Capítulo 4. Pruebas y resultados

4.1 Introducción

Una vez que se entendió cómo trabaja el Kit de Evaluación de *Keeloq*, ahora se trabajará en programar cada integrado para que logren una comunicación eficiente con la estación base y en desarrollar nuestros propios circuitos para que también se comuniquen con la estación base, basados en la tecnología de *Keeloq*.

4.2 Programación y Ventanas de Dialogo

Para la programación de la estación base existe un software que viene con la paquetería del Kit de Evaluación. En lo primero que se trabajó fue en la familiarización con este software que es muy sencillo para beneficio del usuario. En la Figura 4.1 se muestra la pantalla del programa y se puede ver que es sencillo de utilizar para el usuario.

Figura 4.1 Software del kit de Evaluación

Para empezar se instaló el software (que venía en un disco flexible de 3½") en una de las computadoras del laboratorio. Como lo primero que se tenía que programar era la estación base, esta se conectó a uno de los puertos seriales de la computadora que no estaba ocupado y también se conectó a la corriente alterna para que estuviera debidamente alimentada. Una vez hecho esto se "corrió" el programa y se verificó que

efectivamente “reconociera” a la estación base escogiendo el puerto al que había sido conectada (*Setup* > *Serial Port*). Estos pasos se pueden visualizar en la Figura 4.2.

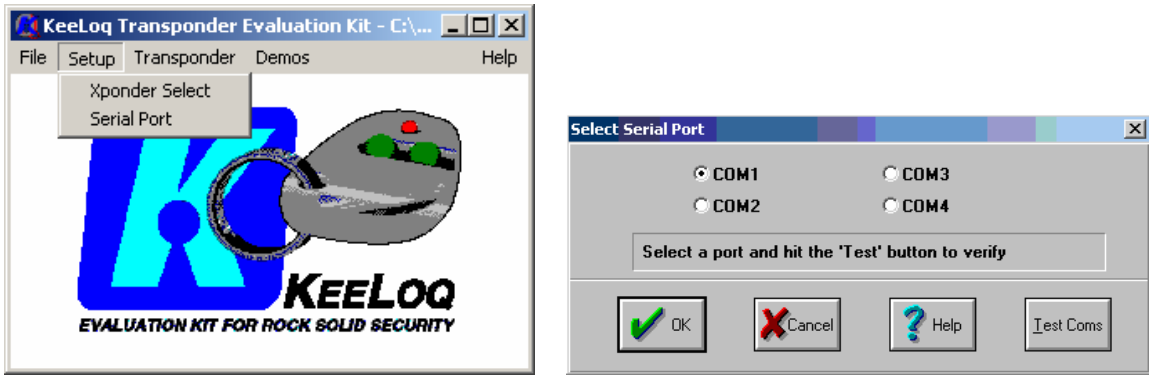


Figura 4.2 Elección del puerto

Después se eligió del menú la opción para programar la estación base (*Transponder* > *Program*) y se especificaron los detalles de cómo se deseaba que esta fuera programada y cuando estos estaban listos se apretaba el botón de ***“Program Base”*** para transferir esta información a la estación base. En la Figura 4.2.1 se pueden visualizar los pasos para programar la base, así como las opciones que se encuentran en la pantalla para que trabaje en la forma que se desee. NOTA: Las especificaciones de cada opción para programar la estación base ya fueron detalladas en el capítulo 3.

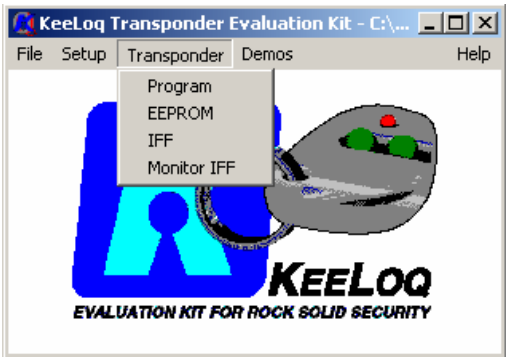
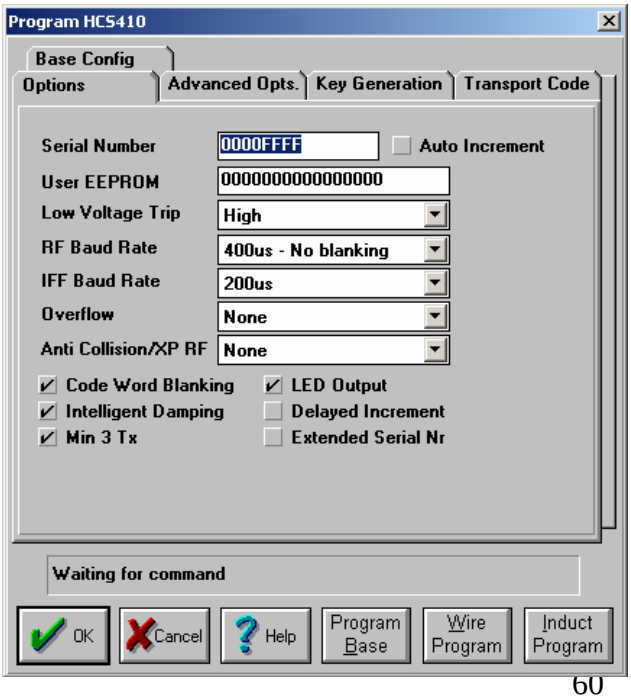


Figura 4.2.1 Programación de la Estación Base



Una vez programada la estación base ahora se tienen que programar los diferentes integrados que vienen con el kit de evaluación. Existen dos formas de programar estos integrados, la primera es por medio de cableado (*wire program*), esto significa conectar directamente el integrado a la estación base por medio de los *jumpers* (J2) y así existe una comunicación directa por la línea de PWM como se puede ver en la Figura 4.2.3. Y la segunda opción para programar es por medio inductivo (*induct program*) a través del campo magnético que genera la estación base, el único requerimiento es que el integrado se encuentre dentro del campo generado por la estación base que es aproximadamente de 6 cm. máximo alrededor de la base.

