone* (12 símbolos para sincronización), por lo que la duración de la señal de retrodispersión $T_{TagRN16}$ con el *RN16* (6 símbolos de preambulo FM0 + 16 símbolos de RN16 + 1 símbolo *dummy*) se obtiene con la ecuación 5.1.4:

$$\begin{aligned} T_{TagRN16} &= T_{PTone} + T_{RN16} \\ T_{TagRN16} &= 12 \times T_{pri} + 6 \times T_{pri} + 16 \times T_{pri} + 1 \times T_{pri} \\ T_{TagRN16} &= 35T_{pri} \end{aligned} \tag{5.1.4}$$

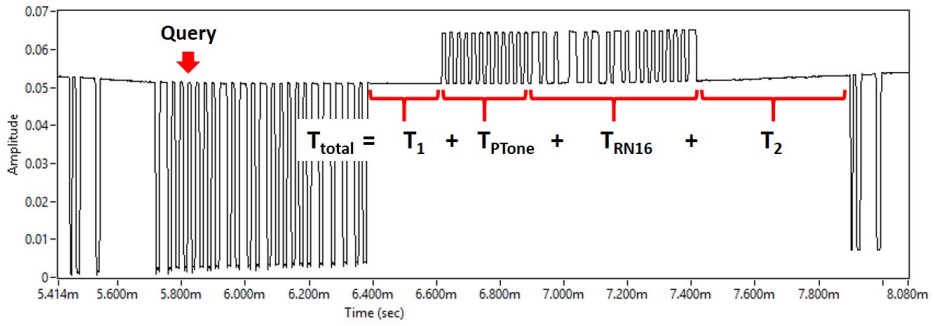


Figura 5.6: Envolvente compleja de una señal de retrodispersión de etiqueta con RN16 , se señalan cada uno de los tiempos que forman T_{total}.

5.1.2. Comunicación con etiquetas pasivas Alien 9654 y USRP

Se logró activar una etiqueta pasiva Alien 9654 usando como equipo de RF al USRP-2920 y se obtuvo el RN16 de la etiqueta. Con el emulador es posible configurar distintos parámetros del enlace, sin embargo para las pruebas se utilizaron parámetros para obtener tasas de transmisión BLF más bajas posibles, esto debido a que con tasas de transmisión altas se necesita una frecuencia de muestreo mayor, y para el procesamiento de la señal tanto en demodulación como en decodificación la memoria de la computadora host se satura, congelando la ejecución del programa.

Además a medida que aumenta la tasa de transmisión BLF, $T_{pri} = \frac{1}{BLF}$ disminuye, por lo que el tiempo $T_2 < 20T_{pri}$ (tiempo en el que se debe de responder a la etiqueta con un ACK) también disminuye.

En la figura 5.7 se muestra la envolvente compleja de la señal recibida por el emulador, donde se puede distinguir los comandos *Select* y *Query* enviados por el emulador para iniciar una ronda de inventario. También se muestra la respuesta de la etiqueta donde envía el RN16. La configuración usada en el emulador para obtener estas señales se muestra en el cuadro 5.1.3.

Enlace de subida	
Factor símbolo 1	1.5 Tari
Tari (sec)	25u
RTcal	2.5 Tari
TRcal	3 RTcal
Enlace de bajada	
M (párametro en comando *Query*)	1 (FM0)
DR	8
BLF	42.666k
Modulación	
Muestras/símbolo	10
Modulación	ASK
Equipo USRP	
Gain Tx	25
IQ rate (transmisión)	800k
Gain Rx	15
IQ rate (recepción)	853.33k
Tiempo de adquisición	200u

Cuadro 5.1.3: Configuración del emulador

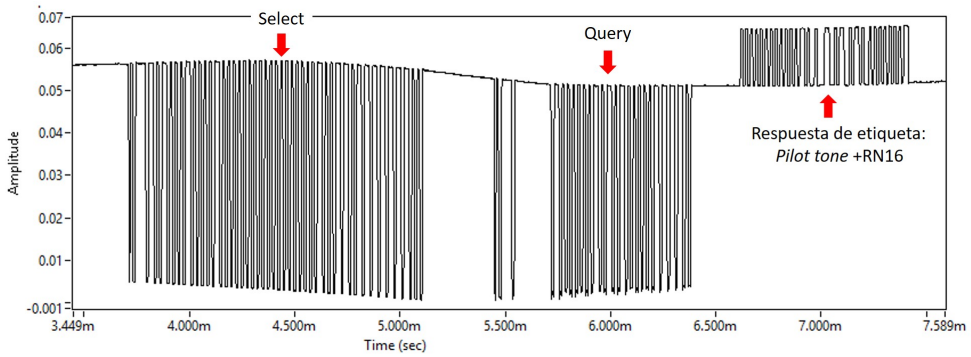


Figura 5.7: Envolvente compleja de las señales recibidas por el emulador

Con la configuración del emulador mostrada en el cuadro 5.1.3 y la información del cuadro 5.1.2 ,se calcula T_{total}:

$$\begin{aligned} T_1 &= MAX(RT_{cal}, 10Tpri) \\ &= MAX(2{,}5 \times 25\mu s, 10/BLF) = 234{,}37\mu s \\ T_2 &= \frac{20}{BLF} = 468{,}76\mu s \\ T_{TagRN16} &= \frac{35}{BLF} = 820{,}32\mu \\ T_{total} &= T_1 + T_{TagRN16} + T_2 = 1{,}523ms \end{aligned} \tag{5.1.5}$$

En la figura 5.8 se muestra la envolvente compleja de la señal recibida por el emulador al intentar completar la ronda de inventario. Se comienza el proceso de inventario enviando un comando *Select*, seguido de un comando *Query*. La etiqueta responde con su *RN16*, pero la confirmación de la recepción del RN16 aparece hasta $49{,}116ms$ después con el comando *ACK*. Como es de esperarse, la etiqueta no responde con el EPC, debido a que el emulador ha sobrepasado el tiempo T_{total} en el que debió confirmar el RN16. Ademas el emulador ha enviado más comandos *Query* (señales que se observan después del primer *Query* y *RN16*). Estos son enviados debido a la programación actual del emulador en la que en caso de no tener el comando ACK listo a tiempo, el emulador debe generar otro comando *Query*. A cada uno de estos comandos, la etiqueta ha contestado con un *RN16* distinto.

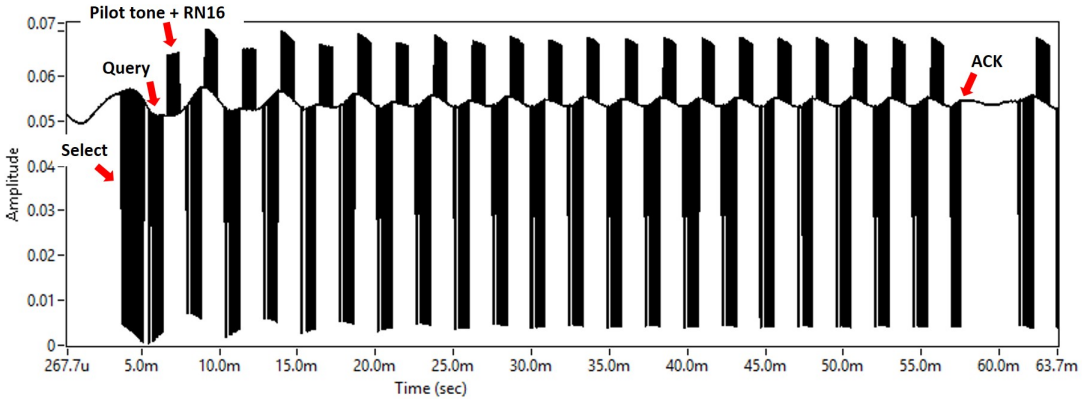


Figura 5.8: Envolvente compleja de las señales recibidas por el emulador al intentar completar la ronda de inventario

Se realizaron pruebas para identificar qué parte del emulador está ocasionando más retrasos. Se prestó mayor atención en el *loop* de procesamiento de la señal recibida, ya que se consideró inicialmente que el proceso de identificación y demodulación pueden estar aportando gran parte del retraso.

Sin embargo, se identificó que la razón del mayor retardo se encuentra en la parte del transmisor del emulador, entre el *loop* de preparación de datos y el *loop* de escritura de datos al USRP. Como se mencionó en el capitulo 3, el *loop* de preparación de datos genera los comandos *Select* y *Query* y espera un tiempo T_e para recibir el $RN16$ del *loop* de procesamiento. Este tiempo T_e debería ser la suma del tiempo T_{total} en que la lectora envia una señal de onda continua para que la etiqueta responda con el RN16. Si después de T_e ms no tiene esta información, el emulador genera otro *Query* y lo envía a la cola de escritura de datos. En el caso contrario, si se cuenta con la información del $RN16$ antes de T_e ms, el emulador forma el comando ACK y lo envía a la cola de escritura de datos al USRP.

