

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

IMPLEMENTACIÓN DE UN PROTOTIPO DE ALARMA COMUNITARIA CONTROLADA POR RADIOFRECUENCIA, CON ENVÍO DE MENSAJE POR CELULAR A LA UNIDAD DE VIGILANCIA PARA AYUDA INMEDIATA

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN
ELECTRÓNICA Y TELECOMUNICACIONES**

VÍCTOR HUGO LANDETA VEGA

e-mail: vhl@hotmail.es

LUZ ADRIANA LAVERDE PAUCAR

e-mail: adrylpq@hotmail.com

DIRECTOR: ING. MÓNICA VINUEZA RHOR

e-mail: monivinueza@yahoo.ar

Quito, Noviembre 2010

DECLARACIÓN

Nosotros, Victor Hugo Landeta Vega y Luz Adriana Laverde Paucar, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Victor Hugo Landeta V.

Luz Adriana Laverde P.

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Victor Hugo Landeta Vega y Luz Adriana Laverde Paucar, bajo mi supervisión.

(Ing. Mónica Vinueza Rhor)
DIRECTOR DE PROYECTO

AGRADECIMIENTO

Esta tesis, si bien a requerido de mucho esfuerzo de los autores queremos agradecer a su directora de tesis, ya que de no serlo no hubiésemos llegado a su finalización, gracias a la Ing. Mónica Vinuesa por su colaboración, ha sido un soporte fuerte en el presente proyecto.

DEDICATORIA

La tesis la dedicamos con todo nuestro cariño a ti Dios que nos has dado la oportunidad de vivir y tener una familia maravillosa.

Con mucho cariño, principalmente a nuestros padres que nos han guiado en la vida y han estado a nuestro lado en todo momento. Gracias papá y mamá por darnos una carrera para nuestro futuro y por creer en nosotros siempre , por apoyarme en todo momento, brindándome su amor, solamente les estamos devolviendo lo que ustedes nos dieron en un principio por todo esto les agradecemos de todo corazón el que estén con nosotros.

Y no podemos terminar sin antes decirles, que sin ustedes a nuestro lado no lo hubiéramos logrado, tantas desveladas sirvieron

CONTENIDO

ÍNDICE GENERAL

DECLARACIÓN	II
CERTIFICACIÓN	III
AGRADECIMIENTOS	IV
DEDICATORIA	V
CONTENIDO	VI
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS	X
ÍNDICE DE FLUJOGRAMAS	X
RESUMEN	XI
PRESENTACIÓN	XII
CAPÍTULO 1	1
CAPÍTULO 2	34
CAPÍTULO 3	48
CAPÍTULO 4	96
BIBLIOGRAFÍA	99
ANEXOS	
1. Manual de Funcionamiento y Mantenimiento	
2. Módulo de radiofrecuencia TLP434A y RLP434A	
3. Decodificador HT12D	
4. Codificador HT12E	
5. Microcontrolador ATmega164P	
6. Microcontrolador ATmega8	
7. Módulo serial RS232	
8. Comandos básicos AT – Nokia	
9. Diagrama Circuito Rmisor y Receptor	
10. Diagramas de flujo de los microcontroladores ATmega8 y ATmega164P	
11. Programa del microcontrolador ATmega8 y ATmega164P	

CAPÍTULO 1

1. FUNDAMENTO TEÓRICO	
1.1 MÉTODOS DE TRANSMISIÓN Y RECEPCIÓN DE DATOS	1
1.1.1 Modulación digital	1
1.1.2 Modulación Ask.....	1
1.1.3 Modulación Fsk	2
FSK de banda reducida o banda angosta.....	4
FSK de banda ancha	4
1.1.4 Modulación Psk	4
1.1.5 Comunicación por RF (Radiofrecuencia).....	6
1.1.5.1 Módulos RF	6
1.1.5.2 Espectro Radioeléctrico	6
La banda UHF	8
Zonas de Fresnel.....	9
1.1.6 Transmisor TLP434 y Receptor RLP434 inalámbrico(RF).....	10

	Transmisor TLP434	10
	Receptor RLP464.....	11
1.1.7	Modos de transmisión.....	14
	Simplex (SX)	14
	Semidúplex (HDX, de Half duplex)	14
	Dúplex total (FDX, de full duplex).....	15
1.2	SISTEMAS DE CODIFICACIÓN Y DECODIFICACIÓN.....	16
	Decodificación.....	16
	Codificación	16
1.2.1	Módulo codificador - HT12E	18
1.2.2	Módulo decodificador - HT12D	20
1.3	MICROCONTROLADORES	21
1.3.1	Módulo microcontrolador - Atmega 164P.....	22
1.3.2	Módulo microcontrolador - Atmega8	25
1.4	TRANSMISIÓN DE MENSAJES VÍA CELULAR.....	30
1.4.1	Teléfonos GSM.....	30
1.4.2	Lenguaje de comunicación	30
	1.4.2.1 Acción realizada por el módem	30
	1.4.2.2 Comandos	32

CAPÍTULO 2

2.	DISEÑO DEL CIRCUITO	
2.1	CIRCUITO DE LA ALARMA.....	34
2.1.1	Fuente	34
2.1.2	Alarma	35
2.2	CIRCUITO DE RADIOFRECUENCIA	37
2.2.1	Circuito receptor	37
2.2.2	Circuito transmisor	38
	2.2.2.1 Fuente.....	38
	2.2.2.2 Transmisor	40
2.3	ACOPLAMIENTO HACIA EL CELULAR.....	42
2.4	CONEXIÓN CON EL MICROCONTROLADOR.....	42

CAPÍTULO 3

3.	IMPLEMENTACIÓN DEL CIRCUITO	
3.1	ACOPLAR LOS CIRCUITOS	48
3.1.1	PROGRAMA DEL TRANSMISOR.....	48
	Configuración inicial	48
	Funcionamiento inicial del programa en el dispositivo físico.....	52
	Manejo de instrucciones mediante el teclado	53

Funcionamiento del teclado en el dispositivo físico.....	54
Subrutina.....	58
Procedimiento para el envío de radiofrecuencia en el dispositivo físico	58
Procedimiento para direccionar la información del decodificador.....	60
3.1.2 PROGRAMA DEL RECEPTOR	61
Configuración inicial	61
Programa principal	65
Comunicación del microcontrolador con el celular.....	66
Funcionamiento físico del celular con el microcontrolador	66
Comunicación del receptor y emisor, mediante radio frecuencia	67
Funcionamiento físico de radio frecuencia.....	67
Comunicación del sensor magnético con el microcontrolador.....	68
Funcionamiento físico del sensor magnético.....	70
Desarrollo del sub procedimiento para comunicar el microcontrolador con el celular	70
Sub procedimiento para esperar la respuesta “OK”	74
Sub procedimiento para limpiar buffer UART2.....	75
Sub procedimiento para enviar mensajes	76
Sub proceso para recibir mensajes.....	77
Subrutina para validar mensaje.....	78
Encendido y apagado de led	80
Sub proceso para revisar datos de radio frecuencia.....	82
Funcionamiento físico cuando recibe radio frecuencia	83
Sub proceso serial	84
3.2 PRUEBAS DE FUNCIONAMIENTO	91
3.2.1 Prueba de distancia	91
3.2.2 Prueba de sonido	92
3.2.3 Prueba de operadora celular	94
3.3 RESULTADOS	95
CAPÍTULO 4	
4. CONCLUSIONES Y RECOMENDACIONES	
4.1 CONCLUSIONES	96
4.2 RECOMENDACIONES	97

ÍNDICE DE FIGURAS

CAPÍTULO 1

Figura. 1.1.	Modulación ASK	2
Figura. 1.2.	Modulación FSK.....	3
Figura. 1.3.	Modulación PSK.....	5
Figura. 1.4.	Zonas de Fresnel	9
Figura. 1.5.	Líneas de vista de la primera zona de fresnel	10
Figura. 1.6.	Distribución de pines transmisor TLP 434A	11
Figura. 1.7.	Circuito básico del transmisor TLP 434A	12
Figura. 1.8.	Distribución de pines receptor RLP 434A	13
Figura. 1.9.	Circuito básico del receptor RLP 434A	13
Figura. 1.10.	Transmisión Simplex	14
Figura. 1.11.	Transmisión Half Dúplex.....	15
Figura. 1.12.	Transmisión Full Dúplex	15
Figura. 1.13.	Proceso de codificación y decodificación.....	16
Figura. 1.14.	Circuito básico del decodificador	17
Figura. 1.15.	Circuito básico del codificador	17
Figura. 1.16.	Módulo codificador HT12E, 8 direcciones y 4 datos	19
Figura. 1.17.	Módulo decodificador HT12D, 8 direcciones y 4 datos.....	21
Figura. 1.18.	CPU – AVR	23
Figura. 1.19.	Diagrama de bloques – Atmega164P.....	24
Figura. 1.20.	Módulo microcontrolador - ATmega164P.....	25
Figura. 1.21.	Diagrama de bloques – Atmega8.....	28
Figura. 1.22.	Módulo microcontrolador – Atmega8	29

CAPÍTULO 2

Figura. 2.1.	Regulador de voltaje a 5 voltios.....	35
Figura. 2.2.	Puntos de prueba para medir voltajes	35
Figura. 2.3.	Activación de la alarma	36
Figura. 2.4.	Indicador de funcionamiento	36
Figura. 2.5.	Sensor magnético	37
Figura. 2.6.	Sensor presencia.....	37
Figura. 2.7.	Funcionamiento del receptor.....	38
Figura. 2.8.	Regulador de voltaje a 5 voltios.....	39
Figura. 2.9.	Puntos de prueba para medir voltajes	40
Figura. 2.10.	Funcionamiento del transmisor.....	41
Figura. 2.11.	Comunicación del microcontrolador y el codificador	41
Figura. 2.12.	Conexión del celular al circuito	42
Figura. 2.13.	Pines del microcontrolador ATmega8	43
Figura. 2.14.	Teclado de dos teclas de pulso.....	43
Figura. 2.15.	Pines del microcontrolador ATmega164P.....	45
Figura. 2.16.	Diagrama del circuito transmisor.....	46
Figura. 2.17.	Diagrama del circuito receptor.....	47

CAPÍTULO 3

Figura. 3.1.	Distancia receptor, transmisor	92
--------------	--------------------------------------	----

ÍNDICE DE TABLAS

CAPÍTULO 1

Tabla. 1-1	Designaciones de banda CCIR.....	7
Tabla. 1-2	Sistemas que funcionan en UHF.....	8
Tabla. 1-3	Características principales TLP 434A	11
Tabla. 1-4	Características DC y eléctricas principales RLP 434A	12
Tabla. 1-5	Características eléctricas HT12E	18
Tabla. 1-6	Descripción de pines del módulo codificador HT12E.....	19
Tabla. 1-7	Características eléctricas HT12D.....	20
Tabla. 1-8	Descripción de pines del módulo codificador HT12D	21
Tabla. 1-9	Descripción del microcontrolador ATmega164P	27
Tabla. 1-10	Descripción de pines del microcontrolador ATmega8	29

CAPÍTULO 3

Tabla. 3-1	Variables presentes en el lenguaje Basic	50
Tabla. 3-2	Variables presentes en el lenguaje Basic	62
Tabla. 3-3	Proceso de encendido de la alarma por distancia.....	92
Tabla. 3-4	Efectividad del sonido de la bocina con fuente externa.....	93
Tabla. 3-5	Efectividad del sonido de la bocina solo con batería interna.....	93
Tabla. 3-6	Tiempos de respuesta de operadora celular Movistar.....	94
Tabla. 3-7	Tiempos de respuesta de operadora celular Porta.....	94

ÍNDICE DE FLUJOGRAMAS

CAPÍTULO 3

Flujograma. 3.1.	Animación inicial	55
Flujograma. 3.2.	Manejo de instrucciones mediante teclado.....	57
Flujograma. 3.3.	Envío de radiofrecuencia	59
Flujograma. 3.4.	Dirección del receptor	60
Flujograma. 3.5.	Comunicación con el teléfono celular	68
Flujograma. 3.6.	Verificación de radiofrecuencia	69
Flujograma. 3.7.	Gabinete abierto.....	71
Flujograma. 3.8.	Gabinete cerrado.....	72
Flujograma. 3.9.	Validar mensaje	81
Flujograma. 3.10.	Intermitencia de led	82
Flujograma. 3.11.	Revisar radio frecuencia	85
Flujograma. 3.12.	Sub proceso serial.....	86
Flujograma. 3.13.	Almacenamiento de datos en la EEPROM.....	88
Flujograma. 3.14.	Diagrama completo del microcontrolador ATmega8.....	89
Flujograma. 3.15.	Diagrama completo del microcontrolador ATmega 164P.....	90

RESUMEN

El presente trabajo desarrolla un prototipo de alarma comunitaria basada en la tecnología de radio frecuencia que funciona en la banda UHF cuyos dispositivos trabajan en los 433.9 Mhz.

La primera parte de este proyecto consiste en identificar los elementos que se pueden utilizar en el proyecto y se establece la comunicación entre transmisor y receptor de radio frecuencia con un alcance de 50 mts, y dependiendo de la línea de vista y el medio se puede superar esta distancia,

La segunda parte se acopla el circuito de la alarma a un teléfono celular ubicada en la cabina mediante comandos AT, los cuales permiten manipular el modem del teléfono; y de esta manera se logra enviar mensajes cortos hacia otro teléfono celular el cual se encuentra en una unidad de auxilio(Policía).

La tercera parte trata de la alimentación, el cual trabaja con 120V de manera constante; en caso de corte de energía eléctrica, la cabina (receptor, sirena, celular) está prevista en su interior una batería recargable de 12 volt, el cual provee de energía durante un lapso de tiempo en espera que la energía eléctrica retorne.

La cuarta parte hace referencia a conclusiones y recomendaciones observadas en el transcurso del diseño y montaje del sistema.

PRESENTACIÓN

La sociedad ha estado siempre en busca de un medio de ayuda frente a la delincuencia, por ello las personas han incluido en su entorno, diferentes tipos de tecnología para este fin.

En este caso se ha visto varias debilidades en las alarmas comunitarias, y se ha realizado un prototipo para solventar estos inconvenientes vistos; por lo cual se ha diseñado una alarma comunitaria con un dispositivo inalámbrico que permite a las personas que lo utilicen, activar de forma personalizada un sistema de prevención en caso de presentarse algún contratiempo.

La alarma comunitaria activa un elemento sonoro y al mismo tiempo envía un mensaje de auxilio a un número celular específico a un centro de vigilancia inmediata. El sistema de alarmas comunitarias, tiene un alcance de 50 m de radio la cual se logra superar en lugares abiertos.

CAPÍTULO 1

1. FUNDAMENTO TEÓRICO

1.1 MÉTODOS DE TRANSMISIÓN Y RECEPCIÓN DE DATOS

1.1.1 MODULACIÓN DIGITAL^{[1][2]}

La modulación tiene como objetivo adaptar una señal digital al medio radioeléctrico por el cual se va a transmitir. Es decir, una señal portadora y una señal moduladora (señal de datos) son multiplicadas, dando como resultado una señal modulada, donde la señal portadora es modificada en función de la señal moduladora.

La señal portadora permite el envío de una señal, gracias a una frecuencia alta, característica de la señal portadora.

Para realizar este proceso se cuenta con tres métodos, que son: ASK (*Amplitude Shift Keying*), FSK (*Frequency Shift Keying*) y PSK (*Phase Shift Keying*), los cuales permiten según la necesidad manipular la amplitud, frecuencia y fase respectivamente.

1.1.2 MODULACIÓN ASK (Modulación por desplazamiento de amplitud)^[3]

ASK, consiste en variar la amplitud de la señal portadora entre dos valores posibles 1L ó 0L según la señal moduladora, y de esta manera nos permite observar presencia o ausencia de amplitud, en la señal portadora.

La señal modulada no es más que la multiplicación de las dos señales (portadora y moduladora), la cual se puede observar en la Figura 1.1

Este tipo de modulación es usada en distintos módulos de transmisión de radiofrecuencia.

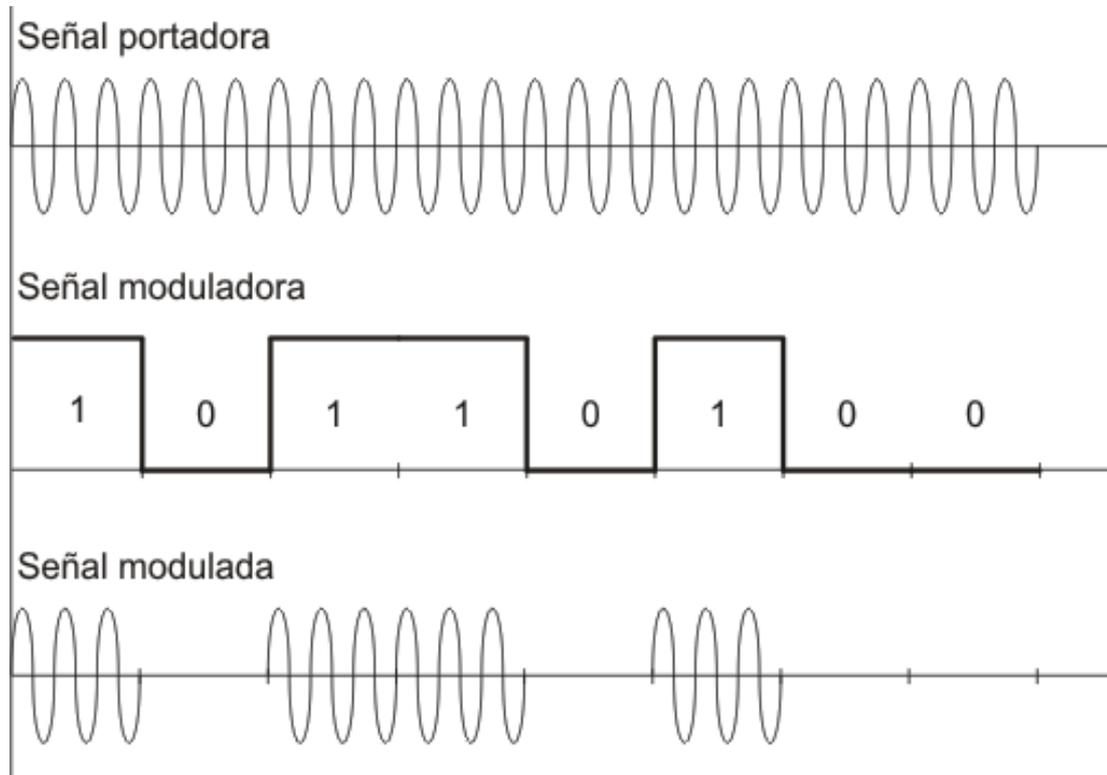


Figura 1.1 Modulación ASK^[4]

Entre las ventajas se puede mencionar que esta tecnología es más barata, y de bajo consumo de potencia, debido al poco uso de ancho de banda con respecto a otros métodos de modulación.

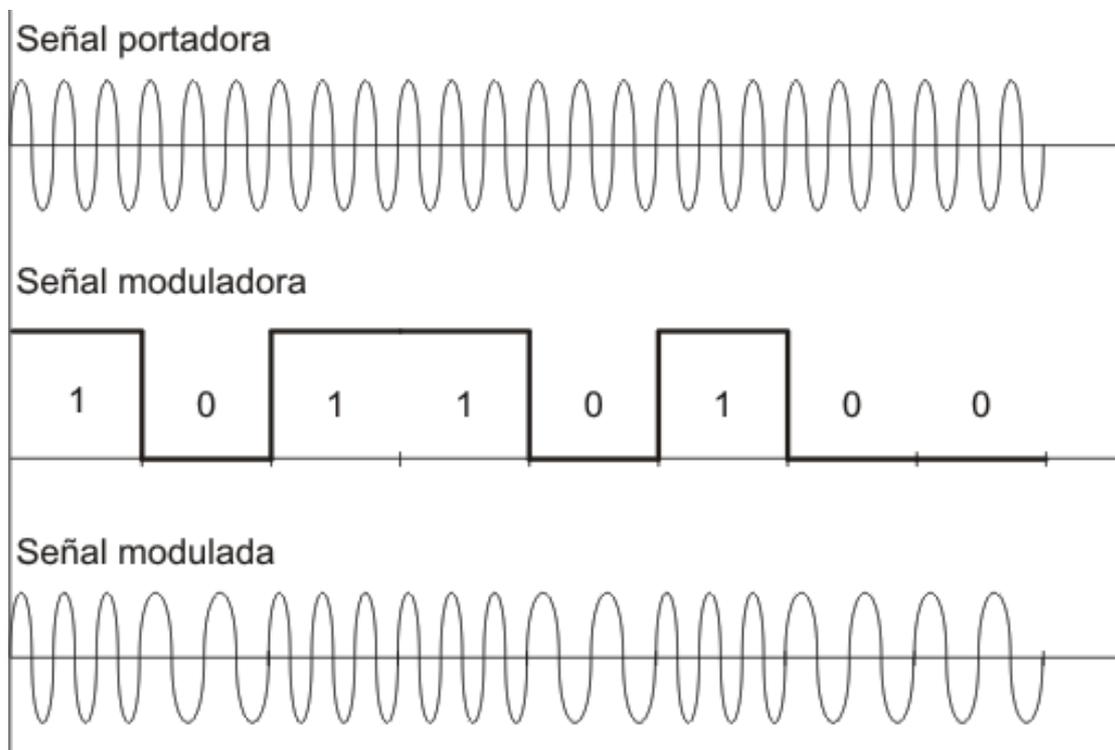
Entre las desventajas está la fragilidad de la señal modulada frente al ruido, ya que el ruido interfiere con la señal, provocando errores en su transmisión, y no se recuperan los datos enviados de manera correcta al finalizar su proceso.

1.1.3 MODULACIÓN FSK (Modulación por desplazamiento de frecuencia)^[5]

FSK, es una forma de modulación de ángulo donde la amplitud es constante y funciona como un conmutador en respuesta a la señal moduladora.

La señal moduladora es una señal binaria, que varía entre dos valores discretos $1L = 1$ y $0L = -1$, la cual permite desviar la frecuencia de la portadora según la desviación máxima que acepte la señal. Es decir, al tener en la señal moduladora un 1L o 0L la frecuencia de salida varía entre dos frecuencias, una mínima y una máxima de igual magnitud pero en sentidos opuestos.

La desviación máxima permite establecer la distancia máxima a la que se situará la frecuencia, $(f_c \pm \Delta f)$ con respecto a la frecuencia central de la portadora. La cual se puede observar en la Figura 1.2



Figur

a 1.2 Modulación FSK^[6]

FSK, en bajas velocidades se usa para comandos de teleimpresoras y apta para vía telefónica, radio frecuencia y en telemetría. Su velocidad límite es aproximadamente de 1.800 bps (bits por segundo).

FSK, es menos sensible al ruido con respecto a ASK, y tecnológicamente costosa debido a su complejidad, tanto para la implementación de circuitos como para la transmisión.

En FSK, existe un uso ineficiente del ancho de banda.

El índice de modulación tiene gran incidencia en la señal modulada y determina los dos tipos fundamentales de FSK.

1.1.3.1 FSK de banda reducida o banda angosta ^[7]

“Si el índice de modulación es pequeño, $m_f < (\pi/2)$, esto significa que la variación de frecuencia de la señal modulada produce una diferencia de fase menor que $\pi/2$. La única diferencia es que en este caso, la amplitud de las armónicas se ve afectada por la frecuencia o sea, se tiene una pequeña modulación de amplitud, superpuesta a la FSK.”

1.1.3.2 FSK de banda ancha ^[7]

“Las ventajas de FSK sobre ASK se hacen notables cuando el índice de modulación es grande es decir $m_f > (\pi/2)$. Con esta condición se aumenta la protección contra el ruido y las interferencias, obteniendo un comportamiento más eficiente respecto a ASK, puesto que en este caso la pequeña modulación de amplitud mencionada en el caso de FSK de banda angosta, se hace despreciable.

La desventaja es que es necesario un mayor ancho de banda, debido a la mayor cantidad de bandas laterales (un par por cada armónica).”

1.1.4 MODULACIÓN PSK (Modulación por desplazamiento de fase) ^[8]

PSK, Consiste en variar la fase de la señal portadora de acuerdo a la señal moduladora.

Existen dos alternativas de modulación PSK:

PSK convencional, donde se tienen en cuenta el valor de salto de la fase. Pero este necesita tener una portadora en el receptor para sincronización o un código para auto sincronización por lo cual necesita de un receptor más complejo.

PSK diferencial, en la cual se considera el valor del salto, con respecto al salto anterior, no necesita recuperar la señal portadora en el emisor, y la información no se encuentra en la fase sino en las transiciones.

Las consideraciones que se observan en la Figura 1.3 son válidas para ambos casos.

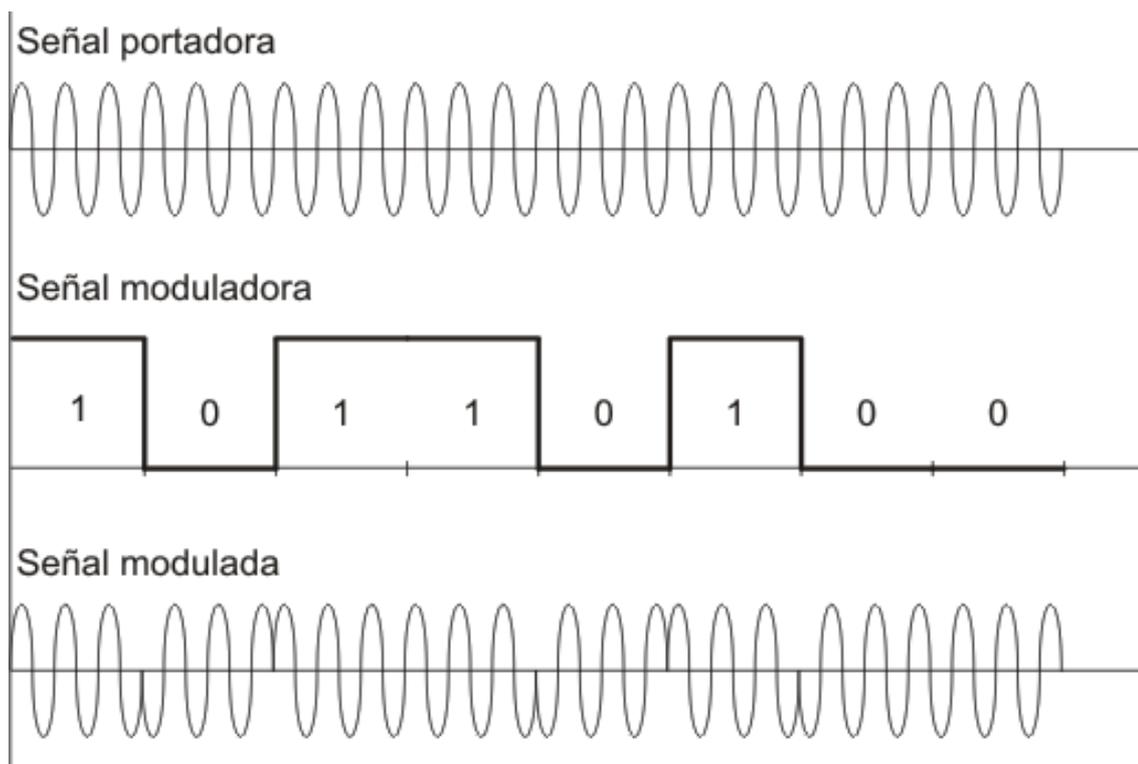


Figura 1.3 Modulación PSK^[9]

La ventaja de este tipo de modulación es que aumenta la velocidad de transmisión tomando en cuenta las características del canal por el cual se transmite.

La desventaja es la alta complejidad que presenta en el circuito receptor y por ende un mayor costo en razón de su tecnología.

1.1.5 COMUNICACIÓN POR RF (Radiofrecuencia)

La transmisión de datos en RF utiliza el espacio como medio de transmisión, es decir no necesita de un cableado para la transmisión, ya que la información está contenida en ondas electromagnéticas y para ello es necesario el empleo de módulos RF.

1.1.5.1 Módulos RF ^[10]

Los módulos RF están conformados por un transmisor, un receptor y una antena, permitiendo la comunicación de datos analógicos como digitales según el modelo que se use. Trabajan en frecuencias de 315MHz, 418MHz, 433.92MHz en la banda UHF (*Ultra high frequency*) y no requieren un protocolo estándar, alcanzando buenas distancias.

Además consta de un condensador variable o bobina variable para ajustar la sintonía (frecuencia) para una mejor respuesta del sistema pero no es aconsejable variar, ya que de no tener los instrumentos necesarios para calibrar la sintonía, la comunicación se perderá entre los módulos.

Estos módulos son utilizados para controles remotos, sistemas de seguridad, identificación y transmisión periódica de datos y utilizan modulación digital ASK, FSK y PSK dependiendo del módulo a usarse y la aplicación para el cual se requiera.