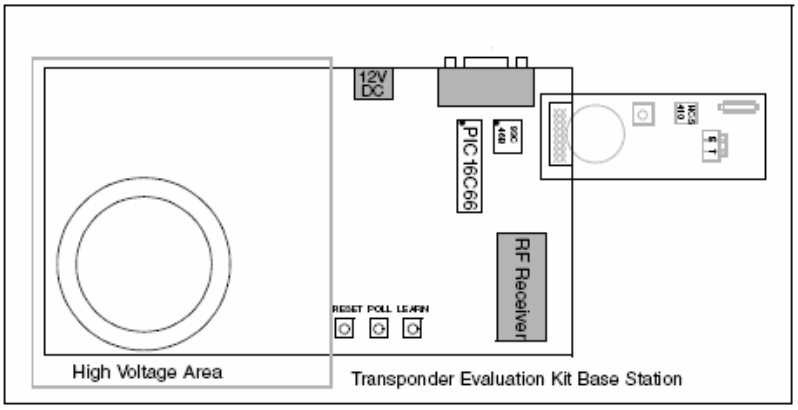


Figura 4.2.2: *Wire Programming* de un transmisor/*transponder* [6]

Lo primero que se tiene que hacer para programar los integrados es escoger el integrado que se va a programar, ya sea el HCS410 o el HCS412 del menú principal del programa (*Setup > Xponder Select*). Una vez escogido el integrado que se desea programar, se selecciona el menú de programar y se dan las especificaciones con las que se quiere programar este integrado que son principalmente el número de serie (que es el "nombre" con el que se le va a reconocer), lo que se le quiere escribir en la EEPROM (64 bits) que viene dentro del integrado, la velocidad a la que se va a transmitir, y las demás

opciones que ya fueron detalladas en el capitulo 3. En la Figura 4.2.4 se muestra cómo se elige el integrado dentro del programa de *Keeloq*.

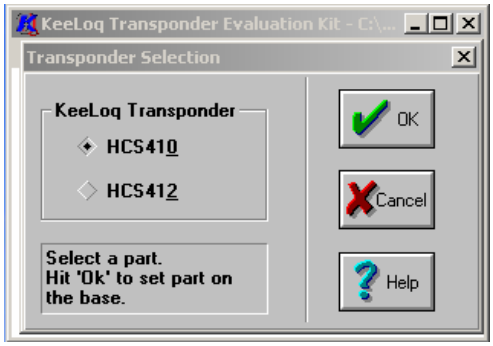


Figura 4.2.3 Elección del *Transponder*

Una vez que los integrados estaban programados se probaron para comprobar que estos se comunicaran eficazmente con la estación base. Con la ayuda del software del Kit de Evaluación se pudieron monitorear estas comunicaciones (*Transponder >Monitor IFF*). Esta ventana nos arroja información como el número de serie del transmisor que se está comunicando con la estación base (para identificación), la palabra mandada por la estación base (*challenge*), la respuesta en forma encriptada del transmisor que se está utilizando y la misma respuesta pero desencriptada. También se pueden ver las transmisiones RF si es que se están utilizando, y en el caso de este tipo de transmisiones, si se pone atención, también muestra el botón que se esta oprimiendo, ya que en los 4 bits más significativos de la respuesta desencriptada se manda un 0010 (2 decimal) si es el botón S0 el que se esta oprimiendo, un 0100 (4 decimal) si es el botón S1 el que se esta oprimiendo y un 1000 (8 decimal) si es el botón S2 el que se esta oprimiendo. En la Figura 4.2.5 se muestra la ventana *Monitor IFF* y se puede observar una comunicación *Code hopping* en la que el número de serie del *transponder* es 000000FE0 y se muestra el *challenge* mandado por la estación base (31AA0CA3), la respuesta encriptada (EBE3EBA4) y la respuesta desencriptada (31AA0CA3). Como se puede ver, esta fue

una comunicación exitosa ya que la respuesta desencriptada coincide con la palabra *Challenge* que se le mandó.

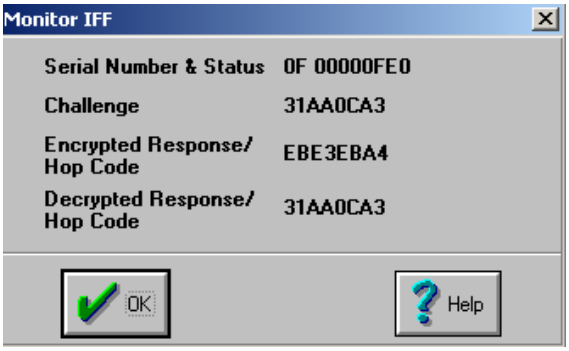


Figura 4.2.4 Ventana *Monitor IFF*

Asimismo existe la opción de leer la información que está guardada dentro de la memoria EEPROM de cualquier *transponder.* Cuando se programa un *transponder* existen 64 bits de memoria EEPROM (USR3:USR0) libres para guardar la información que el usuario desee (por ejemplo: el nombre del usuario de la tarjeta, las veces que la ha utilizado, etc.). NOTA: esta información debe ser proporcionada en forma hexadecimal. También se puede escribir sobre esta memoria que viene dentro de los integrados, así como también existe la posibilidad de cambiar el número de serie de estos *transponders* sin tener que programar de nuevo el integrado, con la opción *Transponder > EEPROM* (ver Figura 4.2.1). Lo único que se tiene que cumplir es que el *transponder* se tiene que encontrar dentro al rango del campo electromagnético generado por la base que son aproximadamente 6 cm alrededor de la base y también se tiene que usar el código de transporte correcto (*Transport Code*). El *Transport Code* es asignado cuando se programa el integrado, y para poder leer o escribir en la EEPROM o cambiar el numero de serie de un *transponder* en especial, se debe escoger dentro de la ventana *User EEPROM* el *transport code* con que fue programado este *transponder* antes de iniciar la operación de