Sin embargo, con el análisis de tiempos se detectó que para esa prueba, la diferencia en tiempo ΔT entre el primer comando *Query* que generó el *loop* de preparación y el primer comando *Query* que se detectó en el *loop* de procesamiento en el receptor es de aproximadamente un segundo. Probablemente el USRP tardó un segundo en iniciar la transmisión, sin embargo el *loop* de preparación ya había comenzado a generar comandos *Query*. Como el equipo USRP aún no envió ninguna señal,el *loop* de procesamiento no detectó ningún comando, y no generó la información del $RN16$. En el *loop* de preparación el tiempo T_e se venció, se generó otro comando *Query* y se agregó a la cola de escritura de datos. Se configuró $T_e = 50ms$ para esta prueba, por lo que antes de que el USRP comenzara la transmisión, se generaron $\frac{\Delta T}{50ms} \approx 20$ comandos *Query* que fueron enviados a cola de escritura de datos.

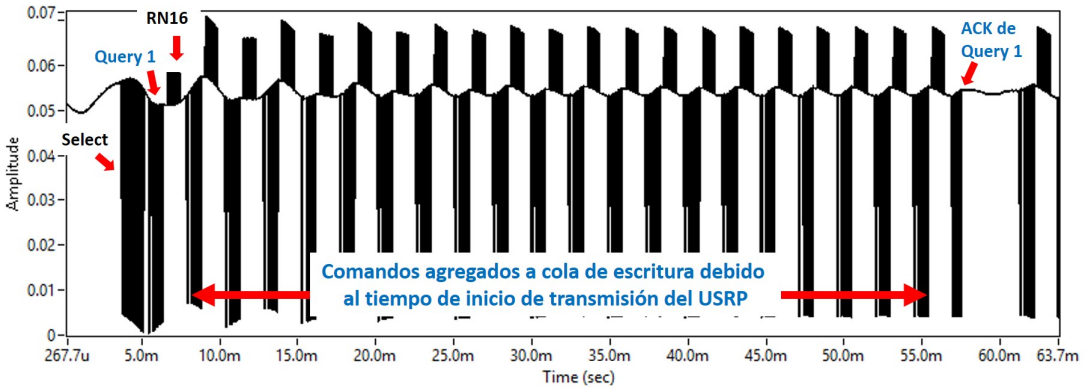


Figura 5.9: Comandos agregados por el emulador debido a la falta de sincronización entre el *loop* de preparación de datos y *loop* de escritura de datos a USRP.

Por el lado del receptor, en el *loop* de procesamiento se obtuvo el tiempo de procesamiento T_p que tarda el emulador desde identificar un comando *Query*, aislar la señal que contiene información de la etiqueta, demodular y decodificar esta señal hasta formar el comando *ACK*. Este tiempo se obtuvo con la formula 5.1.6.

$$T_p = t_d - t_r \tag{5.1.6}$$

Donde t_d es la marca del tiempo en que el receptor detecta un comando *Query* y t_r es la marca de tiempo en que el transmisor tiene listo el comando *ACK* con la confirmación del RN16 de la etiqueta. Para obtener las marcas de tiempo se utilizó la función *Get Date/Time In Seconds* de LabView durante la ejecución de la prueba.

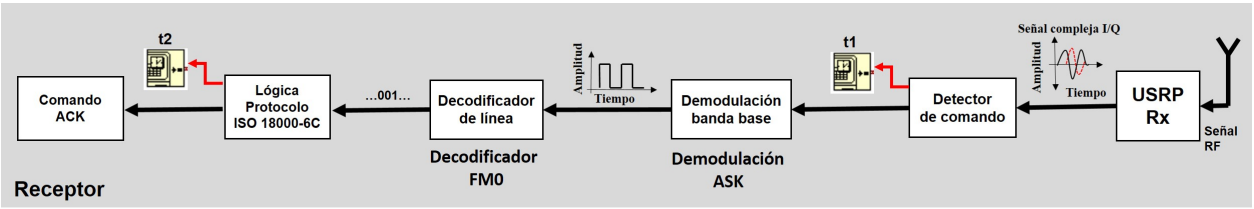


Figura 5.10: Diagrama de bloques del receptor del Emulador

En la figura 5.11 se muestra una gráfica con los tiempos T_p registrados. Se encontró que el tiempo de procesamiento T_p tiene en promedio una duración de 2.4 ms, lo cual no es suficiente para los requisitos del protocolo ($T_p < T_{total}$). Es importante mencionar que el emulador no generó el comando *ACK* de todas las respuestas obtenidas de la etiqueta, esto debido a la configuración actual en la que si la longitud en bits de la señal demodulada y decodificada del *RN16* es menor a 16, la lectora toma esto como un error y deshecha la información.