1.1.5.2 Espectro Radioeléctrico ^[11]

El campo de las telecomunicaciones tiene como objetivo convertir la información a energía electromagnética para luego ser transmitida. El espectro total útil se divide en bandas de frecuencia, a las que se dan nombres y números descriptivos, y algunas de ellas se subdividen a su vez en diversos tipos de servicios. Las designaciones de bandas según el Comité Consultivo Internacional de radio (CCIR) se muestran en la Tabla 1-1

Designación de bandas CCIR			
Número de Banda	Intervalo de frecuencias	Designación	Usos
2	30 Hz – 300 Hz	ELF (frecuencias extremadamente bajas)	Distribución eléctrica, telemetría de baja frecuencia.
3	0.3 kHz – 3 kHz	VF (frecuencias de voz)	Canales de frecuencia de voz.
4	3 kHz – 30 kHz	VLF (frecuencias muy bajas)	Enlaces de radio a gran distancia.
5	30 kHz – 300 kHz	LF (bajas frecuencias)	Enlaces de Radio a gran distancia, ayuda a la navegación aérea y marítima.
6	0.3 MHz – 3 MHz	MF (frecuencias intermedias)	Radiodifusión AM.
7	3 MHz – 30 MHz	HF (frecuencias altas)	Comunicaciones de todo tipo a media y larga distancia.
8	30 MHz – 300 MHz	VHF (frecuencias muy altas)	Enlaces de radio a corta distancia, televisión FM
9	300 MHz – 3GHz	UHF (frecuencias ultra altas)	Enlaces de radio, Radar, TV, ayuda a la navegación aérea.
10	3 GHz – 30 GHz	SHF (frecuencias súper altas)	Radar, enlaces de radio
11	30 GHz – 300 GHz	EHF(frecuencias extremadamente altas)	
12	0.3THz – 3 THz	Luz infrarroja	
13	3 THz – 30 THz	Luz infrarroja	
14	30 THz – 300 THz	Luz infrarroja	
15	0.3 PHz – 3 PHz	Luz visible	
16	3 PHz – 30 PHz	Luz ultravioleta	
17	30 PHz – 300 PHz	Rayos X	
18	0.3 EHz – 3 EHz	Rayos gamma	
19	3 EHz – 30 EHz	Rayos cósmicos	

Tabla 1-1 Designaciones de banda CCIR (Comité Consultivo Internacional de Radiocomunicaciones)^[11]

Sistemas en banda UHF	
Rango de Frecuencia	Usos
300 a 420 MHz	Meteorología y dos vías federales de uso
420 a 450 MHz	De radio localización Gobierno y 70cm de radioaficionados
450 - 470 MHz	UHF Banda de Negocios, GMRS(<i>General movil radio service</i>), FRS (<i>Family radio service</i>), de seguridad pública
470 a 512 MHz	Los canales de TV 14-20
512 a 698 MHz	21-51 canales de TV, el canal 34 se utiliza a veces para el radar, el canal 37 se utiliza para la radioastronomía
698 - 806 MHz	Anteriormente utilizada para los canales de TV 52-69
806 a 824 MHz	Busca personas, anteriormente utilizada para los canales de TV 70-72
824 - 849 MHz	Terminal (teléfono móvil), utilizado anteriormente para los canales de TV 73-77
849 a 869 MHz	La seguridad pública 2-way (bomberos, policía, ambulancia), utilizado anteriormente para los canales de TV 77-80
869 a 894 MHz	Estación de base, utilizado anteriormente para AMPS (Sistema telefónico móvil avanzado), inicialmente utilizada para los canales de TV 80-83
902 - 928 MHz	La banda ISM, teléfonos inalámbricos y equipo de música, RFID(Identificación por radiofrecuencia), enlaces de datos, 33cm de radioaficionados de banda
928 a 960 MHz	Mezcla de estudio de los enlaces de transmisión, el móvil 2-Way, de paginación
1240 -1300 MHz	23cm de radioaficionados de banda
1850 - 1910 MHz	Teléfono móvil PCS
1920 - 1930 MHz	Teléfonos inalámbricos DECT (Digital Enhanced Cordless Telecommunications)
1930 - 1990 MHz	Teléfono móvil PCS
2300 - 2310 MHz	13cm de radioaficionados de banda - segmento inferior
2310 - 2360 MHz	De radio por satélite (Sirius y XM)
2390 - 2450 MHz	13cm de radioaficionados de banda - segmento superior
2400 a 2483,5 MHz	ISM, IEEE 802.11, 802.11b, 802.11g Wireless LAN, IEEE 802.15.4

Tabla 1-2. Sistemas que funcionan en UHF(Ultra High Frequency)^[12]

La banda UHF^[12], tiene una atenuación de 1dB si la primera zona de Fresnel está despejada, y por ser una frecuencia relativamente alta, la antena necesaria es

igualmente pequeña. Son señales entre los límites de 300 MHz a 3GHz, exclusivamente la propagación directa, posibilidad de enlaces por reflexión o a través de satélites artificiales y las usa la emisión comercial de televisión, en los canales 14 a 83, en los servicios móviles de comunicación terrestre, teléfonos celulares algunos sistemas de radar y de navegación, radiofrecuencia y los sistemas de radio por microondas y por satélite. En la Tabla 1-2, se indica el rango de frecuencia en UHF y el uso asignado para las mismas.

Zonas de fresnel^[13], se deriva de la teoría electromagnética respecto de la expansión de la misma al viajar en el espacio libre, donde se difunde una onda luego de ser emitida por una antena. Como resultado de los factores que afectan la propagación se tiene la absorción, reflexión, difracción e Interferencia, resultando un aumento o disminución en el nivel de intensidad de la señal recibida.

Donde el desfase entre las ondas en dicho volúmen no debe superar los 180°, debido a que estamos transmitiendo en la tierra y no en el espacio eso ocasiona rebotes, pero estos rebotes no siempre son favorables ya que pueden llegar en fase o en contrafase. Las zonas de fresnel pares contribuyen negativamente y las zonas de fresnel pares contribuyen positivamente.

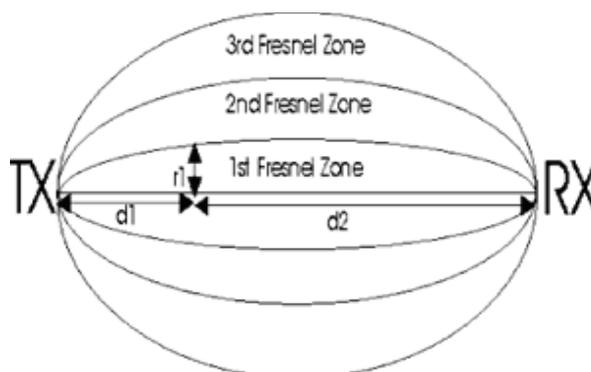


Figura 1.4 Zonas de Fresnel^[13]

En el caso de la radiofrecuencia, es necesario al menos el 80% de la primera zona de fresnel libre. Para tener mejores resultados es necesario cumplir con las siguientes indicaciones:

- 1.- Utilización de antenas correctas.
- 2.- Ausencia de condiciones climatológicas adversas.
- 3.- Visión directa.
- 4.- Altura correcta donde se colocarán las antenas.

Y mientras menos obstáculos haya en el camino hacia el receptor es mejor.

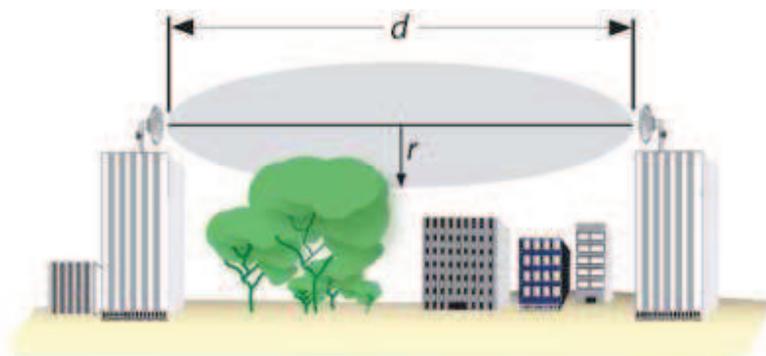


Figura 1.5 Línea de vista de la primera zona de fresnel^[13]

1.1.6 TRANSMISOR TLP434 Y RECEPTOR RLP434 INALÁMBRICO (RF)^{[14][15]}

El transmisor TLP434, y receptor RLP434 permiten comunicación inalámbrica a una velocidad de 9600 bps de un punto a otro evitando la tediosa comunicación por medio de un cableado y más aun cuando estos puntos están lejanos.

Trabajando juntos entablan comunicación hasta de 160m en vista directa y 30m en interiores. Entre las aplicaciones más usadas por los módulos de radiofrecuencia, receptor y transmisor son:

Sistemas de seguridad inalámbrica, alarmas de seguridad e incendios, teledetección, comunicación de datos, sistema de buscapersonas, sistemas de puertas (libre de llaves), apertura de garajes y compuertas, control de luminosidad, sistemas de monitoreo médico, sistemas de llamado, transferencia periódica de datos, automatización de uso industrial y residencial.

Transmisor TLP434.^[16] Módulo transmisor que emite señales de radiofrecuencia a una frecuencia fija de 433 MHz, la forma de transmitir es modulación por

desplazamiento de amplitud (ASK), de esta forma los datos digitales se transmiten por amplitud, este módulo de un tamaño de 13.3x10.3mm, consta de 4 pines que se muestra en la Figura 1.6 y sus características en la Tabla 1-3.

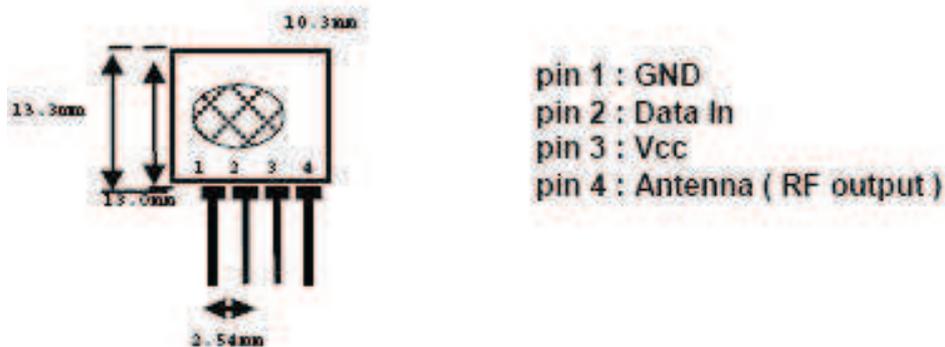


Figura 1.6 Distribución de pines^[16]

Símbolo	Parámetro	Condiciones	Min	Typ	Max	Unid
Vcc	Voltaje de operación		2.0	-	12.0	V
Icc1	Pico de corriente (2V)		-	-	1.64	mA
Icc2	Pico de corriente (12V)		-	-	19.4	mA
Vh	Entrada alto voltaje	Idata=0μA(high)	Vcc-0.5	Vcc	Vcc+0.5	V
Vl	Entrada bajo voltaje	Idata=0μA(low)	-	-	0.3	V
FO	Frecuencia absoluta	315MHz module	314.8	315	315.2	MHz
PO	RF Potencia de salida – 50ohm	Vcc=9V+12V	-	16	-	dBm
		Vcc=5V+6V	-	14	-	dBm
DR	Velocidad de datos	Codificación externa	512	4.8K	200K	Bps

Tabla 1-3 Características principales^[16]

En la Figura 1.7 se observa la conexión del módulo transmisor TLP434 en conjunto a varios elementos que recomienda el fabricante para su funcionamiento básico.

Receptor RLP434. ^[17] Módulo receptor, que permite recibir información a una frecuencia que puede ser ajustada, la forma de recibir es demodulación por desplazamiento de amplitud (ASK), de esta forma los datos digitales se reciben por amplitud. Tiene 8 pines, y el tamaño de 43.42x11.5mm.

En la Figura 1.8, se observa la distribución de pines y en la Tabla 1-4 sus características principales.

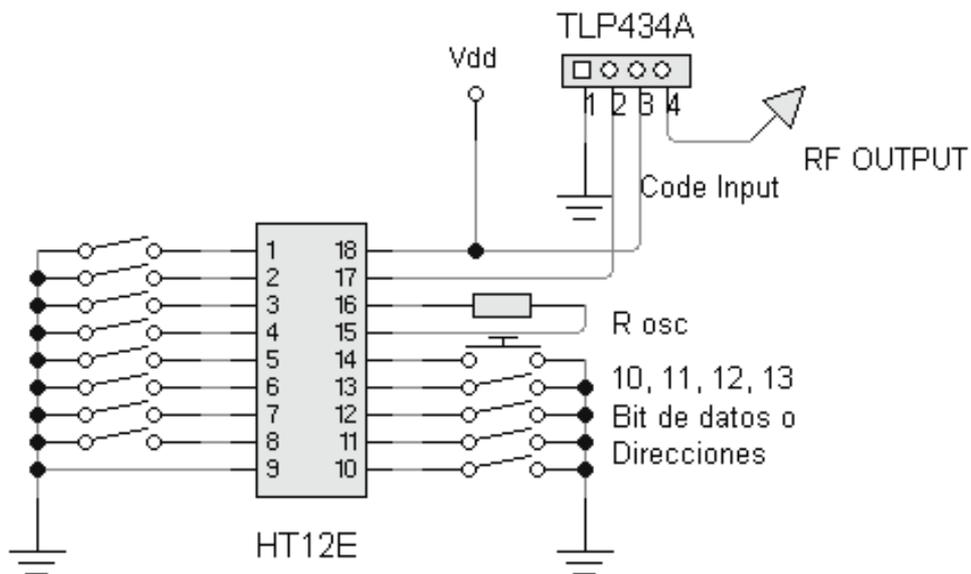


Figura 1.7 Circuito básico del transmisor^[17]

Características DC						
Símbolo	Parámetros	Condiciones	Min	Typ	Max	Unid
Vcc	Operación de voltaje		3.3	5.0V	6.0	V
Itot	Pico de corriente (2V)		-	4.5		mA
Vdata	Salida de datos	I _{data} =-10μA(high)	V _{cc} -0.5	-	V _{cc}	mA
		I _{data} =-10μA(low)	-	-	0.3	V
Características Eléctricas						
Características	SYM	Min	Typ	Max	Unid	
Operación de radio frecuencia	FC	315,418 and 433.92			MHz	
Sensibilidad	Pref		-110		dBm	
Ancho de canal			+/-500		KHz	
Ruido equivalente BW			4		KHz	
Receptor tiempo de subida			5		Ms	
Temperatura de operación	Top	-20	-	80	C	
Velocidad de datos de la placa base			4.8		KHz	

Tabla 1-4 Características DC y Eléctricas, principales RLP434^[17]

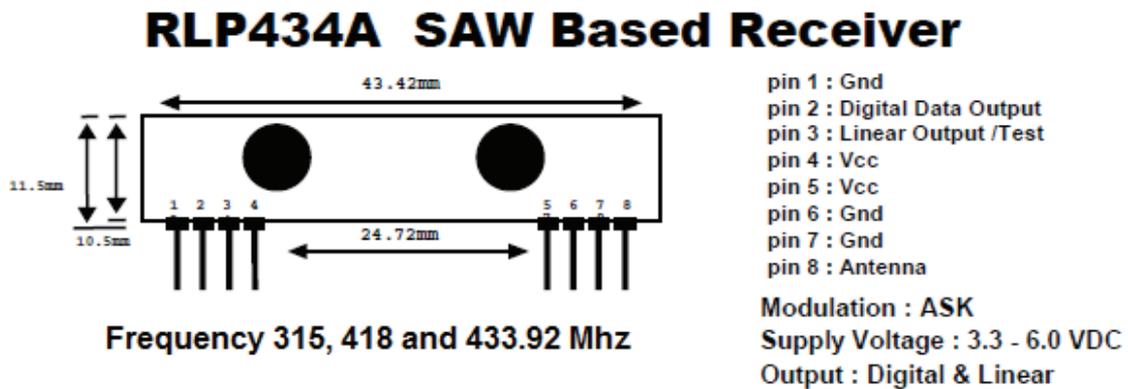


Figura 1.8 Distribución de pines ^[17]

En la figura 1.9 se observa la conexión del módulo receptor RLP434 en conjunto con varios elementos que recomienda el fabricante para su funcionamiento básico.

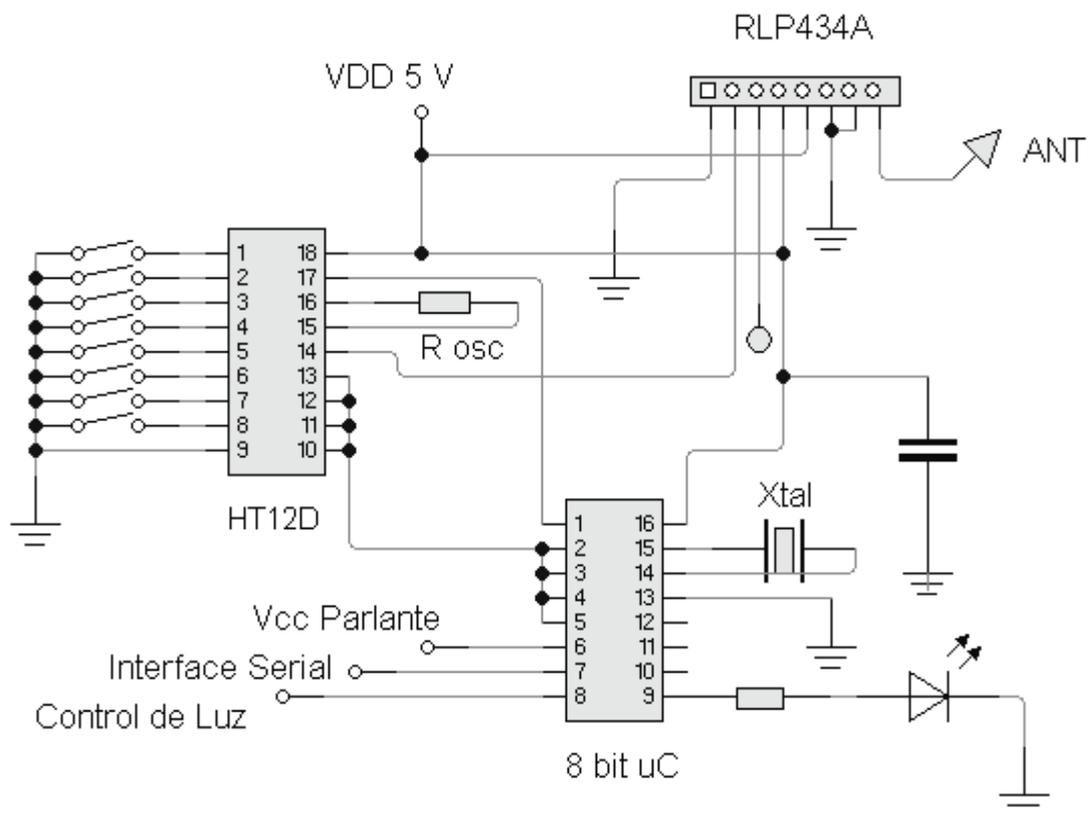


Figura 1.9 Circuito básico del receptor ^[17]

1.1.7 MODOS DE TRANSMISIÓN^[18]

Los sistemas electrónicos de comunicaciones requieren modos para transmitir, pero debido a la necesidad para la cual se requieran, se derivan algunos modos para satisfacer estas necesidades, y estas pueden ser en una o dos direcciones, a estos se les llama modos de transmisión.

Hay tres modos de transmisión posibles: simplex, semiduplex y full duplex.

- *Simplex (SX)*. Este modo de transmisión se hace en un solo sentido. Donde la estación base puede ser un transmisor o un receptor, pero no ambos a la vez, como ejemplo se tiene la emisión comercial de radio o televisión, ya que estas solo transmiten y en otro punto se recibe, como se observa en la Figura 1.10.



Figura 1.10 Transmisión Simplex^[19]

- *Semidúplex (HDX, de Half duplex)*. Este modo de transmisión se realiza entre dos estaciones y en dos sentidos, es decir se puede transmitir y recibir, pero no al mismo tiempo. Esta transmisión requiere que la información sea enviada y recibida de forma alternada en los dos sentidos donde la estación puede ser transmisora y receptora, pero no al mismo tiempo. Como ejemplo se tiene sistemas de radio en dos sentidos que usan botones para hablar (PTT, de

push-to-talk) para conectar sus transmisores, como son los radios de banda civil y de policía, que se observa en la Figura 1.11.

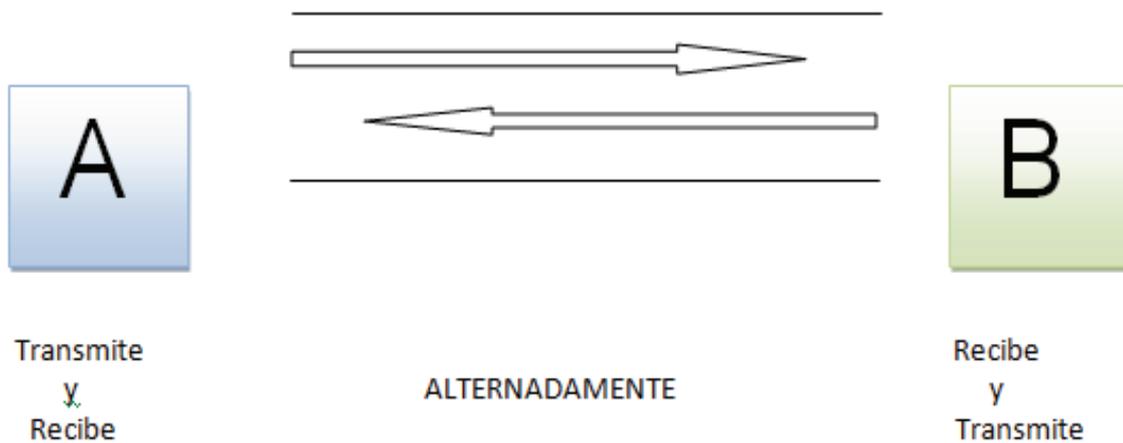


Figura 1.11 Transmisión Half Duplex^[19]

- *Dúplex total (FDX, de full duplex)*. Este modo de transmisión se realiza entre dos estaciones y puede hacerse en dos sentidos y al mismo tiempo. Es decir, las estaciones pueden transmitir y recibir en forma simultánea. Como ejemplo tenemos el sistema telefónico normal. Y se observa en la Figura 1.12

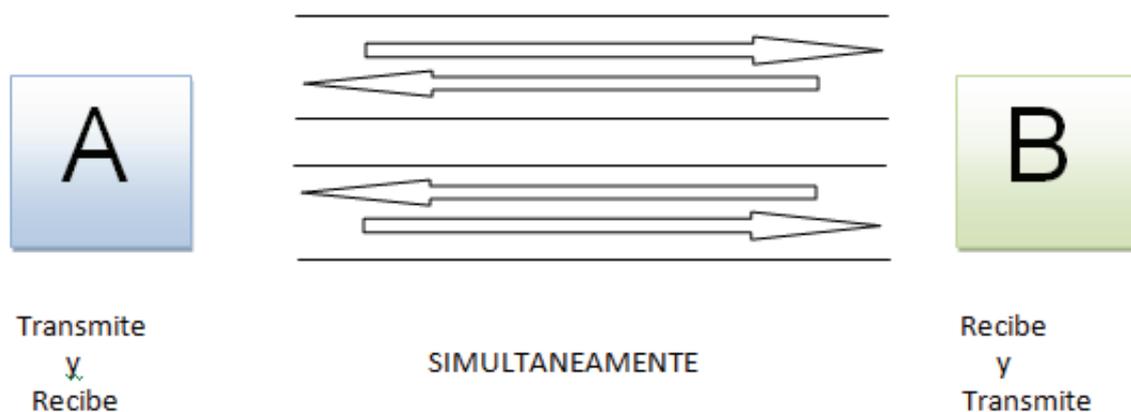


Figura 1.12 Transmisión Full Dúplex^[19]

1.2 SISTEMAS DE CODIFICACIÓN Y DECODIFICACIÓN

Un sistema de codificación y decodificación trabaja mediante códigos, estos son un conjunto de unos y ceros que se usan para representar a un cierto mensaje mediante un código binario.

El proceso de codificación y decodificación es simple e intuitivo, si se ve en la Figura 1.11 se observa lo siguiente, el codificador recibe un mensaje "m1" y emite un código C1, el decodificador recibe el código y emite un mensaje "m2", donde m1 debe ser igual a m2, es decir, el código es decodificable.

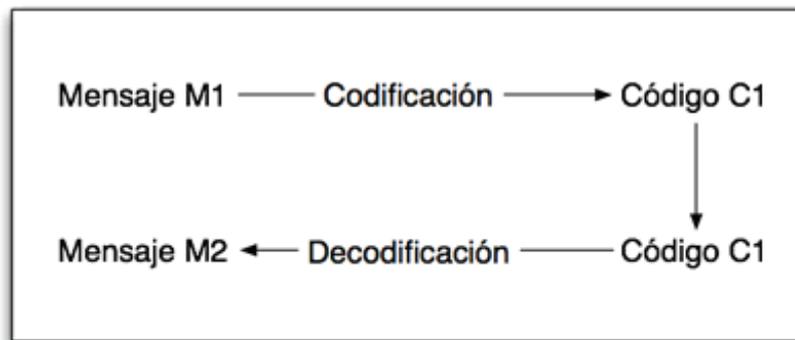


Figura 1.13 Proceso de codificación y decodificación. ^[20]

Decodificación. ^[21] Un decodificador tiene N líneas de entradas que corresponden a códigos y a su salida M líneas donde solo una es activada dependiendo del código que haya ingresado al decodificador, donde $M=2N$ dando así el número de combinaciones posibles a la entrada como se muestra en la Figura 1.14

Codificación. ^[22] Un codificador hace el trabajo contrario a la decodificación y tiene un número M de líneas de entrada donde solo una se activa y produce a la salida un código de N bits dependiendo de la entrada que haya sido activada, es decir, tiene $2M$ entradas y N salidas como se muestra en la Figura 1.15

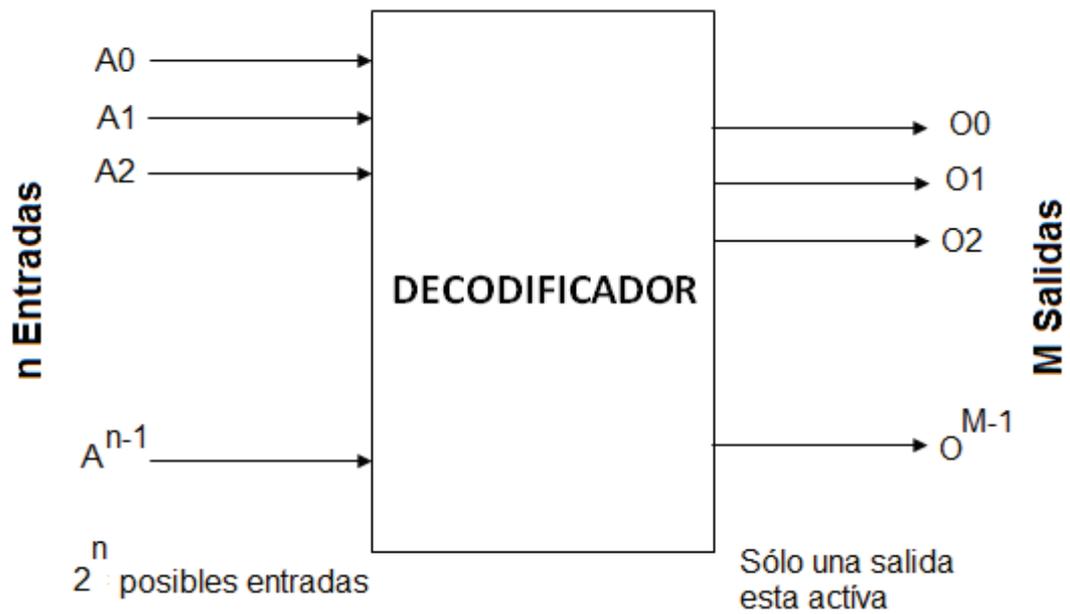


Figura 1.14 Circuito básico del decodificador. ^[20]

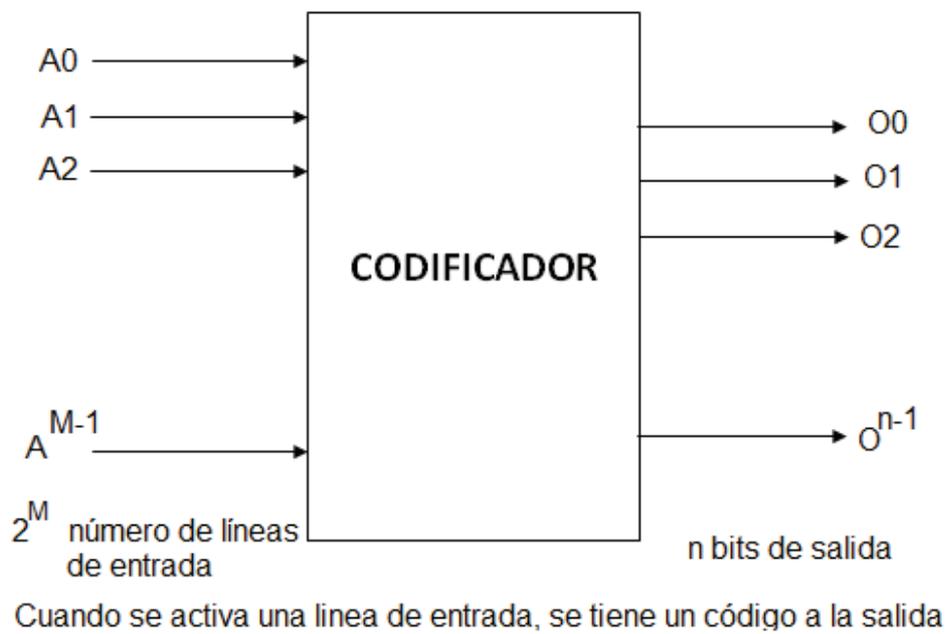


Figura 1.15 Circuito básico del codificador ^[22]

1.2.1 MÓDULO CODIFICADOR - HT12E^[23]

Símbolo	Parámetros	Condición de Prueba		Min	Typ.	Max.	Unid
		V _{DD}	Condición				
V _{DD}	Voltaje funcionamiento	-	-	2.4	5	12	V
I _{STB}	Corriente de espera	3V	<i>Detiene Oscilador</i>	-	0.1	1	μA
		5V		-	2	4	μA
I _{DD}	Corriente de funcionamiento	3V	<i>Sin carga</i>	-	40	80	μA
		5V	F _{OSC} =455kHz	-	150	300	μA
I _{DOUT}	Salida de corriente	5V	V _{OH} =0.9V _{DD} (<i>Source</i>)	-1	-1.6	-	mA
			V _{OL} =0.1V _{DD} (<i>Sink</i>)	1	1.6	-	mA
V _{IH}	"H" Entrada de voltaje	-	-	0.8 V _{DD}	-	V _{DD}	V
V _{IL}	"L" Entrada de voltaje	-	-	0	-	0.2 V _{DD}	V
f _{OSC}	Frecuencia del oscilador	5V	R _{OSC} =1.1MΩ	-	3	-	KHz
R _{TE}	TE <i>Pull-high Resistance</i>	5V	V _{TE} =0V	-	1.5	3	kΩ

Tabla 1-5 Características Eléctricas HT12E^[23]

Esta serie de codificadores de 2¹² codifica una palabra de 12 bits las cuales están compuestas por una dirección de 8 bits y una sección de datos de 4 bits, con este número de bits se puede comandar 256 dispositivos diferentes, enviándoles hasta 16 comandos distintos a cada uno.