lectura o escritura, si no coinciden, la operación fallara. Nota: se debe tener mucho cuidado, ya que al cambiar el numero de serie de un *transponder*, en el caso de que los 10 bits menos significativos del numero de serie de este estén siendo tomados como los bits de discriminación, estos son tomados directamente para la encriptación de los datos y si estos bits de discriminación son modificados en el integrado, la estación base no estará "enterada" de estos cambios y estos bits de discriminación no coincidirán y por lo tanto no podrá haber comunicación. En la Figura 4.2.6 se muestra la ventana *User EEPROM* para que se pueda visualizar todo lo explicado anteriormente, en este caso se realizo una lectura exitosa, el numero de serie es 000000FE0 y la información guardada en la EEPROM es DAD0DED0FEDEC0C0.

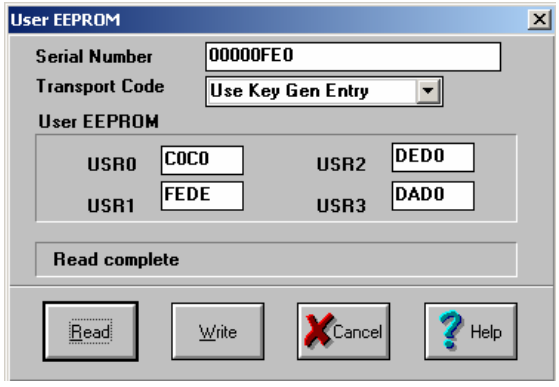


Figura 4.2.5 Ventana *User EEPROM*

Por último la opción IFF (*Identify Friend or Foe*) permite visualizar cómo el integrado encripta un "*challenge*" mandado directamente por el usuario y no uno aleatorio mandado por la estación base y cómo posteriormente el transponder desencripta esta palabra y manda la respuesta desencriptada. Esta opción también nos muestra el número de serie para saber cuál es el integrado con el que estamos trabajando. Lo único que se tiene que especificar es la palabra *challenge* (la cual es un número hexadecimal de 32 bits) así como también se deben especificar la "llave" y el algoritmo de encriptación

que se utilizaran en la operación. NOTA: las dos “llaves” y los algoritmos de encriptación son asignados durante la programación de los transponders.

Esta opción nos puede servir para identificar qué transponder es el que se encuentra dentro del campo electromagnético de la estación base, ya que nos arroja el número de serie del transponder y también nos sirve para saber la respuesta encriptada a un *challenge* en especial. En la Figura 4.2.7 se encuentra una transmisión IFF para visualizar todo lo explicado anteriormente, en esta se manda un *challenge* (00000001) y se ve como resultado que el transponder dentro del campo electromagnético generado por la base es 00000FE0. También se puede observar cómo el algoritmo de encriptación “ensució” la información, ya que la respuesta encriptada (C55F7BD8) es muy diferente a la información mandada (00000001).

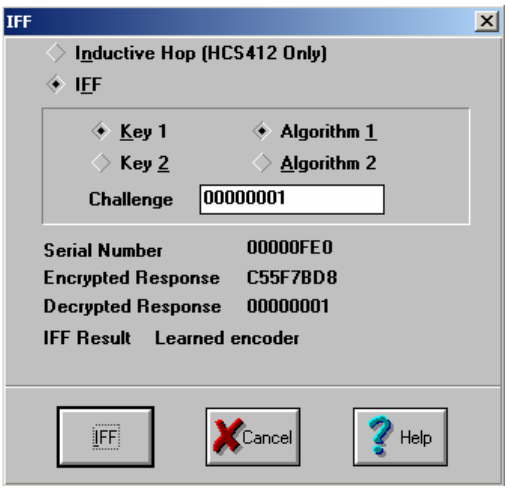


Figura 4.2.6 Ventana IFF (*Identify Friend or Foe*)

Cada que existe una transmisión entre la estación base y un *transponder*, se manda un “*challenge*” (Transmisión Code Hopping) y este siempre es diferente por el “salto” del código. Se hicieron pruebas para comprobar que cada transponder programado tiene diferente encriptación, y aunque se le mande la misma palabra de “*challenge*” la

respuesta encriptada siempre será diferente. Se mandó la misma palabra (*challenge*) a tres integrados HCS410, y en Tabla 5 se pueden observar las respuestas obtenidas.

Tabla 5. Tabla de respuestas de los integrados HCS410

Numero de serie	*Challenge*	Respuesta encriptada
FEFE	**12345678**	**EB1EB166**
FAFA	**12345678**	**F64EF349**
FFFF	**12345678**	**516ACABF**

Como se puede observar los algoritmos de encriptación son diferentes puesto que las respuestas obtenidas también lo son aunque el *challenge* fue el mismo mandado a los diferentes *transponders.*

4.3 Desarrollo de los integrados HCS410

Como la tablilla del *transponder* HCS410 es la más pequeña, es al que más utilidad se le puede encontrar; así que fue la que desarrollamos y duplicamos, ya que por ser pequeña puede ir dentro de una llave o credencial para autentificación de usuario. En la Figura 4.3 se muestra la tablilla que se encuentra en el laboratorio y no mide más de 2cm x 2cm.