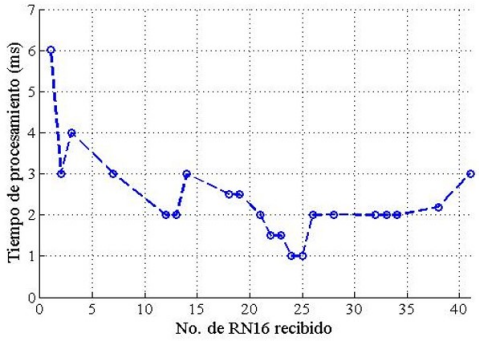


Figura 5.11: Tiempo de procesamiento T_p que tarda el emulador desde que identifica el comando *Query* hasta formar el comando *ACK* .

Entonces se tiene identificados las siguientes limitantes para el emulador RFID con el equipo USRP.

- El problema consiste en sincronizar el comienzo de transmisión del USRP con el comienzo de la lógica del protocolo. Una limitante es que el USRP debe de contar con muestras en el buffer de transmisión, es decir, si se inicia la transmisión pero no se le proporcionan muestras para ser enviadas, o si el USRP transmitió lo que tiene en el buffer de transmisión, el USRP envía un error de "El buffer de transmisión se vació antes de que datos nuevos fueran proporcionados" (*buffer underflow*) y el programa se detiene.
- El tiempo de procesamiento de la señal en el receptor supera el tiempo de espera especificado por el protocolo para el envío del comando *ACK*.
- No se consideró el tiempo que el programa tarda en modular los datos a enviar, unicamente se consideró dentro de la medición de tiempos la parte de la lógica del protocolo y codificación PIE.

5.2. Emulador + PXI-5646R

Una de las ventajas de LabView, es que la programación gráfica puede agilizar el aprendizaje del lenguaje y desarrollo de software, y en algunos casos, facilita el proceso de configuración de hardware cuando se utiliza equipo que cuenta con librerías en LabView. También es relativamente sencillo reutilizar bloques de código entre aplicaciones. En este caso, se reutilizó parte del código del emulador RFID utilizado con USRP para transmitir y recibir las señales por medio de un equipo PXI-5646R, con el objetivo de buscar alternativas de hardware para mejorar los tiempos de respuesta del emulador.

A diferencia del equipo USRP, el PXI es un equipo calibrado, y la potencia máxima de transmisión que el fabricante señala en sus hojas de especificaciones es de 10 dBm (onda continua), por lo que es capaz de energizar a etiquetas pasivas Alien 9654.

5.2.1. Adaptación del código del emulador para equipo PXI

A continuación se describen los VIs que se utilizaron para el Emulador RFID + PXI. Al igual que en el Emulador RFID + USRP, se puede identificar código de la lógica del emulador, y código para la configuración y control del Hardware.

Transmisor

- **Lógica del Emulador:**

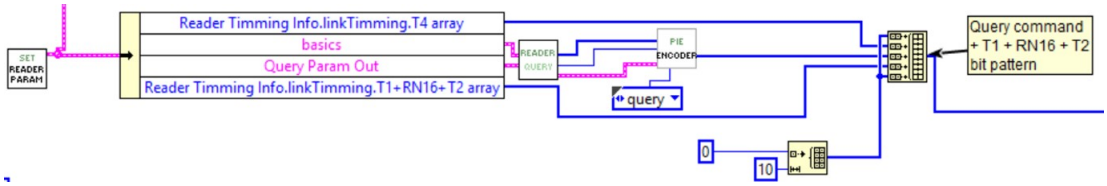


Figura 5.12: Diagrama de bloques de código de transmisor, lógica del Emulador RFID

  - Set Reader Param: Establece todos los parámetros del enlace de subida.
  - Reader Query: Crea el arreglo de datos binarios que conforman el comando *Query*.
  - Pie Encoder: Realiza codificación PIE.

