

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

“Utilización de módulos GPS en combinación con módulos de radio frecuencia para la transmisión de datos de posicionamiento hacia control central empleando microcontroladores”

TESINA DE SEMINARIO

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

PRESENTADO POR:

Valeria Carolina Aguirre Salas

Dennys Magdalena Zambrano Rosado

GUAYAQUIL – ECUADOR

2010

AGRADECIMIENTO

Afloran sentimientos de gratitud a mi familia por su acompañamiento en la consecución de mis logros, al Ing. Carlos Valdivieso por su valiosa orientación profesional y desde luego al Creador que hace posible la cristalización de nuestras metas.

Valeria Aguirre Salas

DEDICATORIA

Con afecto, a todas las personas con ideas innovadoras, que sea este trabajo un estímulo para lograr mejores desempeños gracias a las competencias tecnológicas.

Valeria Aguirre Salas

AGRADECIMIENTO

Ante todo a Dios por permitirme culminar esta etapa estudiantil, a mis padres por estar siempre a mi lado.

Al Ing. Carlos Valdivieso por su acertada gestión y apoyo.

Dennys Zambrano Rosado

DEDICATORIA

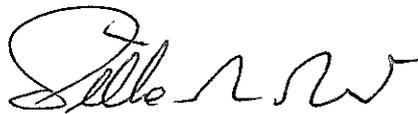
A mis padres, a mis hermanos y a mis amigos que han sido siempre mi pilar y mi fuente de fuerzas para seguir adelante con mis objetivos.

Dennys Zambrano Rosado

TRIBUNAL DE SUSTENTACIÓN



Ing. Carlos Enrique Valdivieso A.
PROFESOR DEL SEMINARIO DE GRADUACIÓN



Ing. Hugo Villavicencio V.
DELEGADO DEL DECANO

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de esta Tesina, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL".

(Reglamento de exámenes y títulos profesionales de la ESPOL).



Valeria Carolina Aguirre Salas



Dennys Magdalena Zambrano Rosado

RESUMEN

En la actualidad se tiene a la disposición una gran cantidad de recursos tecnológicos cubriendo necesidades prioritarias, una de ellas es la orientación, para lo cual se utiliza el Sistema de Posicionamiento Global GPS. El campo de aplicación de dicha tecnología es bastante extenso, va desde sistemas de posicionamiento para control de aplicaciones robóticas y móviles, para ubicación de vehículos, para fines deportivos, etc.

Los datos de posicionamiento que se van a adquirir en el presente trabajo serán combinados con otras tecnologías como los microcontroladores y la transmisión por radiofrecuencia lo cual va a utilizarse según le convenga al usuario final.

ABREVIATURAS

GPS Global Positioning System (Sistema de Posicionamiento Global)

I2C (Inter Integrated Circuit) Circuito Inter Integrado

NMEA National Marine Electronics Association (Asociación Nacional de Electrónica Marina)

DC Direct Current (Corriente directa)

LCD Graphic Liquid Crystal Display (Pantalla de cristal líquido)

RAM Random-access Memory (Memoria de Acceso Aleatorio)

RCIF (Receive Interrupt Flag bit) Bit de Bandera de Recepción de Interrupción

RS232 Recomend Standard 232 (Estándar Recomendado 232)

SPI (Serial Peripheral Interface) Interfaz Periférica Serial

USART Universal Synchronous Asynchronous Receiver Transmitter (Receptor Transmisor Síncrono Asíncrono Universal)

INDICE GENERAL

INTRODUCCIÓN

1	Descripción General del Sistema	1
1.1	Alcance y Limitaciones del Proyecto.....	1
1.1.1	Descripción del Proyecto	2
1.1.2	Estrategia Implementada	2
1.1.3	Limitaciones del Proyecto	3
1.2	Análisis de Soluciones existentes en el mercado	4
2	Fundamentación Teórica.....	5
2.1	Sistema de Posicionamiento Global (GPS).....	5
2.2	El protocolo NMEA.....	6
2.3	Transmisión por Radiofrecuencia.....	9
2.4	Comunicación serial.....	9
2.5	Conceptos Básicos de Programación	10
2.6	Herramientas de software	11
2.6.1	IDE PICC	11
2.6.2	Proteus.....	12
2.6.3	Eagle 5.10.....	12
2.6.4	HyperTerminal	12
2.7	Virtual Serial Port Driver.....	13
3	Diseño del proyecto.....	14
3.1	Diseño general.....	14
3.2	Diseño de la aplicación	15
3.2.1	Especificaciones funcionales	15
3.2.1.1	Diseño de los componentes de software	15
3.2.1.1.1	Código PIC 16F887A	16
3.2.1.1.2	Código PIC 18F452	27
3.2.1.2	Diseño de los componentes de hardware	40
3.2.1.2.1	Principales características del PIC16F877A	40

3.2.1.2.2	El Microcontrolador PIC18F452.....	40
3.2.1.2.3	Módulo receptor GPS 18LVC.....	41
3.2.1.2.4	Módulos HOPE RF	42
3.2.1.2.5	Pantalla LCD 16 x 2	43
3.2.1.2.6	MAX232CPE.....	45
4	Simulación y Pruebas Experimentales	15
4.1	Recepción de Datos del módulo GPS.....	156
4.2	Simulaciones Realizadas	47
4.2.1	Proyecto en Proteus.....	47
CONCLUSIONES		
RECOMENDACIONES		
ANEXOS		
BIBLIOGRAFÍA		

INDICE DE FIGURAS

Figura 2.1 Comunicación serial: Simplex.....	9
Figura 2.2 Comunicación serial: Half Duplex.....	9
Figura 2.3 Comunicación serial: Full Duplex.....	10
Figura 3.1 Diagrama de bloques del proyecto	15
Figura 3.2 El microcontrolador 16F877A	40
Figura 3.3 El microcontrolador 18F452.....	41
Figura 3.4 Receptor GPS 18LVC.....	42
Figura 3.5 Modulo HOPE RF	43
Figura 3.6 Configuración del MAX232	45
Figura 4.1 Formato GGPGA requerido	15
Figura 4.2 Prueba GPS 18LVC.....	47
Figura 4.3 Simulación con Proteus	48
Figura 4.4 Simulación de acceso a permiso de conexión con GPS.....	48
Figura 4.5 Simulación en enlace con módulo GPS.....	49
Figura 4.6 Simulación envío de datos a estación central.....	49

INTRODUCCION

El proyecto a presentar consiste en una combinación efectiva de algunas de las tecnologías que actualmente benefician al ser humano y al desarrollo de la sociedad, tecnologías como el sistema de posicionamiento global, transmisión mediante radiofrecuencia y microcontroladores. Gracias al receptor GPS, en combinación con módulos RF y dos microcontroladores es posible determinar los datos de posicionamiento de un objeto para luego, poder manipularlos a conveniencia del usuario, y maniobrar con las coordenadas GPS desde una estación de control lejana al lugar donde se hizo la toma de dichos datos según unos de los objetivos iniciales del trabajo.

La estrategia implementada, alcances y limitaciones se detallan en el Capítulo 1 junto a algunas soluciones ya existentes en el mercado destinadas para el mismo propósito.

En el Capítulo 2, el objeto de estudio será la fundamentación teórica de cada uno de los dispositivos y elementos utilizados.

El diseño general, la implementación física, la programación de los dispositivos, se tratará en el Capítulo 3.

Las simulaciones realizadas, implementación, datos obtenidos se muestran en el Capítulo 4.

Para finalizar, se tratan las conclusiones y recomendaciones.

CAPÍTULO 1

1 Descripción General del Sistema

En el presente trabajo, se estudia la posibilidad de combinar varias tecnologías modernas que el ser humano actualmente a su disposición, desarrollando un dispositivo que esté en la capacidad de transmitir por medio de radiofrecuencia datos de posicionamiento obtenidos de un módulo de GPS hacia una estación que emplea microcontroladores.

Para mejor comprensión del sistema a desarrollar, es necesario conocer que va a ser estructurado en dos etapas, una de transmisión, donde el módulo GPS recibe datos de posicionamiento para enviarlos a través de radiofrecuencia por el transmisor del módulo HOPE RF, y la etapa de recepción donde el receptor del modulo HOPE RF lo recibe y se envía dicha información a un microcontrolador que transmitirá vía serial los datos de posicionamiento hacia una estación de control.

1.1 Alcance y Limitaciones del Proyecto

Las ambiciones que se tiene con respecto al desarrollo e implementación de este proyecto son bastante extensas, a continuación

se analizará el alcance del mismo y las limitantes de hardware y software que se presenten.

1.1.1 Descripción del Proyecto

La finalidad del proyecto en estudio es extraer información de posicionamiento del módulo GPS en un determinado sitio y transmitirla por medio de radiofrecuencia hacia la estación de control a fin de que ésta maneje los datos según su conveniencia

1.1.2 Estrategia Implementada

El módulo que se ha utilizado para la implementación de este proyecto es el receptor GPS 18LVC, de GARMIN, el cual cuenta con un diseño de prácticas dimensiones y con elementos que facilitan la comunicación con otros dispositivos. Dicho módulo GPS luego de recibir la información satelital va a enviarla a un PIC18F452, microcontrolador que está en la capacidad de establecer comunicación con cualquier dispositivo que soporte comunicación serial SPI o I2C, que adecuará los datos recibidos para transmitirla al HOPE RF, módulo transmisor de radiofrecuencia, para que éste se enlace con su par receptor y le envíe la información de posicionamiento que luego pasará a un PIC16F877A que es el puente de transmisión serial hacia la estación central.

1.1.3 Limitaciones del Proyecto

El proyecto está sujeto a las limitaciones que aporta cada uno de los elementos utilizados como el GPS, el modulo HOPE RF, y entre otras limitaciones que provee la madre naturaleza.

Por parte del GPS, la mayor limitación a combatir en el desarrollo de este proyecto va a ser la capacidad de recepción de los datos de posicionamiento debido a la cantidad de acceso visual que la antena receptora tenga a su disposición, ya que las señales que envían los satélites se comportan como la luz, de tal modo tenemos las mismas restricciones que una señal óptica que no puede pasar a través de montañas, edificios, túneles y algunas superficies metálicas, por lo cual se requiere conexión visual entre el receptor y los satélites.