Cada dirección o entrada de datos se puede establecer en uno de los dos estados lógicos 1L o 0L.

La transmisión se realiza cuando TE (*transmission enable*) del HT12E es activado con 0L. Una vez codificada la palabra se realiza la transmisión.

Las Características Eléctricas a T_a=25°C se muestra en la Tabla 1-5

El codificador HT12E, consta de 18 pines, los cuales se muestra en la Figura 1.16 y su descripción se indica en la Tabla 1-6

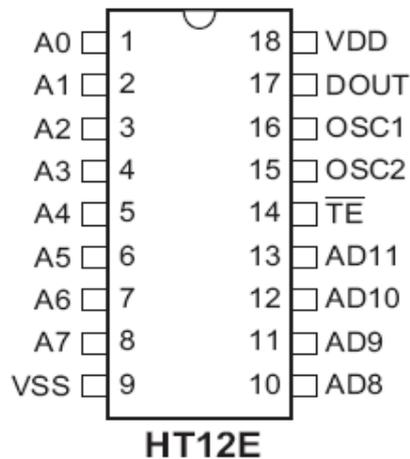


Figura 1.16 Módulo codificador HT12E, 8 Direcciones y 4 Datos. ^[23]

Nombre del Pin	I/O	Conexión Interna	Descripción
A0~A7	I	<i>NMOS Transmission gate protection diode (HT12E)</i>	Entrada en las direcciones A0~A7 establecidas. Los pines se configura externamente VSS o dejar abiertos
AD8~AD11	I	<i>NMOS Transmission gate protection diode (HT12E)</i>	Entrada en direcciones/datos AD8~AD11 establecidos. Los pines configura externamente VSS o dejar abiertos
D8~D11	I	<i>CMOS IN Pull-high</i>	Entrada de datos D8~AD11 establecidos y Transmisión permite, active bajo. Los pines se configura externamente VSS o dejar abiertos
DOUT	O	<i>CMOS OUT</i>	Codificador de datos de salida en serie
L/MB	I	<i>CMOS IN Pull-high</i>	Cierre/transmisión momentánea de formato. Cierre: flotante o VDD; Momentáneo: VSS
TE	I	<i>CMOS IN Pull-high</i>	Transmisión activada, activa en bajo
OSC1	I	<i>OSCILLATOR 1</i>	Pin de entrada del oscilador
OSC2	O	<i>OSCILLATOR 1</i>	Pin de salida del oscilador
X1	I	<i>OSCILLATOR 2</i>	455kHz resonador de entrada del oscilador
X2	O	<i>OSCILLATOR 2</i>	455kHz resonador de entrada del oscilador
VSS	I	-	Suministro de energía negativo
VDD	I	-	Suministro de energía positivo

Tabla 1-6 Descripción de pines del módulo codificador HT12E. ^[23]

1.2.2 MÓDULO DECODIFICADOR - HT12D^[24]

El decodificador de 2^{12} decodifica palabras de 12 bits de los cuales 8 corresponden a direcciones y 4 bits de datos.

Los decodificadores reciben direcciones de serie y los datos que se transmiten por radio frecuencia. El decodificador compara los datos de entrada de serie tres veces continuamente con las direcciones de su sitio. Si no hay ningún error, los códigos de datos de entrada son decodificados y luego transferido a los pines de salida.

El pin VT (transmisión válida) pasa a 1L para indicar una transmisión válida. Las Características Eléctricas HT12D, a $T_a=25^\circ\text{C}$ se muestra en la Tabla 1-7

Símbolo	Parámetros	Condiciones de Prueba		Min	Typ.	Max.	Unid
		V _{DD}	Condiciones				
V _{DD}	Voltaje de Operación	-	-	2.4	5	12	V
I _{STB}	Corriente de espera	5V	Detiene oscilador	-	0.1	1	μA
		12 V		-	2	4	μA
I _{DD}	Corriente de funcionamiento	5V	Sin carga f _{OSC} =150kHz	-	200	400	μA
I _O	Salida de datos, Unidad de salida de corriente (D8–D11)	5V	V _{OH} =4.5V	-1	-1.6	-	mA
	Datos de salida del variador de corriente de absorción (D8–D11)	5V	V _{OL} =0.5V	1	1.6	-	mA
I _{VT}	VT Salida de fuente de corriente	5V	V _{OH} =4.5V	-1	-1.6	-	mA
	VT Salida de corriente de absorción		V _{OL} =4.5V	1	1.6	-	mA
V _{IH}	"H" Entrada de voltaje	5V	-	3.5	-	5	V
V _{IL}	"L" Entrada de voltaje	5V	-	0	-	1	V
f _{OSC}	Frecuencia de oscilación	5V	R _{OSC} =51kΩ	-	150	-	KHz

Tabla 1-7 Características Eléctricas HT12D. ^[24]

El decodificador HT12D, consta de 18 pines, los cuales se muestra en la Figura 1.17 y su descripción se indica en la Tabla 1-8

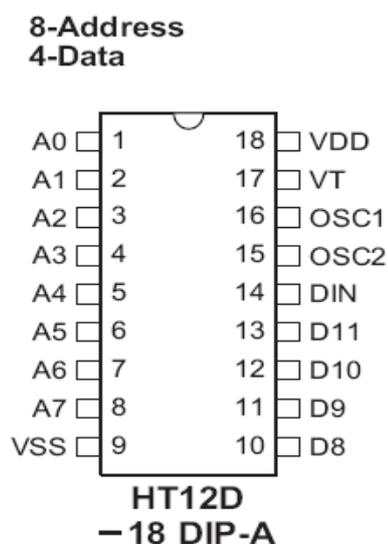


Figura 1.17 Módulo decodificador HT12E, 8 direcciones y 4 Datos. ^[24]

Nombre del Pin	I/O	Conexión Interna	Descripción
A0~A7	I	NMOS Transmisión de la puerta	Entrada de pines por direcciones A0~A11, ajuste puede ser extremadamente estable para VDD ó VSS.
D8~D11	0	CMOS OUT	Salida de datos por los pines
DIN	I	CMOS IN	Entrada de datos serial del pin
VT	0	CMOS OUT	Transmisión valida, Alto activo
OSC1	I	OSCILLATOR	Salida del oscilador del pin
OSC2	0	OSCILLATOR	Salida del oscilador del pin
VSS	I	-	Fuente de alimentación negativa (GND)
VDD	I	-	Fuente de alimentación positiva

Tabla 1-8 Descripción de pines del módulo codificador HT12D ^[24]

1.3 MICROCONTROLADORES ^[25]

Un microcontrolador es un circuito integrado de alta escala de integración que incorpora la mayor parte de los elementos que configuran un controlador: unidad central de proceso, memoria, unidades de E/S.

Dispone normalmente de los siguientes componentes:

- Procesador o UCP (Unidad Central de Proceso).
- Memoria RAM para Contener los datos.
- Memoria para el programa tipo ROM/PROM/EPROM.
- Líneas de E/S para comunicarse con el exterior.
- Control de periféricos (temporizadores, Puertas Serie y Paralelo, CAD Conversor Analógico/Digital, CDA: Conversor Digital/Analógico).
- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

La arquitectura de la UCP de los AVR (*RISC virtual avanzado*) se muestra en la Figura 1.18

1.3.1 MÓDULO MICROCONTROLADOR - ATmega 164P^[26]

El AVR es otro tipo de microcontrolador, su CPUs usa arquitectura Harvard. Tiene 32 registros de propósito general de 8 bits. Los registros de entrada/salida y la memoria de datos forman un solo espacio de localidades, que se acceden mediante operaciones de carga y de almacenamiento. Sus instrucciones se ejecutan en un ciclo de máquina. El diagrama de bloques del Atmega164P muestra en la siguiente Figura 1.19 y la Tabla 1-9 muestra la descripción de pines.

El ATmega164P provee las siguientes características:

- 16/32/64K bytes en el sistema de Flash Programable con capacidad de lectura y escritura de 512B/1K/2K bytes en la EEPROM.
- 1/2/4K bytes en la SRAM.
- 32 pines de E/S para propósito general.
- 32 registros de propósito general.
- Contador en Tiempo real (RTC).

- Tres Timer/Contadores flexibles con modo de Comparación y PWM.
- 2 USARTs.
- Un byte orientado a la Interfaz Serial de 2 hilos.
- 8 canales ADC de 10 bits.
- Watchdog Timer programable con oscilador interno.
- Un Puerto serial SPI.
- Interface de prueba JTAG.
- Seis modos de programación seleccionable para ahorro de energía.

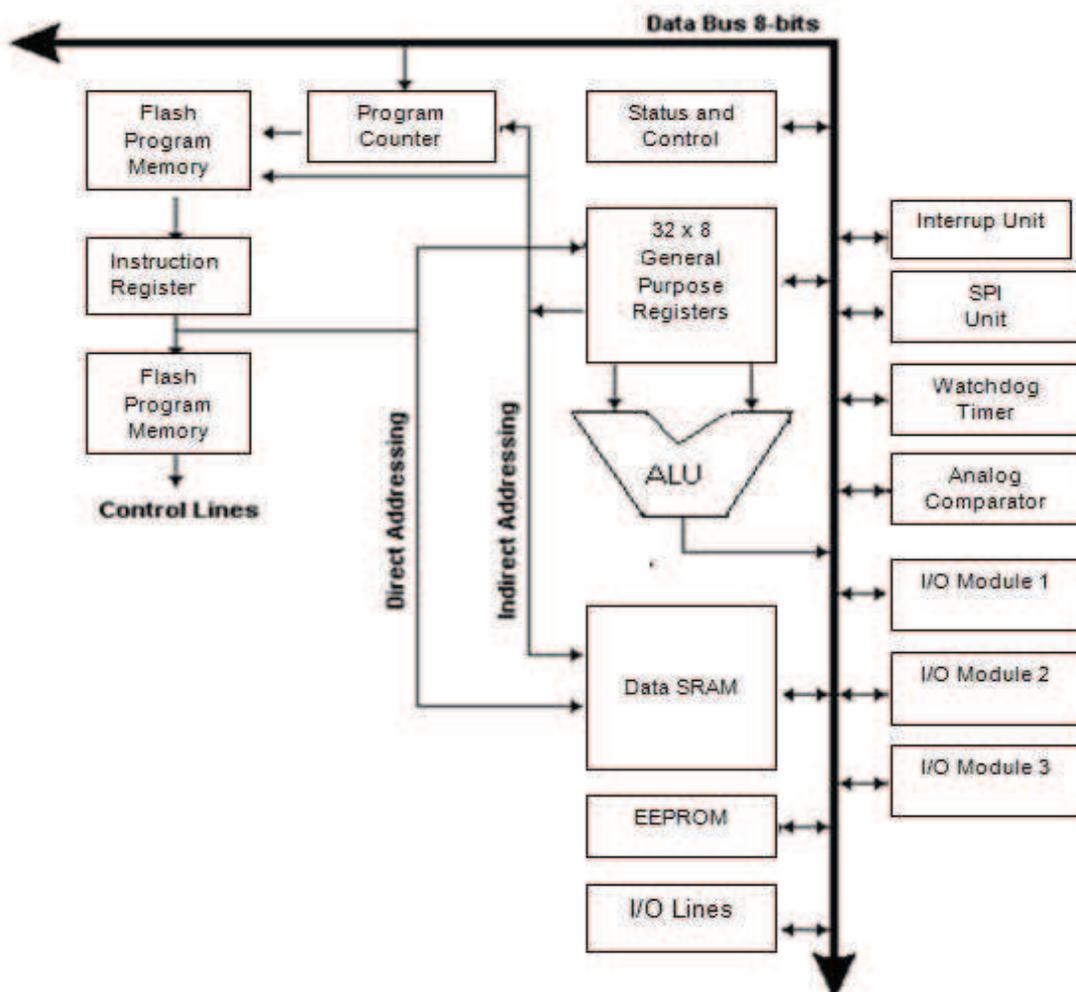


Figura 1.18 CPU – AVR (*RISC virtual avanzado*)^[25]

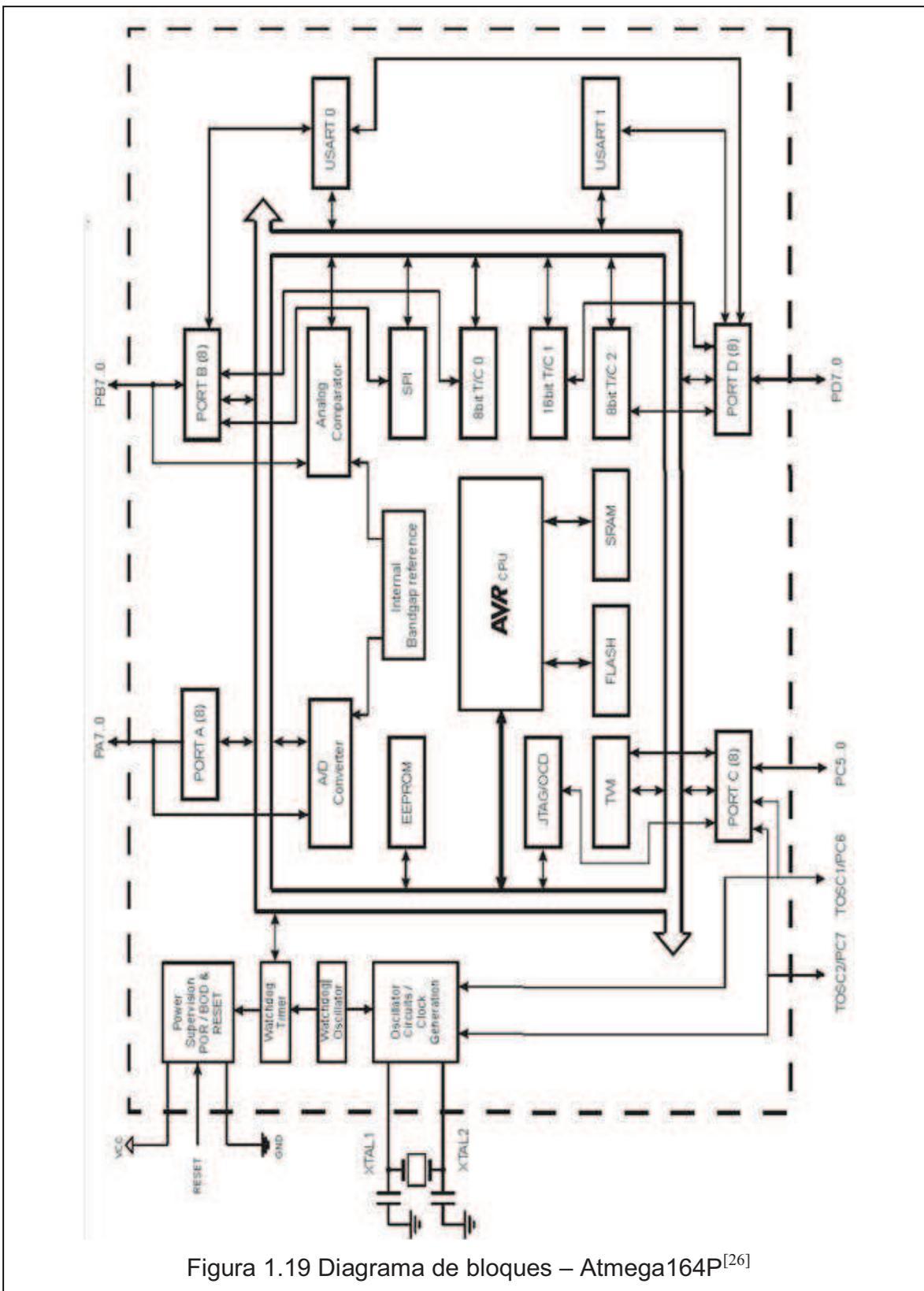


Figura 1.19 Diagrama de bloques – Atmega164P^[26]

El chip interno ISP de la FLASH permite a la memoria de programa ser reprogramada a través del puerto interno ISP mediante un programador convencional no volátil o mediante un programa interno en el dispositivo AVR.

La Descripción de Pines del Módulo microcontrolador ATmega164P, se muestra en la Figura 1.20

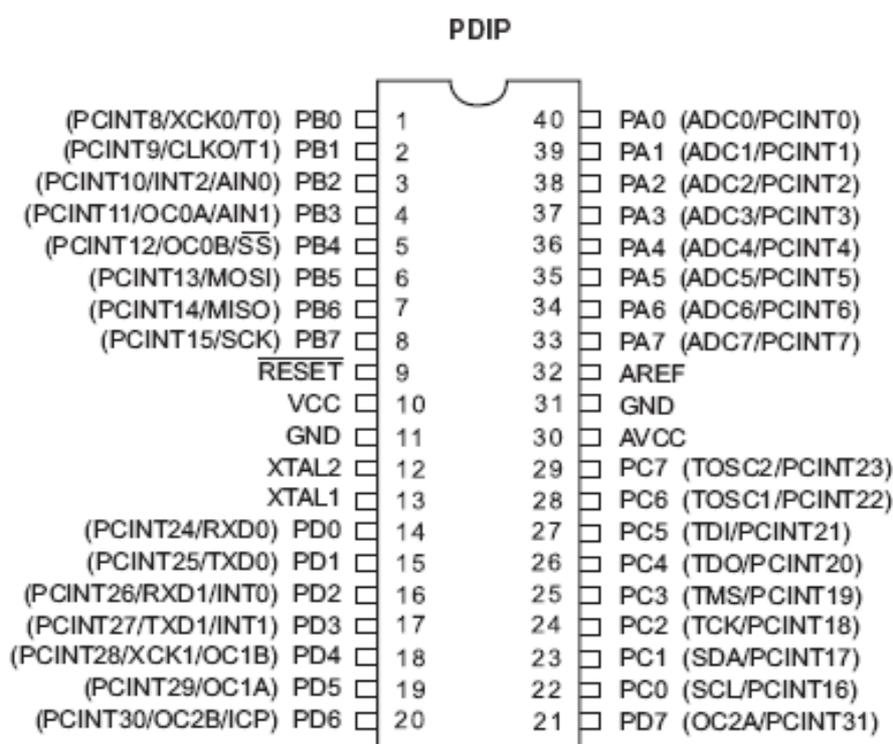


Figura 1.20 Módulo Microcontrolador - ATmega164P. ^[26]

La descripción del microcontrolador ATmega164P se muestra en la tabla 1-9

1.3.2 MÓDULO MICROCONTROLADOR - ATmega8^[27]

El ATmega8 es un microcontrolador CMOS basado en la arquitectura AVR RISC (*Reduced Instruction Set Computer*), ofrece un repertorio de 130 instrucciones, 32 registros de 8 bits de propósito general.

El diagrama de bloques del Atmega8 muestra en la Figura 1.21

El ATmega8 provee las siguientes características:

- Una memoria flash de 8K bytes para programas que permite arrancarse y auto programarse.
- Una EEPROM de 512 bytes.
- Una SRAM de 1K bytes para datos.
- También cuenta con un USART programable en serie.
- TWI (two wire interfase).
- Un ADC de seis canales con una precisión de 10 bits.
- Un watchdog programable.
- Un puerto SPI.
- Tres contadores.
- Tres canales PWM.
- Un contador Real Time con oscilador independiente.
- Un comparador analógico.
- Además de 23 líneas de entrada/salida.
- El microcontrolador soporta cinco modos de ahorro de energía.
- Interrupciones internas y externas.
- Un oscilador RC interno.

La memoria es no volátil y la memoria de programa puede ser reprogramada a través de un interface serial ISP.

La Descripción de Pines del Módulo microcontrolador ATmega8, el cual se muestra en la Figura 1.22

Microcontrolador ATmega 164P	
Nombre del pin	Descripción
VCC	Alimentación de voltaje digital
GND	Tierra
RESET	Entrada del Reset. Un pulso de nivel bajo 0L lo activa
XTAL1	Entrada para el amplificador del oscilador invertido y entrada para el circuito de operación del reloj interno.
XTAL2	Salida del Oscilador
AVCC	Es la alimentación de voltaje para el pin del Puerto F y el Conversor Análogo a Digital. Este debe ser conectado externamente a VCC, siempre y cuando el ADC no sea usado. Si el ADC es usado, este deberá ser conectado a VCC a través de un filtro paso bajo.
AREF	Esta es la referencia para el pin de la conversión Análoga a Digital.
Port A (PA7:PA0)	El pórtilo A es bidireccional de 8 bits sirve como E/S de datos. Entrada del conversor Análogo/ Digital, interrupciones externas adicionales que se registra en el PCIO.
Port B (PB7:PB0)	El pórtilo B bidireccional de 8 bits sirve como E/S de datos. Comparador analógico. SPI (Interfase a Pórtilos seriales) los cuales son full dúplex tiene tres líneas para comunicación sincrónica con siete velocidades programables y bandera de fin de transmisión. Interrupción externa INT2. Temporizadores T0, T1. Contadores OC0A, OC0B, USART0 XCK0, Salida de reloj CLK0. Interrupciones externas adicionales.
Port C (PC7:PC0)	El pórtilo C es un puerto bidireccional de 8 bits sirve como E/S de datos. Interface JTAG para sistema de depuración en la cual se tiene acceso a todos los periféricos y permite la programación de la Flash. EEPROM. Fusibles y Bits de seguridad. Temporizadores y contador. Interfase serial con 2 líneas TWI. Interrupciones externas adicionales.
Port D (PD7:PD0)	El pórtilo D bidireccional sirve como E/S de datos. Interrupciones externas
INT0, INT1	Contadores OC1B, OC1A, OC2B, OC2A. USART1. Transmisor TX0, TX1 y Receptor RX1, RX2. Interrupciones externas adicionales.

Tabla 1-9 Descripción del Microcontrolador ATmega164P^[26]

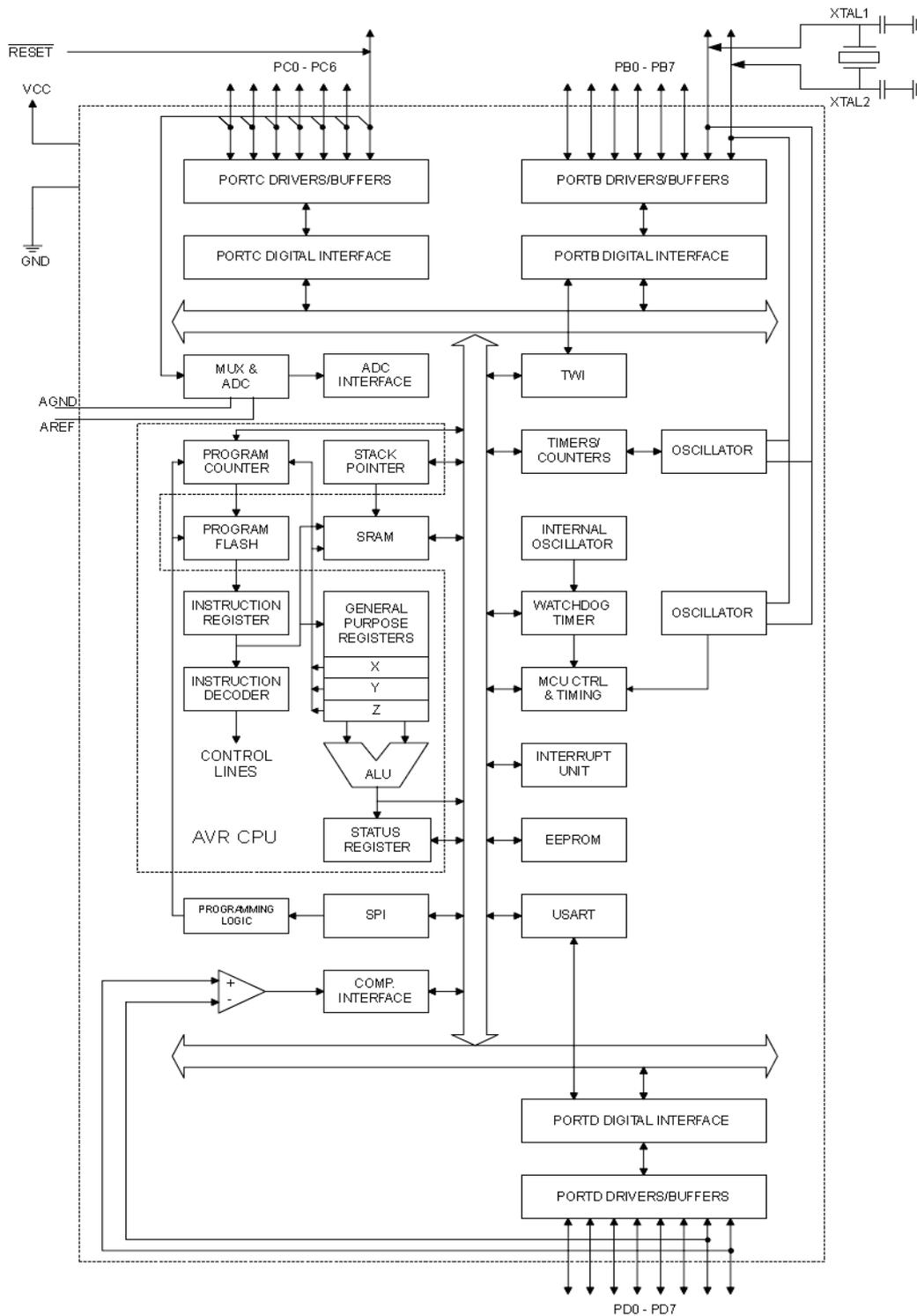


Figura 1.21 Diagrama de bloques – Atmega8.^[27]

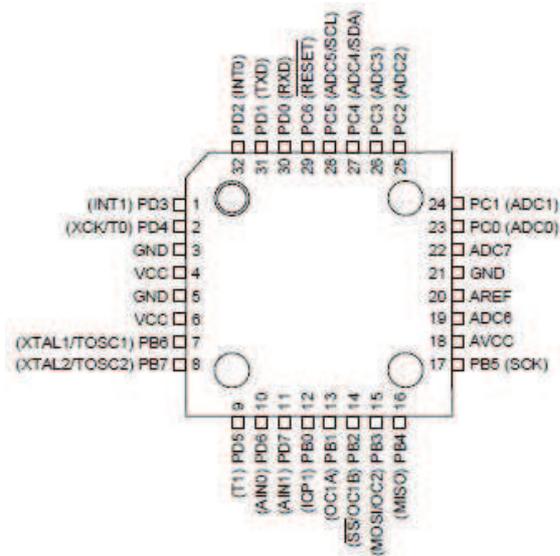


Figura 1.22 Módulo Microcontrolador – Atmega8. [27]

Descripción de pines del ATmega8 se muestra en la tabla 1-10

Microcontrolador ATmega8	
Nombre del pin	Descripción
VCC	Corriente de alimentación.
GND	Corriente de tierra.
PortB (PB7...PB0)	El pòrtico B es un puerto bidireccional de 8 bits de/S de datos
PB6 y PB7	Tienen funciones especiales (I/O del oscilador o TOSC del async Timer/Counter2).
PortC (PC6...PC0)	El pòrtico C es un puerto de 7 bits de E/S de datos. Puerto de 7 bits. PC6 puede ser usado como reset en 0L. PC0-PC6 son entradas de ADC.
PortD (PD7...PD0)	El pòrtico D es un puerto de 8 bits de E/S de datos. Comparador analògico AIN0, AIN1. Timer T0, T1. Contador 0 y 1. Interrupciones externas INT0, INT1. USART
Reset	Se activa con 0L.
AVCC	Corriente de alimentaci3n para el conversor A/D. Debe conectarse a VCC con un filtro de paso bajo.
AREF	Referencia anal3gica para el conversor A/D.
AVR	Es una arquitectura Harvard, es decir, con memoria de programas y datos separados.

Tabla 1-10 Descripci3n de pines del Microcontrolador ATmega8 [27]

1.4 TRANSMISIÓN DE MENSAJES VÍA CELULAR^[28]

1.4.1 Teléfonos GSM

Los teléfonos GSM usan el lenguaje AT para comunicarse con sus terminales, es decir tienen un juego de comandos AT específico para configurar y proporcionar instrucciones a los terminales, dichos comandos permiten realizar llamadas de datos o de voz, leer y escribir en la agenda de contactos y enviar mensajes.

1.4.2 LENGUAJE DE COMUNICACIÓN.^[29]

Los comandos AT son instrucciones codificadas que en principio fueron creadas para comunicar las terminales de los modems pero como el desarrollo de tecnología ha aumentado, estos comandos fueron acondicionados para ser usados por GSM.

“El lenguaje de comunicación más utilizado y básico es el siguiente:

1.4.2.1 Acción realizada por el modem

ATA

1. *Se pone en modo respuesta y espera una señal portadora del modem remoto.*
2. *Espera 7 segundos y colgará si no se detecta portadora.*

ATDnúmero

1. *Descuelga y llama al número de teléfono solicitado.*
 2. *Espera un tono de llamada antes de marcar.*
- *Si no se detecta ese tono en 6 segundos, el modem devuelve código de resultado "no dial tone"*
 - *Si se detecta el tono el modem espera 7 segundos*
 - *Si no establece conexión el modem vuelve al estado de comandos*
 - *Si se establece conexión el modem entra en el estado on-line.*

ATE Eco. Los comandos introducidos en el modem vuelven por eco al PC (por defecto).