- **Control y configuración de Hardware:** Se utilizó el programa *MT RFSG Generate ASK*, que se encuentra en los ejemplos del kit de herramientas de modulación de LabView. Este ejemplo muestra cómo generar una señal modulada ASK de fase continua y las capacidades de la configuración de forma de onda arbitraria de NI-RFSG. El módulo de lógica del emulador alimenta este programa con parámetros como tasa de símbolo, arreglo de datos binarios del comando de la lectora a enviar, frecuencia de muestreo, entre otros. Este programa configura el equipo, escribe la señal que se va a transmitir al dispositivo, comienza la transmisión, y una vez que ha terminado de transmitir la señal, cierra la sesión del PXI. Este módulo transmite una señal finita.

  Este programa no ejecuta la lógica completa del emulador RFID + USRP, únicamente genera un comando *Query* y termina la transmisión.

Receptor

- **Lógica del emulador** Se reutilizó todo el *loop* de procesamiento de señal, que consta de los siguientes subVIs:

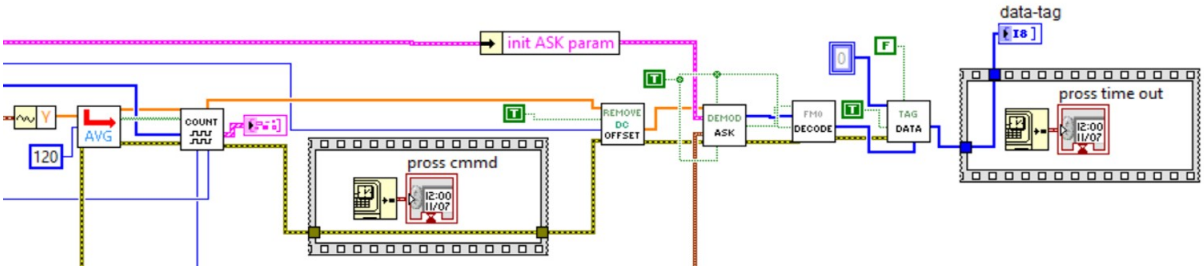


Figura 5.13: Diagrama de bloques de código de receptor, lógica del Emulador RFID

- AVG: Obtiene información de pulsos bajos/pulsos altos de la envolvente compleja de la señal recibida.
- Detector de comando: Identifica comandos de la lectora en la envolvente compleja de la señal recibida. La finalidad es localizar el fin del comando *Query* y del comando *ACK*, para aislar la señal de retrodispersión de la etiqueta y enviarla a la siguiente etapa de procesamiento.
- Remove DC Offset: Elimina el *Off-set* de la señal de retrodispersión de la etiqueta.
- ASK Demodulation: Realiza demodulación ASK.
- FM0 decoder: Realiza decodificación FM0
- Tag-Data: Recupera la información enviada por la etiqueta (RN16/EPC).

- **Control y configuración de Hardware:** Se utilizó el programa *RFSA Pulse Trigger*, que se encuentra en los ejemplos de la libreria NI-RFSA. Este programa adquiere datos I/Q usando el analizador vectorial de señales (PXI), y configura el inicio de la adquisición de datos cuando se cumple cierta condición mediante el VI *niRFSA Configure IQ Power Edge Ref Trigger*

En este programa la transferencia de datos de la memoria del equipo PXI a la memoria de la computadora se inicia cuando la señal adquirida cruza cierto umbral de disparo. Los datos adquiridos alimentan los VIs de procesamiento de la señal para obtener el *RN16* de la etiqueta. La duración de la adquisición se configura con el control *Burst Lenght(sec)*.

5.2.2. Comunicación con etiquetas pasivas Alien y PXI

Se logró activar una etiqueta pasiva Alien 9654 y se obtuvo el RN16 de la etiqueta. En la figura 5.14 se muestra la envolvente compleja de la señal con el comando Query enviado por el PXI para iniciar una ronda de inventario. También se muestra la respuesta de la etiqueta donde envía el RN16. La configuración usada en el PXI para obtener estas señales se muestra en el cuadro 5.2.4.

Enlace de subida	
Factor símbolo 1	1.5 Tari
Tari (sec)	25u
RTcal	2.5 Tari
TRcal	3 RTcal
Enlace de bajada	
M (párametro en comando *Query*)	1 (FM0)
DR	8
BLF	42.666k
Modulación	
Muestras/símbolo	10
Modulación	ASK
Equipo PXI Transmisión	
Power Level(dBm)	0
IQ rate (transmisión)	800k
Pre-filter gain (dBm)	-2
Arb:Waveform Repeat Count	1
Equipo PXI Recepción	
Reference Level(dBm)	0
IQ rate(S/s)	853.33k
Trigger Slope	Rising Slope
Trigger Level(dBm)	-15
Burst Length(sec)	62 ms
Reference Position(%)	0
Minimum Quite time(sec)	0