Por otra parte también tenemos las limitantes que se presentan al querer transmitir por radiofrecuencia como reducción de señal, capacidad de transmisión, alcance, rango de frecuencia y también por la presencia de otros transmisores de radiofrecuencia que producen interferencia.

1.2 Análisis de Soluciones existentes en el mercado

El campo de aplicación del Sistema de Posicionamiento Global es en la actualidad es muy extenso ya que va desde el uso militar hasta un software en un dispositivo de uso cotidiano como el celular. Así mismo la variedad de dispositivos existentes en el mercado permite hacer una evaluación de lo requerido, llegando a establecer prioridades de costo, funcionalidad y lo más importante, cuan práctico es, según el campo de aplicación. En vista del uso que se le va a dar al presente proyecto, se estima que lo adecuado es utilizar módulos simples, prácticos y portátiles, por lo cual se prefiere el receptor GPS 18LVC de Garmin, sobre la placa multifuncional Smart GPS de mikroElektronika.

Aunque el uso de tecnología GPS ha crecido descomunalmente se puede decir que no hay en el mercado ninguna aplicación existente que realice todas las operaciones y adquisiciones de datos como las planteadas en el presente proyecto, lo cual justifica la propuesta de desarrollo del mismo, impulsando a nuevas tecnologías y al aporte del factor humano para la sociedad.

CAPÍTULO 2

2 Fundamentación Teórica

Este capítulo procura abarcar en detalle la fundamentación teórica de cada uno de los temas que constituyen un pilar fundamental en el desarrollo del proyecto en estudio.

2.1 Sistema de Posicionamiento Global (GPS)

GPS es un sistema de posicionamiento global que permite determinar en todo el mundo la posición de un objeto. Se conoce que fue mentalizado por los franceses y belgas, pero fueron los estadounidenses quienes lo desarrollaron y lo implementaron, hoy por hoy es operado por el Departamento de Defensa de los Estados Unidos. Actualmente se cuenta con 27 satélites artificiales, 24 de ellos operativos y 3 de respaldo, con trayectorias sincronizadas para cubrir todo el globo.

Para que un receptor pueda determinar su ubicación requiere de la información de al menos 3 satélites de la red, de los que recibe señales de la ubicación y del reloj de cada uno de ellos. Con base en estas

señales, el aparato sincroniza el reloj del GPS y calcula el retraso de las señales (es decir, la distancia al satélite). Por "triangulación" calcula la posición en que éste se encuentra. En el caso del GPS, la triangulación se basa en determinar la distancia de cada satélite respecto al punto de medición. Conocidas las distancias, se determina fácilmente la propia posición relativa respecto a los tres satélites. Conociendo además las coordenadas o posición de cada uno de ellos por la señal que emiten, se obtiene la posición absoluta o las coordenadas reales del punto de medición. También se consigue una exactitud extrema en el reloj del GPS, similar a la de los relojes atómicos que llevan a bordo cada uno de los satélites.

2.2 El protocolo NMEA

NMEA 0183 (o NMEA de forma abreviada) es una especificación combinada eléctrica y de datos entre aparatos electrónicos marinos y, también, en su mayoría, entre receptores GPS.

El protocolo NMEA 0183 es un medio a través del cual los instrumentos marítimos y también la mayoría de los receptores GPS pueden comunicarse los unos con los otros. Ha sido definido, y está controlado, por la organización estadounidense National Marine Electronics Association.

El protocolo NMEA se basa en cadenas. Cada cadena se inicia con el signo \$ (código ASCII 36) y termina con una secuencia de signos que comienza con una nueva línea, tales como CR (código ASCII 13) y LF (código ASCII 10). El significado de toda la cadena depende de la primera palabra. Por ejemplo, una cadena que comienza con \$GPGLL da información acerca de la latitud y la longitud, la hora exacta (Tiempo Universal Coordinado), la validez de los datos (A - Activo o V - Prohibido) y la suma de verificación ("checksum") que permite comprobar si los datos se recibieron correctamente. Los datos individuales de cada elemento están separados por una coma ",".

Cada segundo se envía un conjunto de cadenas NMEA al microcontrolador. En el caso de que los datos sobre latitud y longitud no sean fijos (por ejemplo, si un receptor GPS falla a la hora de determinar su posición) o cuando los datos no sean determinados, el receptor GPS mantendrá en su salida el mismo juego de cadenas, dejando de lado cualquier dato perdido. El dispositivo utilizado, LEA-5S, muestra las siguientes cadenas de códigos:

\$GPRMC Recommended minimum specific GPS/Transit data

\$GPVTG Track made good and ground speed

\$GPGGA Global Positioning System Fix Data

\$GPGSA GPS DOP and active satellites

\$GPGSV GPS DOP and active satellites

\$GPGLL Geographic position, latitude / longitude and time

A continuación se muestra un ejemplo de una cadena NMEA completa:

\$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47

123519	Lectura tomada a las 12:35:19 UTC
4807.038,N	Latitud 48° 07.038' N
01131.000,E	Longitud 11° 31.000' E
1	Fix quality: 0 - 8
08	Número de satélites en sincronía
0.9	Horizontal dilution of position
545.4,M	Altitud sobre el nivel del mar
46.9,M	Altura
*47	Checksum data

2.3 Transmisión por Radiofrecuencia

La radiocomunicación posibilita la transmisión de señales mediante la modulación de ondas electromagnéticas, las cuales pueden propagarse tanto a través del aire como del espacio vacío.

2.4 Comunicación serial

La comunicación serial utilizando el estándar RS232 consiste en la transmisión de una serie de bits, uno detrás del otro, existen tres diferentes formas de transmisión en serie.

Simplex: Un equipo transmite, el otro recibe, para visualizarlo de manera gráfica, véase figura 2.1.

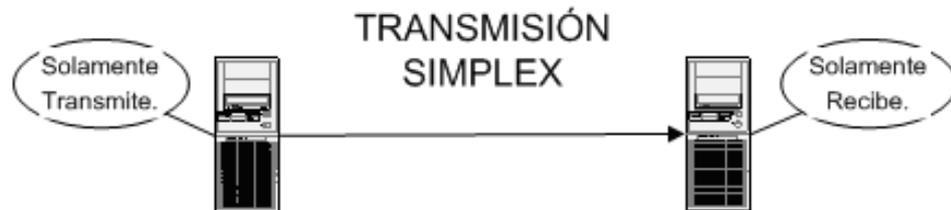


Figura 2.1 Comunicación serial: Simplex

Half-duplex: La transmisión se da por parte de ambos equipos, de manera no simultánea; los equipos se alternan en la transmisión, uno transmite mientras el otro recibe. Véase figura 2.2

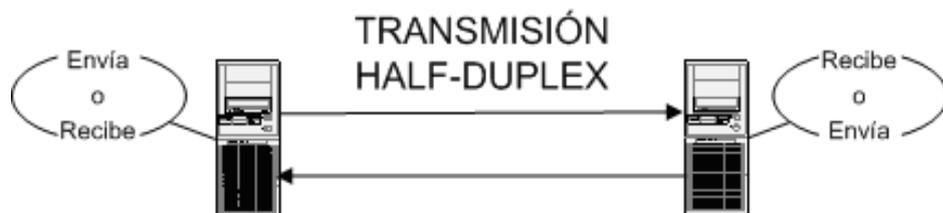


Figura 2.2 Comunicación serial: Half Duplex

Full-duplex: La transmisión por parte de ambos equipos es simultánea. Posee las líneas de transmisión y recepción independientes de tal forma que la línea de transmisión de un equipo se conecte a la entrada de recepción del otro y viceversa. Los puertos serie del PC son capaces de utilizar este modo. Véase figura 2.3.



Figura 2.3 Comunicación serial: Full Duplex

2.5 Conceptos Básicos de Programación

A continuación se describirá brevemente los conceptos de programación necesarios en el desarrollo del presente trabajo.

Una variable global, es accesible en todos los ámbitos de un programa. Los mecanismos de entorno global interactúan con dichas variables. El entorno global se contrapone al entorno local ya que en éste último todas las variables son locales y sin memoria compartida.

Las variable locales trabajan en un ámbito local, ya que solamente se puede acceder a ellas desde la función o bloque de instrucciones en donde se declaran. En la mayoría de lenguajes de programación, éstas son variables automáticas almacenadas directamente en la pila de

llamadas lo cual significa que cuando una función recursiva se llama a sí misma, las variables locales reciben espacio para el direccionamiento de memoria separado, en cada instancia de la función, de tal manera, éstas se pueden declarar, reescribir y leer sin riesgos para otros procesos fuera del bloque en el que son declarados.

Las funciones, subprogramas o subrutinas se presentan como un sub-algoritmo, parte del algoritmo principal, con la única finalidad de resolver una tarea específica y devolver el valor especificado. Las funciones son muy similares a los procedimientos, la diferencia entre ambos es que los últimos mencionados no devuelven un valor, se utilizan para procesar información sobre las variables de ámbito global.

2.6 Herramientas de software

A continuación se detalla todas las herramientas de software que se utilizaron en el desarrollo del tema propuesto y que ayudaron con la programación del hardware y la simulación del proyecto.

2.6.1 IDE PICC

Para el desarrollo de la programación que será parte del controlador del proyecto se utiliza el lenguaje de programación C, el cual es orientado a la implementación de sistemas operativos, al hablar de éste, se habla de un lenguaje de medio nivel que tiene acceso a las

estructuras típicas de alto nivel. EL compilador de lenguaje C utilizado es PICC de la casa HITECH.

PICC trabaja básicamente con lenguaje de alto nivel, tiene una interfaz muy amigable para el usuario y ciertamente responde de buena forma al momento que el programador realiza su tareas.

2.6.2 Proteus

Se utiliza para la simulación de los circuitos necesarios en la implementación física del proyecto, se usa proteus ya que es una herramienta de software con muchas capacidades y también porque el modo de operarlo es de gran facilidad, claro conociendo los fundamentos básicos de circuitos electrónicos.

2.6.3 Eagle 5.10

Este software es una gran herramienta que se utiliza para el diseño de esquemáticos y de diagrama de pistas, en el desarrollo de este proyecto se recurrió a dicho software debido a su extensa librería de elementos y sus respectivos empaquetamientos, de tal manera se facilitó el bosquejo de algunos componentes cuyos pines están distribuidos dependiendo de la casa electrónica que los elabore.