ATH. Descuelga el teléfono

ATI. Revisa la ROM del modem (checksum)

ATL. Programa el volumen del altavoz

ATM. Programa conexión/desconexión del altavoz

ATO. Vuelve a estado on-line desde el estado de comandos.

ATQ. Programa los códigos de resultado a ON/OFF

ATS. Visualiza/cambia contenidos de los registros S

ATV. Envía códigos de resultado en palabras o números

ATW. Envía "códigos del progreso de la negociación"

ATX. Programa códigos de resultado

ATZ. Reset

AT&C. Programa detección de portadora

AT&D. Programa control de DTR

AT&K. Programa control de flujo

AT&W. Almacena perfil configuración del usuario

AT&Y. Especifica que perfil de configuración usuario en los almacenados se va a utilizar

1.4.2.2 Comandos

Comandos generales

AT+CGMI: Identificación del fabricante

AT+CGSN: Obtener número de serie

AT+CIMI: Obtener el IMSI.

AT+CPAS: Leer estado del modem

Comandos del servicio de red

AT+CSQ: Obtener calidad de la señal

AT+COPS: Selección de un operador

AT+CREG: Registrarse en una red

AT+WOPN: Leer nombre del operador

Comandos de seguridad

AT+CPIN: Introducir el PIN

AT+CPINC: Obtener el número de reintentos que quedan

AT+CPWD: Cambiar password

Comandos para la agenda de teléfonos

AT+CPBR: Leer todas las entradas

AT+CPBF: Encontrar una entrada

AT+CPBW: Almacenar una entrada

AT+CPBS: Buscar una entrada

Comandos para SMS

AT+CPMS: Seleccionar lugar de almacenamiento de los SMS

AT+CMGF: Seleccionar formato de los mensajes SMS

AT+CMGR: Leer un mensaje SMS almacenado

AT+CMGL: Listar los mensajes almacenados

AT+CMGS: Enviar mensaje SMS

AT+CMGW: Almacenar mensaje en memoria

AT+CMSS: Enviar mensaje almacenado

AT+CSCA: Establecer el Centro de mensajes a usar

AT+ WMSC: Modificar el estado de un mensaje.”

CAPÍTULO 2

2. DISEÑO DEL CIRCUITO.

2.1 CIRCUITO DE LA ALARMA.

2.1.1 FUENTE.

La Figura 2.1 es la parte básica para el encendido del circuito, uno de los primeros pasos a seguir es la regulación de voltaje que ingresa a cada elemento principal, para esto se tiene una fuente a la entrada de 12 voltios, la misma que alimenta a la totalidad del circuito.

Se ha colocado a la entrada un diodo D1 (1N4007) para proteger al circuito de una conexión opuesta de polos de la fuente de 12 voltios y evitar que la batería recargue la fuente externa y cause daños.

El capacitor C1 (1000 μ F) permite estabilizar el voltaje a la entrada, cuando la sirena se enciende, permitiendo al condensador mantener estabilidad de voltaje en el integrado IC1 (7805).

El integrado de regulación de voltaje IC1 (7805) permite regular el voltaje de salida, este integrado soporta voltajes de entrada de 36 a 7 voltios y entrega a la salida 5 voltios.

El led D2 indica la existencia de voltaje a 5 voltios, mientras que el capacitor C2 (10 μ F) mantiene el voltaje, y evita variaciones del mismo en los elementos que trabajan a 5 voltios.

En la Figura 2.2, se tiene puntos de prueba, JP1, JP2, JP3, JP4 donde se puede tomar medidas de voltaje con la ayuda de un multímetro en VCCin, Vccd (5 voltios), GND, para verificar que el voltaje en estos puntos sea el indicado.

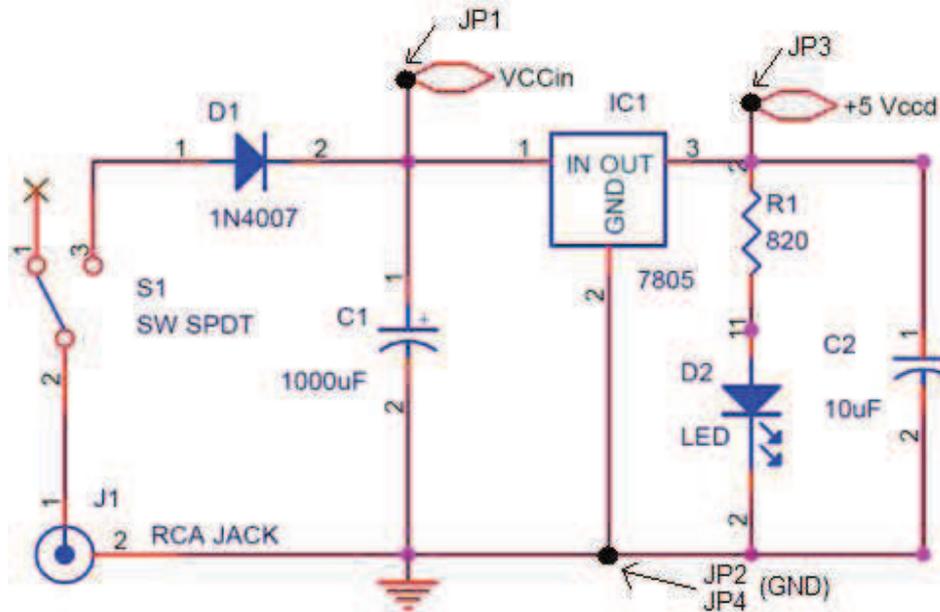


Figura 2.1 Regulador de voltaje a 5 voltios

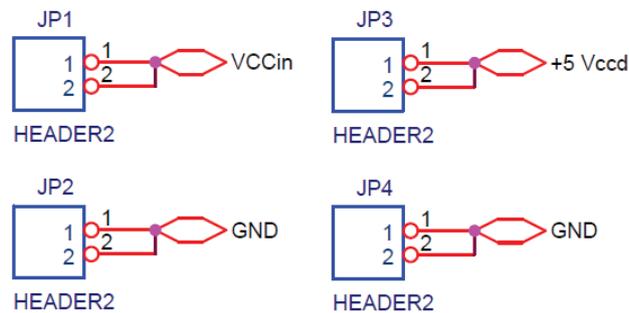


Figura 2.2 Puntos de prueba para medir voltajes.

2.1.2 ALARMA

En la Figura 2.3, se tiene la sirena JP7, que trabaja a 12 voltios, la bocina va conectada a 12 V y se controla mediante un relay a 12 voltios, al activarse permite un punto de referencia con tierra para ser activada.

El Relay es controlado por la salida PD4 del microcontrolador, al enviar una señal en 1L permite que el transistor Q1 (2N3904) active el relay y a su vez active la sirena, el

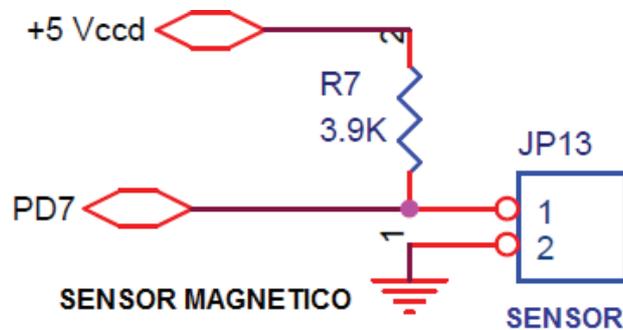


Figura 2.5 Sensor Magnético

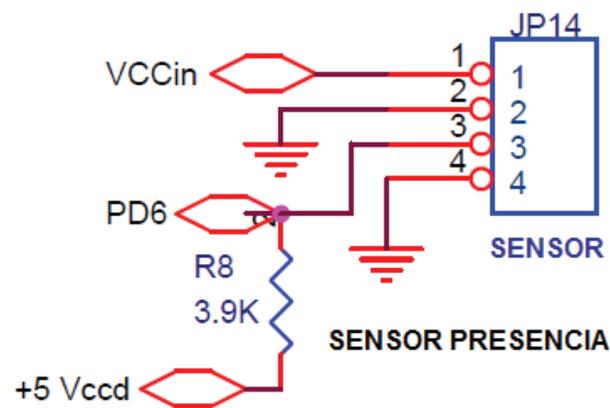


Figura 2.6 Sensor Presencia

2.2 CIRCUITO DE RADIOFRECUENCIA

2.2.1 CIRCUITO RECEPTOR

La Figura 2.7, detalla la conexión del receptor, para esto se utiliza un decodificador HT12D, el mismo que las entradas A0, A1, A2, A3, A4, A5, A6, A7, están conectadas a un switch SW DIP-8 el mismo que permite la manipulación manual de las direcciones dadas en el receptor y evita interferencias de otros transmisores que utilicen la misma frecuencia.

En la entrada del HT12D – Din ingresa la información recibida en el receptor, para ser decodificada y entregada al microcontrolador por las salidas del D11, D10, D9, D8 y este procesa una instrucción.

La salida VT funciona en 1L, solo cuando la transmisión es válida y va conectado en PA4 del microcontrolador.

Las salidas del HT12D van conectadas al microcontrolador de la siguiente manera: D0 → PA0, D1 → PA1, D2 → PA2, D3 → PA3, VT → PA4.

El receptor RLP434a funciona con voltaje de 5 voltios y una antena para aumentar el alcance de comunicación con el transmisor.

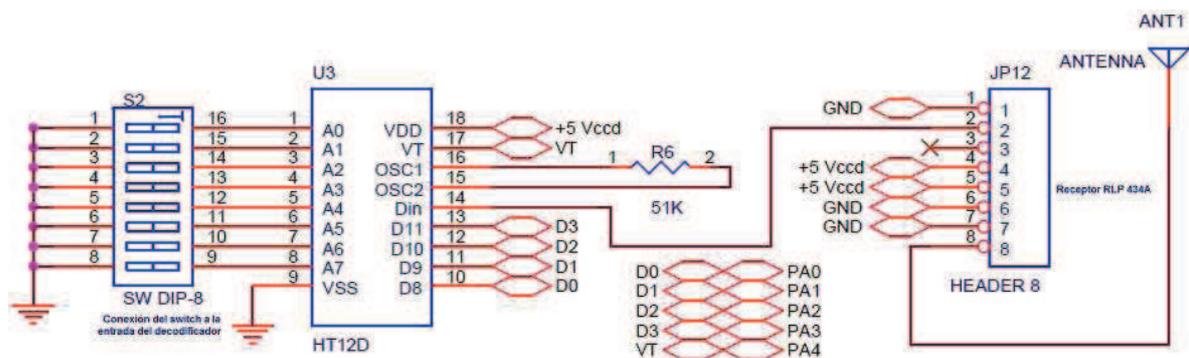


Figura 2.7 Funcionamiento del receptor

2.2.2 CIRCUITO TRANSMISOR.

2.2.2.1 Fuente

La Figura 2.8 es la parte básica del encendido del circuito, uno de los primeros pasos a seguir es la regulación de voltaje que ingresa a cada elemento principal, para esto, se alimenta con una fuente de 9 voltios a la entrada, alimentando a la totalidad del circuito.

Se ha colocado a la entrada un diodo D1 (1N4007) para proteger al circuito de una conexión opuesta de polos de la batería de 9 voltios.

El capacitor C2 (100 μ F/25V) permite estabilizar el voltaje a la entrada, permitiendo al condensador mantener estabilidad de voltaje en el integrado IC1 (7805).

El integrado de regulación de voltaje IC1 (7805) permite regular el voltaje de salida, este integrado soporta voltajes de entrada de 36 a 7 voltios y entrega a la salida 5 voltios.

El led D2 indica la existencia de voltaje a 5 voltios, mientras que el capacitor C3 (10 μ F) mantiene el voltaje, y evita variaciones del mismo en los elementos que trabajan a 5 voltios.

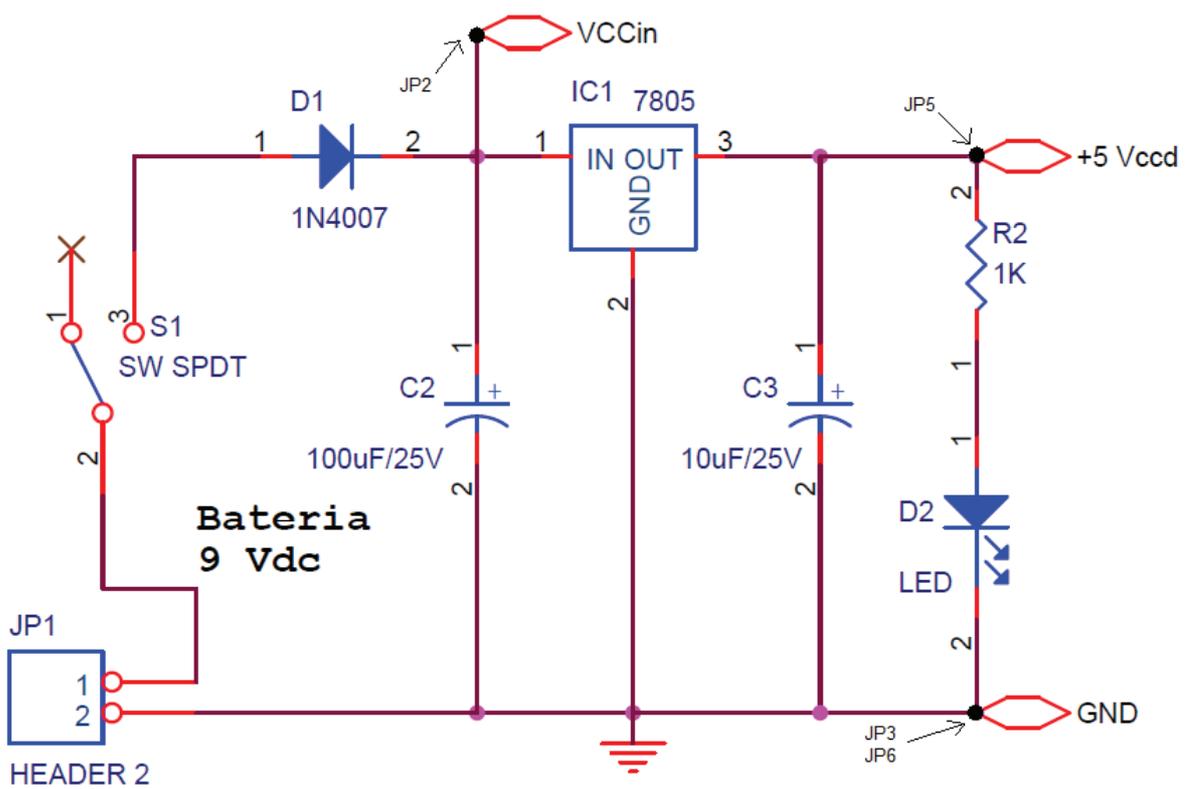


Figura 2.8 Regulador de voltaje a 5 voltios

En la Figura 2.9, se tiene varios puntos de prueba, JP2, JP3, JP5, JP6 donde se logra tomar medidas de voltaje con la ayuda de un multímetro en VCCin, Vccd (5 voltios), GND, para verificar que el voltaje en estos puntos, sea el indicado.



Figura 2.9 Puntos de prueba para medir voltajes.

2.2.2.2 Transmisor

La Figura 2.10, detalla la conexión del transmisor, para esto se ha utilizado un codificador HT12E, el mismo que las entradas A0, A1, A2, A3, A4, A5, A6, A7 van conectadas al microcontrolador en los pines PD0, PD1, PD2, PD3, PD4, PD5, PD6, PD7 respectivamente, los mismos que mediante una dirección dada desde el microcontrolador permite la comunicación con el decodificador del receptor y evita interferencias de otros transmisores que utilicen la misma frecuencia.

En la salida del HT12E – Dout envía la información que se codifica en el HT12E, esta información se ingresa por las entradas AD8, AD9, AD10, AD11 del codificador, mediante el uso de transistores de la Figura 2.11, que se conectan al colector.

La salida TE funciona en bajo lógico (0L), este envía los datos en radio frecuencia al transmisor.

Las entradas del HT12E van conectadas al transistor de la siguiente manera: D0 ← AD8, D1 ← AD9, D2 ← AD10, D3 ← AD11, TE ← TE.

El transmisor TLP434a funciona con voltaje de 5 voltios y una antena para aumentar el alcance de comunicación con el receptor.

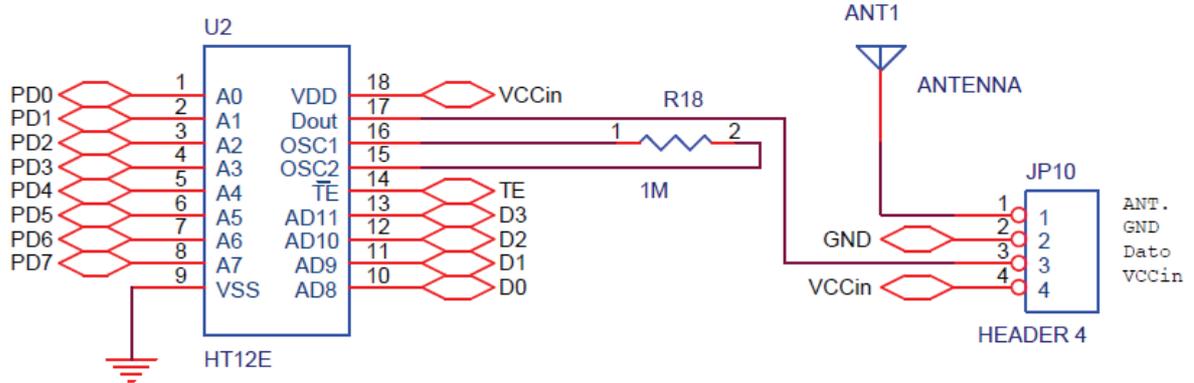


Figura 2.10 Funcionamiento del transmisor

En la Figura 2.11, se observa varios transistores, su función es comunicar al microcontrolador con las entradas del codificador, el codificador funciona a 12 voltios por lo que se conecta en el colector del transistor, mientras que los pines del microcontrolador funcionan a 5 voltios, conectados en la base del transistor, trabajando en corte o saturación y permitiendo enviar los datos desde el microcontrolador al codificador.

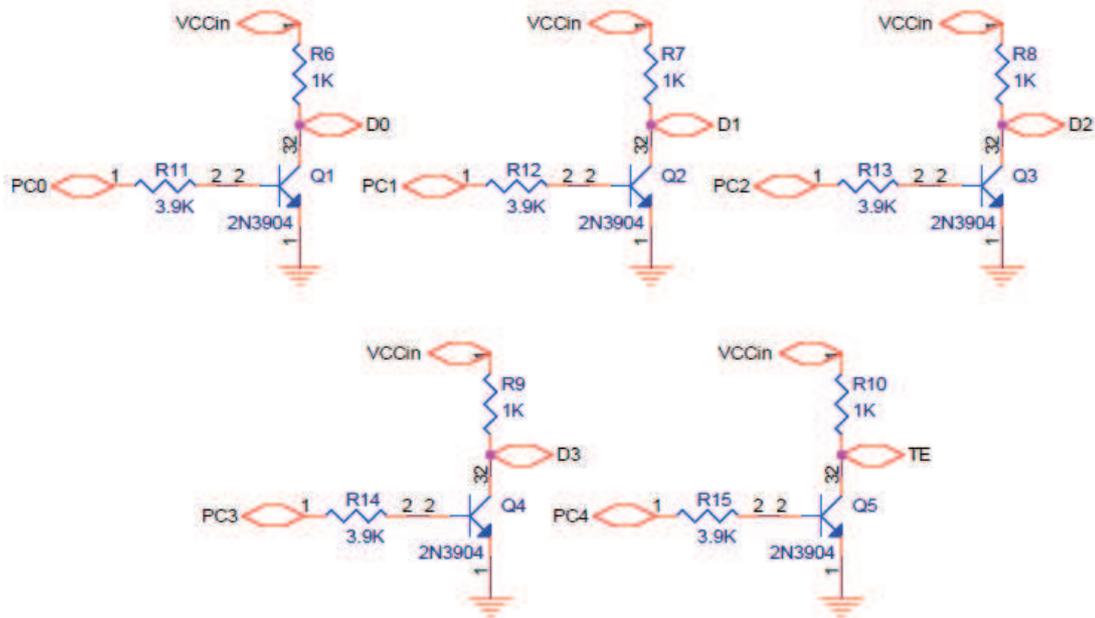


Figura. 2.11 Comunicación del microcontrolador y el codificador

2.3 ACOPLAMIENTO HACIA EL CELULAR.

En la Figura 2.12, se observa la conexión del celular (Nokia 3220), el mismo que mediante un cable de datos DK-5, usando los pines del celular Rx, Tx, y GND se logra acoplar con el microcontrolador AVR ATmega164P, mediante los pines PD2, PD3 los mismos que mediante comandos AT, logran comunicar al circuito con el celular, mediante mensajes cortos.

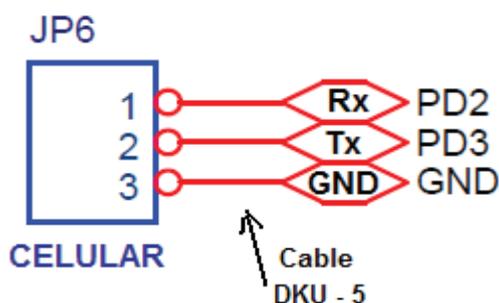


Figura 2.12 Conexión del Celular al Circuito

2.4 CONEXIÓN CON EL MICROCONTROLADOR.

El microcontrolador ATmega 8, se alimenta con un voltaje de 5 voltios, utiliza un cristal de 8MHz para sincronizar las instrucciones del microcontrolador, la cual se observa en la Figura 2.13.

La comunicación entre el usuario y el circuito transmisor es mediante un teclado de pulso que consta de dos teclas, las cuales van conectadas a los pines del microcontrolador, los cuales son PB0 y PB1. Su circuitería se observa en la Figura 2.14.

Su función, es el controlar las distintas instrucciones de los diferentes procesos que deben cumplir los elementos que conforman el circuito de transmisión.

Su ocupación principal es controlar la comunicación, que se envía desde el transmisor al receptor, para esto se ha implementado un codificador con una

dirección establecida a su entrada, Los pines usados desde el microcontrolador con el codificador van conectados de la siguiente manera: PD0→A0, PD1→A1, PD2→A2, PD3→A3, PD4→A4, PD5→A5, PD6→A6, PD7→A7.

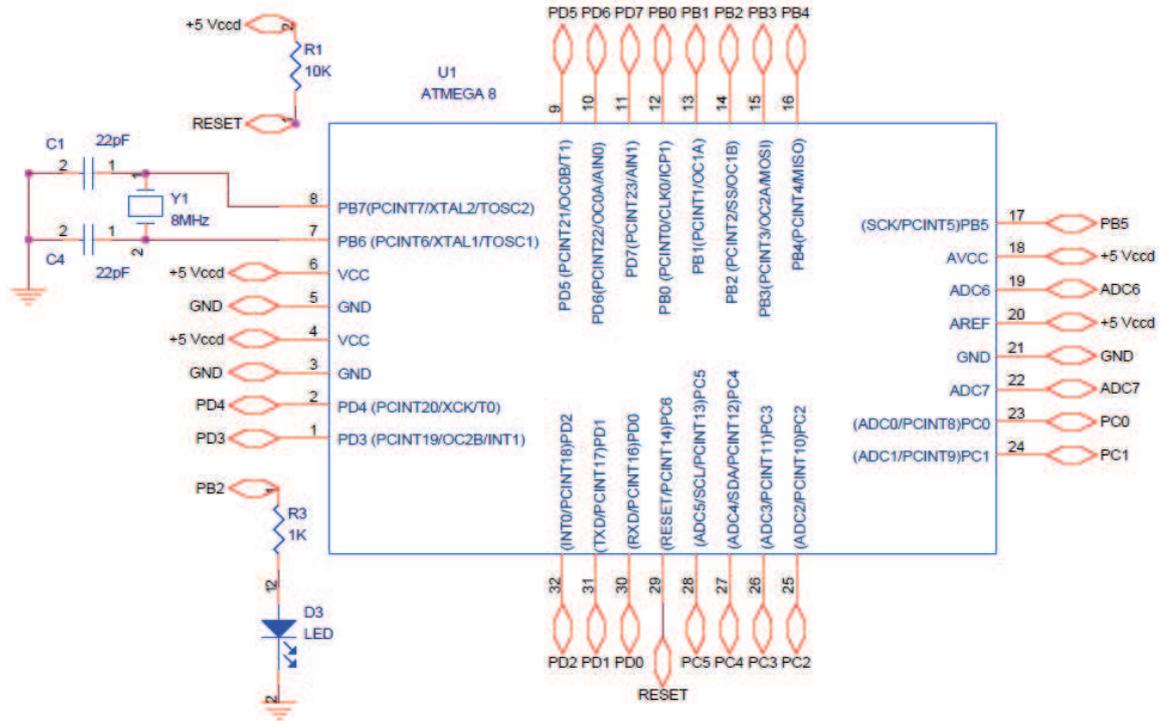


Fig 2.13 Pines del microcontrolador ATmega8

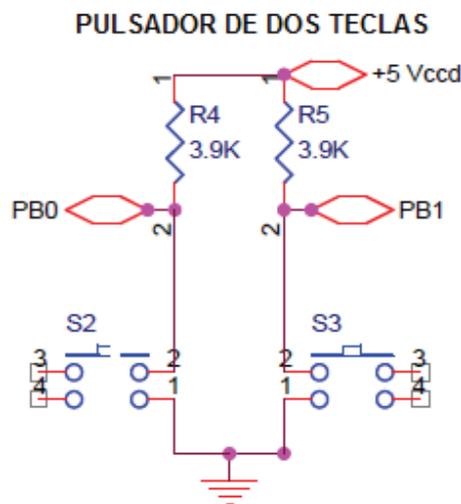


Figura 2.14 Teclado de dos teclas de pulso.

Estos pines del microcontrolador son de salida, mientras que los del codificador son solo de entrada, de esta forma el microcontrolador envía un código binario al codificador indicando que se puede comunicar con un decodificador que maneje el mismo código binario.

La manera de enviar información o datos desde el microcontrolador al codificador (HT12E) es usando los pines PC0, PC1, PC2, PC3, PC4 que están conectados a la base del transistor (2N-3904) y desde el colector del transistor (D0, D1, D2, D3, TE) va conectado a los pines del codificador (AD8, AD9, AD10 y TE), enviando por estos pines, datos que luego se enviarán por medio de radiofrecuencia con ayuda del transmisor como se mostro tanto en la Figura 2.10 como la Figura 2.11 relacionados a esta información.

El microcontrolador ATmega164P, trabaja con un voltaje de 5 voltios así como también con un cristal de 8 MHz

El microcontrolador recibe los datos recibidos desde el transmisor, así como también se comunica con el celular para realizar una acción predeterminada.

El decodificador del circuito receptor, recibe la información por medio del módulo receptor lo decodifica y envía al microcontrolador, esto lo hace mediante los pines del decodificador que envía al microcontrolador de la siguiente manera D8→PA0, D9→PA1, D10→PA2, D11→PA3, VT→PA4. Estos pines se observan en la Figura 2.7 y Figura 2.15.

Los datos ingresados desde el decodificador ingresan en un conjunto de instrucciones internas y establecen una instrucción específica en el desempeño de las alarmas a activarse.

Por medio del Pin PD6 del microcontrolador se conecta a un sensor de presencia ya indicado en la Figura 2.5, y el microcontrolador procede mediante una instrucción el envío de una notificación del evento a un celular.

El Celular va conectado en los pines PD2, PD3 del microcontrolador por donde recibe las ordenes que debe ejecutar para el envío de mensajes a otro elemento móvil específico que se observa en la Figura 2.12. Esta comunicación se realiza mediante un cable serial DKU-5.

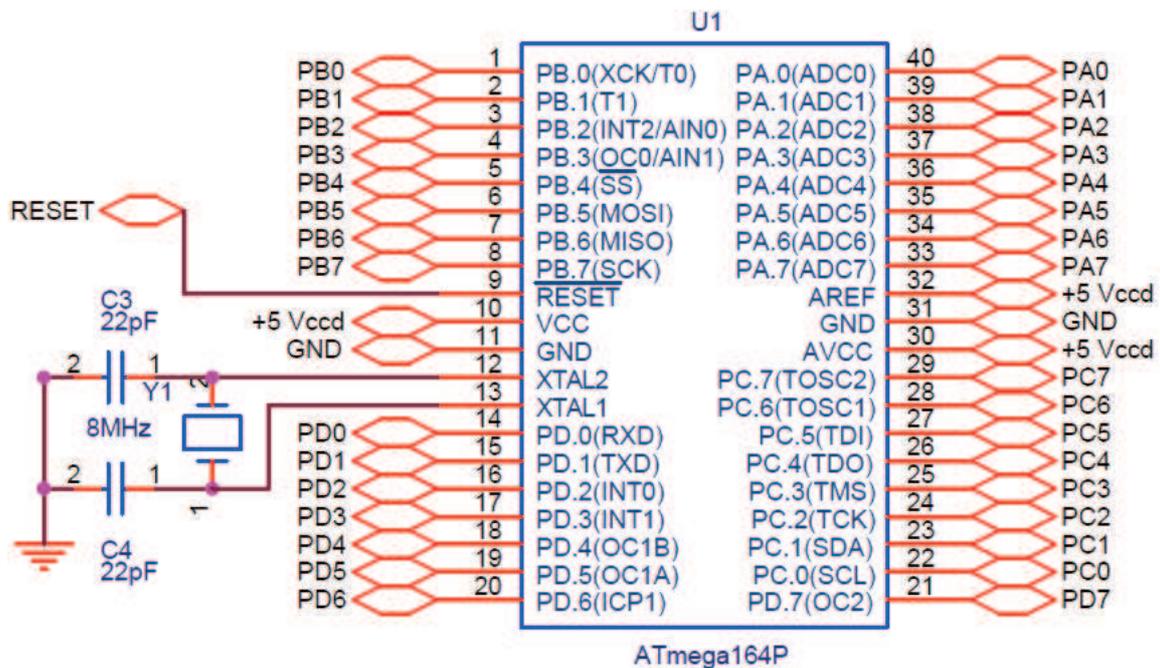


Figura 2.15 Pines del microcontrolador ATmega 164P

La parte de la sirena o alarma sonora se realiza mediante el Pin PD6 que va conectado a la base de un transistor 2N3904 por medio del cual se activa un relé (Figura 2.3) y este a su vez activa la alarma sonora.