Figura 4.3 Tablilla del *transponder* HCS410

Lo primero que se hizo fue buscar en la Guía de Usuarios del kit de evaluación de *KEELOQ* para ver cuáles eran los dispositivos que se iban a necesitar para armar este

circuito, pero sólo aparecía el esquemático del integrado pero sin los valores nominales de los dispositivos (resistencias e inductores) así que estos se tuvieron que calcular para que funcionara correctamente. El inductor que se utilizó para que se generara la comunicación entre la base y el integrado se calculó para que trabajara a 125 Khz. Esta comunicación se logra por medio del campo capacitor/inductor generado por el *transponder* y el circuito de recepción análoga en la estación base. Para calcular el valor necesario del inductor se investigaron las fórmulas para la resonancia armónica y se encontró la fórmula que se encuentra en la Ecuación 1.

$$W_r = \sqrt{1/LC} = 2\pi f_r \qquad \text{(Ecuación 1)}$$

Donde $\mathbf{f_r}$ es la frecuencia de resonancia (este dato se tiene) y es de 125 Khz. que es a la frecuencia a la que queremos llegar y se fija el valor del capacitor a 1.5nF por lo tanto sólo se despeja la **L** de la fórmula y se encuentra que el inductor debe ser de .9 mH y ya con este dato se armó el circuito que se muestra en la Figura 4.3.1 y se probó para ver si este circuito funcionaba y se comunicara correctamente con la Estación Base. NOTA: como en el laboratorio no hubo existencia de capacitores de 1.5nF se utilizó uno de 1.8 nF que era el que más se acercaba al valor calculado.

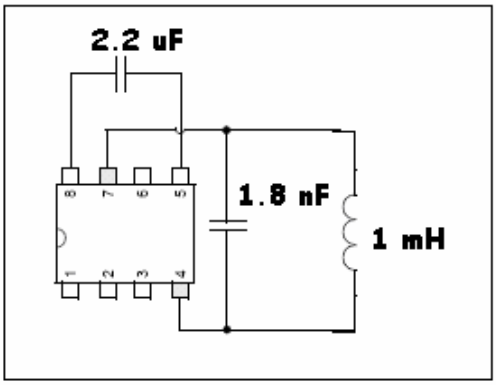


Figura 4.3.1 Circuito armado del HCS410

Una vez armado el circuito, se probó pero no funcionó inmediatamente, ya que al estar montado en un *protoboard* existían muchas capacitancias parásitas que afectaban y no dejaban que el circuito trabajara correctamente, provocando que funcionara por ratos. Así que se armó de nuevo el circuito pero ahora en una tablilla y soldando los componentes y se volvió a probar acercándolo a la estación base para ver si respondía al campo electromagnético generado por este, y entonces sí se obtuvieron los resultados esperados, es decir, que se comunicara con la estación base. Una vez que la estación base reconoció al *transponder* en su campo, este se programó asignándole un número de serie, datos en la memoria EEPROM, etc. Y se programó también la estación base para que posteriormente se "aprendiera" este *transponder* como uno de los 4 que puede reconocer al mismo tiempo. Y como quedaba otro integrado HCS410 se armó otro circuito idéntico al que funcionó y también se programó para que se terminara contando con tres tablillas funcionando como *transponders* inductivos sin necesidad de pila Por lo tanto se trabajaron con estos tres transponders para hacer las pruebas. Una vez que estos *transponders* estaban correctamente programados, se revisó la comunicación que existía entre estos *transponders* y la base.

4.4 Pruebas

Con la ayuda de un programa de comunicación serial en el control de comunicaciones, se monitorearon las transmisiones *Code hopping* que se generaban cada vez que el *transponder* entraba en el campo magnético generado por la Base y se pudo observar que, efectivamente, estas eran diferentes en cada transmisión debido al Algoritmo de Encriptación con que trabajan estos integrados. A continuación se muestran

tres transmisiones *Code hopping* de cada transponder programado, cada una de estas transmisiones se muestra encriptada en forma decimal, que es como trabaja este programa.

Número de serie: 00000FE0

248 128 128 248 128 128 248 128 128 120 030 120 030 120 224 248 248 128 120 000 000 120 254 128 120 000 240 000 248 000 000 000 000 248 248 128 120 224 120 000 240 120 254 120 000 015 248 248 128 120 224 120 000 240 120 254 120 000 015

248 128 128 248 128 128 248 128 128 120 030 120 030 248 248 000 128 248 248 248 120 000 000 248 248 000 000 000 000 128 128 128 120 254 128 000 120 254 000 128 248 128 128 128 120 254 128 000 120 254 000 128 248

248 128 128 248 128 128 248 128 128 120 030 120 030 248 120 030 000 120 224 000 128 128 000 248 000 000 000 000 248 000 128 128 000 000 120 254 248 128 248 248 000 128 128 000 000 120 254 248 128 248

Figura 4.4.1 Transmisión *Code Hopping* del Transponder 1

En la Figura 4.4.1 se muestran tres transmisiones *Code hopping* del *transponder* número 1 con el que se trabajó. Este *transponder* tiene como numero de serie: 00000FE0 y se puede observar que hay una parte dentro de la información encriptada que se repite en las tres transmisiones (parte resaltada), en este caso 248 000 000 000. Con base en estas observaciones se puede deducir que este es el número de serie, ya que como se explicó en el capitulo tres, el número de serie no se manda encriptado, pero sí se transmite cada que existe una comunicación *Code Hopping* con la estación base, y esto se puede corroborar viendo estas transmisiones.

En la Figura 4.4.2 se muestran las tres transmisiones *Code Hopping* del transponder número 2 que se monitorearon con la ayuda del programa y en este caso se puede observar que el número de serie 000C0C0A es interpretado como 120 000 240 000 000 000 000 000 en decimal, ya que esta secuencia de números se repite en cada transmisión.