2.6.4 HyperTerminal

Es un programa que se utiliza para conectar con otros equipos, sitios Telnet, servicios en línea y equipos host, mediante un módem, un

cable de módem nulo o una conexión con protocolo Internet. Permite además visualizar los datos en formato de texto que recibe un computador por su puerto serial.

2.7 Virtual Serial Port Driver

Ya que actualmente no todos los computadores disponen de un puerto RS232 hay una manera de utilizar comunicación serial, mediante la aplicación que permite controlar puertos virtuales se puede configurar uno de esos puertos USB como puerto RS232.

CAPÍTULO 3

3 Diseño del proyecto

El presente capítulo ha sido destinado a describir las etapas necesarias para la elaboración del proyecto, las cuales son de diseño, Implementación y programación.

3.1 Diseño general

El propósito del proyecto consiste en la utilización de módulos GPS en combinación con módulos de radio frecuencia para la transmisión de datos de posicionamiento hacia control central empleando microcontroladores.

Debido a la complejidad del trabajo propuesto se decidió trabajar por etapas, en la primera solamente se trabaja con el GPS, en ésta se analiza el modo con el cual se puede extraer los datos de posicionamiento lo puede ser visualizado gracias al PIC 18F452 con una programación que permite visualizar en una pantalla LCD las tramas que nos provee el GPS, y en el caso de no poseer una pantalla, con la ayuda de alguna herramienta o interfaz diseñada para el uso de

receptores GPS se podría también observar directamente en la computadora los datos que se están recibiendo, una de las opciones para dichos fines es el software Qblox. La segunda etapa consiste en el envío y transmisión de datos, y está conformada por un PIC de la familia 16F887A, y dos pares de módulos RF cuyo tipo de modulación es FSK. La tercera fase de trabajo corresponde a la comunicación del PIC 18F452 con la computadora o estación de control. Véase figura 3.1

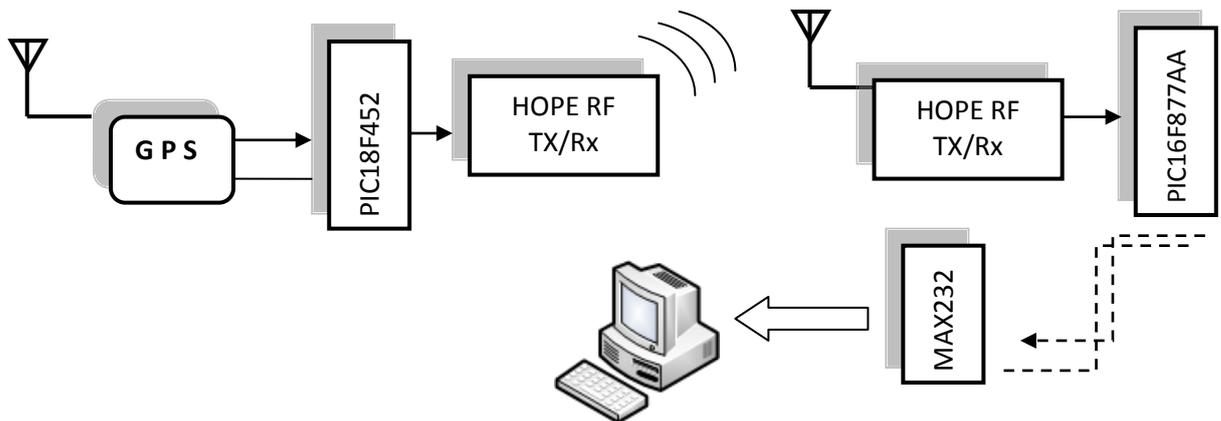


Figura 3.1 Diagrama de bloques del proyecto

3.2 Diseño de la aplicación

3.2.1 Especificaciones funcionales

Esta sección está destinada al análisis del software que controla el funcionamiento del presente proyecto.

3.2.1.1 Diseño de los componentes de software

3.2.1.1.1 Código PIC 16F887A

Archivo .c

En las siguientes líneas de código se hace un llamado a las librerías que van a ser usadas en la programación.

```
#include "C:\Documents and Settings\uc\Escritorio\PROYECTO  
GRADO\COMPU_GPS\GPS.h"
```

```
#include <ctype.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
#include <stdlibm.h>
```

```
#include "input.c"
```

A continuación se declaran variables a utilizar en el desarrollo del proyecto.

```
char bufferGPS[80];
```

```
int1 flg_mensaje=0;
```

```
char opcion=0;
```

```
int8 len,m,x,y;
```

```
char *DatoGPS,*cadenaLON,*cadenaLAT;
```

En el fragmento de código siguiente se configuran las características del PIC como puertos, entradas, salidas, se habilitan y deshabilitan opciones según las necesidades del código.

```
void main()
```

```
{
```

```
    setup_adc_ports(NO_ANALOGS);
```

```
    setup_adc(ADC_OFF);
```

```
    setup_psp(PSP_DISABLED);
```

```
    setup_spi(SPI_SS_DISABLED);
```

```
    setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
```

```
    setup_timer_1(T1_DISABLED);
```

```
    setup_timer_2(T2_DISABLED,0,1);
```

```
    setup_comparator(NC_NC_NC_NC);
```

```
    setup_vref(FALSE);
```

```
    set_tris_b(0X0f);
```

```
set_tris_a(0B00000010);
```

Lo siguiente indica la asignación de los datos de posicionamiento a la variable char DatoGPS, además la configuración de ciertos pines como salida.

```
DatoGPS=malloc(90);
```

```
cadenaLON = malloc(12);
```

```
cadenaLAT = malloc(12);
```

```
while(TRUE){
```

```
    inicio:
```

```
    flg_mensaje=0;
```

```
    output_bit (pin_C4, 0);
```

```
    output_bit (pin_D2, 0);
```

```
    output_bit (pin_D3, 0);
```

```
    fprintf(COMPU,"\fBienvenido al Sistema de Posicionamiento  
Global\n\r");
```

```
    fprintf(COMPU,"          MENU\n\r");
```

```
    fprintf(COMPU,"¿Desea iniciar la recepcion de datos de  
posicionamiento?\n\r");
```

```

fprintf(COMPU,"                Presione S/N    \n\r");

output_bit (pin_C4, 1);

opcion=fgetc(COMPU);

fprintf(COMPU,"                %c    \n\r",opcion);

```

Las líneas escritas a continuación permiten que el usuario decida acceder o no al enlace con el GPS.

```

if (opcion=='S' || opcion=='s'){

    InitBufferGPS();

    output_bit (pin_D2, 1);

    delay_ms(100);

    fprintf(COMPU,"\f  COMENZANDO ENLACE CON EL GPS...");

    delay_ms(300);

    fprintf(GPS,"ENVIANDO1\n\r");

    fgets(bufferGPS,GPS);

    len=strlen(bufferGPS);

    output_bit(pin_D3,1);

    delay_ms(200);

```

```
if(len>=15){  
  
    fprintf(COMPU,"%f      POSICIONAMIENTO GPS");  
  
    for(m=0;m<len;m++){  
  
        DatoGPS[m]=bufferGPS[m];  
  
    }  
  
    DatoGPS[m++]='\0';  
  
    output_bit(pin_D3,1);  
  
    delay_ms(500);  
  
    output_bit(pin_D3,0);  
  
    delay_ms(500);  
  
    output_bit(pin_D3,1);  
  
    delay_ms(500);  
  
    output_bit(pin_D3,0);  
  
    fprintf(COMPU,"\r\n%S",DatoGPS);  
  
    x=0;  
  
    for(m=0;x!=2;m++){
```

```
if(DatoGPS[m]==',')  
  
    x++;  
  
}  
  
cadenaLAT=InitBuffer1(12,cadenaLAT);  
  
x=0;  
  
for(y=0;x!=2;y++){  
  
    cadenaLAT[y]=DatoGPS[m];  
  
    m++;  
  
    if(DatoGPS[m]==','){  
  
        x++;  
  
    }  
  
}  
  
cadenaLON=InitBuffer1(12,cadenaLON);  
  
x=0;  
  
m++;  
  
for(y=0;x!=2;y++){
```

```
cadenaLON[y]=DatoGPS[m];

m++;

if(DatoGPS[m]==','){

    x++;

}

}

fprintf(COMPU,"\r\n COORDENADAS GPS \n\r");

fprintf(COMPU,"\r\nLATITUD: %S\n\r",cadenaLAT);

fprintf(COMPU,"LONGITUD: %S",cadenaLON);

delay_ms(5000);

fprintf(COMPU,"\r\nPresione R para retornar: ");

sigue:

opcion=fgetc(COMPU);

fprintf(COMPU,"                %c          ",opcion);

if (opcion=='R' || opcion=='r'){

    goto inicio;
```

```
}else{  
  
    goto sigue;  
  
    delay_ms(500);  
  
}  
  
}else{  
  
    output_bit (pin_C4, 1);  
  
    output_bit (pin_D2, 1);  
  
    output_bit (pin_D3, 1);  
  
    delay_ms(500);  
  
    output_bit (pin_C4, 0);  
  
    output_bit (pin_D2, 0);  
  
    output_bit (pin_D3, 0);  
  
    delay_ms(500);  
  
    output_bit (pin_C4, 1);  
  
    output_bit (pin_D2, 1);  
  
    output_bit (pin_D3, 1);
```

```
delay_ms(500);

output_bit (pin_C4, 0);

output_bit (pin_D2, 0);

output_bit (pin_D3, 0);

fprintf(COMPU, "\f ERROR EN RECEPCION DE TRAMA");

fprintf(COMPU, "\r\nPresione R para retornar: ");

sigue1:

opcion=fgetc(COMPU);

fprintf(COMPU, "                %c                ",opcion);

if (opcion=='R' || opcion=='r'){

    goto inicio;

}else{

    goto sigue1;

    delay_ms(500);

}

}
```

```
    }  
  
    delay_ms(500);  
  
    }  
  
}  
  
void InitBufferGPS(void){  
  
    int8 l;  
  
    for(l=0;l<=80;l++){  
  
        bufferGPS[l]='\0';  
  
    }  
  
char* InitBuffer1(int8 length, char *array){  
  
    int8 y;  
  
    for(y=0;y<=length;y++){  
  
        array[y]='\0';  
  
    }  
  
    return array;  
  
}
```