En la Figura 2.16 se observa el circuito completo del sistema de alarma comunitaria relacionada directamente al circuito transmisor, de donde se envía una señal de radio frecuencia, la misma que activa al circuito receptor que se muestra en la Figura 2.17, cada uno de estos diagramas han sido explicados más detalladamente en el Capítulo 2, para un mayor entendimiento.

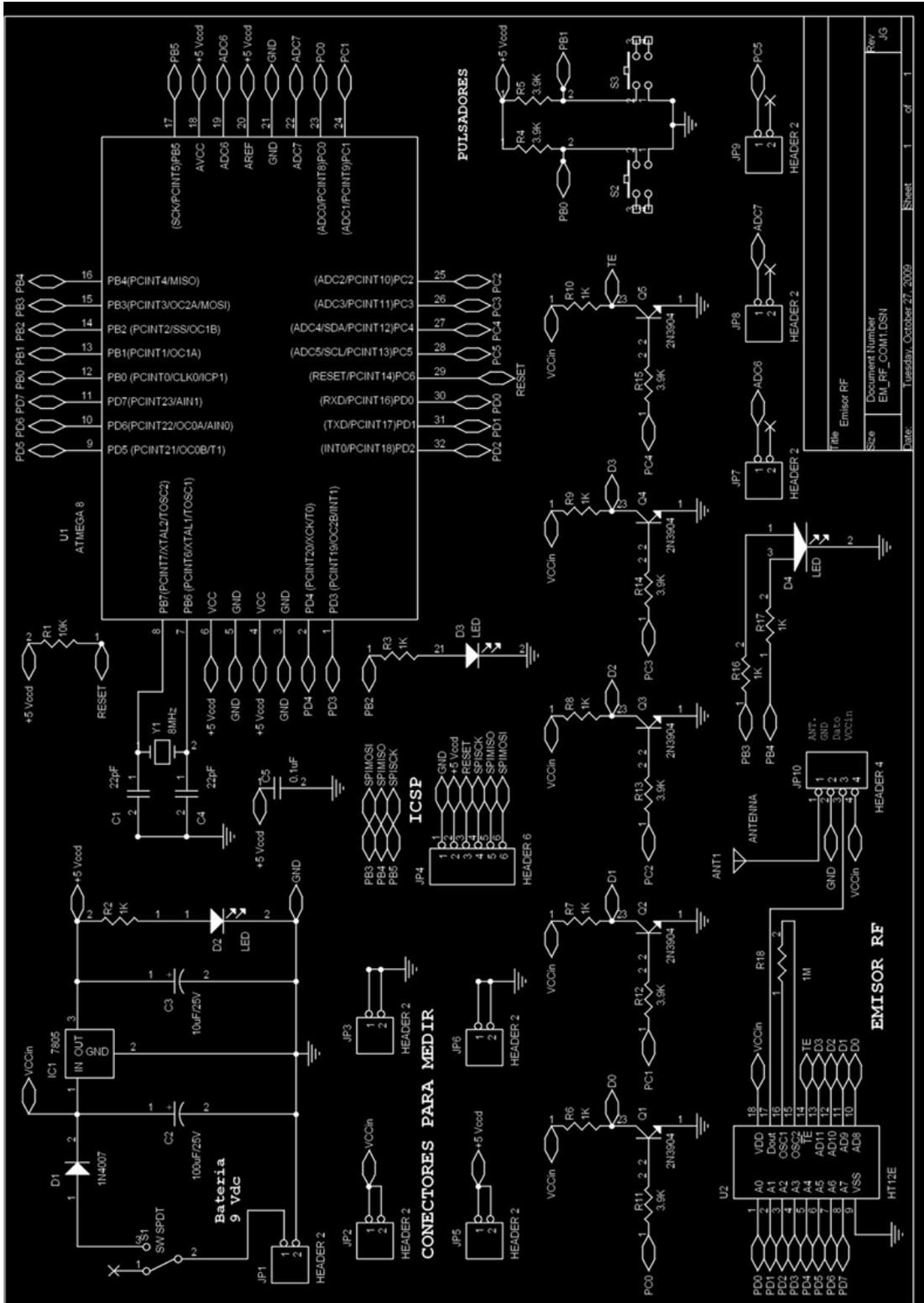


Figura 2.16 Diagrama del Circuito Transmisor

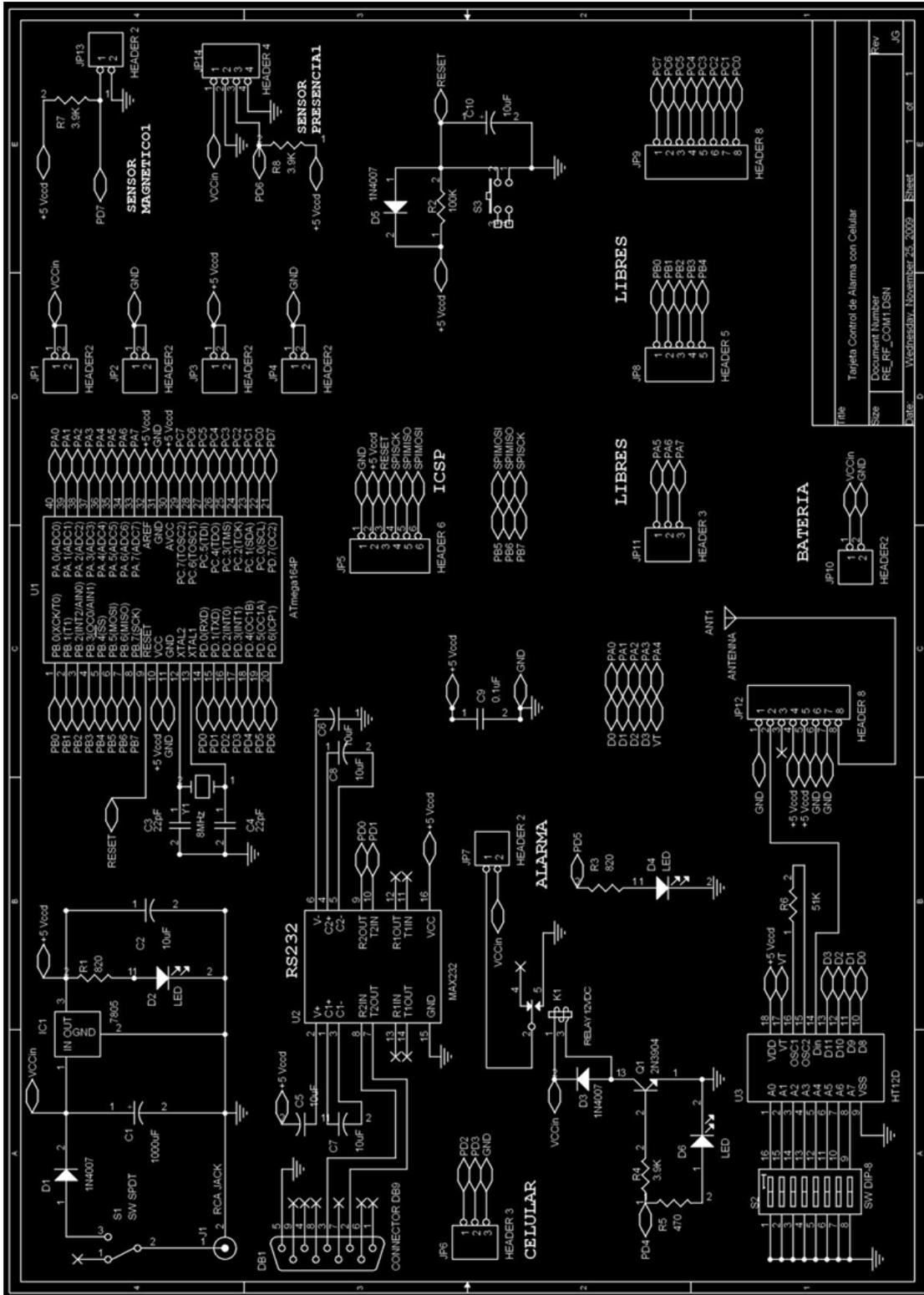


Figura 2.17 Diagrama del Circuito Receptor

CAPÍTULO 3

3. IMPLEMENTACIÓN DEL CIRCUITO

3.1. ACOPLAR LOS CIRCUITOS.

3.1.1. PROGRAMA DEL TRANSMISOR

El microcontrolador ATmega8, de la familia de los AVR, es usado en el circuito emisor, para ello se ha incluido un programa en BASCOM-AVR, basado en lenguaje Basic, dado por el fabricante de este tipo de microcontroladores AVR.

Para iniciar, se observa en el programa la definición del tipo de microcontrolador, frecuencia del cristal, y la velocidad serial.

Para esto se ha usado las instrucciones básicas del programa BASCOM-AVR, en la que indica un conjunto de instrucciones al inicio del programa para detallar la configuración inicial.

- *Configuración inicial*

Para iniciar la programación se debe definir el tipo de microcontrolador a ser usado, para este propósito se usa la instrucción “\$regfile”, esta instrucción se encarga de direccionar el microcontrolador que se va a usar, así también como la instrucción “\$crystal”, que permite señalar el tipo de cristal que el microcontrolador está usando.

\$regfile = "m8def.dat"	'Definición del microcontrolador a utilizar.
\$crystal = 8000000	'Definición de la velocidad del cristal

La definición de entradas y salidas de los pines del microcontrolador se deben seleccionar en el programa, para que el hardware funcione a la necesidad del proyecto.

Los registros DDR, PORT, PIN son usados como entradas o salidas de datos, para esto se describe estos registros de la siguiente manera: "DDR", configura el pin como entrada o salida de datos. "PORT", es el registro de salida de datos y "PIN", es el registro de entrada de datos.

"Alias", permite dar un nombre general dentro de un proyecto, ya sea este un puerto de salida o un puerto de entrada.

Ddrc.0 = 1	
Portc.0 = 1	
Ddrc.1 = 1	
Portc.1 = 1	'El DDR indica si el pin actúa como entrada o salida
Ddrc.2 = 1	(1L = Salida, 0L = entrada)
Portc.2 = 1	
Ddrc.3 = 1	'El Port es un registro de salida de datos.
Portc.3 = 1	
Ddrc.4 = 1	
Portc.4 = 0	
Te Alias Portc.4	'Se da un alias al Port de salida "c", pörtico 4

Para las direcciones de radiofrecuencia se usa el pörtico "d", dándoles el valor de DDR de 1 lógico en todos sus pörticos, es decir que todos los pörticos van a funcionar como salidas.

De igual manera el pörtico "d", va a tomar el valor de 0 lógico en todas sus salidas.

El tipo de variables que se usan deben ser dimensionadas, esto se alcanza con la instrucción "Dim", que sirve para dimensionar el tipo de variable que se utiliza, este tipo de variable, se indica en la Tabla 3-1.

En las siguientes instrucciones se observa, la dimensión de las variables que se van a usar en el programa.

Tipo	Dimensión	Tipo	Dimensión
Bit	0-1	Word	0 a 65535
Byte	0 a 255	String	Cadena de caracteres máximo 254

Tabla 3-1. Variables presentes en el lenguaje Basic

Ddrd = 255	'Ddrd indica que el pÓrtico actúa como salida
Portd = 0	'Portd nos indica el valor asignado el pÓrtico de salida "d", y se deshabilita con 0L
Dirtotal Alias Portd	'Se da un alias al Port de salida "d".
Dim Datorf As Byte	'Variable inicial del programa (0 a 255)
Dim Enviar As Byte	'Variable inicial del programa (0 a 255)
Dim Direccion As Byte	'Variable inicial del programa (0 a 255)
Dim X As Byte	'Variable inicial del programa (0 a 255)
Dim Y As Word	'Variable inicial del programa (0 a 65535)
Const Rebote = 60	'Declara una constante simbÓlica

En el circuito físico, se utiliza un led indicador, para esto al pÓrtico b.2 se maneja como salida mediante el DDR, el cual se coloca en 1 lógico, y en el pÓrtico el valor de 0 lógico a la salida para deshabilitar el encendido del led, a este pÓrtico se le da un alias de "Ledazul".

Ddrb.2 = 1	'Ddrb indica que el pÓrtico actúa como salida
Portb.2 = 0	'Portb indica el valor asignado al pÓrtico de salida "b" y se deshabilita con 0L
Ledazul Alias Portb.2	'Se da un alias al Port de salida "b", pÓrtico 2

Tanto en el pÓrtico b.3 y pÓrtico b.4 se busca controlar un led bicolor, para esto los pÓrticos mencionados mediante el DDR se le da el valor de 1 lógico, para que actúen como pÓrticos de salida del microcontrolador, y para mantenerlos apagados se les da el valor de 0 lógico.

Tanto al pÓrtico b.4 como b.3 tienen un alias denominado Ledrojo y Ledverde respectivamente.

Ddrb.3 = 1 Portb.3 = 0	'Ddrb indica que el p�rtico act�a como salida 'Portb indica el valor asignado al p�rtico de salida "b" , y se deshabilita con 0L
Ddrb.4 = 1 Portb.4 = 0	'Ddrb indica que el p�rtico act�a como salida 'Portb indica el valor asignado al p�rtico de salida "b" , y se deshabilita con 0L
Ledrojo Alias Portb.4	'Se da un alias al Port de salida "b", p�rtico 4
Ledverde Alias Portb.3	'Se da un alias al Port de salida "b", p�rtico 3

Las teclas que se usan en el emisor, van conectadas a los p rticos del microcontrolador b.0 y b.1, estos p rticos trabajan como entradas, los mismos que se han colocado en 0 l gico mediante el DDR, y se ha dado el valor de 1 l gico en sus entradas.

Las entradas de los p rticos denominados Pinb.0 y Pinb.1 se les ha dado un alias llamado *"Tecla1"* y *"Tecla2"* respectivamente.

El valor de *"Portc.4"* o *"Te"* se le da el valor de cero l gico, as  como a los valores de *"Direccion"*, para posteriormente ir a la subrutina *"Dirreceptor"*.

Ddrb.0 = 0 Portb.0 = 1	'Ddrb indica que el p�rtico act�a como entrada 'Portb indica el valor asignado al p�rtico de salida "b" , y se deshabilita con 1L
Ddrb.1 = 0 Portb.1 = 1	'Ddrb indica que el p�rtico act�a como entrada 'Portb indica el valor asignado al p�rtico de salida "b", y se deshabilita con 1L
Tecla1 Alias Pinb.0	'Se da un alias al Pin de entrada "b"
Tecla2 Alias Pinb.1	'Se da un alias al Pin de entrada "b"
Te = 0	'Valor "Te" es igual a cero
Direccion = 0	'Valor "Direccion" es igual a cero
Gosub Dirreceptor	'Ejecuta Subrutina "Dirreceptor"

Para iniciar el programa, cuando se enciende el dispositivo f sico, se tiene un set de animaci n inicial, para esto se ha utilizado un conjunto de instrucciones.

Utilizando la variable “X”, se condiciona para que sus valores vayan de 0 a 5 veces su ejecución.

Cuando “X” esta dentro del rango de trabajo, los pódicos toman el valor de 1 lógico, los cuales se les ha dado un alias “Ledverde”, “Ledrojo”, “Ledazul”, durante un tiempo de 200 mili-segundos. Luego se coloca un 0 lógico en los pódicos llamados mediante un alias “Ledverde”, “Ledrojo”, “Ledazul”, espera nuevamente 200 milisegundos y retorna, mediante la instrucción de programa “Next” a la instrucción “For”, hasta cumplir la variable seleccionada. Cuando termina el set de instrucciones, espera un segundo y continúa trabajando en el programa.

For X = 0 To 5	‘Ejecuta un bloque de instrucciones, un número de veces.
Ledverde = 1	‘Valor Te es igual a cero
Ledrojo = 1	‘Valor Te es igual a cero
Ledazul = 1	‘Valor Te es igual a cero
Waitms 200	‘Espera un tiempo
Ledverde = 0	‘Valor Ledverde es igual a cero
Ledrojo = 0	‘Valor Ledrojo es igual a cero
Ledazul = 0	‘Valor Ledazul es igual a cero
Waitms 200	‘Espera 200 mili – segundos
Next X	‘Ejecuta la variable X un número de veces
Wait 1	‘Espera 1 segundo

- *Funcionamiento inicial del programa en el dispositivo físico.*

El transmisor al ser encendido tiene una animación inicial, la cual consiste en encender y apagar un periodo de cinco veces los leds de colores del transmisor. Esto se logra con una variable denominada “X”, la misma que toma valores consecutivos del 0 al 5, al tomar cada uno de estos valores los pódicos denominados “Ledverde”, “Ledrojo”, “Ledazul”, toman el valor de 1 lógico, el cual físicamente enciende los leds de colores, luego espera 200 milisegundos encendidos los leds, y procede en los

pórticos “Ledverde”, “Ledrojo”, “Ledazul” a colocar el valor de 0 lógico durante otros 200 milisegundos, realizando esta instrucción durante 5 veces.

Esto se aprecia en el Diagrama de flujo 3.1, de forma didáctica.

- *Manejo de instrucciones mediante el teclado*

Cuando el programa continua, espera hasta que el pórtilo llamado mediante un alias “Tecla1”, tome el valor de 0 lógico, Coloca los valores de 1 lógico en el pórtilo “Ledrojo”, y 0 lógico en el pórtilo “Ledverde”. Espera el valor de “Rebote” en milisegundos, es decir 60 milisegundos, con lo cual se evita inestabilidad en el dato que ingresa, espera nuevamente 60 milisegundos y en “Enviar” se coloca el valor de 01 lógico. El valor de “Direccion” se coloca 0 lógico y se dirige a la subrutina “Dirreceptor” y luego cuando retorna a la subrutina “Enviarrf”, espera 100 milisegundos, en el pórtilo “Enviar” coloca un 1 lógico y en “Direccion” le da el valor de 255, es decir coloca 1 lógico en todos los pórtilos del alias “Direccion”, luego ejecuta nuevamente las sub rutinas “Dirreceptor” y “Enviarrf”.

Cuando el programa continúa espera hasta que el pórtilo llamado mediante un alias “Tecla2”, tome el valor de 0 lógico, coloca los valores de 0 lógico en el pórtilo “Ledrojo”, y 1 lógico en el pórtilo “Ledverde”.

Espera el valor de “Rebote” en milisegundos, es decir 60 milisegundos, con lo cual se evita inestabilidad en el dato que ingresa, espera nuevamente 60 milisegundos y en “Enviar” se coloca el valor de 10 lógico.

El valor de “Direccion” se coloca 0 lógico y se dirige a la subrutina “Dirreceptor” y luego cuando retorna a la subrutina “Enviarrf”, espera 100 milisegundos, en el pórtilo “Enviar” coloca un 10 lógico y en “Direccion” le da el valor de 255, es decir coloca 1 lógicos en todos los pórtilos del alias “Direccion”, luego ejecuta nuevamente las sub rutinas “Dirreceptor” y “Enviarrf”.

Para finalizar vuelve a la instrucción “If”, de tal forma, que se mantiene en un lazo infinito.

Do	
If Tecla1 = 0 Then	'Permite la ejecución condicional.
Ledrojo = 1	'Valor Ledrojo es igual a uno
Ledverde = 0	'Valor Ledverde es igual a cero
Waitms Rebote	'Espera 60 mili – segundos
While Tecla1 = 0	'Espera que se suelte la tecla
Wend	'Ejecuta una serie de instrucciones mientras es verdadera
Waitms Rebote	'Elimina rebote al soltar la tecla
Enviar = 1	'Valor Enviar es igual a uno
Direccion = 0	'Valor Direccion es igual a cero
Gosub Dirreceptor	'Ejecuta Subrutina Dirreceptor
Gosub Enviarrf	'Ejecuta Subrutina Enviarrf
Waitms 100	'Espera 100 mili – segundos
Enviar = 1	'Valor Enviar es igual a 1
Direccion = 255	'Valor de Direccion es igual a 255
Gosub Dirreceptor	'Ejecuta Subrutina Dirreceptor
Gosub Enviarrf	'Ejecuta Subrutina Enviarrf
End If	'Retorna al programa

- *Funcionamiento del teclado en el dispositivo físico.*

Las teclas del dispositivo físico permiten enviar un dato mediante radiofrecuencia para activar o desactivar el receptor, para esto se ha diseñado un conjunto de instrucciones en el transmisor que funciona de la siguiente manera.

El programa se mantiene en espera hasta recibir un dato ya sea este de la tecla 1 o la tecla 2.

Cuando se presiona una de las dos teclas, el programa prende el led rojo y apaga el led verde.

En el caso de la tecla 2 apaga el led rojo y enciende el led verde.

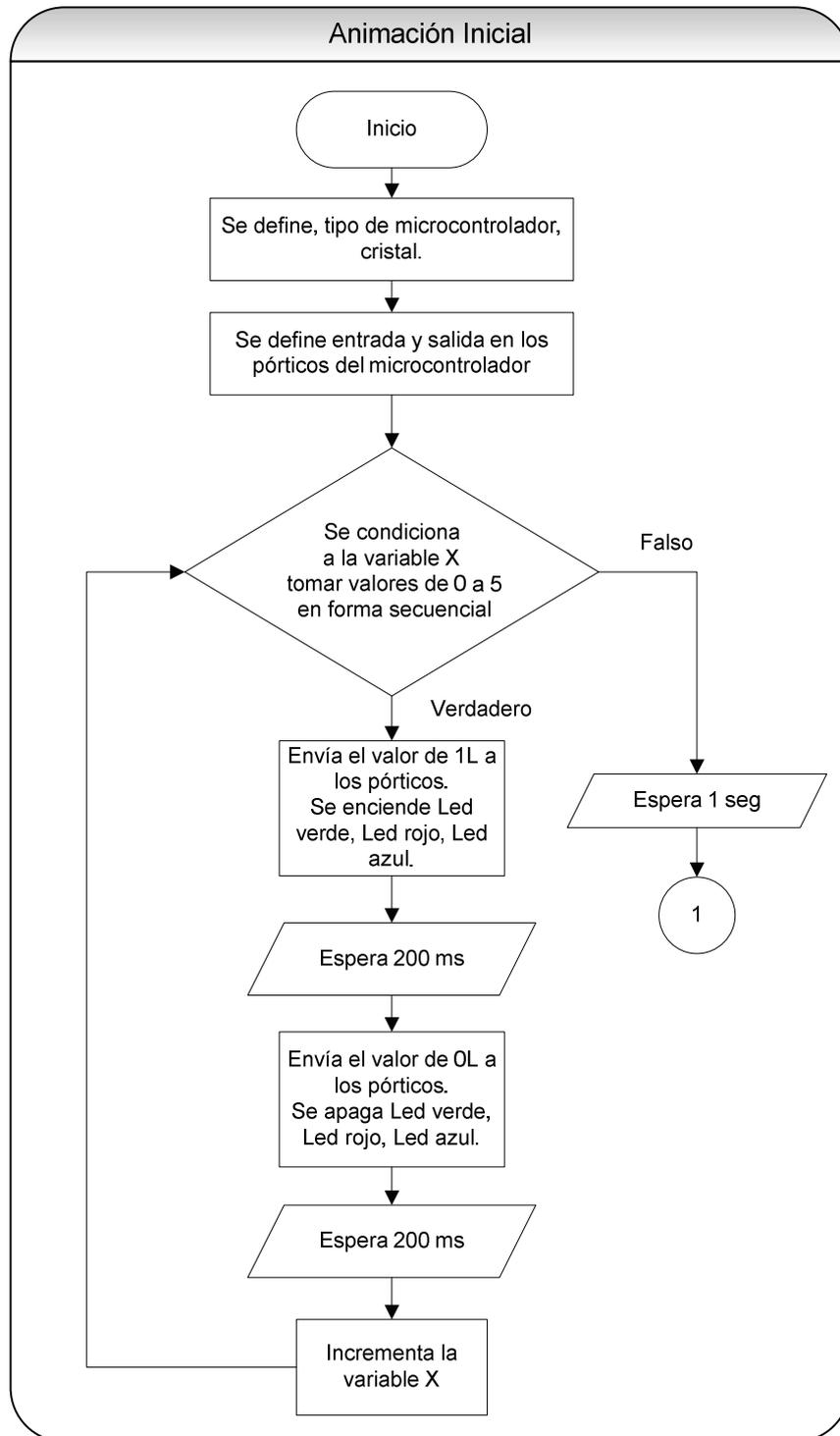


Diagrama de flujo 3.1 Animación Inicial

Cuando termina el proceso del indicador de luces, el programa procede a dar un tiempo llamado de rebote, el cual espera que el dato ingresado mientras es presionado este estable y al ser soltada la tecla no exista confusión en los datos recibidos.

En cada uno de los casos se envía un dato, ya sea este 01 o 10 binario, coloca una dirección con la cual se comunicará con el decodificador del receptor y se enviara el dato mediante radio frecuencia, luego esperará un lapso de tiempo y procede nuevamente a enviar la dirección a otro receptor y nuevamente los datos mediante radiofrecuencia. Ver Diagrama de flujo 3.2

If Tecla2 = 0 Then	'Permite la ejecución condicional.
Ledrojo = 0	'Valor Ledrojo es igual a uno
Ledverde = 1	'Valor Ledverde es igual a uno
Waitms Rebote	'Espera 60 mili – segundos
While Tecla2 = 0	'Espera que se suelte la tecla
Wend	'Ejecuta una serie de instrucciones mientras es verdadera
Waitms Rebote	'Elimina rebote al soltar la tecla
Enviar = 2	'Valor Enviar es igual a dos
Direccion = 0	'Valor Direccion es igual a cero
Gosub Dirreceptor	'Ejecuta Subrutina Dirreceptor
Gosub Enviarrf	'Ejecuta Subrutina Enviarrf
Waitms 100	'Espera 100 mili – segundos
Enviar = 2	'Valor Enviar es igual a 2
Direccion = 255	'Valor de Direccion es igual a 255
Gosub Dirreceptor	'Ejecuta Subrutina Dirreceptor
Gosub Enviarrf	'Ejecuta Subrutina Enviarrf
End If	'Retorna al programa
Loop	'Inicia un lazo infinito hasta cumplir la sub rutina

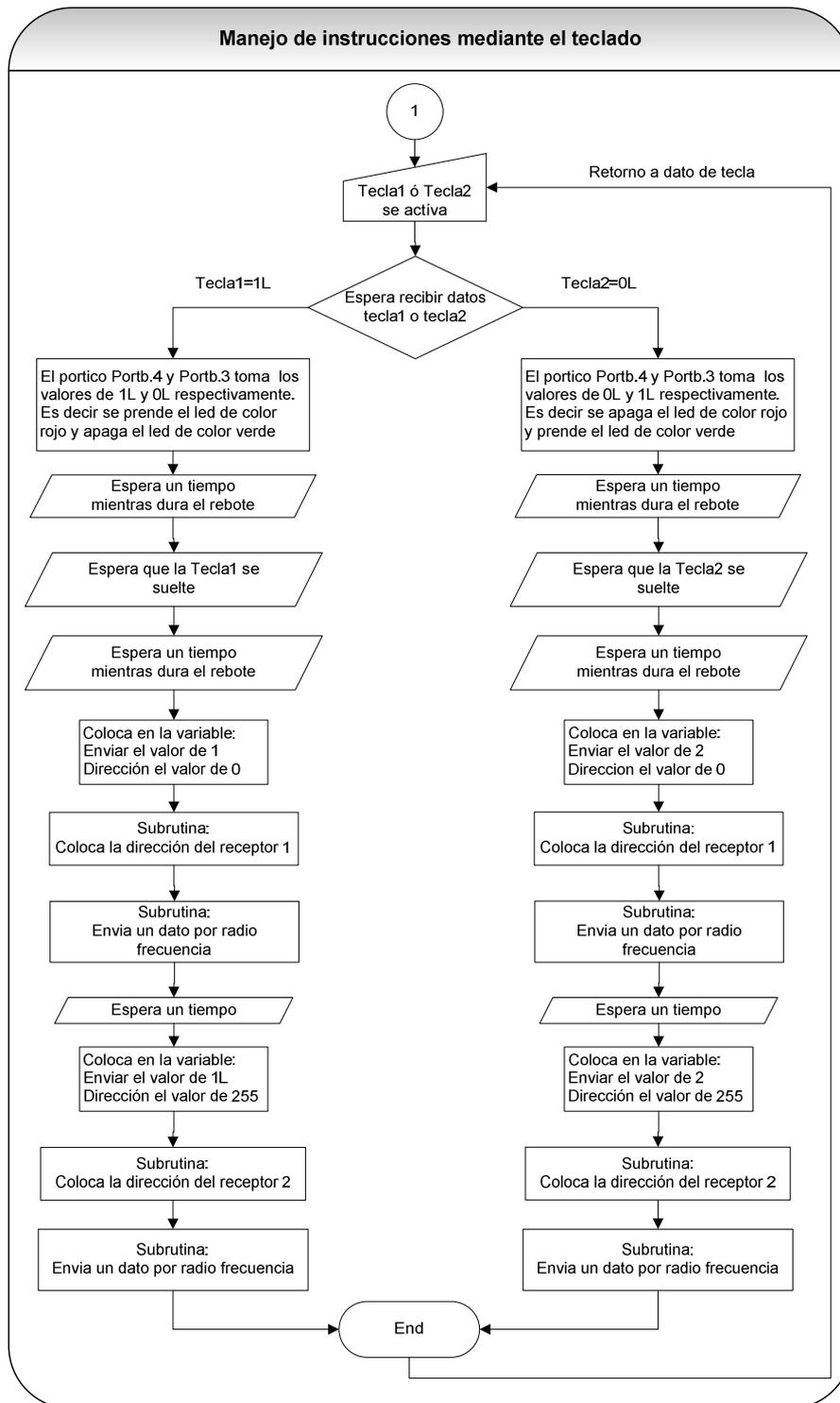


Diagrama de flujo 3.2 Manejo de instrucciones mediante el teclado

- *Subrutina*

En la subrutina “*Enviarrf*”, el p rtico “*Ledazul*” toma el valor de 1 l gico, mientras que en el p rtico “*c*”, en los p rticos menos significativos ingresa un c digo el cual ser  enviado mediante radio frecuencia en el dispositivo f sico. Cabe recalcar que los p rticos mencionados act an como salidas del microcontrolador.