Número de serie: 000C0C0A

248 128 128 248 128 128 248 128 128 120 030 120 030 120 254 128 128 128 248 248 248 000 248 120 224 120 000 240 000 000 000 000 000 000 248 248 120 030 248 248 248 128 248 128 128 120 254 248 120 030 248 248 248 128 248 128 128 248

248 128 128 248 128 128 248 128 128 120 030 120 030 128 248 120 030 128 128 248 120 000 015 248 128 120 000 240 000 000 000 000 000 000 248 120 030 128 128 248 128 248 128 248 248 120 030 128 128 248 128 248 128 248

248 128 128 248 128 128 248 128 128 120 030 120 030 120 224 000 000 000 120 224 248 128 120 000 240 000 000 000 000 000 000 248 120 254 128 248 120 000 000 120 254 120 254 120 254 120 254 128 248 120 000 000 120 254 120 254 248

Figura 4.4.2 Transmisión *Code Hopping* del Transponder 2

En la Figura 4.4.3 se muestran las tres transmisiones *Code Hopping* que se monitorearon con el transponder número tres con el que se trabajó. En esta oportunidad se puede apreciar que el número serie 0000FEDE es interpretado en forma decimal como 128 248 128 000 000 ya que esta es la parte que se repite en las tres transmisiones.

Número de serie: 0000FEDE

248 128 128 248 128 128 248 128 128 120 030 120 030 128 120 224 248 120 000 240 128 128 248 000 128 128 248 128 000 000 000 248 248 248 128 120 030 128 120 254 128 248 128 248 248 248 128 120 030 128 120 254 128 248 128

248 128 128 248 128 128 248 128 128 120 030 120 030 248 120 000 255 120 254 128 128 128 000 128 128 248 128 000 000 000 248 000 248 248 120 030 120 030 128 120 030 248 000 248 248 120 030 120 030 128 120 030

248 128 128 248 128 128 248 128 128 120 030 120 030 248 000 120 000 240 128 128 000 128 128 128 248 128 000 000 000 120 030 120 254 120 000 000 128 120 224 120 254 000 248 128 120 030 120 254 120 000 000 128 120 224 120 254 000 248 128

Figura 4.4.3 Transmisión *Code Hopping* del Transponder 3

Estas observaciones son muy interesantes ya que se puede corroborar que efectivamente el número de serie es transmitido cada que existe una comunicación *Code Hopping* y que no es fácil distinguirlo dentro de la comunicación. Es importante que el número de serie sea transmitido en cada comunicación porque de esta forma se puede

reconocer el *transponder* con el que se está trabajando en ese momento Esto es de gran utilidad ya que, si esto se empleara para una puerta, sólo tendríamos que utilizar un *transponder* programado previamente y "aprendido" por la estación base, y con sólo acercarlo al cerrojo (donde tendría que estar la base) este se activaría y se abriría. Para nuestro caso, utilizando tres llaves, se podría llevar un registro de cuántas veces entra cada persona con su llave y saber al momento quién es la persona que está entrando.

Una vez que se sabe cómo trabaja la comunicación entre los *transponders* y la base, sólo falta encontrar una aplicación. Basados en la idea de una puerta, tendríamos que lograr que cuando el *transponder* se encuentre dentro del rango del campo electromagnético generado por la base, se active un mecanismo, por ejemplo el de quitar un seguro y que cuando este se aleje, el seguro se vuelva a activar automáticamente. Así que se necesita una etapa de potencia y otra digital. La etapa digital se toma directamente de la estación base, ya que se acondicionó una señal de 0 a 5 volts cuando el *transponder* estuviera dentro del campo de la base, y la etapa de potencia es para activar cualquier dispositivo que se alimente de corriente alterna.

Con el objetivo de hacer pruebas en el laboratorio, se conectó un foco para que cuando un *transponder* programado previamente y "aprendido" por la base, se encontrara dentro del campo magnético generado por esta, el foco se prendiera y apagara indicando que está reconociendo al *transponder*. Para esto se necesitó de un TRIAC para darle el disparo de encendido al foco y de un optoacoplador para aislar la parte digital de la de potencia. En la parte digital se colocó un registro, ya que este nos brindaba la oportunidad de recibir una señal de reloj y activarse sólo cuando esta se activara. A continuación en la Figura 4.4.4 se muestra el circuito que se diseñó e implementó en el laboratorio:

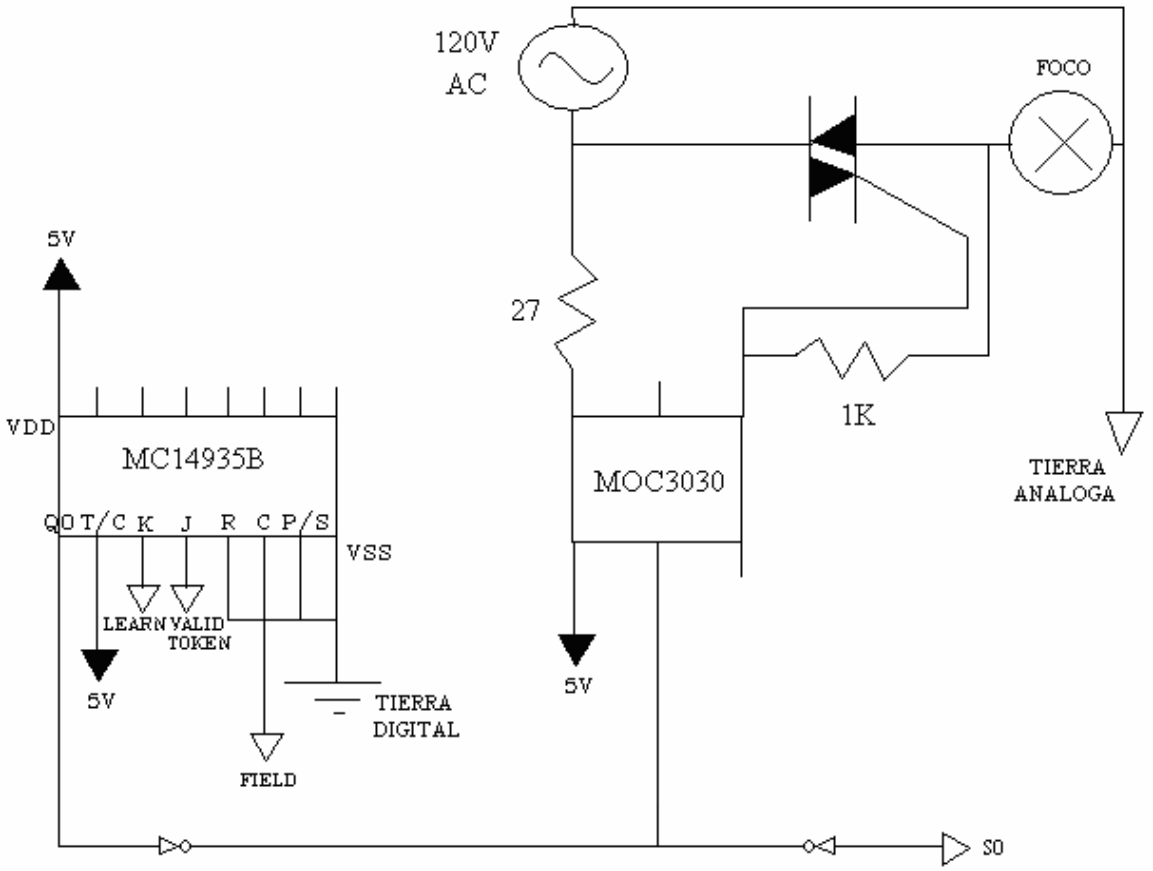


Figura 4.4.4 Circuito implementado en el laboratorio

Las señales LEARN, FIELD y VALID TOKEN fueron tomadas directamente de la Estación Base y mandadas a un registro, obteniendo así una señal acondicionada que es mandada directamente al MOC3030 para dar el disparo que recibe el TRIAC, el cual activa directamente el foco (que es la señal de salida) cada vez que el *transponder* se encuentra dentro del campo electromagnético generado por la estación base. La señal acondicionada (Q0) es negada porque el MOC tiene una habilitación negativa, es decir, se activa cuando la señal está en bajo (0v). La señal de FIELD es mandada a la entrada CLOCK del registro, este toma las demás señales cuando existe una transición positiva (subida de 0 a 5v). La señal de S0 también fue tomada directamente del PIC16C66 que se encuentra en la Estación Base, esta señal se ocupa para prender el foco cuando existe una

transmisión RF de cualquiera de los *transponders* RF de largo alcance (cuando se oprime el botón S0 de la tablilla de los transmisores), esta señal también va negada por la misma razón sobre la habilitación negativa del MOC. Cabe mencionar que el foco se mantiene encendido, es decir, la señal se mantiene habilitada todo el tiempo que el botón se mantenga oprimido. La conFiguración del MOC fue investigada para obtener el mejor desempeño de este y para que el TRIAC fuera "disparado" adecuadamente.

4.5 Resultados

Ahora una vez que los transmisores/transponders estaban programados y comunicándose eficientemente con la estación base, y que el circuito armado para el disparo de la aplicación estaba trabajando correctamente se tomaron diferentes gráficas para mostrar los resultados obtenidos. A continuación se muestran estas gráficas obtenidas de las señales de habilitación que se utilizaron en el circuito y también la de la señal de salida. Cabe mencionar que para obtener las señales de potencia se necesitó un aislador de voltaje para no dañar el osciloscopio con que se obtuvieron las señales, también es importante mencionar que este estaba conectado a una tierra flotante para no tener problemas de tierras.

En la Figura 4.5.1 se muestra la grafica de la señal de FIELD, la cual nos indica el periodo de muestreo, es decir, nos indica que cada 1.24 segundos la estación base busca por transponders que se encuentren dentro del campo electromagnético generado por la base. Si la estación base encuentra un transponder de forma inductiva se genera una comunicación *Code Hopping*, y cuando el transponder es retirado del campo electromagnético la estación base regresa a seguir muestreando.

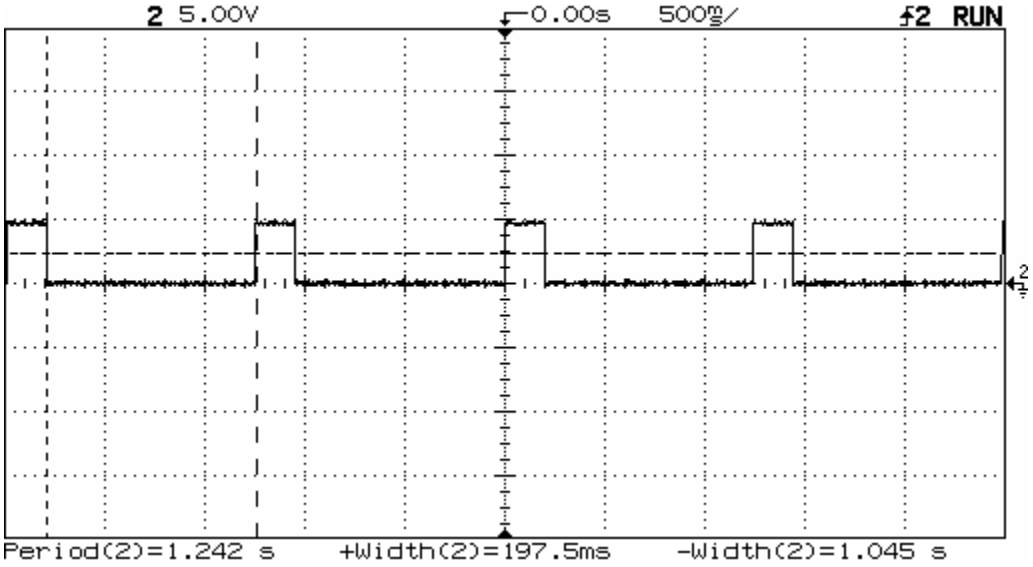


Figura 4.5 Gráfica del periodo de muestreo de la estación base (FIELD)