A continuación se realiza la programación para comunicación serial a una tasa de 9600 baudios

```
#include <16F877A.h>
```

```
#device adc=8
```

```
#FUSES NOWDT //No Watch Dog Timer
```

```
#FUSES HS //High speed Osc (> 4mhz)
```

```
#FUSES PUT //Power Up Timer
```

```
#FUSES NOPROTECT //Code not protected from reading
```

```
#FUSES NODEBUG //No Debug mode for ICD
```

```
#FUSES NOBROWNOUT //No brownout reset
```

```
#FUSES NOLVP //No low voltage prgming B3 used for I/O
```

```
#FUSES NOCPD //No EE protection
```

```
#FUSES NOWRT //Program memory not write protected
```

```
#use delay(clock=20000000)
```

```

#use
rs232(baud=9600,parity=N,xmit=PIN_C6,rcv=PIN_C7,bits=8,stream=G
PS,restart_wdt)

#use
rs232(baud=9600,parity=N,xmit=PIN_B1,rcv=PIN_B0,bits=8,stream=CO
MPU,restart_wdt)

void InitBufferGPS(void);

char* InitBuffer1(int8 length, char *array);

```

3.2.1.1.2 Código PIC 18F452

Archivo .c

En las siguientes líneas de código se hace un llamado a las librerías que van a ser usadas en la programación.

```

#include "C:\Documents and Settings\uc\Escritorio\PROYECTO
GRADO\GPS_COMPU\main.h"

#include "C:\Documents and Settings\uc\Escritorio\PROYECTO
GRADO\GPS_COMPU\lcd.c"

#include <ctype.h>

```

```
#include <stdio.h>

#include <stdlib.h>

#include <stdlibm.h>

#include <string.h>

#include "input.c"
```

A continuación se declaran variables a utilizar en el desarrollo del proyecto.

```
char bufferCOMPU[15];

char bufferGPS[90];

char *cadenaGPS,*cadenaLON,*cadenaLAT;

int8 len=0,m=0,x=0,y=0;
```

En esta primera parte del bloque principal se procede al la configuración de algunos parámetros del PIC, como puertos, timers, comparadores, etc.

```
void main()

{

    setup_adc_ports(NO_ANALOGS);
```

```
setup_adc(ADC_OFF);

setup_psp(PSP_DISABLED);

setup_spi(SPI_SS_DISABLED);

setup_wdt(WDT_OFF);

setup_timer_0(RTCC_INTERNAL);

setup_timer_1(T1_DISABLED);

setup_timer_2(T2_DISABLED,0,1);

setup_timer_3(T3_DISABLED|T3_DIV_BY_1);

//Setup_Oscillator parameter not selected from Intr Oscillotar Config tab

set_tris_d(0X00);

set_tris_c(0B10000000);

set_tris_b(0B10000000);

lcd_init();

cadenaGPS=malloc(90);

cadenaLON=malloc(12);

cadenaLAT=malloc(12);
```

```
output_bit (pin_D2, 0);

output_bit (pin_D3,0);

inicio:

lcd_putc('\f');

lcd_putc(" TRANSMISOR ");

lcd_putc('\n');

lcd_putc(" POSICION RF ");

delay_ms(1000);

//Setup_Oscillator parameter not selected from Intr Oscillator Config tab

while(true){

    fgets(bufferCOMPU,RF);

    lcd_putc('\f');

    lcd_putc(" PROCESANDO ");

    lcd_putc('\n');

    lcd_putc(" INFORMACION ");

    delay_ms(1000);
```

```
output_bit (pin_D2, 0);

output_bit (pin_D3,0);

len=strlen(bufferCOMPU);

if(len>=9){

    InitBufferCOMPU();

    output_bit (pin_D2, 1);

    delay_ms(500);

    output_bit (pin_D3,0);

    fgets(bufferGPS,GPS);

    len=strlen(bufferGPS);

    if(len>=15){

        output_bit (pin_D2, 0);

        output_bit (pin_D3, 1);

        delay_ms(500);

        cadenaGPS=InitBuffer1(80,cadenaGPS);

        for(m=0;m<len;m++){
```

```
    cadenaGPS[m]=bufferGPS[m];  
  
}  
  
fprintf(RF,"%S\n\r",cadenaGPS);  
  
x=0;  
  
for(m=0;x!=2;m++){  
  
    if(cadenaGPS[m]==',')  
  
        x++;  
  
}  
  
cadenaLAT=InitBuffer1(12,cadenaLAT);  
  
x=0;  
  
for(y=0;x!=2;y++){  
  
    cadenaLAT[y]=cadenaGPS[m];  
  
    m++;  
  
    if(cadenaGPS[m]==','){  
  
        x++;  
  
    }  
  
}
```

```
}  
  
cadenaLON=InitBuffer1(12,cadenaLON);  
  
x=0;  
  
m++;  
  
for(y=0;x!=2;y++){  
  
    cadenaLON[y]=cadenaGPS[m];  
  
    m++;  
  
    if(cadenaGPS[m]==''){  
  
        x++;  
  
    }  
  
}  
  
lcd_putc('\f');  
  
printf(lcd_putc,"LAT: %S",cadenaLAT);  
  
lcd_putc('\n');  
  
printf(lcd_putc,"LON:%S",cadenaLON);  
  
delay_ms(5000);
```

```
output_bit (pin_D2, 0);

output_bit (pin_D3, 0);

goto inicio;

}else{

fprintf(RF,"ERROR\n\r");

lcd_putc('\f');

printf(lcd_putc,"  ERROR EN  ");

lcd_putc('\n');

printf(lcd_putc," CONEXION GPS ");

delay_ms(2000);

output_bit (pin_D2, 0);

output_bit (pin_D3,0);

delay_ms(500);

output_bit (pin_D2, 1);

output_bit (pin_D3,1);

delay_ms(500);
```

```
output_bit (pin_D2, 0);

output_bit (pin_D3,0);

delay_ms(500);

output_bit (pin_D2, 1);

output_bit (pin_D3,1);

delay_ms(500);

output_bit (pin_D2, 0);

output_bit (pin_D3,0);

goto inicio;

}

}else{

fprintf(RF,"ERROR\n\r");

lcd_putc('\f');

printf(lcd_putc," ERROR EN ");

lcd_putc('\n');

printf(lcd_putc," CONEXION PC ");
```

```
delay_ms(2000);

output_bit (pin_D2, 0);

output_bit (pin_D3,0);

delay_ms(500);

output_bit (pin_D2, 1);

output_bit (pin_D3,1);

delay_ms(500);

output_bit (pin_D2, 0);

output_bit (pin_D3,0);

delay_ms(500);

output_bit (pin_D2, 1);

output_bit (pin_D3,1);

delay_ms(500);

output_bit (pin_D2, 0);

output_bit (pin_D3,0);

goto inicio;
```

```
    }  
  }  
}  
  
void InitBufferCOMPU(void){  
  
    int8 l;  
  
    for(l=0;l<=15;l++)  
  
        bufferCOMPU[l]='\0';  
  
}  
  
void InitBufferGPS(void){  
  
    int8 l;  
  
    for(l=0;l<=90;l++)  
  
        bufferGPS[l]='\0';  
  
}  
  
char* InitBuffer1(int16 length, char *array){  
  
    int16 y;  
  
    for(y=0;y<=length;y++)
```

```
    array[y]='\0';  
  
    return array;  
  
}
```

A continuación se realiza la programación para comunicación serial a una tasa de 9600 baudios

```
#include <18F452.h>
```

```
#device adc=8
```

```
#FUSES NOWDT    //No Watch Dog Timer
```

```
#FUSES WDT128  //Watch Dog Timer uses 1:128 Postscale
```

```
#FUSES HS      //Resistor/Capacitor Osc
```

```
#FUSES NOPROTECT //Code not protected from reading
```

```
#FUSES NOOSCSN //Oscillator switching is disabled, main  
oscillator is source
```

```
#FUSES NOBROWNOUT //Reset when brownout detected
```

```
#FUSES BORV20 //Brownout reset at 2.0V
```

```
#FUSES PUT     //Power Up Timer
```

```
#FUSES NOSTVREN //Stack full/underflow will not cause reset

#FUSES NODEBUG //No Debug mode for ICD

#FUSES NOLVP //No low voltage prgming, B3(PIC16) or B5(PIC18)
used for I/O

#FUSES NOWRT //Program memory not write protected

#FUSES NOWRTD //Data EEPROM not write protected

#FUSES NOWRTB //Boot block not write protected

#FUSES NOWRTC //configuration not registers write protected

#FUSES NOCPD //No EE protection

#FUSES NOCPB //No Boot Block code protection

#FUSES NOEBTR //Memory not protected from table reads

#FUSES NOEBTRB //Boot block not protected from table reads

#use delay(clock=20000000)

#use
rs232(baud=9600,parity=N,xmit=PIN_C6,rcv=PIN_C7,bits=8,stream=RF
)

#use rs232(baud=9600,parity=N,rcv=PIN_B0,bits=8,stream=GPS)
```

```
void InitBufferGPS(void);
```

```
char* InitBuffer1(int16 length, char *array);
```

```
void InitBufferCOMPU(void);
```

3.2.1.2 Diseño de los componentes de hardware

En los siguientes puntos se hará mención de las características y beneficios que otorgan los elementos principales en el desarrollo del proyecto.