Mediante la funci n “*Waitms*”, el programa espera 10 milisegundos y coloca en el p rtico “*Te*”, el valor de 1 l gico. Este espera 200 milisegundos y coloca nuevamente a “*Te*” en el valor de 0 l gico, espera 10 milisegundos y coloca el “*Ledazul*” en 0 l gico, para luego retornar al programa principal.

Enviarrf:	'Ingresa a la subrutina Enviarrf
Ledazul = 1	'Led azul es igual a 1
Portc.0 = Not Enviar.0	
Portc.1 = Not Enviar.1	'Env�a Contador a P�rtico B (Los cuatro d�gitos menos significativos)
Portc.2 = Not Enviar.2	
Portc.3 = Not Enviar.3	
Waitms 10	'Espera 10 mili-segundos
Te = 1	'Env�a dato por RF
Waitms 200	'Espera 200 mili segundos
Te = 0	'Apaga TE=0
Waitms 10	'Espera 10 mili segundos
Ledazul = 0	'Ledazul es igual a cero
Return	'Retorna al programa principal

- Procedimiento para el env o de radiofrecuencia en el dispositivo f sico.

Para el env o de radiofrecuencia en el dispositivo f sico, el programa permite el encendido del led azul en el circuito, luego env a los datos al p rtico c, los cuales son negados, es decir se invierte su valor de tal forma que el codificador env e los datos al decodificador, se espera un lapso de tiempo y se activa la salida de datos en el pin del decodificador Te, mediante el valor de 1L, luego espera un tiempo y procede a

desactivar Te enviando un 0L, espera un tiempo y se apaga el led azul, colocando un 0L para retornar al programa inicial. Ver Diagrama de flujo 3.3

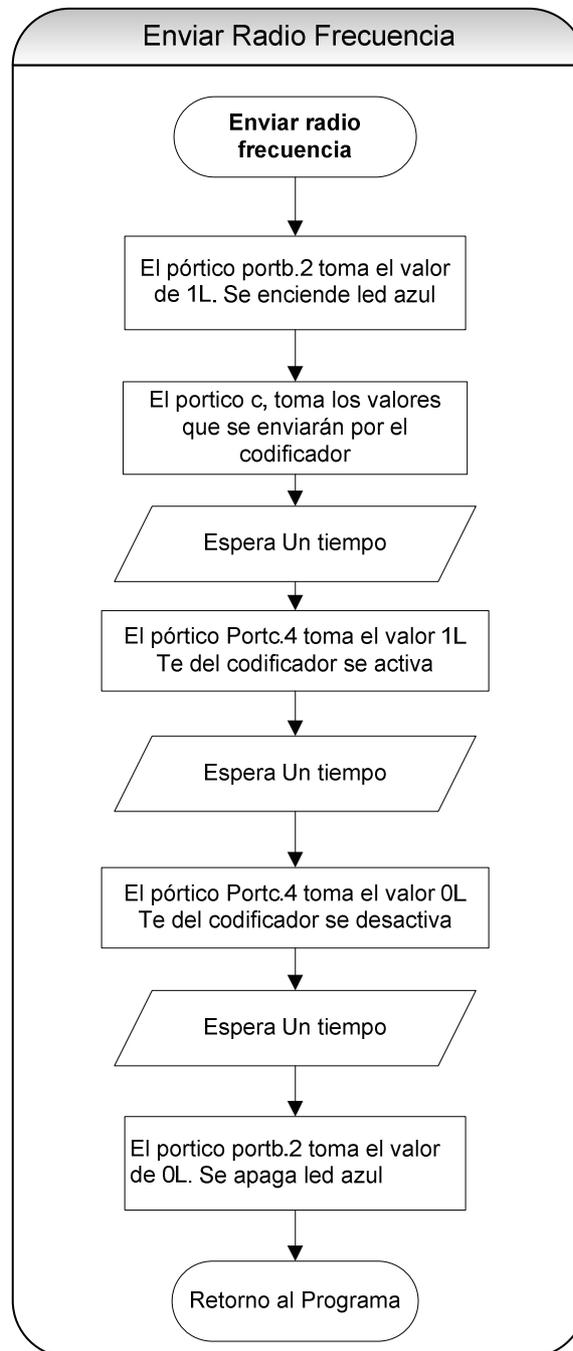


Diagrama de flujo 3.3 Env\u00edo de radiofrecuencia

En la subrutina llamada “Dirreceptor”, los valores que tiene “Dirtotal”, van a ser iguales a “Direccion”, luego del mismo el programa espera 1 milisegundo y retorna al programa principal.

Dirreceptor:	'Ingresa a la sub rutina Dirreceptor
Dirtotal = Direccion	'Dirtotal es igual a Dirección
Waitms 1	'Espera 1 mili-segundo
Return	'Retorna al programa principal
End	'Finaliza el programa

- *Procedimiento para direccionar la información del decodificador*

La subrutina para colocar la dirección a donde debe ser enviado los datos se maneja mediante la variable Dirección, a esta variable en el programa principal se le da un valor ya sea esta de 0 ó 255.

De esta forma se direcciona a los receptores que van a recibir la información generada por el programa. Ver Diagrama de flujo 3.4

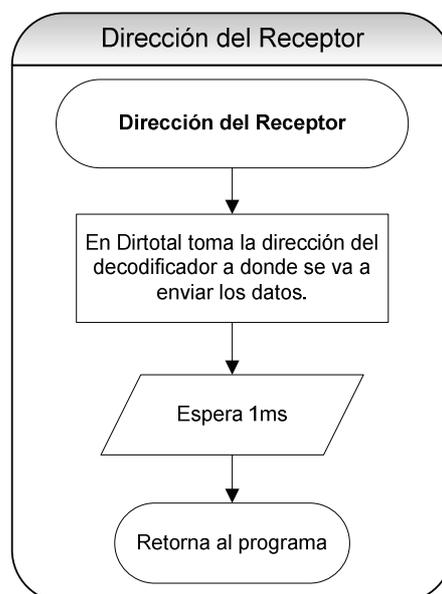


Diagrama de flujo 3.4 Dirección del receptor

3.1.2. PROGRAMA DEL RECEPTOR.

El microcontrolador ATmega 164P, de la familia de los AVR, es usado en el circuito receptor, para ello se ha incluido un programa en BASCOM-AVR, basado en lenguaje Basic, dado por el fabricante de este tipo de microcontroladores AVR.

Para iniciar, se observa en el programa la definición del tipo de microcontrolador, frecuencia del cristal, y la velocidad serial.

Para esto se ha usado las instrucciones básicas del programa BASCOM-AVR, en la que indica un conjunto de instrucciones al inicio del programa para detallar la configuración inicial.

- *Configuración inicial*

Para iniciar la programación se debe definir el tipo de microcontrolador a ser usado, para este propósito se usa la instrucción “\$regfile”, esta instrucción se encarga de direccionar el microcontrolador que se va a usar, así también como la instrucción “\$crystal”, que permite señalar el tipo de cristal que el microcontrolador está usando, también, la función “\$baud”, sirve para determinar la velocidad de transmisión a la que se van a comunicar los dispositivos.

\$regfile = "m164pdef.dat"	'Definición del microcontrolador a utilizar
\$crystal = 8000000	'Definición de la velocidad del cristal
\$baud = 9600	'Velocidad de comunicación serial PC
\$baud1 = 1200	'Velocidad de comunicación serial celular

Se puede observar que el microcontrolador, usado es el Atmega 164P, con un cristal de 8MHz .

El microcontrolador se comunica de forma serial tanto con el computador a una velocidad de transmisión de 9600 bps, y al celular a una velocidad de transmisión de 1200 bps.

Luego se debe dimensionar el tipo de variables que se va a utilizar, esto se logra con la instrucción “*Dim*”, que sirve para dimensionar el tipo de variable que se utiliza, este tipo de variable, se indica en la Tabla 3-2.

Tipo	Dimensión	Tipo	Dimensión
Bit	0-1	Word	0 a 65535
Byte	0 a 255	String	Cadena de caracteres máximo 254

Tabla 3-2. Variables presentes en el lenguaje Basic

En las siguientes instrucciones se observa, la dimensión de las variables que se van a usar en el programa.

Dim Datin As Bit	'Variable inicial del programa (0/1)
Dim Senin As Bit	'Variable inicial del programa (0/1)
Dim B0 As Byte	'Variable inicial del programa (0 a 255)
Dim I As Byte	'Variable inicial del programa (0 a 255)
Dim A As Byte	'Variable inicial del programa (0 a 255)
Dim B As Byte	'Variable inicial del programa (0 a 255)
Dim Sret As String * 20	'Guarda la respuesta del celular en 20 caracteres
Dim Mensa As String * 150	'Guarda el mensaje a enviar, hasta 150 caracteres
Dim Mensaje As String * 150	'Guarda el mensaje recibido, hasta 150 caracteres
Dim Numero1 As String * 14	'Guarda un número de celular de hasta 14 dígitos
Dim Numero2 As String * 14	'Guarda un número de celular de hasta 14 dígitos
Dim Temporal As String * 1	'Guarda un número de hasta 1 dígito
Dim Bandera1 As Bit	'Variable inicial del programa (0/1)
Dim Bandera2 As Bit	'Variable inicial del programa (0/1)
Config Com1 = Dummy, Synchron = 0, Parity = None, Stopbits = 1, Databits = 8, Clockpol = 0	
Config Com2 = Dummy, Synchron = 0, Parity = None, Stopbits = 1, Databits = 8, Clockpol = 0	
Open "com2:" For Binary As #1	'Abre el segundo UART (Transmisor-Receptor Asíncrono Universal) por hardware

La función “*Declare Sub*” permite iniciar la declaración de subrutinas en el programa, estas subrutinas están basadas en el envío, recepción y validación de mensajes, los mismos que se usan en el programa.

Declare Sub Config_inicial()	'Declara una sub rutina para configuración inicial del celular
Declare Sub Getok(s As String)	'Declara una sub rutina para obtener ok
Declare Sub Limpiarbuffer()	'Declara una sub rutina para limpiar buffer
Declare Sub Enviarmensaje(s As String , N As String)	'Declara una sub rutina para enviar mensaje
Declare Sub Recibirmensaje()	'Declara una sub rutina para recibir mensaje
Declare Sub Validarmensaje(s As String)	'Declara una sub rutina para validar mensaje

La interrupción del p rtico serial, se basa en una sub rutina donde se almacena un dato de llegada, que proviene de una fuente externa.

Para esto se usa el comando *“On Urxc Rec_isr”* que activa la sub rutina llamada *“Rec_isr”*, cuando recibe un dato. Con la funci n *“Enable Urxc”* permite habilitar la interrupci n del Puerto serial y *“Enable Interrupts”* permite las interrupciones presentes en el microcontrolador.

Dim Texto2 As String * 200	'Guarda el mensaje recibido, hasta 200 caracteres
Dim Control As Byte	'Guarda un dato de 0 hasta 255 caracteres
Dim Z As String * 3	'Guarda un dato de hasta 3 caracteres
On Urxc Rec_isr	'Activa la subrutina serial, en caso de recibir un dato
Enable Urxc	'Habilita la interrupci�n de recepci�n del puerto serial
Enable Interrupts	'Habilita las interrupciones presentes en el microcontrolador

La definici n de entradas y salidas de los p rticos del microcontrolador se deben seleccionar en el programa, para que el hardware funcione a la necesidad del proyecto.

Los registros DDR, PORT, PIN son usados como entradas o salidas de datos, para esto se describe estos registros de la siguiente manera: *“DDR”*, configura el pin como

entrada o salida de datos. “*PORT*”, es el registro de salida de datos y “*PIN*”, es el registro de entrada de datos.

”*Alias*”, permite dar un nombre general dentro de un proyecto, ya sea este un puerto de salida o un puerto de entrada.

'RECEPTOR DE RF

Ddra.0 = 0 : Porta.0 = 1	
Ddra.1 = 0 : Porta.1 = 1	'El DDR indica si el pin actúa como entrada o salida (1L = Salida, 0L = entrada)
Ddra.2 = 0 : Porta.2 = 1	
Ddra.3 = 0 : Porta.3 = 1	'El Port es un registro de salida de datos.
Ddra.4 = 0 : Porta.4 = 1	
Rfdata0 Alias Pina.0	
Rfdata1 Alias Pina.1	
Rfdata2 Alias Pina.2	'Se da un alias al pin de entrada “a”, en los pórticos 1,2,3
Rfdata3 Alias Pina.3	
Vt Alias Pina.4	'Se da un alias al pin de entrada “a”, pórtico 4

'ALARMA (SIRENA)

Ddrd.4 = 1	'Al pin “d”, pórtico 4 se le da el valor de salida 1L
Portd.4 = 0	'Al pin “d”, pórtico 4, inicia en 0L
Alarma Alias Portd.4	'Se da un alias al pin “d”, pórtico 4.

'LED INDICADOR

Ddrd.5 = 1	'Al pin “d”, pórtico 5 se le da el valor de salida 1L
Portd.5 = 0	'Al pin “d”, pórtico 5, inicia en 0L
Led1 Alias Portd.5	'Se da un alias al pin de salida “d”, pórtico 5.

'SENSOR PRESENCIA

Ddrd.6 = 0	'Al pin “d”, pórtico 6 se le da el valor de entrada 0L
Portd.6 = 1	'Al pin “d”, pórtico 6, inicia en 1L
Senpre1 Alias Pind.6	'Se da un alias al pin de entrada “d”, pórtico 6.

'SENSOR MAGNÉTICO

Ddrd.7 = 0	'Al pin “d”, pórtico 7 se le da el valor de entrada 0L
Portd.7 = 1	'Al pin “d”, pórtico 7, inicia en 1L
Senmag1 Alias Pind.7	'Se da un alias al pin de entrada “d”, pórtico 7.

Ahora se ingresa el número de celular, al cual se enviará los mensajes necesarios.

Para esto en “numero1” y “numero2”, se asegura no exista algún dato que cambie algún valor que se almacene.

En el “numero1” se toma las direcciones donde está el “numero1”, que van desde la localidad 1 hasta 9 leyendo la eeprom el valor de la variable “B”, la misma que se guarda en una localidad llamada “Temporal” y esta a su vez se suma al “numero1” y se repite esta secuencia el valor condicionado por I, de igual manera sucede esto para el “numero2”, tan solo que va desde la localidad 12 hasta 20

Numero1 = ""	'Se deja en blanco el número de celular
For I = 1 To 9	'Constante “I”, que va de la localidad 1 a 9
Readeeprom B , I	'Dirección que genera un dato y se lee en la eeprom.
Temporal = Chr(b)	'En temporal se almacena el nuevo dato.
Numero1 = Numero1 + Temporal	'El número se aumenta hasta que “I” llega a 9
Next I	'Repite la operación hasta que “I” llegue a 9
Numero2 = ""	'Se deja en blanco el número de celular
For I = 12 To 20	'Contante “I”, que va de la localidad 12 a 20
Readeeprom B , I	'Dirección que genera un dato y se lee en la eeprom.
Temporal = Chr(b)	'En temporal se almacena el nuevo dato.
Numero2 = Numero2 + Temporal	'El número se aumenta hasta que “I” llega a la localidad 20
Next I	'Repite la operación hasta que “I” llegue a la localidad 20

- *Programa principal*

El programa principal consta de tres partes principales: control del celular, radiofrecuencia, y sensor magnético.

Para esto se ha ejecutado varias órdenes en el programa con el fin de controlar todos estos aspectos.

En cada una de estas instrucciones se ha creado una condición, la misma que al cumplirse activa una orden, caso contrario pasa a la siguiente.

- *Comunicación del microcontrolador con el celular.*

La primera instrucción se basa en el control de la comunicación con el celular, para esto se espera el ingreso de un carácter al celular para activar esta instrucción, luego si la variable “A” es igual a 1, se dirige al sub proceso “*recibirmensaje*”, luego al sub proceso “*validarmensaje*” espera 250 milisegundos y finaliza su ejecución, volviendo a un lazo infinito, hasta recibir una instrucción.

La instrucción “*Ischarwaiting(#channel)*”, retorna un 1, cuando un carácter está esperando en el hardware UART del buffer, es decir en este caso en el com2.

Mientras que “*#channel*” es un número constante que identifica el canal abierto.

Cuando “A” es igual a 1, se recibe el mensaje, pero si es diferente continua a la siguiente instrucción. En caso de ser 1, la instrucción recibe el mensaje y pasa al sub proceso Toggle_led1, y posteriormente al sub proceso de validar el mensaje, espera 250 milisegundos y retorna a la instrucción “If”.

Do	
A = Ischarwaiting(#1)	'Espera la existencia de un carácter en el celular com2.
If A = 1 Then Recibirmensaje	'Si A=1, recibe el mensaje, si es diferente continua.
Gosub Toggle_led1	'Se dirige a la subrutina toggle_led1
Validarmensaje Mensaje	'Se dirige a la subrutina de validar mensaje
Waitms 250	'Espera 250 mili-segundos
End If	'Retorna al programa

- *Funcionamiento físico del celular con el microcontrolador.*

El microcontrolador espera hasta que el celular indique que tiene un carácter en la memoria del celular, cuando esto ocurre comunica al modem colocando el valor de

1L en la variable "A", lo cual permite al microcontrolador recibir el mensaje y posteriormente validar el mensaje para realizar una instrucción específica. Ver Diagrama de flujo 3.5

- Comunicación del receptor y emisor, mediante radio frecuencia

Para la recepción de radiofrecuencia, el decodificador espera que en el Pina.4 o Vt, ingrese un 1 lógico, para activar la instrucción, es decir que "Datin" sea igual a 1L.

Cuando se tiene un 1L, el programa se dirige a un sub proceso llamado "Revisarrf", vuelve al programa principal y finaliza volviendo a la instrucción "If".

Para la recepción de radiofrecuencia, el decodificador espera que en el Pina.4 o Vt, ingrese un 1 lógico, para activar la instrucción, es decir que "Datin" sea igual a 1L.

Cuando se tiene un 1L, el programa se dirige a un sub proceso llamado "Revisarrf", vuelve al programa principal y finaliza volviendo a la instrucción "If".

Datin = Vt	'Datin toma el valor de Vt, para activar el decodificador necesita ser igual a 1L
If Datin = 1 Then	'Si Datin es igual a 1L, la instrucción continua al siguiente paso.
Gosub Revisarrf	'Se dirige a la subrutina revisarrf
End If	'Retorna al programa

- *Funcionamiento físico de Radio frecuencia.*

Cuando el transmisor envía un dato, con los cuatro dígitos menos significativos, el receptor da el valor de 1L en Vt del decodificador, y procede a activar una instrucción dada desde el transmisor. Ver Diagrama de flujo 3.6

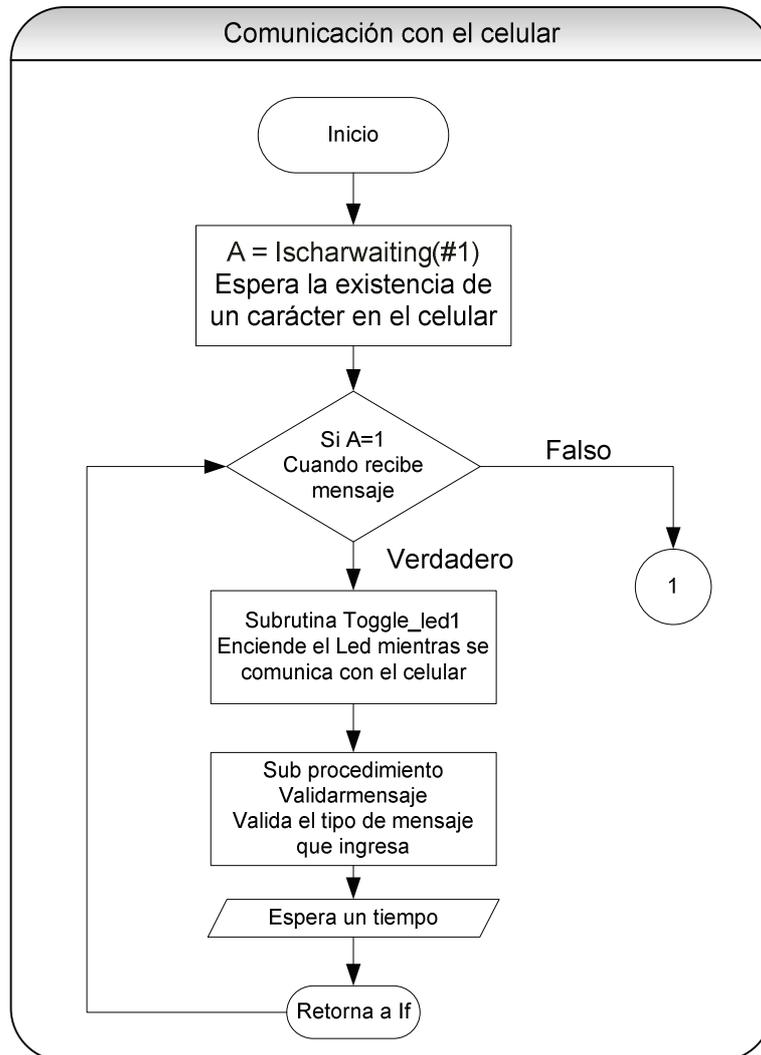


Diagrama de flujo 3.5 Comunicación con el teléfono celular

- *Comunicación del sensor magnético con el microcontrolador*

De igual forma, para trabajar con los sensores se toma el mismo procedimiento, es decir “*Senin*” espera tomar el valor de “*Senmag1*”.

Si “*Senin*” es igual a 1L, la “*bandera1*” es igual a 0L, pasa a 1L, la “*bandera2*” es igual a 1L, se envía nuevamente el mensaje de puerta abierta a “*mensa*” y se envía la frase a un número de celular específico.

Se limpia el buffer y se espera un valor en “Mensa”.

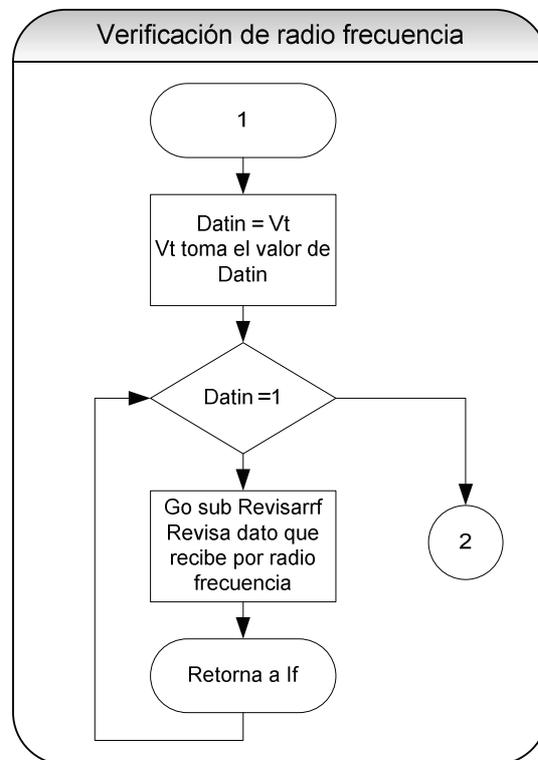


Diagrama de flujo 3.6 Verificación de radio frecuencia

“Mensa” toma nuevamente el valor de mensaje, se envía al celular “Numero2” seleccionado.

Se limpia el buffer, se deja en blanco mensa y finaliza la instrucción retornando a la instrucción “If”.

Senin = Senmag1	'Indica que Senin toma el mismo valor que Senmag1
If Senin = 1 Then	'Revisar el sensor magnético, si es 1L realiza una actividad.
If Bandera1 = 0 Then	'Si la Bandera1 es 0L, se activa
Bandera1 = 1	'La bandera 1 pasa a 1L
Bandera2 = 1	'La bandera 2 pasa a 1L, evitando nuevamente el mensaje.
Mensaje = "Puerta Abierta1"	'Indica que Mensaje tiene el mismo valor que “Puerta Abierta 1”

Mensa = Mensaje	'Indica que Mensa tiene el mismo valor que Mensaje.
Enviarmensaje Mensa , Numero1	'Envía el mensaje al número de celular 1
Limpiarbuffer	'Limpia el buffer
Mensa = ""	'Espera un valor en mensa
Mensaje = "Puerta Abierta1"	'Indica que Mensaje tiene el mismo valor que "Puerta Abierta 1"
Mensa = Mensaje	'Indica que Mensa tiene el mismo valor que Mensaje.
Enviarmensaje Mensa , Numero2	'Envía el mensaje al número de celular 2
Limpiarbuffer	'Limpia el buffer
Mensa = ""	'Se limpia el mensaje en Mensa
End If	'Retorna al programa

- Funcionamiento físico del Sensor magnético.

El sensor al activarse, coloca un 1L en el pórtilo, indicando al microcontrolador comunicarse con el dispositivo celular mediante un mensaje de gabinete abierto o gabinete cerrado.

Se debe observar que el gabinete se encuentra cerrado cuando el pórtilo toma el valor de 0L. Ver Diagrama de flujo 3.7 y 3.8

- *Desarrollo del sub procedimiento para comunicar el microcontrolador con el celular.*

Para iniciar la comunicación, se debe definir el sub procedimiento, que está compuesto por la función "SUB" seguido de un nombre, el cual no se ha usado en alguna otra parte del programa.

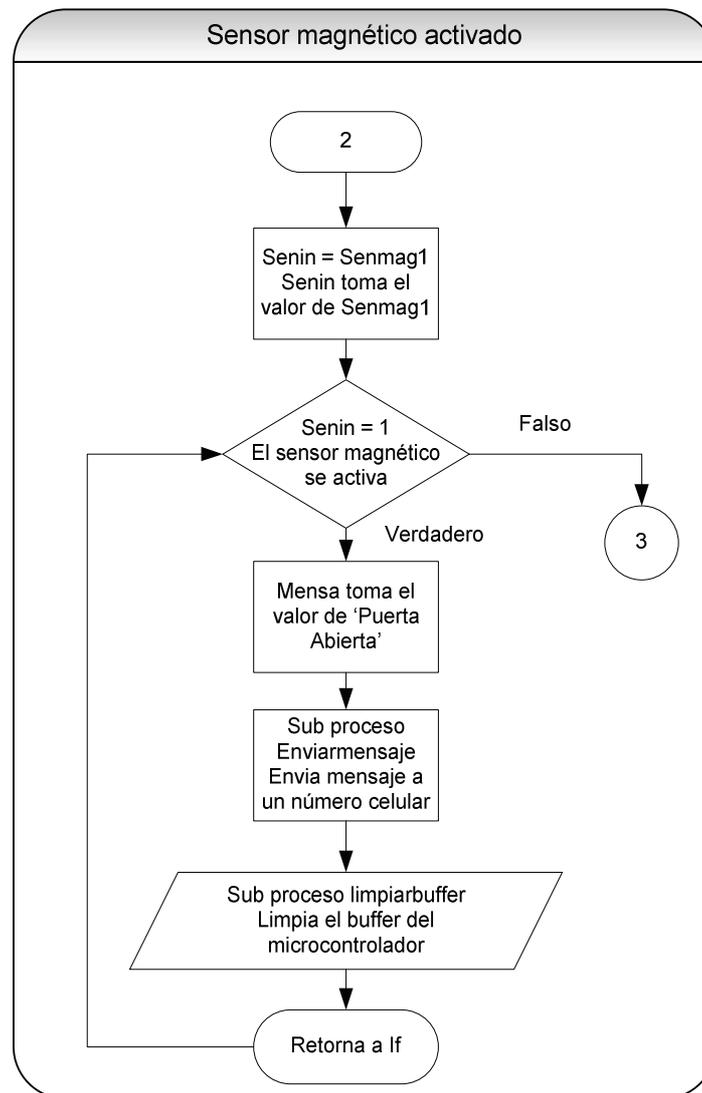
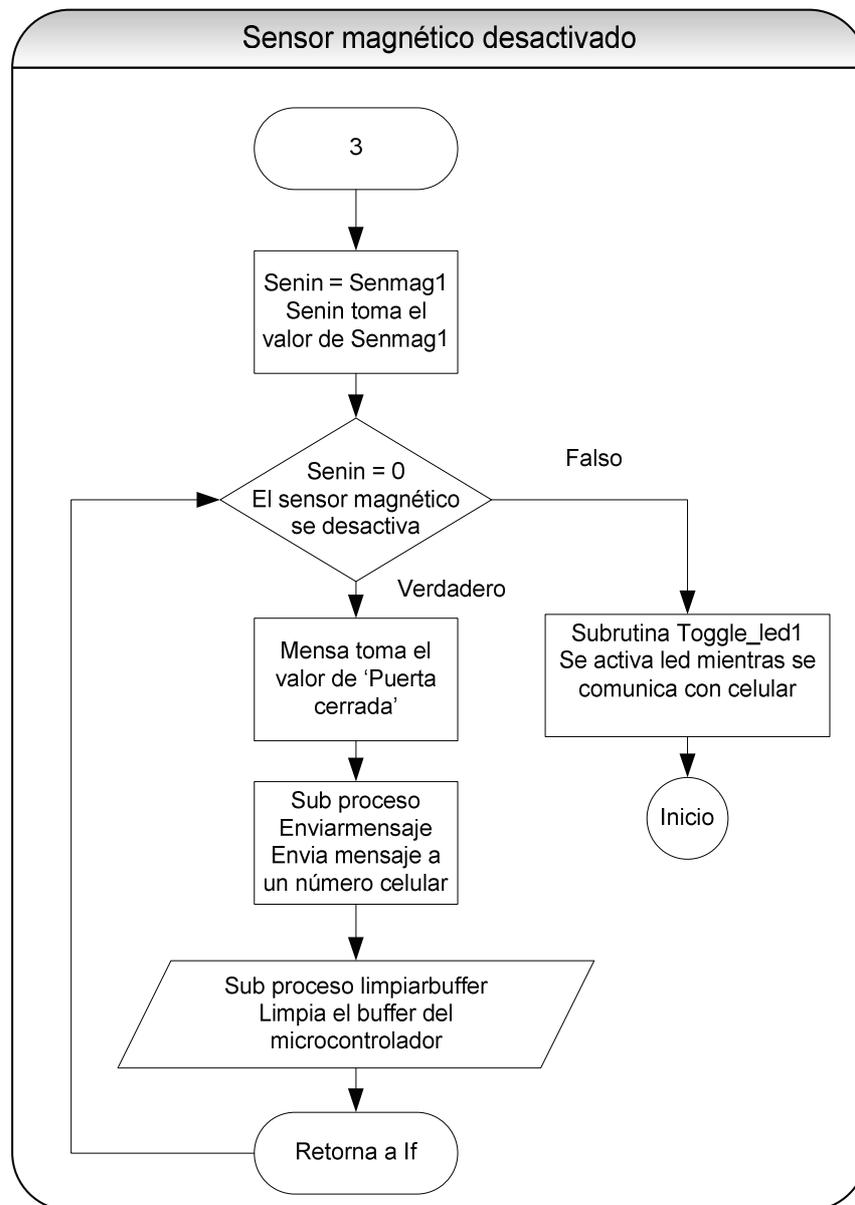


Diagrama de flujo 3.7 Gabinete Abierto

Luego se usa *“Print”* el cual se encarga de enviar al pórtico serial RS-232, una cadena de datos de un archivo, en este caso se envía AT, y el celular recibe como *“atención”*.