Cuadro 5.2.4: Configuración del emulador

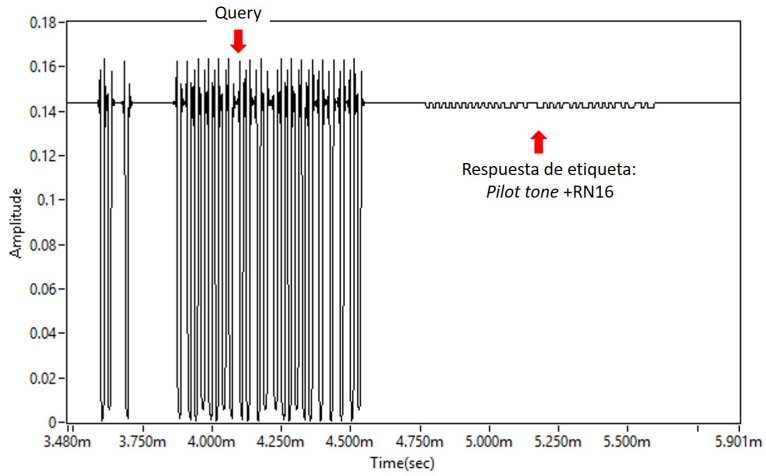


Figura 5.14: Envolvente compleja de las señales recibidas por el PXI

Para esta prueba se obtuvo el tiempo T_{p-pxi} que tarda el emulador desde identificar un

comando *Query*, aislar la señal que contiene información de la etiqueta, demodular y decodificar esta señal hasta obtener el RN16. Este tiempo se obtuvo con la formula 5.2.1.

$$T_{p-pxi} = t_{dpxi} - t_{rpxi} \approx 2ms \tag{5.2.1}$$

Donde t_{dpxi} es la marca del tiempo en que el receptor detecta un comando *Query* y t_{rpxi} es la marca de tiempo en que se obtuvo el RN16 de la etiqueta. Para obtener las marcas de tiempo se utilizó la función *Get Date/Time In Seconds* de LabView durante la ejecución de la prueba.

El tiempo T_{total} es el obtenido por la ecuación 5.1.3, ya que se utilizaron los mismos parámetros del enlace para las pruebas con el USRP. El tiempo T_{p-pxi} aunque es menor que el obtenido con Emulador + USRP, sigue siendo mayor al requerido por el protocolo para completar la ronda de inventario.

Capítulo 6

Conclusiones

Los sistemas de radio definido por software (*SDR* por sus siglas en inglés *Software Defined Radio*) son sistemas de radio frecuencia en los que algunas de las operaciones de transmisión o recepción son son efectuadas por medio de software, cuando normalmente se realizan por medio de *hardware* especializado. Debido a esto, es posible construir prototipos y herramientas para el estudio de protocolos de comunicación, dando la libertad de crear pruebas y análisis personalizados a la necesidades de los usuarios. Para este proyecto, se desarrolló una herramienta para la emulación RFID ISO-180006C, programado en LabView y utilizando equipo de radio definido por software de la marca *National Instruments*. A continuación se presentan las conclusiones en base a los resultados obtenidos.

Lógica del Emulador RFID

Se construyó en LabView, la lógica propia del emulador. Es posible configurar parámetros del protocolo ISO 18000-6C para el enlace de subida y de bajada. A diferencia de las lectoras comerciales disponibles, es posible configurar parámetros de transmisión personalizados. La lógica del emulador, se encarga de crear los comandos de la lectora, contiene también una máquina de estados con la que se implementó la lógica del estándar, identifica comandos enviados por el emulador e identifica la señal de retrodispersión de la etiqueta, demodula, decodifica y obtiene el RN16 de la misma.

Este programa puede ser utilizado con diferente equipo de hardware, ya que los componentes del emulador están encapsulados en programas independientes dependiendo de su función, y se puede identificar el código de la lógica del emulador y código para la configuración y control del Hardware. Para utilizar un equipo, únicamente se tiene que adaptar los datos que alimentan la etapa de transmisión del *hardware* utilizado.

Como trabajo a futuro se podrían realizar las siguientes mejoras y adaptaciones al programa actual:

- Algunos componentes del programa pueden ser modificados para mejorar el tiempo de respuesta del emulador. Por ejemplo en la parte de modulación ASK, habría la posibilidad de tener pre-guardados los comandos de la lectora ya codificados y modulados, para restarle tiempo de procesamiento a esta tarea. Sin embargo se encuentra la restricción de que durante la ronda de inventario, algunos valores de estos comandos cambian dependiendo del estado en que se encuentre la comunicación con las etiquetas.