3.2.1.2.1 Principales características del PIC16F877A

Las principales características de esta familia de microcontroladores son el control de dirección individual de los pines entrada salida, interrupción por cambio de nivel, resistencias Pull Ups programables individualmente, conversor ADC, tres timers, comunicación serial. Véase figura 3.2

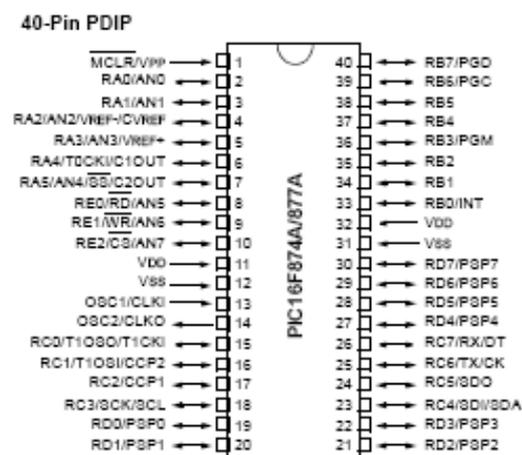


Figura 3.2 El microcontrolador 16F877A

3.2.1.2.2 El Microcontrolador PIC18F452

Este microcontrolador tiene 8 bits del tipo flash, sus memorias de programa y RAM son de 32kB y 1536 Bytes respectivamente, característica importantes en el momento de selección, de manera que es apropiado para los fines en que se va a usar. Cuenta con 40 pines, de los cuales, 35 están destinados a entrada y salida configurables. Véase figura 3.3

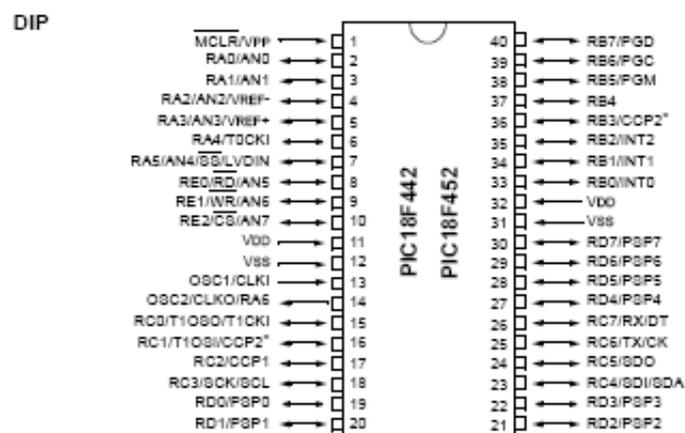


Figura 3.3 El microcontrolador 18F452

Cuenta además con un puerto EUSART mejorado que provee comunicación serial compatible con RS232 y RS485, un puerto maestro de comunicación serial que maneja formatos I2C y SPI lo cual valida la posibilidad de entablar comunicación con otros dispositivos compatibles.

Cuenta además con pines que facilitan el proceso de grabación del programa fuente por medio de programadores como los PIC KIT.

3.2.1.2.3 Módulo receptor GPS 18LVC

Este producto entabla comunicación de tipo RS 232 con un puerto serial, los niveles de voltaje son totalmente compatibles con los de RS232, cuenta con protección contra polarización inversa. Adicionalmente cuenta con un cable con cinco terminales destinados dos a polarización, y los tres restantes para recepción, transmisión y pulso de salida. Véase figura 3.4.



Figura 3.4 Receptor GPS 18LVC

Hay ciertas características que deben conocerse como el voltaje de entrada que está ubicado en un rango de 4.0 – 5.5 V, corriente de entrada máxima de 60mA. Este receptor tiene sensibilidad mínima de -165 dBW, trabaja en temperaturas mayores a -30°C y menores a -80°C respondiendo en tiempos entre 2 a 45 minutos dependiendo de condiciones climatológicas. Su margen de error es de aproximadamente 15 metros.

3.2.1.2.4 Módulos HOPE RF

Para cumplir uno de los requerimientos del proyecto fue necesario usar dos módulos de transmisión RF, los cuales trabajan con modulación FSK, su alimentación es de 5V y la frecuencia de operación está en el rango de 315.418 y 433.92 MHz. Para obtener resultados según lo que indican las especificaciones no es necesario tener un punto de vista despejado, estos módulos trabajan como lo haría un dispositivo WIFI, van irradiando la información que desean transmitir, no se necesita tampoco ubicar las antenas de los mismos de una manera específica, en la única situación que no transmitiría es si las antenas están separadas más del máximo alcance que es de 100 ft.

Los módulos que se usan en el presente trabajo tienen integrado en uno sólo la etapa de recepción y transmisión entre sus pines están los de polarización, los de entrada y salida de datos y los destinados a funcionar como antena. Véase figura 3.5



Figura 3.5 Modulo HOPE RF

3.2.1.2.5 Pantalla LCD 16 x 2.

Una pantalla de cristal líquido o LCD (acrónimo del inglés liquid crystal display) es una pantalla delgada y plana formada por un número de píxeles en color o monocromos colocados delante de una fuente de luz o reflectora. A menudo se utiliza en dispositivos electrónicos de pilas, ya que utiliza cantidades muy pequeñas de energía eléctrica.

Cada píxel de un LCD típicamente consiste de una capa de moléculas alineadas entre dos electrodos transparentes, y dos filtros de polarización, los ejes de transmisión de cada uno que están (en la mayoría de los casos) perpendiculares entre sí. Sin cristal líquido entre el filtro polarizante, la luz que pasa por el primer filtro sería bloqueada por el segundo (cruzando) polarizador.

La superficie de los electrodos que están en contacto con los materiales de cristal líquido es tratada a fin de ajustar las moléculas de cristal líquido en una dirección en particular. Este tratamiento suele ser normalmente aplicable consiste en una fina capa de polímero que es unidireccionalmente frotada utilizando, por ejemplo, un paño. La dirección de la alineación de cristal líquido se define por la dirección de frotación.

La pantalla LCD que se usa en el presente proyecto es LCD 16 x 2, las características de este tipo de pantalla se puede asemejar con las

dimensiones de una matriz 16 x 2, donde 16, o las columnas, corresponden al número máximo de caracteres que se pueden mostrar en pantalla y 2 comprende el número de filas a presentar en la pantalla.

3.2.1.2.6 MAX232CPE.

Debido a que los niveles de voltaje de la tarjeta GPS y de los módulos RF no son suficientes para entablar comunicación con la PC fue necesario utilizar un MAX232 a fin de convertir el voltaje a 10V. El MAX232 tiene función emisor y receptor integrados y tiene la capacidad de administrar dos puertos seriales y obedece a la configuración mostrada en la figura 3.5

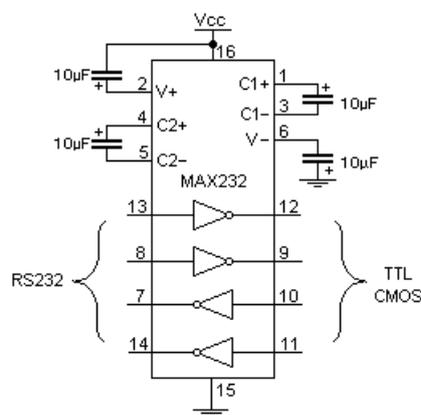


Figura 3.6 Configuración del MAX232

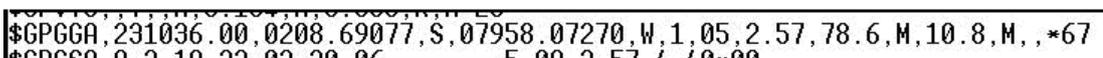
CAPÍTULO 4

4 Simulación y Pruebas Experimentales

En este capítulo se trata la puesta en marcha del proyecto, simulaciones y pruebas que se realizaron a lo largo del desarrollo del proyecto.

4.1 Recepción de Datos del módulo GPS

El primer paso a dar es verificar el correcto funcionamiento del receptor GPS, es decir, que entable comunicación con los satélites de posicionamiento y que las tramas NMEA tengan el formato requerido GGPGA como lo indica la figura 4.1



```
$GPGGA,231036.00,0208.69077,S,07958.07270,W,1,05,2.57,78.6,M,10.8,M,,*67
```

Figura 4.1 Formato GGPGA requerido

Debido a que el módulo GPS en uso solamente tiene un cable con terminales se procedió a usar un adaptador de tipo USB a fin de conectarlo al computador, se configuró un nuevo puerto serial virtual a

9600 baudios por medio del Hyperterminal que es una aplicación disponible en el sistema operativo Windows, y al cabo de unos segundos establece comunicación y en la pantalla del hyperterminal se empiezan a visualizar las tramas NMEA. Véase figura 4.1

```

$GPVTG , T , M , 0.161 , N , 0.298 , K , A * 26
$GPGGA , 231035.00 , 0208.69068 , S , 07958.07253 , W , 1.05 , 2.57 , 78.5 , M , 10.8 , M , , * 68
$GPGSA , A , 3 , 19 , 32 , 03 , 20 , 06 , , , , , , 5.09 , 2.57 , 4.40 * 00
$GPGSV , 3 , 1 , 11 , 03 , 66 , 156 , 35 , 06 , 64 , 115 , 21 , 11 , 08 , 229 , , 14 , 29 , 116 , * 77
$GPGSV , 3 , 2 , 11 , 16 , 39 , 004 , , 19 , 40 , 184 , 46 , 20 , 31 , 293 , 25 , 22 , 03 , 146 , * 7D
$GPGSV , 3 , 3 , 11 , 23 , 08 , 333 , , 31 , 11 , 044 , , 32 , 57 , 274 , 33 * 43
$GPGLL , 0208.69068 , S , 07958.07253 , W , 231035.00 , A , A * 6B
$GPRMC , 231036.00 , A , 0208.69077 , S , 07958.07270 , W , 0.164 , , 200410 , , , A * 7A
$GPVTG , T , M , 0.164 , N , 0.305 , K , A * 26
$GPGGA , 231036.00 , 0208.69077 , S , 07958.07270 , W , 1.05 , 2.57 , 78.6 , M , 10.8 , M , , * 67
$GPGSA , A , 3 , 19 , 32 , 03 , 20 , 06 , , , , , , 5.09 , 2.57 , 4.40 * 00
$GPGSV , 3 , 1 , 11 , 03 , 66 , 156 , 35 , 06 , 64 , 115 , 22 , 11 , 08 , 229 , , 14 , 29 , 116 , * 74
$GPGSV , 3 , 2 , 11 , 16 , 39 , 004 , , 19 , 40 , 184 , 46 , 20 , 31 , 293 , 25 , 22 , 03 , 146 , * 7D
$GPGSV , 3 , 3 , 11 , 23 , 08 , 333 , , 31 , 11 , 044 , , 32 , 57 , 274 , 32 * 42
$GPGLL , 0208.69077 , S , 07958.07270 , W , 231036.00 , A , A * 67
$GPRMC , 231037.00 , A , 0208.69085 , S , 07958.07289 , W , 0.033 , , 200410 , , , A * 73
$GPVTG , T , M , 0.033 , N , 0.061 , K , A * 24
$GPGGA , 231037.00 , 0208.69085 , S , 07958.07289 , W , 1.05 , 2.57 , 78.7 , M , 10.8 , M , , * 6C
$GPGSA , A , 3 , 19 , 32 , 03 , 20 , 06 , , , , , , 5.09 , 2.57 , 4.40 * 00
$GPGSV , 3 , 1 , 11 , 03 , 66 , 156 , 35 , 06 , 64 , 115 , 23 , 11 , 08 , 229 , , 14 , 29 , 116 , * 75
$GPGSV , 3 , 2 , 11 , 16 , 39 , 004 , , 19 , 40 , 184 , 46 , 20 , 31 , 293 , 24 , 22 , 03 , 146 , * 7C
$GPGSV , 3 , 3 , 11 , 23 , 08 , 333 , , 31 , 11 , 044 , , 32 , 57 , 274 , 31 * 41
$GPGLL , 0208.69085 , S , 07958.07289 , W , 231037.00 , A , A * 6D

```

Figura 4.2 Prueba GPS 18LVC

4.2 Simulaciones Realizadas

A continuación, las simulaciones propias del tema del proyecto.