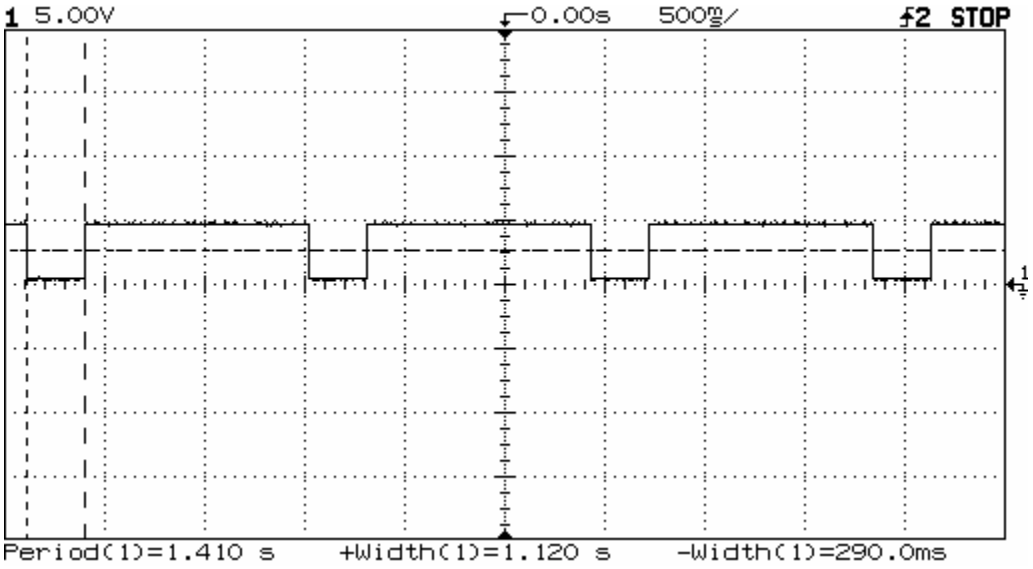


Figura 4.5.1 Gráfica de la frecuencia de oscilación (Q0)

En la Figura 4.5.2 se muestra la gráfica de la señal que sale del registro y es mandada al MOC para que este le mande el disparo directamente al TRIAC. Por lo tanto la gráfica nos indica que el foco se mantendrá encendido durante 1.12 segundos y apagado sólo 290 ms. Volviendo al ejemplo de la puerta, la señal nos indica que el seguro

se quitaría durante 1.12 segundos dando tiempo para que el usuario abra y pase, y cuando este cerrara, el *transponder* se alejaría de la base, cerrándose así el cerrojo automáticamente.

A continuación en las Figuras 4.5.2 y 4.5.3 se muestran las gráficas de la salida ya conectadas al foco

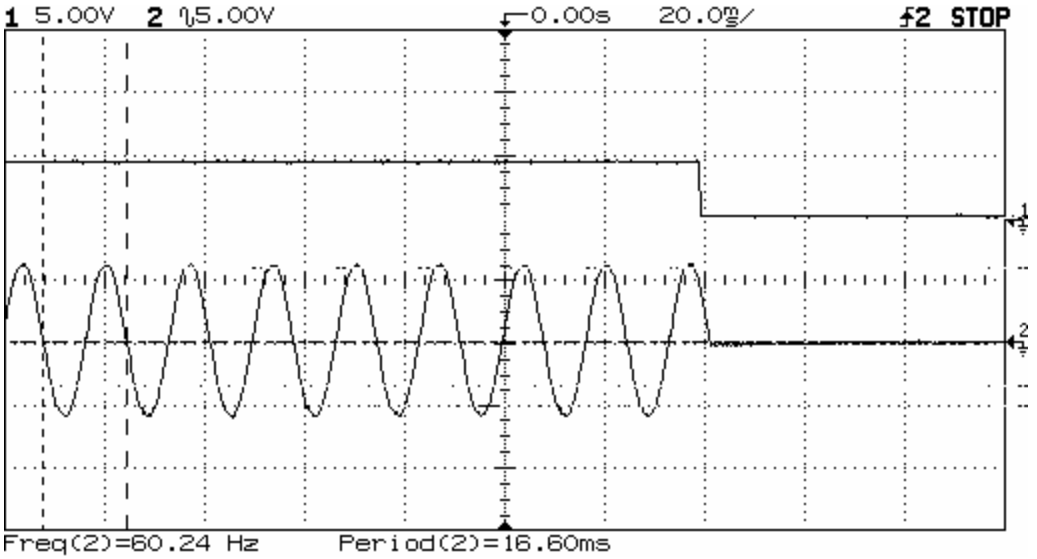


Figura 4.5.2 Gráfica de la salida (apagado)

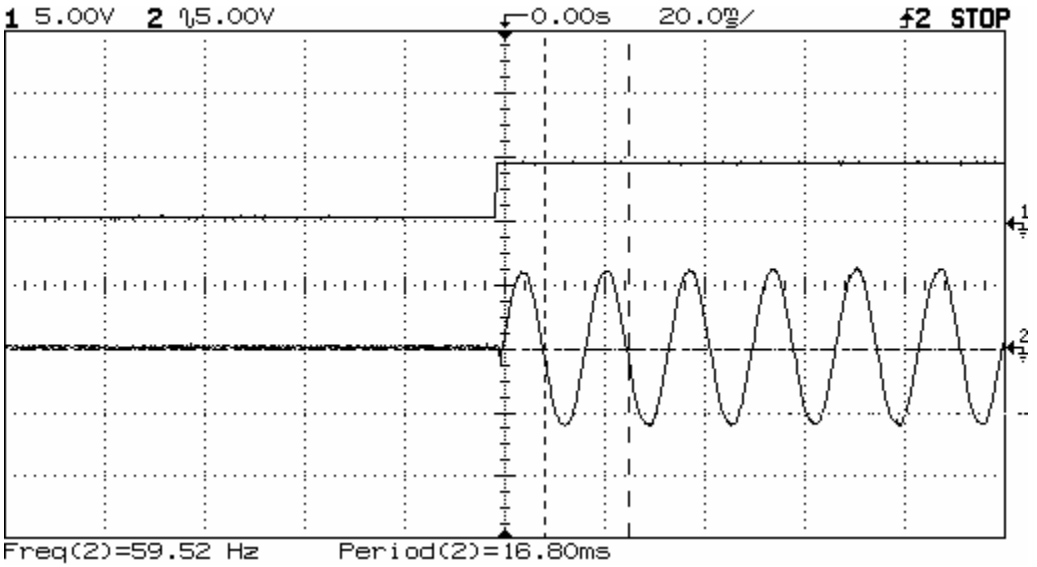


Figura 4.5.3 Gráfica de la salida (encendido)