- Mejorar el rango de lectura, podrían utilizarse antenas direccionales y en el receptor implementar nuevos métodos de rechazo de la señal enviada por la lectora.
- Ampliar la libreria del programa para la simulación de otros protocolos de RFID.
- Agregar modulación Miller (en inglés *Miller Modulated Subcarrier*)

Emulador + USRP

Se utilizó el USRP para transmitir los comandos que guían la comunicación con etiquetas comerciales de RFID. El USRP es un equipo de radio definido por software.

Se logró activar etiquetas pasivas Alien 9654, y se obtuvo el RN16 de las etiquetas. A pesar de los esfuerzos realizados, el emulador no es capaz de cumplir con las restricciones de la temporización del enlace dictada por el protocolo. Dentro de las limitaciones del programa se encuentran la sincronización del comienzo de transmisión del USRP con el comienzo de la lógica del protocolo, y también el control de la aparición del error *buffer underflow* del USRP. También es posible que la comunicación entre el equipo USRP y computadora host, que se realiza por medio de Gigabit Ethernet, contribuya a los retrasos del emulador.

Debido a que el programa de lógica del emulador RFID se ejecuta en la computadora Host, es posible que se obtengan resultados distintos al utilizar computadoras con mas capacidad, o diferente sistema operativo.

A pesar de estas limitaciones, el programa actual podría servir como herramienta para la enseñanza del protocolo, realizar mediciones y estudio de sistemas RFID en aplicaciones especificas. Podría ser herramienta para mediciones de potencia de la señal de retrodisperción recibida, tiempo de respuesta de etiquetas, y temporización del enlace de subida/bajada entre otras.

Emulador + PXI

Se utilizó el PXI-5646R para transmitir los comandos que guían la comunicación con etiquetas comerciales Alien 9654. Se logró activar a las etiquetas y obtener el RN16. A pesar de los esfuerzos realizados, el emulador no es capaz de cumplir con las restricciones de la temporización del enlace dictada por el protocolo.

El equipo PXI es más costoso que el equipo USRP, pero tiene la ventaja de tener FPGA programable. La solución al problema del tiempo de respuesta del emulador podría estar en la programación de FPGA para que algunas funciones se realicen por hardware, sin eliminar la flexibilidad que brinda el software para realizar modificaciones y mejoras al protocolo.

Para realizar la prueba, se utilizaron como base los programas *MT RFSG Generate ASK* para transmitir los comandos del protocolo y *RFSA Pulse Trigger* para recibir las señales. Estos programas no ejecutan la lógica completa del protocolo ISO-18000-6C, solo realiza una transmisión de un comando *Query*. La lógica de la obtención del *RN16* no está conectada al transmisor, por lo que el programa no envía el comando *ACK*. Como trabajo a futuro, se podría completar la lógica del protocolo utilizando como referencia los siguientes programas de ejemplo de la librería de LabView *RFSG*:

- ***RFSG Arbitrary Waveform Streaming:*** El ciclo de transmisión es alimentado por colas (*Queues*), similar a la estructura actual del emulador + USRP.

- ***RFSG Getting Started Script:*** Demuestra cómo usar *scripts* para dictar el comportamiento de la generación de forma de onda. Las formas de onda se guardan previamente en la memoria de PXI, y el script controla la generación de las mismas.

En cualquiera de los dos casos, es necesario primero lograr los requisitos de temporización del enlace para completar la ronda de inventario.

A pesar de estas limitaciones, el programa actual podría servir como herramienta para la enseñanza del protocolo, para realizar mediciones y estudio de sistemas RFID en aplicaciones especificas. Podría ser herramienta para mediciones de potencia de la señal de retrodisperción recibida, tiempo de respuesta de etiquetas, y temporización del enlace de subida/bajada entre otras.

Bibliografía

[1] Xiaoqiang Zhang and Manos Tentzeris. Applications of Fast-Moving RFID Tags in High-speed Railway Systems. *International Journal of Engineering Business Management*, 3(1):27–31, 2011.

[2] A. Buffi and P. Nepa. An rfid-based technique for train localization with passive tags. In *2017 IEEE International Conference on RFID (RFID)*, pages 155–160, May 2017.

[3] J. Kim, B. Cho, B. Lee, H. Lee, C. Park, and J. Bang. Antenna design for high speed rfid system. In *2013 IEEE Antennas and Propagation Society International Symposium (APSURSI)*, pages 1738–1739, July 2013.