4.2.1 Proyecto en Proteus

En la simulación, figura 4.3, a fin de visualizar la recepción de datos, el enlace entre los microcontroladores, y la respectiva transmisión de los mismos hasta la estación de control se utilizó la opción Virtual Terminal

en Proteus. Véase figuras 4.4, 4.5 y 4.6. En la implementación física, se puede ubicar de manera opcional una pantalla LCD que suplirá el virtual terminal que verifica la recepción del módulo GPS.

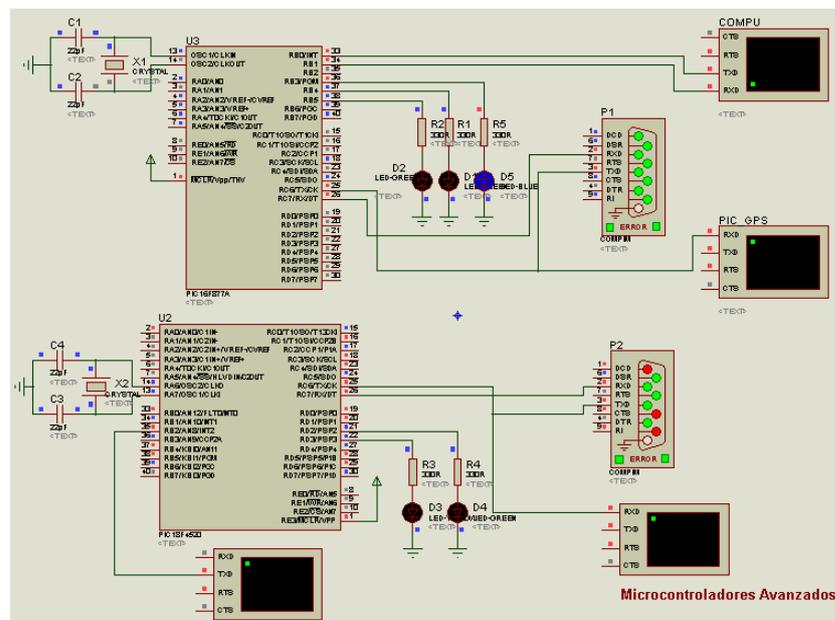


Figura 4.3 Simulación con Proteus

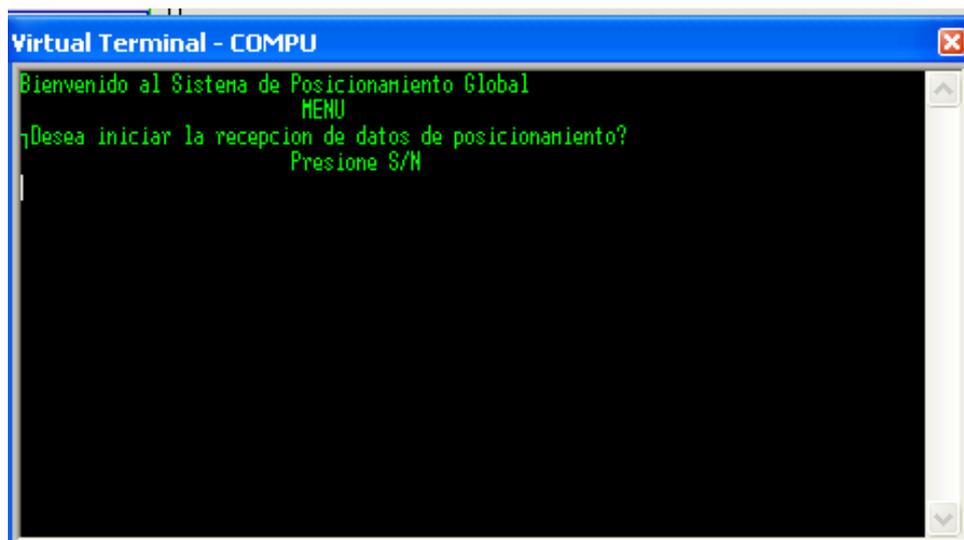


Figura 4.4 Simulación de acceso a permiso de conexión con GPS

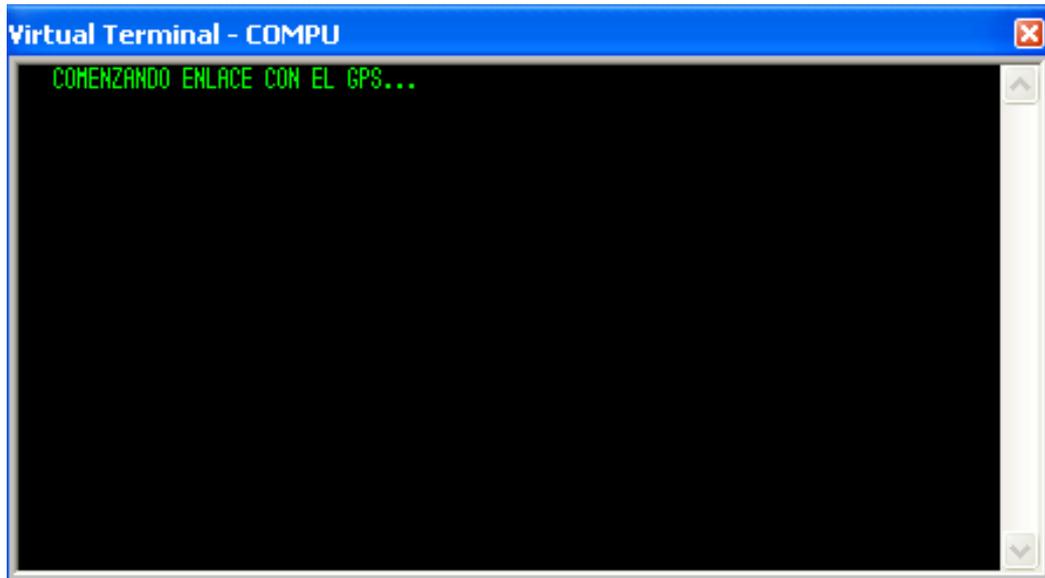


Figura 4.5 Simulación en enlace con módulo GPS

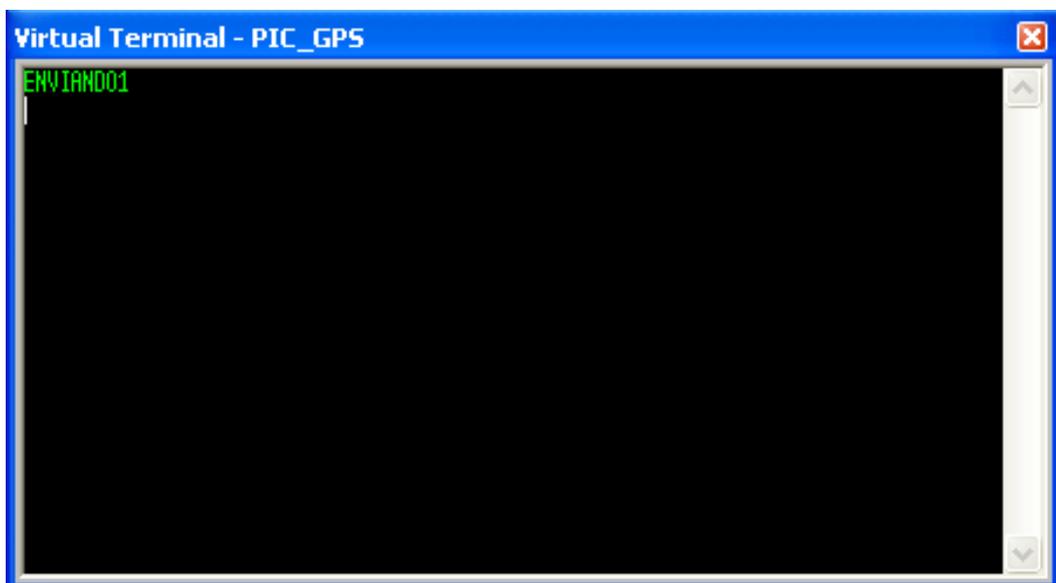


Figura 4.6 Simulación envío de datos a estación central

CONCLUSIONES Y RECOMENDACIONES

1. El desarrollo de nuevas tecnologías es realizable gracias a la combinación de otras ya existentes, en este caso, comprobamos que luego de adquirir un dato de posicionamiento desde un receptor GPS, podemos monitorear dicha recepción en una estación de control, la cual se encuentra ubicada en un sitio distante al lugar donde se realizó la extracción de datos GPS.
2. Este proyecto representa una respuesta a ciertas necesidades existentes en la actualidad, demostrando que con un poco de ingenio podemos resolver problemas de recepción GPS y realizar ejercicios didácticos de comunicaciones que nos permitan enriquecer nuestra experiencia en dicho campo.
3. Aunque se cuenta con componentes de primera calidad, en base a pruebas se establece que los datos de posicionamiento no son precisos, tienen un error límite de 15 m. a la redonda, lo cual sólo puede mejorarse con un módulo receptor de mejores capacidades.
4. Los resultados obtenidos han estado de acuerdo a los datos esperados, no obstante si queremos aumentar la distancia de transmisión y recepción entre los módulos de radiofrecuencia la única solución es cambiar dichos módulos por unos con mejores

características, pero se debe hacer con mucha cautela ya que sus capacidades dependen del tipo de modulación que usen.