Luego un *“enter”* con lo cual se inicia la comunicación entre el microcontrolador y el equipo celular.



Flujo grama 3.8 Gabinete cerrado

El microcontrolador espera la respuesta del celular mediante la instrucción “*Getok Sret*”, la cual espera la respuesta “*OK*” del celular, manteniéndose en un lazo infinito mediante la instrucción “*loop*”, cuando recibe “*OK*” limpia el buffer, espera 25 milisegundos y continua.

Sub Config_inicial()	'Define un sub procedimiento
Do	
Print #1 , "AT" ; Chr(13);	'Se envía "AT" + ENTER, se inicia la comunicación.
Getok Sret	'Espera que el celular devuelva "OK"
Loop Until Sret = "OK"	'Espera hasta que Sret sea igual a "OK"
Limpiarbuffer	'Se limpia el buffer
Waitms 25	'Espera 25 mili-segundos

Ahora se debe apagar el eco de respuesta, para esto se usa la siguiente instrucción, mediante el *"Print"*, que envía mediante el pòrtico serial RS-232, la palabra *"ATE0"* y se da un enter, con el fin de detener el eco de respuesta del celular, esta instrucción viene dada por el fabricante del celular Nokia.

Mediante *"Getok Sret"*, el microcontrolador espera la respuesta del celular, hasta recibir un *"OK"*, luego limpia el buffer, espera 25 milisegundos y continúa.

Do	
Print #1 , "ATE0" ; Chr(13);	'Se recibe "OK", y apaga el eco de respuesta
Getok Sret	'Espera que el celular devuelva "OK"
Loop Until Sret = "OK"	'Espera hasta que Sret sea igual a "OK"
Limpiarbuffer	'Se limpia el buffer
Waitms 25	'Espera 25 milisegundos

Cuando se apaga el eco de respuesta, se procede a configurar la respuesta en modo de texto.

Usando la instrucción *"Print #1, AT+CMGF=1"* y un enter, para configurar la respuesta en modo de texto, espera que el celular responda *"OK"*, y luego limpia el buffer, espera 25 milisegundos y continúa.

```

Do
Print #1 , "AT+CMGF=1" ; Chr(13); 'Se espera "OK", Configura respuesta en modo de texto
Getok Sret 'Espera que el celular devuelva "OK"
Loop Until Sret = "OK" 'Espera hasta que Sret sea igual a "OK"
Limpiarbuffer 'Se limpia el buffer
Waitms 25 'Espera 25 mili-segundos

```

El sub procedimiento para finalizar se encarga de enviar el mensaje por el cable serial, espera respuesta de "OK" del celular, limpia el buffer, espera 25 milisegundos y repite el procedimiento con el fin de enviar nuevamente el mensaje, y finaliza el sub procedimiento.

```

Do
Print #1 , "AT+CNMI=1,2,0,0,0" ; Chr(13); 'Espera "OK", mientras ingresa mensaje, se
envía por cable.
Getok Sret 'Espera que el celular devuelva "OK"
Loop Until Sret = "OK" 'Espera hasta que Sret sea igual a "OK"
Limpiarbuffer 'Se limpia el buffer
Waitms 25 'Espera 25 mili segundos
Do
Print #1 , "AT+CNMI=1,2,0,0,0" ; Chr(13); 'Espera "OK", mientras ingresa mensaje, se
envía por cable.
Getok Sret 'Espera que el celular devuelva "OK"
Loop Until Sret = "OK" 'Espera hasta que Sret sea igual a "OK"
Limpiarbuffer 'Se limpia el buffer
Waitms 25 'Espera 25 mili segundos
End Sub 'Fin del sub procedimiento

```

- Sub procedimiento para esperar la respuesta "OK"

El sub procedimiento "Getok", permite esperar la respuesta "OK" del celular, Para iniciar, la variable "S" toma el valor de cero en todos sus caracteres, luego mediante la instrucción "Inkey (#1)", se logra leer carácter por carácter en la variable "B".

Mediante la instrucción “*Select Case*”, es una sentencia que se puede ejecutar dependiendo de una variable de selección, esta variable se va sumando de carácter en carácter en la variable “S”, finalizando la selección hasta completar la palabra “OK” y finaliza el sub procedimiento.

Sub Getok(s As String)	'Sub procedimiento para obtener “OK”
S = ""	'La variable “S” recibe el valor de cero
Do	
B = Inkey(#1)	'Se lee carácter por carácter
Select Case B	'Depende de una variable de selección.
Case 0	'Valor “Nulo”
Case 13	'Valor “Enter”
Case 10	'Valor “Avance de línea”
If S <> "" Then Exit Do	'Espera el valor de “S”
Case Else	'Condiciona el valor de case
S=S+Chr (b)	'Suma en la variable “S”
End Select	'Finaliza la selección
Loop	'Inicia un lazo infinito hasta cumplir el sub procedimiento
End Sub	'Finaliza el sub procedimiento

- Sub procedimiento para limpiar buffer UART2

Se activa el sub procedimiento llamado “*limpiarbuffer*”, encargado de limpiar los canales de comunicación entre el micro controlador y el celular, para esto espera 50 mili segundos, ingresa en variable “B” carácter por carácter en un lazo infinito hasta que “B” tenga el valor de cero y finaliza el sub procedimiento.

Sub Limpiarbuffer()	'Sub procedimiento para limpiar buffer
Waitms 50	'Espera 50 milisegundos
Do	
B = Inkey(#1)	'Se lee carácter por carácter
Loop Until B = 0	'Espera hasta que B sea igual a 0
End Sub	'Finaliza el sub proceso

- Sub procedimiento para enviar mensajes.

Se activa el sub procedimiento “*Enviarmensaje*”, mientras espera la respuesta de “*OK*” vía el cable serial. Mientras tanto con la función “*Getok Sret*”, espera que el celular responda “*OK*” en un lazo infinito hasta cumplir con la condición.

Luego se dirige al sub procedimiento de limpiar el buffer y continua en el sub proceso principal de enviar mensaje.

El celular espera la respuesta del microcontrolador con el valor de carácter “>”, se deja en blanco “*Sret*”. Ahora en la variable “*B*” se lee de carácter en carácter, cada una de las variables de selección, hasta que a la variable “*Sret*” cumpla la condición de recibir un carácter de avance de línea, luego salta a una etiqueta específica llamada “*Envio1*”, la cual limpia el buffer, espera 100 mili segundos vuelve a limpiar el buffer, envía el mensaje y espera la respuesta del celular mediante un “*OK*”, después se dirige a un sub proceso el cual da el valor de “*OK*”, para salir de un lazo de espera, limpia el buffer, espera 500 milisegundos y finaliza el sub proceso.

Sub Enviarmensaje(s As String , N As String)	'Sub procedimiento para enviar mensaje
Do	
Print #1 , "AT" ; Chr(13);	'Espera respuesta “ <i>OK</i> ”
Getok Sret	'Espera que el celular devuelva “ <i>OK</i> ”
Loop Until Sret = "OK"	'Espera hasta que Sret seas igual a “ <i>OK</i> ”
Limpiarbuffer	'Limpia el buffer
Do	
Print #1,"AT+CMGS=";Chr(34) ; N ;Chr(34) ; Chr(13);	'Espera respuesta “>”
Sret = ""	'Sret se coloca en blanco.
Do	
B = Inkey(#1)	'Se lee caracter por caracter
Select Case B	'Depende de una variable de selección.
Case 0	'Valor “Nulo”
Case 13	'Valor “Enter”

Case 10	'Valor "Avance de línea"
If Sret <> "" Then Exit Do	'Espera el valor de "Sret"
Case 62	'Valor ">"
Goto Envio1	'Salta a la etiqueta especificada
Case Else	'Condiciona el valor de case
Sret = Sret + Chr(b)	'Suma en la variable "Sret"
End Select	'Finaliza la selección
Loop	'Inicia un lazo infinito hasta cumplir el sub procedimiento
Loop Until Sret = ">"	'Espera hasta que Sret tenga el carácter ">"
Envio1:	'Ingreso a etiqueta "envio1"
Limpiarbuffer	'Se dirige al sub proceso "limpiar buffer"
Waitms 100	'Espera 100 mili segundos
Limpiarbuffer	'Se dirige al sub proceso "limpiar buffer"
Print #1 , S ; Chr(26);	'Espera respuesta OK, mensaje + ctrl Z
Do	
Getok Sret	'Sub procedimiento para obtener "OK"
Loop Until Sret = "OK"	'Espera hasta que Sret tenga el valor "Ok"
Limpiarbuffer	'Se dirige al sub proceso "limpiar buffer"
Waitms 500	'Espera 500 mili-segundos
End Sub	'Finaliza el sub proceso

- *Sub proceso para recibir mensajes.*

Al iniciar el sub proceso "Recibir mensaje", la primera instrucción es comunicar al microcontrolador con el celular mediante el cable serial.

La variable "B", empieza a leer de caracter en caracter los diez dígitos del celular gravados en la memoria eeprom manteniendo un lazo hasta cumplir el valor asignado.

La variable Mensaje se deja en blanco para posteriormente colocar un mensaje de texto, para esto se va leyendo la variable "B" tomando variables de selección hasta

tener los valores deseados. Va sumando los valores en la variable Mensaje hasta finalizar la selección y procede a un lazo infinito.

Luego mediante el cable serial se envía un mensaje vía cable serial. En la variable “B” lee de carácter en carácter hasta un valor de cero hasta terminar la rutina de Do, es decir que el valor de “B” sea cero y termina el sub proceso.

Sub Recibirmensaje()	'Sub procedimiento para recibir mensaje
Print #1 , Mensaje	'Envía Mensaje por el cable serial
Do	
B = Inkey(#1)	'Se lee carácter por carácter
Loop Until B = 10	'Espera hasta que B sea igual a 10
Mensaje = ""	'Deja “Mensaje” en blanco
Do	
B = Inkey(#1)	'Se lee carácter por carácter
Select Case B	'Depende de una variable de selección.
Case 13:	'Valor “Nulo”
Case 10:	'Valor “Enter”
Exit Do	'Sale de la estructura “Do” en cualquier momento
Case Else	'Ejecuta case si la selección es falsa
Mensaje = Mensaje + Chr(b)	'Suma los dígitos en “Mensaje”
End Select	'Finaliza la selección.
Loop	'Se mantiene en un lazo infinito
Print #1 , Mensaje	'Envía por cable serial el mensaje
Do	
B = Inkey(#1)	'Se lee carácter por carácter
Loop Until B = 0	'Se mantiene un lazo hasta que B es igual a cero
End Sub	'Finaliza el sub proceso

- *Subrutina para validar mensaje*

Cuando el celular envía un mensaje el programa se encarga de validar el tipo de mensaje que ingresa al microcontrolador.

El microcontrolador espera dos tipos de mensaje los cuales debe validar, “ALARMA OFF” y “ALARMA ON” los mismos que encienden la alarma o apagan la misma, es decir con reset en alarma coloca 0L y con set en alarma 1L.

Coloca en “Mensaje” un valor en texto el cual se enviará como comunicado a un número de celular seleccionado, se limpia el buffer, se limpia “Mensa” y finaliza la selección del mensaje de ingreso, espera un tiempo y retorna al programa. Ver Diagrama de flujo 3.9

Sub Validarmensaje(s As String)	'Sub procedimiento para recibir mensaje
Select Case S	'Depende de una variable de selección
Case "ALARMA OFF":	'Case espera “ALARMA OFF”
Reset Alarma	'Reset un bit a cero
Mensaje = "ALARMA APAGADA ESFOT 1"	'Coloca un valor en caracteres en “Mensaje”
Mensa = Mensaje	'Mensa es igual a Mensaje
Enviarmensaje Mensa , Numero1	'Sub proceso “ <i>Enviarmensaje</i> ”, al “numero1”
Limpiarbuffer	'Limpiar buffer
Mensa = ""	'Deja en blanco Mensa
Mensaje = "ALARMA APAGADA ESFOT 1"	'Coloca un valor en caracteres en “Mensaje”
Mensa = Mensaje	'Mensa es igual a Mensaje
Enviarmensaje Mensa , Numero2	'Sub proceso “ <i>Enviarmensaje</i> ”, al “Numero2”
Limpiarbuffer	'Limpiar buffer
Mensa = ""	'Deja en blanco Mensa
Case "ALARMA ON":	'Case espera “ALARMA ON”
Set Alarma	'Establece un bit a Alarma
Mensaje = "ALARMA ENCENDIDA ESFOT1"	'Coloca un valor en caracteres en “Mensaje”
Mensa = Mensaje	'Mensa es igual a Mensaje
Enviarmensaje Mensa , Numero1	'Sub proceso “ <i>Enviarmensaje</i> ”, al “Numero1”
Limpiarbuffer	'Limpiar buffer
Mensa = ""	'Deja en blanco Mensa
Mensaje = "ALARMA ENCENDIDA ESFOT1"	'Coloca un valor en caracteres en “Mensaje”
Mensa = Mensaje	'Mensa es igual a Mensaje

Enviarmensaje Mensa , Numero2	'Sub proceso "Enviarmensaje", al "Numero2"
Limpiarbuffer	'Limpiar buffer
Mensa = ""	'Deja en blanco Mensa
End Select	'Termina selección
Waitms 100	'Espera 100 mili segundos
Mensaje = ""	'Deja en blanco Mensa
End Sub	'Finaliza sub proceso

- *Encendido y apagado de Led*

Cuando el programa ingresa al sub procedimiento "Toggle_led1", la variable I toma los valores que van de 0 a 2 la misma que se vuelve una condición inicial, luego mediante la instrucción "Toggle" hace que el led se encienda y apague.

Esta instrucción permite cambiar el valor de salida de un pòrtico.

El sub procedimiento espera 25 milisegundos y regresa una y otra vez hasta cumplir la condición dada. Para finalizar este proceso se coloca el valor de 0L en el pòrtico Led1, y retorna al programa. Ver Diagrama de flujo 3.10.

Toggle_led1:	'Se dirige al sub proceso Toggle_led1
For I = 0 To 2	'La variable I va de 0 a 2
Toggle Led1	'Se dirige al sub proceso Toggle_led1
Waitms 25	'Espera 25 mili segundos
Next	'Ejecuta un número de veces "For"
Reset Led1	'Reset el valor de led 1
Return	'Retorna al programa principal

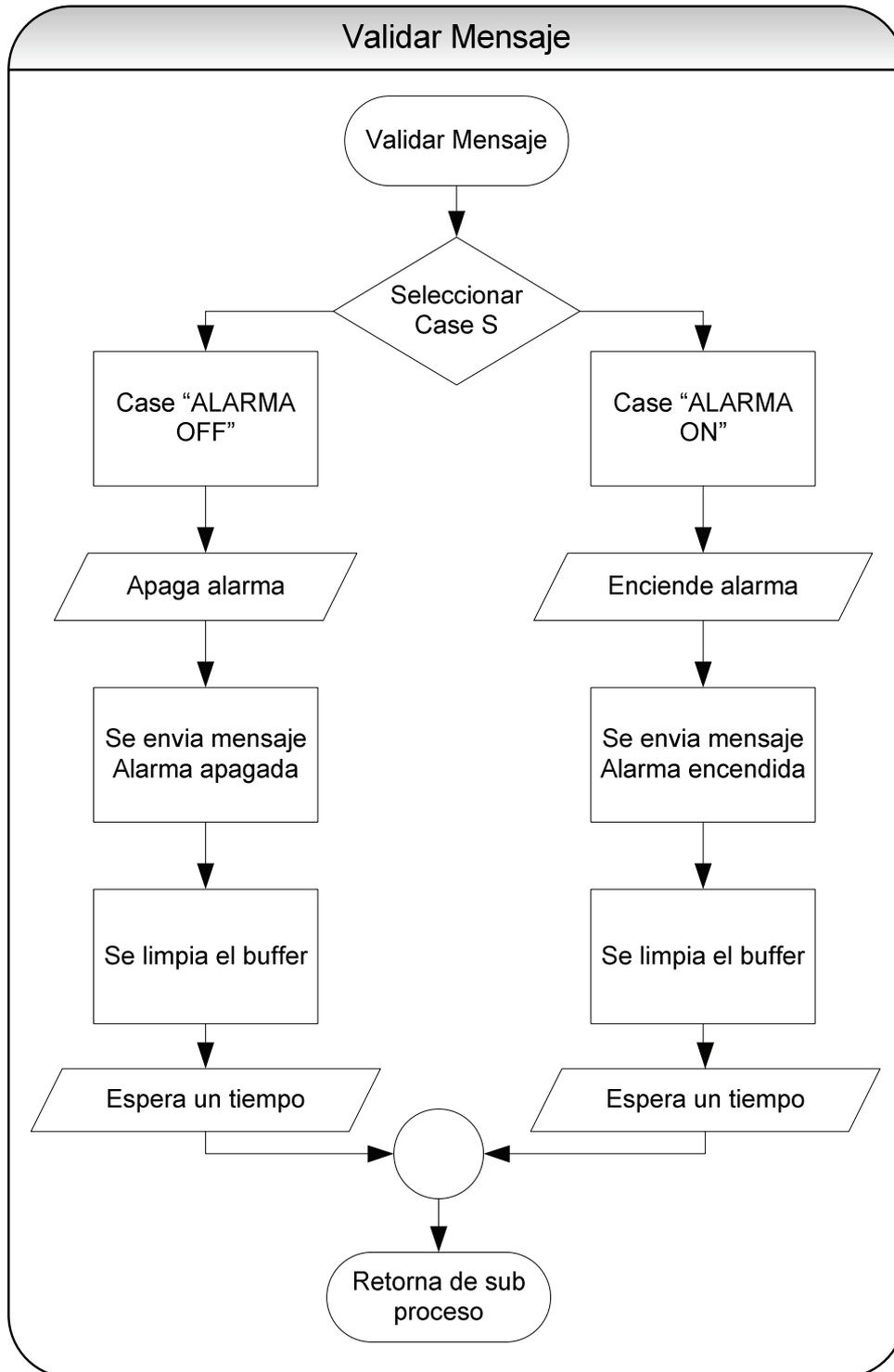


Diagrama de flujo 3.9 Validar mensaje

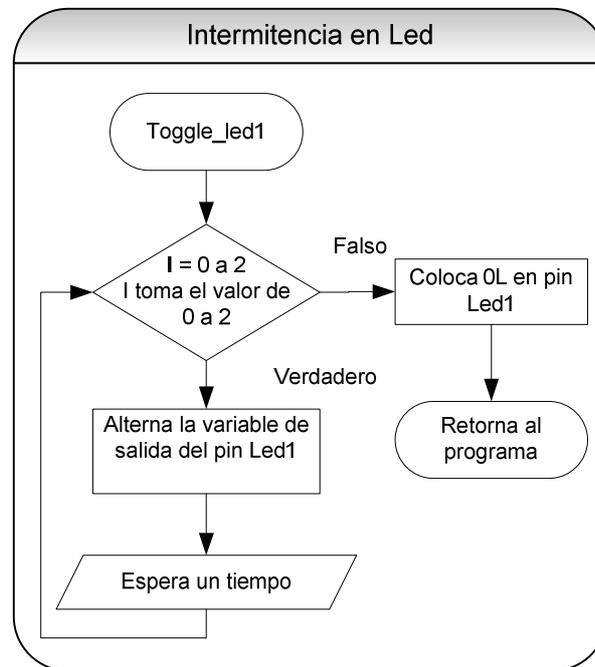


Diagrama de flujo 3.10 Intermittencia de Led

- Sub proceso para revisar datos de radio frecuencia.

Con la instrucción “*While*” ejecuta una variable de instrucciones en un bucle, mientras una instrucción dada es verdad. En este caso se espera que “*Datin*” sea igual a 1, este valor toma “*Vt*”, espera 1 milisegundo y se van dando valores en los pódicos de B0, espera nuevamente 1 milisegundo, e inicia a seleccionar el pódico “*B0*”.

Si se selecciona 1L, se activa la alarma.

En el caso del “Mensaje”, este toma el valor de los caracteres seleccionados, el mismo que se coloca en “Mensa”, y se procede a enviar esto al sub proceso “*Enviarmensaje*”, luego se limpia el buffer, se deja el mensaje en blanco, y se procede a enviar el mensaje de la misma manera a un segundo número.

En otro momento ingresa un nuevo valor en “*Case*”, en este caso es de 2 decimal o 10 binario, cuando ocurre esto la alarma toma el valor de cero, y “*Mensaje*” toma un

nuevo valor en forma de texto, este es igual a “*Mensa*” el mismo que se envía a un número de celular, se limpia el buffer, y se deja en blanco el valor de “*Mensa*”, para luego enviar el mismo valor a un segundo número, para terminar la selección y luego retornar al programa.

- Funcionamiento físico cuando recibe radio frecuencia.

El Decodificador del receptor se comunica mediante radiofrecuencia con el codificador, el mismo que envía los datos.

Para que estos dos dispositivos se comuniquen, ambos tienen una dirección similar, la misma que al ser correcta, coloca un valor de 1L en V_t para que permita el ingreso de los datos.

El sub proceso para revisar la radio frecuencia que viene del transmisor depende de los datos que se envían desde el codificador del emisor, es decir los datos obtenidos por el decodificador se procesan en el microcontrolador y este activa o desactiva las alarmas del sistema receptor, para luego volver al programa principal. Ver Diagrama de flujo 3.11

Revisarrf:

While Datin = 1	'Espera que el dato sea 1 lógico
Datin = Vt	'El dato es igual a Vt
Wend	'Encamina el dato a While
Waitms 1	'Espera 1 mili-segundo
B0 = 0	'Dato en B0 es igual a cero
B0.0 = Rfdato0	'B0.0 es igual a Rfdato0
B0.1 = Rfdato1	'B0.1 es igual a Rfdato1
B0.2 = Rfdato2	'B0.2 es igual a Rfdato2
B0.3 = Rfdato3	'B0.3 es igual a Rfdato3
Waitms 1	'Espera 1 mili-segundo
Select Case B0	'Depende de una variable de selección B

Case 1	'Se da el valor de 01
Alarma = 1	'Alarma se pone en 1 lógico
Mensaje = "ALARMA ON 1"	'Ingreso de mensaje
Mensa = Mensaje	'Mensaje es igual a mensa
Enviarmensaje Mensa , Numero1	'Envía mensaje al número seleccionado
Limpiarbuffer	'Limpia el buffer
Mensa = ""	'Deja en blanco Mensa
Mensaje = "ALARMA ON 1"	'Ingreso de mensaje
Mensa = Mensaje	'Mensaje es igual a mensa
Enviarmensaje Mensa , Numero2	'Envía mensaje al número seleccionado
Limpiarbuffer	'Limpia el buffer
Mensa = ""	'Deja en blanco Mensa
Case 2	'Se da el valor de 10
Alarma = 0	'Alarma se pone en 0 lógico
Mensaje = "ALARMA OFF 1"	'Ingreso de mensaje
Mensa = Mensaje	'Mensaje es igual a mensa
Enviarmensaje Mensa , Numero1	'Envía mensaje al número seleccionado
Limpiarbuffer	'Limpia el buffer
Mensa = ""	'Deja en blanco Mensa
Mensaje = "ALARMA OFF 1"	'Ingreso de mensaje
Mensa = Mensaje	'Mensaje es igual a mensa
Enviarmensaje Mensa , Numero2	'Envía mensaje al número seleccionado
Limpiarbuffer	'Limpia el buffer
Mensa = ""	'Deja en blanco Mensa
End Select	'Finaliza la selección
Return	'Retorna al programa

- Sub proceso serial.

La función *"Disable Urx"*, permite deshabilitar el receptor serial del microcontrolador, mientras se espera la respuesta del celular *"OK"*, el mismo que tiene la función específica para los comandos AT de nokia, llamada *"Noecho"*, esta permite parar los ecos que se envían desde el celular.

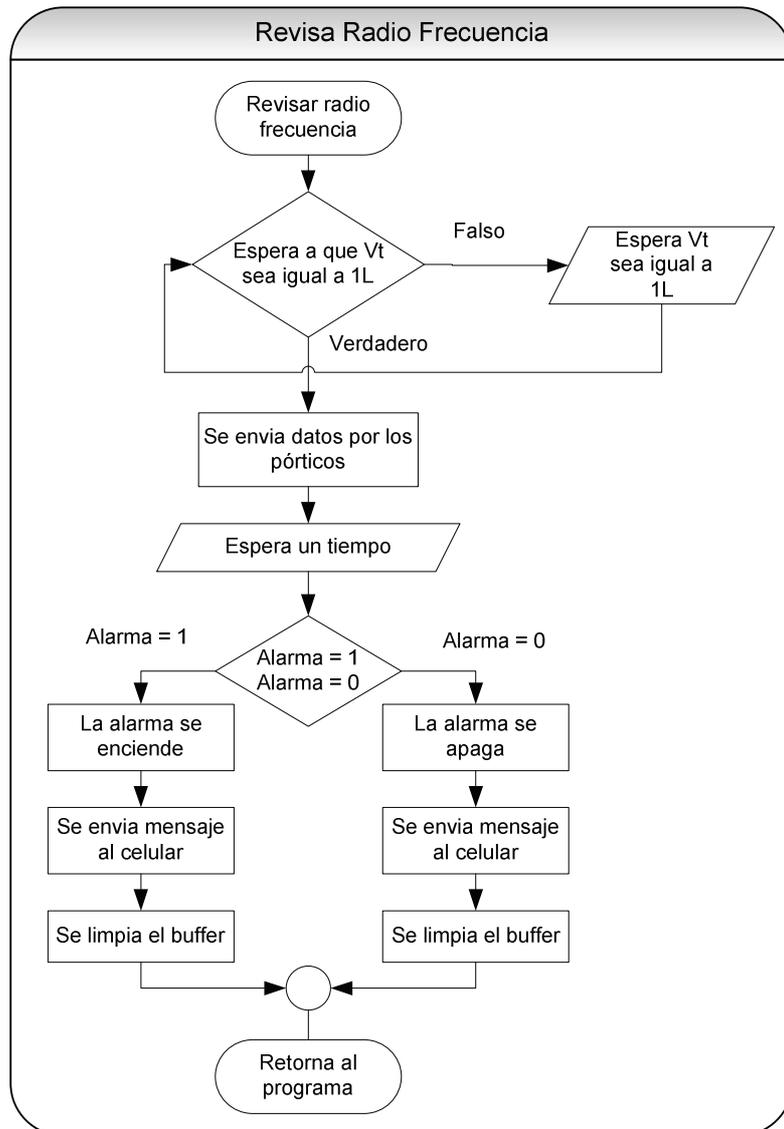


Diagrama de flujo 3.11 Revisar radio frecuencia

Después se selecciona una variable de selección. En “case a”, espera un “OK” del celular y detiene el eco. Luego en “Numero1” toma el valor del “texto2”, esto se repite en las localidades 1 a la 9, la cual se almacena en la variable “Z”, la cual se registrará en la eeprom, hasta completar la condición de la variable Control.

Es decir, ingresan todos los dígitos del celular al que se va a enviar el mensaje establecido.