[4] R. Rupp Angerer, C. Langwieser. Evaluation and exploration of RFID systems by rapid prototyping. *M. Pers Ubiquit Comput (2012)*, 16:309–321, 2012.

[5] B. M. Husain M. Hoffmann S. Bali, M. N. Jazi and T. Kaiser. Development of a configurable and enhanced rfid reader testbed based on epc generation-2 standard. *Open Science Journal of Electrical and Electronic Engineering*, 2(5):84–92, 2015.

[6] D.M. Dobkin. *The RF in RFID: UHF RFID in Practice*. Elsevier Science, 2012.

[7] D. Paret. *RFID at Ultra and Super High Frequencies: Theory and application*. Wiley, 2009.

[8] ISO/IEC 18000-6:2010. http://www.iso.org/iso/catalogue_detail.htm?csnumber=46149.

[9] INDUSTRY SOLUTIONS. https://www.impinj.com/solutions/.

[10] Logistics Reduction: RFID Enabled Autonomous Logistics Management (REALM) (LR-REALM). https://techport.nasa.gov/view/93175.

[11] V. Ravi and R. Aparna. Security in rfid based smart retail system. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 587–592, March 2016.

[12] G. Kulkarni, R. Shelke, R. Sutar, and S. Mohite. âĂIJrfid security issues amp; challengesâĂİ. In *2014 International Conference on Electronics and Communication Systems (ICECS)*, pages 1–4, Feb 2014.

[13] Laura Hervert-Escobar, Neale R. Smith, José Ramón Rodríguez-Cruz, and Leopoldo Eduardo Cárdenas-Barrón. Methods of selection and identification of rfid tags. *International Journal of Machine Learning and Cybernetics*, 6(5):847–857, 2015.

[14] C. A. Albright, S. A. Kaiser, L. W. Oglesbee, and D. W. Engels. Forward error correction in passive uhf gen2 communications. In *2015 IEEE International Conference on RFID (RFID)*, pages 17–24, April 2015.

[15] Voyantic Tagformance Lite. http://voyantic.com/.

[16] CISC RFID MeETS System. https://www.cisc.at/products/rfid-meets/.

[17] M. Buettner and D. Wetherall. A software radio-based uhf rfid reader for phy/mac experimentation. In *2011 IEEE International Conference on RFID*, pages 134–141, April 2011.

[18] N. Kargas, F. Mavromatis, and A. Bletsas. Fully-coherent reader with commodity sdr for gen2 fm0 and computational rfid. *IEEE Wireless Communications Letters*, 4(6):617–620, Dec 2015.

[19] ¿Qué es LabVIEW?. http://www.ni.com/es-mx/shop/labview.html.

[20] Leyre Azpilicueta Fernández de las Heras. *Estudio sobre la codificación conjunta canal-fuente en sistemas UHF RFID*. PhD thesis, INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY, 2009.

[21] How Can I Read 1,000 Tagged Apparel Items Within a Small Area? http://www.rfidjournal.com/blogs/experts/entry?8891.

[22] Klaus Finkenzeller. *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*. Wiley Publishing, 3rd edition, 2010.

[23] Nemai C. Karmakar Stevan Preradovic. Modern RFID Readers. *Microwave Journal*, 2007.

[24] Michaël Nicolas. *Radio logicielle: analyse dárchitectures matérielles et outils informatiques. Electronique*. 2011.

[25] Alejandro Sánchez Lakehal. *La Radio Definida por Software: Diseño de un receptor de banda aeronáutica VHF*. Tesis de licenciatura, Escuela Politécnica Superior de Ingeniería de Vilanova i la Geltrú, 2015.

[26] Radio Frequency Identification : RFID Implementation using National Instruments Hardware. http://www.ni.com/example/30914/en/.

[27] a Goldsmith. Wireless communications. *GLOBECOM 05 IEEE Global Telecommunications Conference 2005*, 3(4):427, 2005.

[28] National Instruments. *NI LabVIEW Modulation Toolkit Help*. June 2014.

[29] What Is NI USRP Hardware? http://www.ni.com/white-paper/12985/en/.

[30] Data Streaming Performance Tips.

[31] NI. Device specifications Ni USRP -2920. Technical report.

[32] ¿Qué es un Transceptor Vectorial de Señales PXI?. http://www.ni.com/es-mx/shop/wireless-design-test/vector-signal-transceivers/what-is-vst.html.