5. Luego de un análisis de costos de materiales usados, versus la gran variedad de campos de aplicación del presente proyecto, se puede decir que además de ser viable la elaboración del prototipo funcional del mismo, económicamente hablando, es una opción accesible para todos, y no únicamente para sectores de la sociedad con amplia capacidad de compra.

Las recomendaciones son:

1. Al empezar a desarrollar la presente trabajo, es fundamental estar bien informado de los temas centrales del proyecto, ya que así se puede tener un buen desenvolvimiento en el mismo.
2. Es recomendable revisar de forma muy minuciosa los manuales de operación y características de los elementos a usar, ya que debido a sus capacidades se podrían tener inconvenientes de voltaje al intentar entablar una comunicación serial, lo cual se resuelve con un MAX232.
3. Al momento de implementar el proyecto en la vida real hay que tomar en cuenta las limitaciones del GPS por tanto es mejor ubicarlo en lugares donde la calidad de recepción no se vea comprometida, la mejor opción es poner la antena receptora al aire libre, pendiente de una ventana o fuera de paredes de concreto.
4. Si se quiere lograr el alcance máximo de los módulos RF, se debe ubicarlos entre ellos en línea de punto de vista.
5. Verificar previamente el correcto funcionamiento de cada elemento y modulo a usar ya que el presente proyecto se desarrolla en cascada y si un elemento falla, la siguiente etapa no va a funcionar de la manera esperada.

ANEXOS

ANEXO 1

Hoja de datos técnicos módulo GPS 18LVC

1.4.3 GPS 18 LVC, GPS 18 LVC-5m, and GPS 18-5Hz

These products interface to a serial port. The units accept TIA-232-F (RS-232) level inputs and transmit voltage levels that swing from ground to the positive supply voltage, TIA-232-F (RS-232) polarity. They also have reverse polarity protection. The cable contains wires for power, ground, receive, transmit, and measurement pulse output.

At the end of the cable, the wires are terminated in a connector that is used by Garmin for testing purposes. Most customers will remove this connector and replace it with another connector of their own choosing. Removing the factory installed connector and/or replacing with another customer-supplied connector will have no effect on the warranty (see Section 1.2 Limited Warranty). The factory-installed connector will mate with JST right-angle PCB-mount connector (model BM06B-SRSS-TBT) or side-entry PCB-mount connector (model SM06B-SRSS-TB). You may obtain technical information on these PCB-mount mating connectors from the JST Web site: <http://www.jst.com/>.



Note: Unless otherwise specified, all references to the GPS 18 LVC also include the GPS 18 LVC-5m.

1.5.2.2 Input Current

- GPS 18 USB: 55 mA @ 5.0 V
- GPS 18 PC: 50 mA @ 13.8 V
- GPS 18 LVC: 60 mA @ 5.0 V
- GPS 18-5Hz: 65 mA @ 5.0 V

1.5.2.3 CMOS Serial Output Levels

- GPS 18 PC: 0 V to 5 V (Asynchronous Serial, TIA-232-F (RS-232) Compatible Polarity)
- GPS 18 LVC and GPS 18-5Hz: 0 V to V_{in} , between 4 and 5.5 V (Asynchronous Serial, TIA-232-F (RS-232) Compatible Polarity)

1.5.2.4 GPS Receiver Sensitivity

-165 dBW minimum

1.5.3 Environmental Characteristics

- Operating Temperature: -30°C to +80°C
- Storage Temperature: -40°C to +90°C

1.5.4 GPS Performance

1.5.4.1 Receiver

WAAS enabled; 12 parallel channel GPS receiver continuously tracks and uses up to 12 satellites (up to 11 with Measurement Pulse Output active) to compute and update your position.

1.5.4.2 Acquisition Times

- Reacquisition: Less than 2 seconds
- Warm: Approx. 15 seconds (all data known)
- Cold: Approx. 45 seconds (initial position, time, and almanac known; ephemeris unknown)
- AutoLocate™: 5 minutes (almanac known; initial position and time unknown)
- SkySearch: 5 minutes (no data known)

1.5.4.3 Update Rate

- GPS 18 USB, PC, and LVC: 1 record per second
- GPS 18-5Hz: 5 records per second

1.5.4.4 Accuracy

- GPS Standard Positioning Service (SPS)
Position: < 15 meters, 95% typical
Velocity: 0.1 knot RMS steady state
- WAAS
Position: < 3 meters, 95% typical
Velocity: 0.1 knot RMS steady state
- Measurement Pulse Output Time: ±1 microsecond at rising edge of the pulse
- Dynamics: 999 knots velocity (only limited at altitude greater than 60,000 feet), 6g dynamics

ANEXO 2

Hoja de datos técnicos PIC18F452

28/40-pin High Performance, Enhanced FLASH Microcontrollers with 10-Bit A/D

High Performance RISC CPU:

- C compiler optimized architecture/Instruction set
 - Source code compatible with the PIC-16 and PIC-17 instruction sets
- Linear program memory addressing to 32 Kbytes
- Linear data memory addressing to 1.5 Kbytes

Device	On-Chip Program Memory		On-Chip RAM (bytes)	Data EEPROM (bytes)
	FLASH (bytes)	# Single Word Instructions		
PIC18F242	16K	8192	768	256
PIC18F252	32K	16384	1536	256
PIC18F442	16K	8192	768	256
PIC18F452	32K	16384	1536	256

- Up to 10 MIPS operation:
 - D/D - 40 MHz osc./clock input
 - 4 MHz - 10 MHz osc./clock input with PLL active
- 16-bit wide instructions, 8-bit wide data path
- Priority levels for Interrupts
- 8 x 8 Single Cycle Hardware Multiplier

Peripheral Features:

- High current sink/source 25 mA/25 mA
- Three external interrupt pins
- Timer0 module: 8-bit/16-bit timer/counter with 8-bit programmable prescaler
- Timer1 module: 16-bit timer/counter
- Timer2 module: 8-bit timer/counter with 8-bit period register (time-base for PWM)
- Timer3 module: 16-bit timer/counter
- Secondary oscillator clock option - Timer1/Timer3
- Two Capture/Compare/PWM (CCP) modules. CCP pins that can be configured as:
 - Capture Input. Capture is 16-bit, max. resolution 6.25 ns (70Y16)
 - Compare is 16-bit, max. resolution 100 ns (70Y16)
 - PWM output. PWM resolution is 1- to 10-bit, max. PWM freq. @: 8-bit resolution = 156 kHz, 10-bit resolution = 39 kHz
- Master Synchronous Serial Port (MSSP) module, two modes of operation:
 - 3-wire SPI™ (supports all 4 SPI modes)
 - I²C™ Master and Slave mode

Peripheral Features (Continued):

- Addressable USART module:
 - Supports RS-485 and RS-232
- Parallel Slave Port (PSP) module

Analog Features:

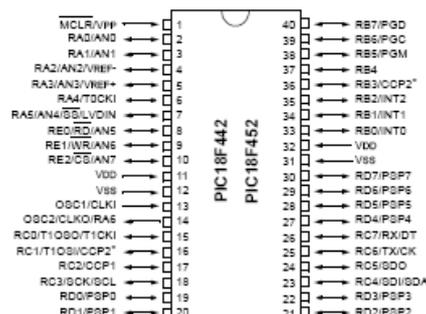
- Compatible 10-bit Analog-to-Digital Converter module (A/D) with:
 - Fast sampling rate
 - Conversion available during SLEEP
 - Linearity ≤ 1 LSB
- Programmable Low Voltage Detection (PLVD)
 - Supports Interrupt on-Low Voltage Detection
- Programmable Brown-out Reset (BOR)

Special Microcontroller Features:

- 100,000 erase/write cycle enhanced FLASH program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory
- FLASH/Data EEPROM Retention: > 40 years
- Self-reprogrammable under software control
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own On-Chip RC Oscillator for reliable operation
- Programmable code protection
- Power saving RI-FPP mode
- Selectable oscillator options including:
 - 4X Phase Lock Loop (of primary oscillator)
 - Secondary Oscillator (32 kHz) clock input
- Single supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins

CMOS Technology:

- Low power, high speed FLASH/EEPROM technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Industrial and Extended temperature ranges
- Low power consumption:
 - < 1.6 mA typical @ 5V, 4 MHz
 - 25 μ A typical @ 3V, 32 kHz
 - < 0.2 μ A typical standby current



ANEXO 3

Hoja de datos técnicos PIC 16F877A

28/40/44-Pin Enhanced Flash Microcontrollers

Devices Included in this Data Sheet:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

High-Performance RISC CPU:

- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches, which are two-cycle
- Operating speed: DC – 20 MHz clock input
DC – 200 ns instruction cycle
- Up to 8K x 14 words of Flash Program Memory,
Up to 368 x 8 bytes of Data Memory (RAM),
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to other 28-pin or 40/44-pin
PIC16CXXX and PIC16FXXX microcontrollers

Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,
can be incremented during Sleep via external
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period
register, prescaler and postscaler
- Two Capture, Compare, PWM modules
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- Synchronous Serial Port (SSP) with SPI™
(Master mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver
Transmitter (USART/SCI) with 9-bit address
detection
- Parallel Slave Port (PSP) – 8 bits wide with
external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for
Brown-out Reset (BOR)

Analog Features:

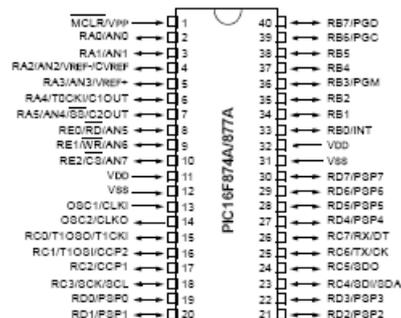
- 10-bit, up to 8-channel Analog-to-Digital
Converter (A/D)
- Brown-out Reset (BOR)
- Analog Comparator module with:
 - Two analog comparators
 - Programmable on-chip voltage reference
(VREF) module
 - Programmable input multiplexing from device
inputs and internal voltage reference
 - Comparator outputs are externally accessible

Special Microcontroller Features:

- 100,000 erase/write cycle Enhanced Flash
program memory typical
- 1,000,000 erase/write cycle Data EEPROM
memory typical
- Data EEPROM Retention > 40 years
- Self-reprogrammable under software control
- In-Circuit Serial Programming™ (ICSP™)
via two pins
- Single-supply 5V In-Circuit Serial Programming
- Watchdog Timer (WDT) with its own on-chip RC
oscillator for reliable operation
- Programmable code protection
- Power saving Sleep mode
- Selectable oscillator options
- In-Circuit Debug (ICD) via two pins

CMOS Technology:

- Low-power, high-speed Flash/EEPROM
technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Commercial and Industrial temperature ranges
- Low-power consumption



ANEXO 4

Módulo HOPE RF

HM-TR Transparent Wireless Data Link Module

1. General

HM-TR series transparent wireless data link module is developed by Hope microelectronics Co. Ltd, dedicated for applications that needs wireless data transmission. It features high data rate, longer transmission distance. The communication protocol is self controlled and completely transparent to user interface. The module can be embedded to your current design so that wireless communication can be set up easily.