En estas graficas se puede observar que la señal de alterna sólo se activa cuando se le manda el disparo al TRIAC por medio de la señal acondicionada (Q0). En la Figura 4.5.2 se muestra cómo la señal de alterna se apaga cuando se le quita la habilitación y en la Figura 4.5.3 se muestra de igual manera cómo la señal de alterna se enciende cuando se le manda la señal de habilitación.

En la Figura 4.5.4 se muestra un ciclo completo de apagado y encendido de la aplicación que se esta utilizando. Se puede observar cómo la señal de alterna sigue a la señal digital de habilitación y se enciende y apaga junto con esta.

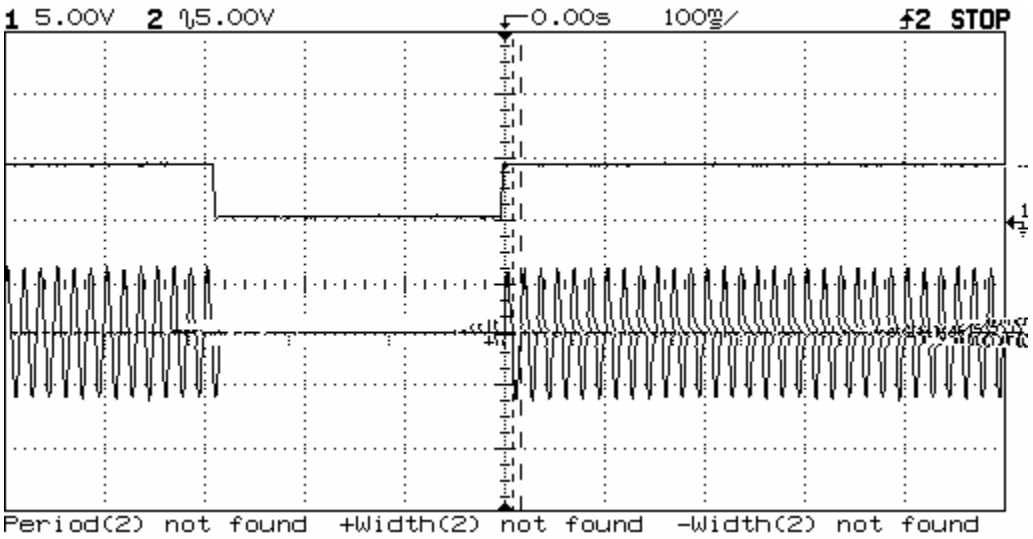


Figura 4.5.4 Gráfica de la salida (1 ciclo)

En la Figura 4.5.5 se encuentra la grafica que nos enseña varios ciclos de apagado y encendido de la aplicación para que se vea a un mas claramente y desde una perspectiva de una mayor frecuencia de muestreo, cómo trabaja el circuito armado.

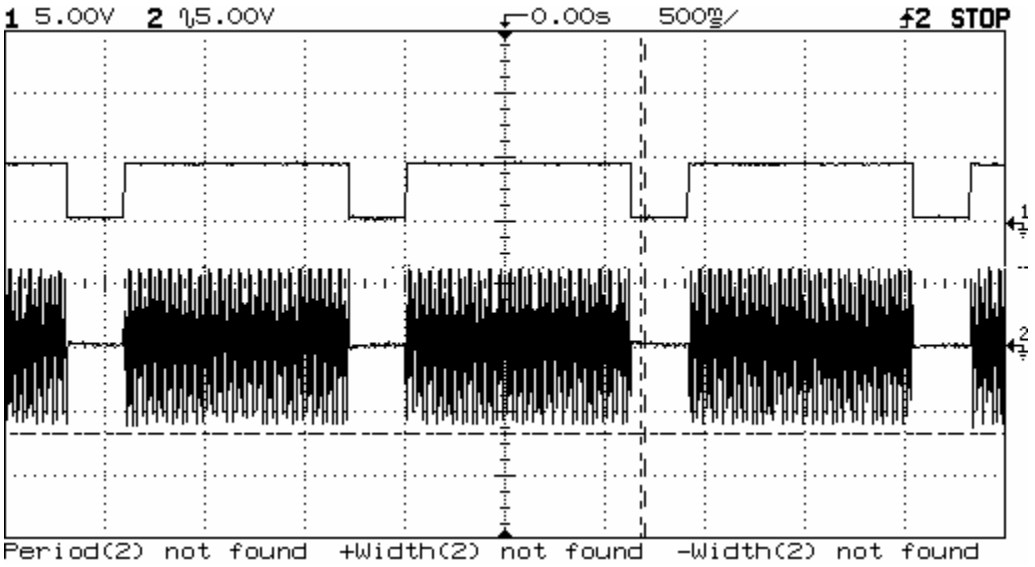


Figura 4.5.5 Gráfica de la salida (varios ciclos)

Como se puede ver en las gráficas anteriores el circuito armado funciona correctamente ya que cuando la señal de habilitación se vuelve cero, inmediatamente la señal analógica se extingue, apagando el foco que es nuestro indicador externo. Como este circuito ya esta trabajando con la aplicación de un foco, ahora se le puede conectar lo que sea a este circuito ya sea que este alimentado por voltaje directo o por corriente alterna ya que este circuito trabajará correctamente para cualquier aplicación.