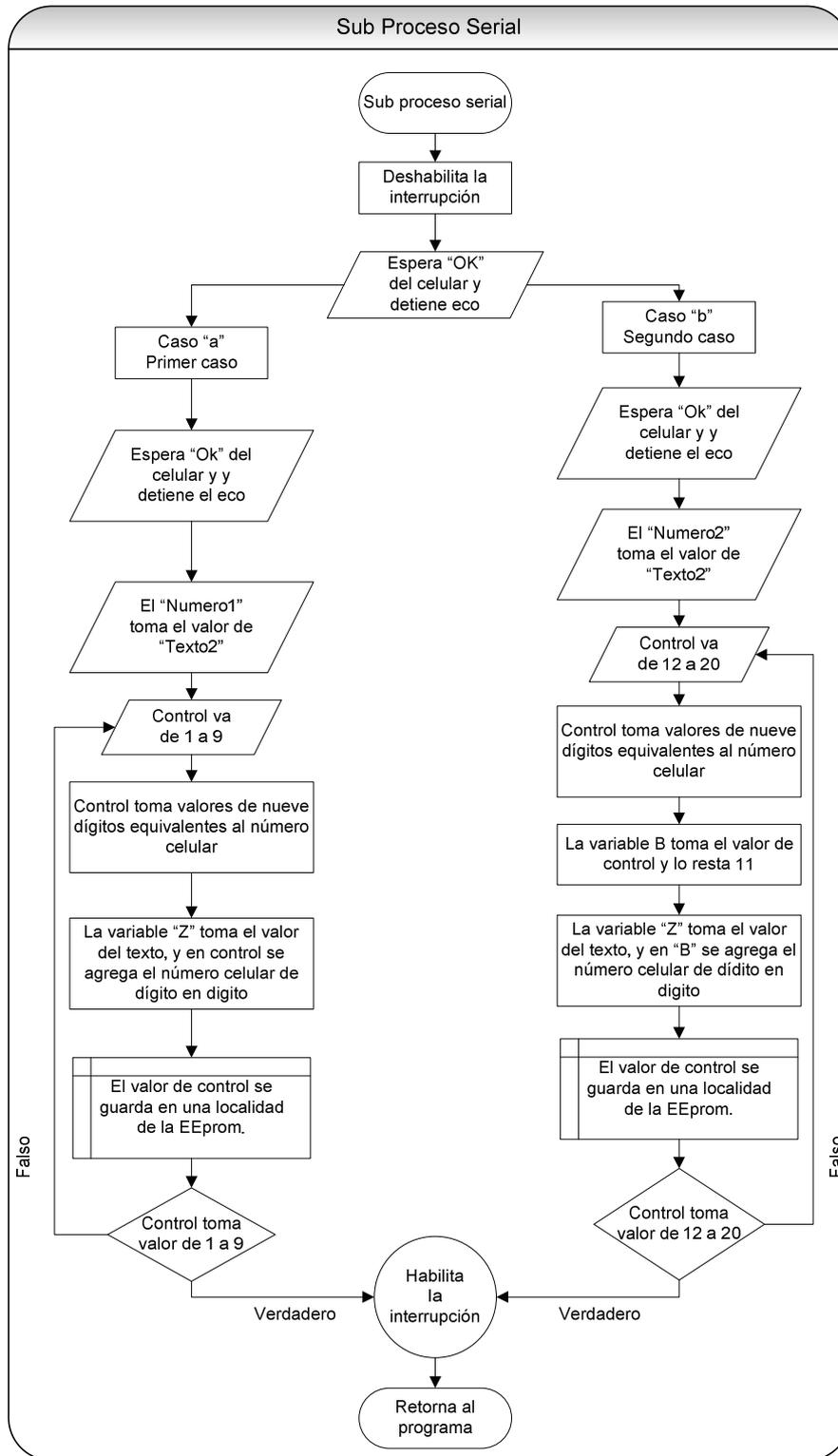


Diagrama de flujo 3.12 Sub proceso serial

Luego se tiene “Case b”, donde se repite el proceso anterior pero desde las localidades de la eeprom 12 a la 20, hasta completar los dígitos del celular, y terminar la selección. Ver Diagrama de flujo 3.12

Rec_isr:	'Sub rutina Serial
Disable Urxc	'Deshabilita la interrupción completa de recepción serial.
Input Texto2 Noecho	'Espera “OK”, y detiene el Eco
Select Case Texto2	'Depende de una variable de selección
Case "a"	'Se da el valor a “a”
Input Texto2 Noecho	'Espera “OK”, y detiene el Eco
Numero1 = Texto2	'El “Numero1” es igual a “Texto2”
For Control = 1 To 9	'La variable Control va de 1 a 9
Z = Mid(texto2 , Control , 1)	'En la variable Z, ingresan los dígitos del celular en control, y cuantos toma en cada lazo.
Writeeeprom Z , Control	'Escribe un contenido en las variables de la eeprom de datos.
Next Control	'Ejecuta un número de veces “For”
Case "b"	'Se da el valor a “b”
Input Texto2 Noecho	'Espera “OK”, y detiene el Eco
Numero2 = Texto2	'El “Numero2” es igual a “Texto2”
For Control = 12 To 20	'La variable Control va de 12 a 20
B = Control - 11	'El valor de la variable “B”, Toma la dirección de “Control” menos 11
Z = Mid(texto2 , B , 1)	'En la variable Z, ingresan los dígitos del celular en control, y cuantos toma en cada lazo.
Writeeeprom Z , Control	'Escribe un contenido en las variables de la eeprom de datos.
Next Control	'Ejecuta un número de veces “For”
End Select	'Finaliza la selección
Enable Urxc	'Habilita la interrupción completa de recepción serial.
Return	'Retorna
End	'Finaliza

La sub función “Dta”, permite ubicar en un lugar de la eeprom los números de celular a donde se enviará los mensajes necesarios desde el microcontrolador a otro dispositivo celular. Ver Diagrama de flujo 3.13

Dta:	'Sub función “Dta”
\$eeprom	'Instruir al compilador que almacene los datos en las líneas de datos de la eeprom
Data 0 , "095348879"	'Devuelve un valor
Data 0 , "083214782"	'Devuelve un valor
\$data	'Instruir al compilador a almacenar los datos en las línea después de \$data

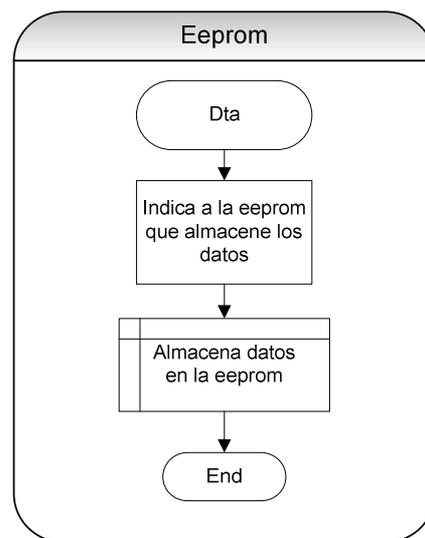


Diagrama de flujo 3.13 Almacenamiento de datos en la eeprom

En el Diagrama de flujo 3.14, se observa el programa completo tanto del circuito transmisor controlado por el microcontrolador ATmega 8, así como el Diagrama de flujo 3.15 del microcontrolador ATmega164P que controla el circuito receptor.

Cada uno de estos Diagramas de flujo, se ha detallado en el transcurso del Capítulo 3, de una manera más ilustrativa, para un mejor entendimiento.

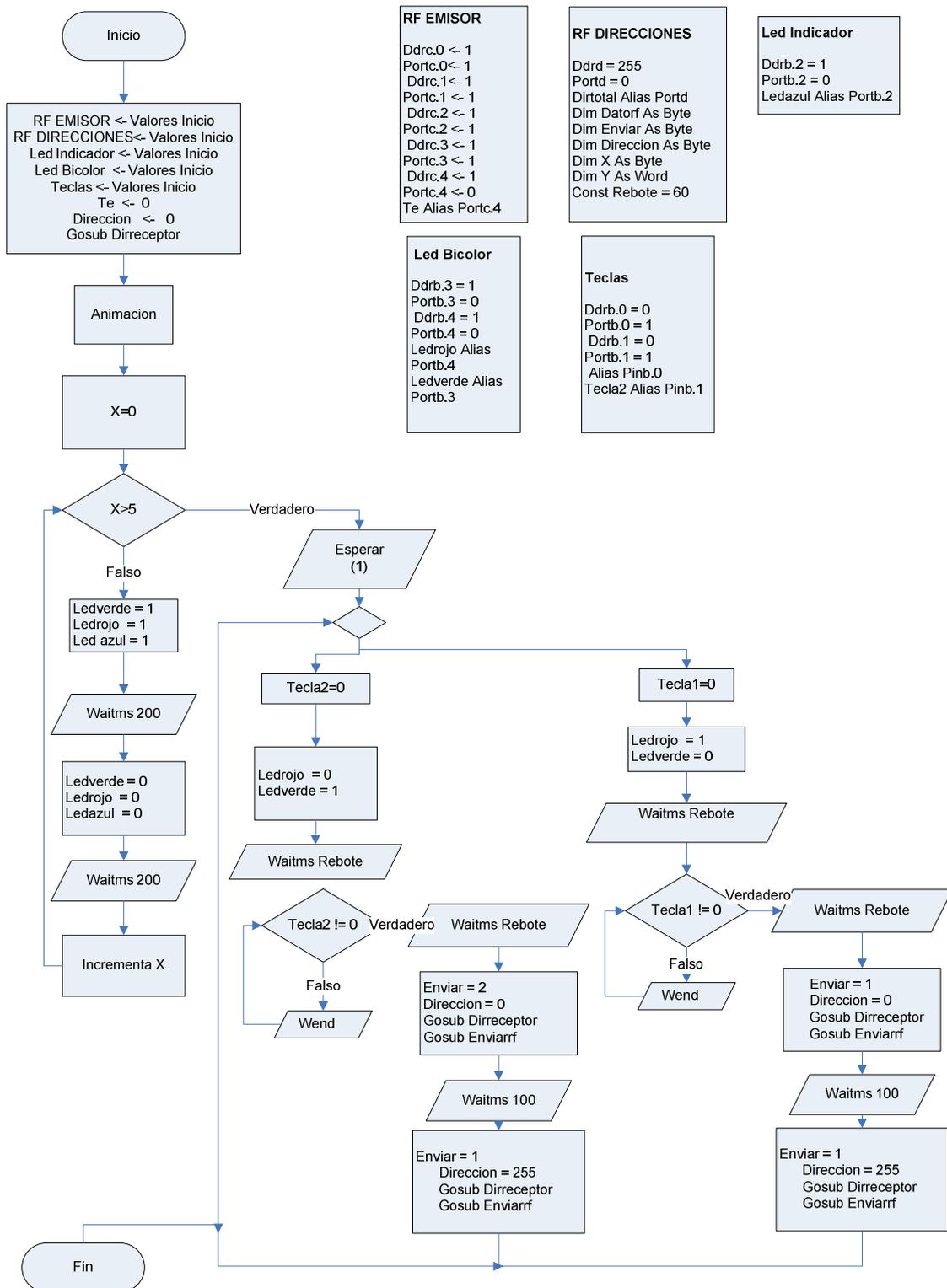


Diagrama de flujo 3.14 Diagrama completo del microcontrolador ATmega8

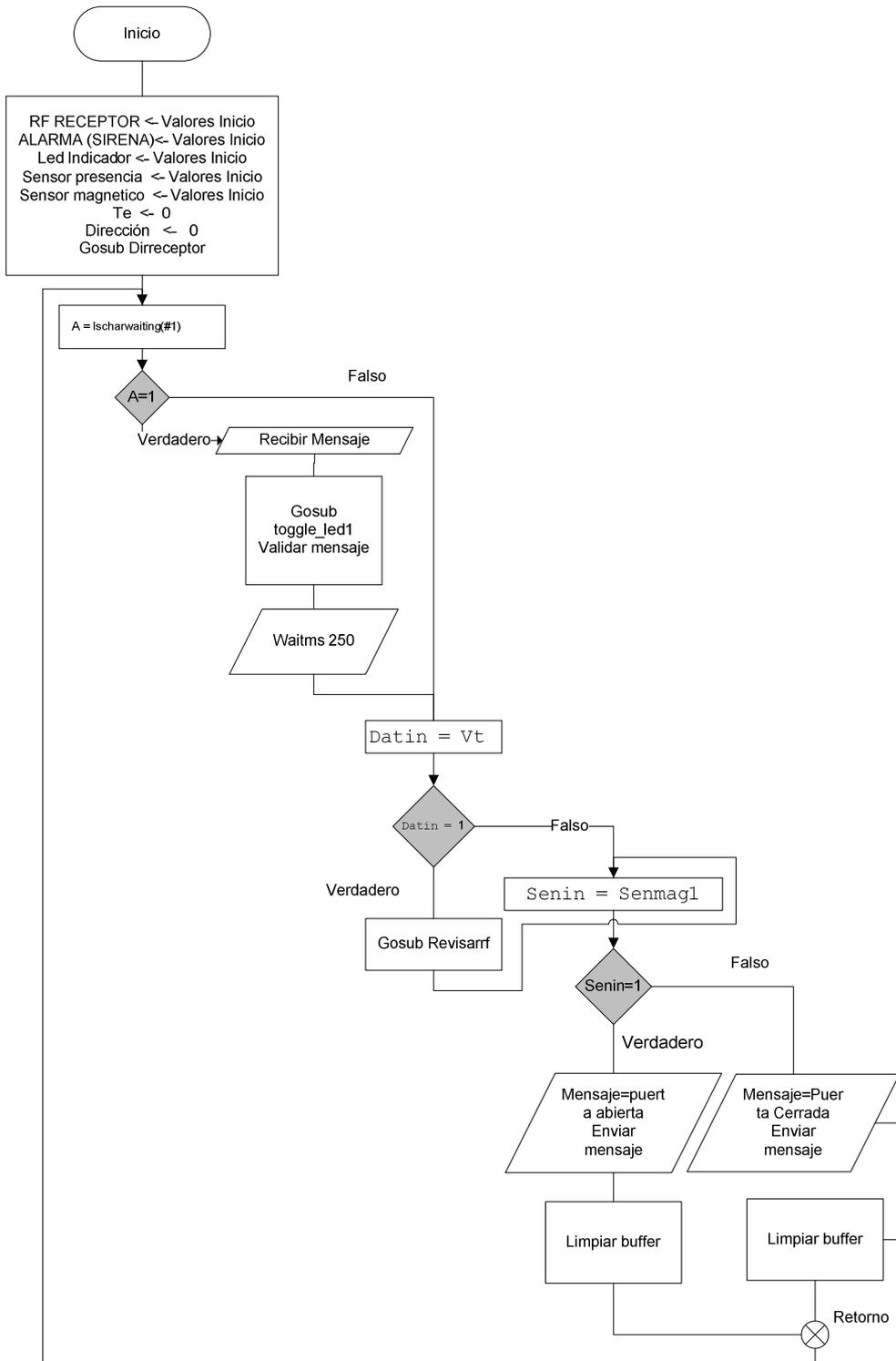


Diagrama de flujo 3.15 Diagrama completo del microcontrolador ATmega164P

3.3. PRUEBAS DE FUNCIONAMIENTO

Entre las pruebas de funcionamiento se debe comprobar la efectividad que existe, para activar la alarma desde una distancia existente entre el receptor y el emisor, comprobando los diferentes inconvenientes que se presenten y el tiempo de duración con las baterías.

3.3.1. PRUEBA DE DISTANCIA

Una de las pruebas más importantes es alcanzar una distancia mínima de 50 metros de radio entre el transmisor y el receptor vista a vista, Para esto se ha colocado el receptor a una distancia de 8 metros de alto (Altura promedio de un poste) y a cada 5 metros de largo se ha ido comprobando su efectividad en un campo abierto.

En la Tabla 3-3 se observa los resultados obtenidos, y en la figura 3.1 como se toman las medidas de la Tabla 3-3.

Resultados:

Al realizar la prueba de distancia uno de los inconvenientes para aumentar la distancia entre el transmisor y el receptor es la altura que debe tomar el equipo que recibe la señal de radiofrecuencia.

Este problema se presenta ya que al aumentar la distancia a nivel del piso se divisan mayor cantidad de obstáculos que atenúan la señal de radio frecuencia que viaja por el medio ambiente hasta el objetivo.

Cuando se realiza la prueba a 8 metros de altura se observa que puede alcanzar una distancia de 73 metros tomando en cuenta la distancia de vista a vista desde el transmisor al receptor, para obtener esta distancia se ha calculado la hipotenusa de un triángulo rectángulo, para obtener la distancia real que logra alcanzar el sistema.

Al pasar esta distancia la señal se atenúa, y tiende a debilitarse, no consiguiendo activar el sistema receptor.

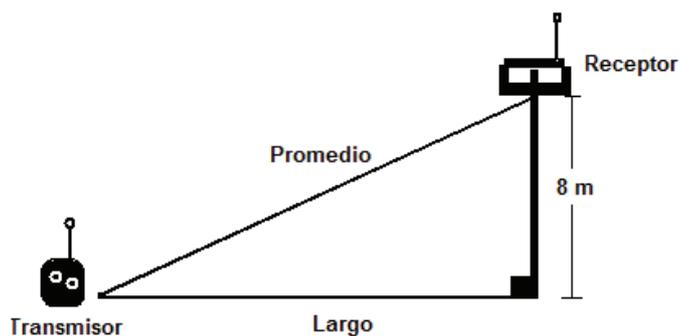


Figura 3.1 Distancia receptor, transmisor

Emisor [Volt]	Receptor[Volt]	Alto (metros)	Largo (metros)	Promedio (metros)	Si	No	A veces
9 (Voltios)	5 (Voltios)	8	1	9	X		
9 (Voltios)	5 (Voltios)	8	5	13	X		
9 (Voltios)	5 (Voltios)	8	10	18	X		
9 (Voltios)	5 (Voltios)	8	15	23	X		
9 (Voltios)	5 (Voltios)	8	20	28	X		
9 (Voltios)	5 (Voltios)	8	25	33	X		
9 (Voltios)	5 (Voltios)	8	30	38	X		
9 (Voltios)	5 (Voltios)	8	35	43	X		
9 (Voltios)	5 (Voltios)	8	40	48	X		
9 (Voltios)	5 (Voltios)	8	45	53	X		
9 (Voltios)	5 (Voltios)	8	50	58	X		
9 (Voltios)	5 (Voltios)	8	55	63	X		
9 (Voltios)	5 (Voltios)	8	60	68	X		
9 (Voltios)	5 (Voltios)	8	65	73	X		
9 (Voltios)	5 (Voltios)	8	70	78		X	
9 (Voltios)	5 (Voltios)	8	75	83		X	
9 (Voltios)	5 (Voltios)	8	80	88		X	
9 (Voltios)	5 (Voltios)	8	85	93		X	
9 (Voltios)	5 (Voltios)	8	90	98		X	
9 (Voltios)	5 (Voltios)	8	95	103		X	
9 (Voltios)	5 (Voltios)	8	100	108		X	

Tabla 3.3 – Proceso de encendido de la alarma por distancia

3.3.2. PRUEBA DE SONIDO

El sistema funciona con una batería externa, como interna, la batería externa es de 110 voltios, regulada por una fuente que alcanza un voltaje máximo de 15 voltios, el

cual alimenta tanto a la batería interna como al circuito, por lo que, se necesita un tiempo estimado de funcionamiento tanto de la alimentación de voltaje externa como de la batería en forma independiente, la cual se indica tanto en la Tabla 3-4 y 3-5.

El sistema de alarma, dispone de una bocina sonora la misma que funciona a 12 voltios. Al usar una fuente externa se observa que su funcionamiento es normal, y puede durar un lapso de tiempo mayor a 120 minutos.

Voltaje In (Volt)	Batería (Volt)	Bocina (Volt)	Tiempo (Minutos)	Sonido Alto	Sonido Medio	Sonido Bajo
15	14.0	12	10	X		
15	14.0	12	20	X		
15	14.0	12	30	X		
15	14.0	12	40	X		
15	14.0	12	50	X		
15	14.0	12	60	X		
15	14.0	12	70	X		
15	14.0	12	80	X		
15	14.0	12	90	X		
15	14.0	12	100	X		
15	14.0	12	110	X		
15	14.0	12	120	X		

Tabla 3-4 Efectividad del sonido de la bocina con fuente externa

Voltaje In (Volt)	Batería (Volt)	Bocina (Volt)	Tiempo (Minutos)	Sonido Alto	Sonido Medio	Sonido Bajo
0	14.0	12	10	X		
0	14.0	12	20	X		
0	13.5	12	30	X		
0	13.3	12	40	X		
0	13.0	12	50	X		
0	12.8	12	60	X		
0	12.7	12	70	X		
0	12.3	12	80	X		
0	12.3	12	90	X		
0	11.4	12	100		X	
0	11.2	12	110		X	
0	11.0	12	120		X	

Tabla 3-5 Efectividad del sonido de la bocina solo con batería interna

Resultados:

Al usar una fuente interna de 12 voltios, la bocina inicia con un sonido fuerte pero al transcurso del tiempo la batería empieza a descargarse, y mientras disminuye el voltaje, la bocina empieza a disminuir el sonido, para aumentar el tiempo de duración del sonido de la bocina se puede colocar baterías internas en paralelo, para que la descarga sea más lenta y el sonido se conserve durante un mayor tiempo.

3.3.3. PRUEBA DE OPERADORA CELULAR

Todo sistema tiene un tiempo de proceso interno, y en este caso un tiempo en el cual envía los mensajes vía celular, se ha realizado pruebas con la operadora Movistar y Porta, teniendo la siguiente respuesta mostrada en la Tabla 3-6 y Tabla 3-7.

Horario	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
8AM	5	5	5	5	5	5	5
10AM	6	6	6	6	6	5	5
12PM	6	5	5	6	6	6	5
02PM	5	5	5	5	5	6	5
04PM	6	6	5	6	6	6	5
06PM	6	6	6	6	6	6	5
08PM	6	6	6	5	6	7	5
10PM	5	5	5	6	6	6	5
12AM	5	5	5	5	5	6	5

Tabla 3-6 Tiempos de respuesta de operadora celular Movistar [segundos]

Horario	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
8AM	5	5	5	5	5	5	5
10AM	6	6	6	6	5	6	5
12PM	5	5	5	5	5	5	5
02PM	5	5	5	5	5	5	5
04PM	6	6	5	6	5	5	5
06PM	6	6	5	6	6	6	5
08PM	6	6	5	6	6	5	5
10PM	5	5	5	5	5	5	5
12AM	5	5	5	5	5	5	5

Tabla 3-7 Tiempos de respuesta de operadora celular Porta [segundos]

Observaciones:

Las pruebas realizadas es enviando un mensaje a cada hora indicada en la tabla 3-6 y Tabla 3-7, el tiempo que se demora en enviar el mensaje y tener la respuesta es de 5 segundos, los mismos que se ejecutan de la siguiente manera: 2 segundo en el proceso interno del microcontrolador y de 3 a 4 segundos en el envío del mensaje, este tiempo no varía considerablemente.

El tiempo total de envío del mensaje es de 6 segundos.

3.1. RESULTADOS

Los resultados del proyecto son alentadores, ya que el sistema tiene un alcance de 50 metros requerido.

Al activar la sirena de la alarma, la fuente entrega mayor energía pero la duración de la misma es mayor a una hora activada, y apagada la sirena dura una semana en reposo, es decir funcionando solo el circuito interno del receptor.

Las pruebas de funcionamiento con la operadora Movistar, ha llevado a concluir que su servicio de mensajería tiene una respuesta óptima, en todo el transcurso del día a igual que la operadora Porta.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES.

- Los módulos de radiofrecuencia, TLP434A y RLP434A necesitan ser direccionados mediante un codificador y un decodificador, para lograr controlar su comunicación de forma segura, ya que, un dispositivo de las mismas características podría accionar el sistema de forma accidental.
- El microcontrolador, ya sea este el ATmega164P o ATmega8 necesitan estar conectados a un cristal, para controlar la velocidad de procesamiento y la comunicación del microcontrolador con otros dispositivos como es el computador o celular.
- El sistema de alarma debe tener una altura mayor a 6 metros, para disponer de mayor visibilidad, de esta forma el sistema logra tener mayor alcance de funcionamiento mientras se acciona el pulsador inalámbrico denominado transmisor. El sistema disminuye su alcance óptimo, al tener mayor cantidad de obstáculos ya que estos atenúan la señal por lo que es necesario que cada dispositivo, tanto el transmisor como el receptor dispongan de una antena visible para mejorar el alcance.
- Al usar el sistema de alarma receptora, el sonido es óptimo mientras usa la energía externa, pero al usar una batería interna su sonido va disminuyendo mientras dicha batería se descarga.
- Cuando el celular accidentalmente recibe una llamada de cualquier otro celular o teléfono fijo tiende a desactivar las funciones del microcontrolador, por lo

que es necesario resetear el circuito receptor, para esto se debe realizar una prueba de funcionamiento semanal, para confirmar que el circuito está funcionando en condiciones óptimas.

- Cuando el teléfono celular no dispone de saldo para el envío de mensajes de texto, este entra en un lazo infinito hasta ser reseteado y disponer de mensajes de la operadora contratada.
- La operadora celular “Movistar”, con la que se trabajó en el proyecto, se observó que no tiene inconvenientes cuando se envía un mensaje de texto, vale destacar que estas pruebas fueron realizadas en días ordinarios o fines de semana pero no en días festivos donde se tiene aglomeración de gente enviando mensajes o realizando llamadas.
- Las baterías del sistema de alarmas deben estar cargadas ya que de estas dependen también el alcance obtenido entre los dispositivos de transmisión y recepción del sistema.

4.2.RECOMENDACIONES

- El transmisor del sistema de alarma comunitaria necesita estar vista a vista con el sistema receptor para tener una mejor comunicación y mejorar la distancia usada entre estos dispositivos.
- Es recomendable llevar un mantenimiento adecuado de las baterías, ya que al descargarse disminuye el alcance de comunicación entre ellas.
- Las antenas del transmisor y receptor deben ser desplegadas en su máxima longitud para lograr un alcance de comunicación mayor.

- La contratación de mensajería corta debe ser de vital importancia ya que de esto depende el funcionamiento del sistema de alarma.
- En la circuitería de los dispositivos de transmisión y recepción se debe tener los elementos necesarios para su instalación ya que las pistas son cercanas y al someter a temperaturas altas se corre el riesgo de dañar las pistas del circuito.
- Los dispositivos que usan baterías de corriente continua deben tener un sistema de reconocimiento de fases o polaridades para evitar un mal funcionamiento o un daño grave, para esto es recomendable el uso de colores estándar o etiquetas.
- Para aumentar el tiempo de duración de las baterías se debe colocar una batería del mismo valor de forma paralela, tanto para el circuito receptor como para el celular.

Bibliografía

N°	Titulo Referencia tomada
[1]	<i>Modulación Digital</i> Tomassi, Wayne” <i>Sistemas de Comunicación Electrónica</i> ”, Comunicaciones Digitales, pág. 468, 4ª ed., 2003
[2]	<i>Modulación Digital</i> http://www.udistrital.edu.co/comunidad/profesores/hjeslavab/Downloads/Modulacion%20Digital.pdf
[3]	<i>Modulador ASK</i> http://webdiee.cem.itesm.mx/web/servicios/archivo/trabajos/comunicaciones/ask/introduccion.html
[4]	<i>Modulación ASK</i> http://www.textoscientificos.com/redes/modulacion/ask
[5]	<i>FSK Desplazamiento de frecuencia</i> Tomassi, Wayne” <i>Sistemas de Comunicación Electrónica</i> ”, Modulación por desplazamiento de frecuencia, Pág. 471, 4ª ed., 2003,
[6]	<i>Modulación FSK</i> http://www.textoscientificos.com/redes/modulacion/FSK
[7]	<i>FSK Desplazamiento de frecuencia</i> http://www.textoscientificos.com/redes/modulacion/FSK
[8]	<i>PSK Desplazamiento de fase</i> Tomassi, Wayne” <i>Sistemas de Comunicación Electrónica</i> ”, Modulación por desplazamiento de fase, Pág. 478, 4ª ed., 2003
[9]	<i>Modulación PSK</i> http://www.textoscientificos.com/redes/modulacion/psk
[10]	<i>Módulos RF</i> http://redalyc.uaemex.mx/redalyc/html/304/30400403/30400403.html
[11]	<i>Espectro radioeléctrico</i> Tomassi, Wayne” <i>Sistemas de Comunicación Electrónica</i> ”, Frecuencias de transmisión, pág. 5,6 4ª ed., 2003
[12]	<i>Usos de UHF</i> http://www.topbits.com/uhf.html
[13]	Zonas de fresnel http://asterion.almadark.com/2008/11/30/las-zonas-fresnel-y-el-alcance-de-los-equipos-de-radio-frecuencia/
[14]	<i>TLP434(A)–RF ASK Módulos Híbridos para Radio Control</i> http://micros.mforos.com/1149902/6829041-introduccion-a-la-radio-frecuencia-con-modulos-de-laipac/?marcar=por+cierto
[15]	<i>Transmisor TLP 434A Receptor RLP 434A</i> http://www.forosdeelectronica.com/f17/manejo-modulos-tlp434-rlp434-441/
[16]	<i>TLP434</i> http://www.alldatasheet.com/datasheet-pdf/pdf/169604/ETC/TLP434A.html
[17]	<i>RLP 434A</i> http://www.alldatasheet.com/datasheet-pdf/pdf/169605/ETC/RLP434A.html

[18]	<i>Modos de Transmisión</i> Tomassi, Wayne” <i>Sistemas de Comunicación Electrónica</i> ”, Modos de transmisión, pág.10, 4ª ed., 2003
[19]	<i>Apuntes de Propagación y Antenas</i>
[20]	<i>Sistemas de Codificación y Decodificación.</i> http://www.textoscientificos.com/informacion/teoria
[21]	<i>Decodificadores</i> Ronald Tocci, “ <i>Sistemas Digitales</i> ”, Codificadores, pág.517, 1993
[22]	<i>Codificadores</i> Ronald Tocci, “ <i>Sistemas Digitales</i> ”, Decodificadores, pág. 504, 1993
[23]	<i>HT12E</i> http://pdf1.alldatasheet.com/datasheet-pdf/view/64409/HOLTEK/HT12E.html
[24]	<i>HT12D</i> http://pdf1.alldatasheet.com/datasheet-pdf/view/64409/HOLTEK/HT12D.html
[25]	<i>Microcontroladores</i> http://www.slideshare.net/jewelarde/07-familia-atmega
[26]	<i>ATmega164P</i> http://rapidshare.com/files/203310262/atmega164p_espa_ol.pdf
[27]	<i>ATmega8</i> http://www.datasheetcatalog.org/datasheet/atmel/2486S.pdf
[28]	<i>Comandos AT</i> http://bluehack.elhacker.net/proyectos/comandosat/comandosat.html
[29]	<i>Comandos AT GSM</i> http://blackinside.wordpress.com/2007/03/28/comandos-at-gsm/