2. Features

1. FSK technology, half duplex mode, robust to interference
2. ISM band, no need to apply frequency usage license
3. Operation frequency can be configured and can be used in FDMA applications
4. Transmitting frequency deviation and receiver bandwidth can be selected.
5. Protocol translation is self controlled, easy to use.
6. Data rate can be select from a wide range.
7. Provide ENABLE pin to control duty-cycle to satisfy different application requirements
8. High sensitivity, long transmission range.
9. Standard UART interface, TTL or RS232 logic level selectable
10. Very reliable, small size, easier mounting.
11. No tuning in producing

3. Application

- | | |
|--|-------------------------|
| 1. Remote control, remote measurement system | 5. Data collection |
| 2. Wireless metering | 6. IT home appliance |
| 3. Access control | 7. Smart house products |
| 4. Identity discrimination | 8. Baby monitoring |

4. Mechanical appearance

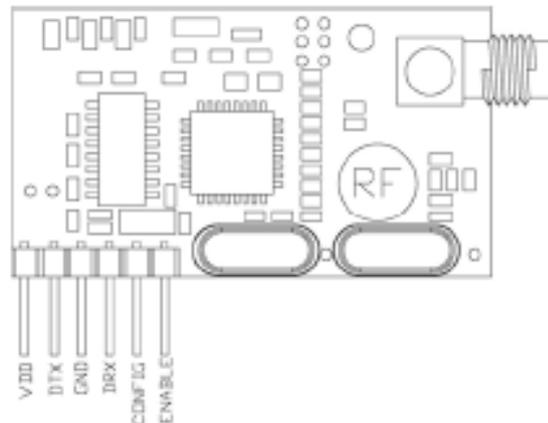


HM-TRXXX-232



HM-TRXXX-TTL

5. Pin definition



Pins	title	description
VCC	Power supply	+5V
DTX	Data transmission	Module data transmission
DRX	Data receiving	Module data receiving
CONFIG	Configure mode	If CONFIG pin is high at power on, the module enter configure mode to set up work parameters.
ENABLE	Working funtion	If config pin is low at power on, the module will enter normal mode for data transmission

6. Module parameters

Basic parameters

Working Voltage	Description	Min.	Typ.	Max.	5V
Working temperature		-35	25	80	℃
Working frequency	4 standard frequency	310.24	-	929.27	MHz
power	Pmax depends on the specific frequency	Pmax-21	Pmax	Pmax	dBm
frequency deviation		15		240	kHz
Receiving bandwidth		67		400	kHz
UART Baud Rate		300	9600	19200	bps
UART data bit		5	8	9	bit
Check bit	No check or Parity check				
Stop bit		1	1	2	bit
Antenna Connector					SMA (female)
Module size					24x43mm

ANEXO 5

Hoja de datos técnicos pantalla LCD 16x2



PIN NUMBER	SYMBOL	FUNCTION
1	Vee	GND
2	Vdd	+ 3V or + 5V
3	Vc	Contrast Adjustment
4	RS	H/L Register Select Signal
5	R/W	H/L Read/Write Signal
6	E	H →L Enable Signal
7	DB0	H/L Data Bus Line
8	DB1	H/L Data Bus Line
9	DB2	H/L Data Bus Line
10	DB3	H/L Data Bus Line
11	DB4	H/L Data Bus Line
12	DB5	H/L Data Bus Line
13	DB6	H/L Data Bus Line
14	DB7	H/L Data Bus Line
15	A/Vee	+ 4.2V for LED/Negative Voltage Output
16	K	Power Supply for B/L (OV)

ANEXO 6

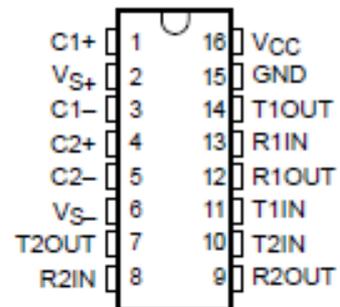
Hoja de datos técnicos MAX232

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047I – FEBRUARY 1989 – REVISED OCTOBER 2002

- Meet or Exceed TIA/EIA-232-F and ITU Recommendation V.28
- Operate With Single 5-V Power Supply
- Operate Up to 120 kbit/s
- Two Drivers and Two Receivers
- ± 30 -V Input Levels
- Low Supply Current . . . 8 mA Typical
- Designed to be Interchangeable With Maxim MAX232
- ESD Protection Exceeds JESD 22 – 2000-V Human-Body Model (A114-A)
- Applications
 - TIA/EIA-232-F
 - Battery-Powered Systems
 - Terminals
 - Modems
 - Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ± 30 -V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

TA	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube	MAX232N	MAX232N
	SOIC (D)	Tube	MAX232D	MAX232
		Tape and reel	MAX232DR	
	SOIC (DW)	Tube	MAX232DW	MAX232
		Tape and reel	MAX232DWR	
SOP (NS)	Tape and reel	MAX232NSR	MAX232	
-40°C to 85°C	PDIP (N)	Tube	MAX232IN	MAX232IN
	SOIC (D)	Tube	MAX232ID	MAX232I
		Tape and reel	MAX232IDR	
	SOIC (DW)	Tube	MAX232IDW	MAX232I
		Tape and reel	MAX232IDWR	

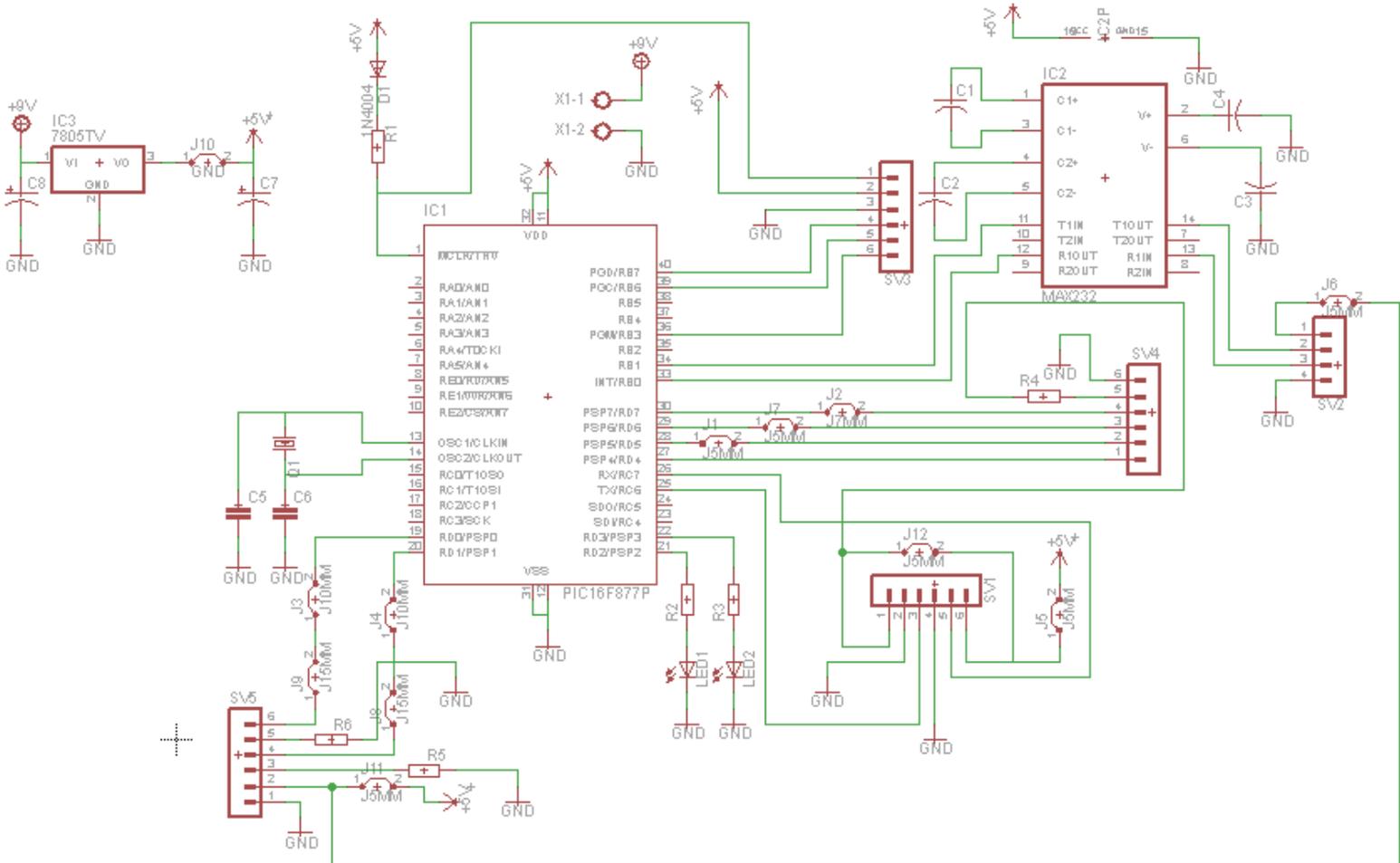
† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.

ANEXO 7

Esquemático y PCB del circuito

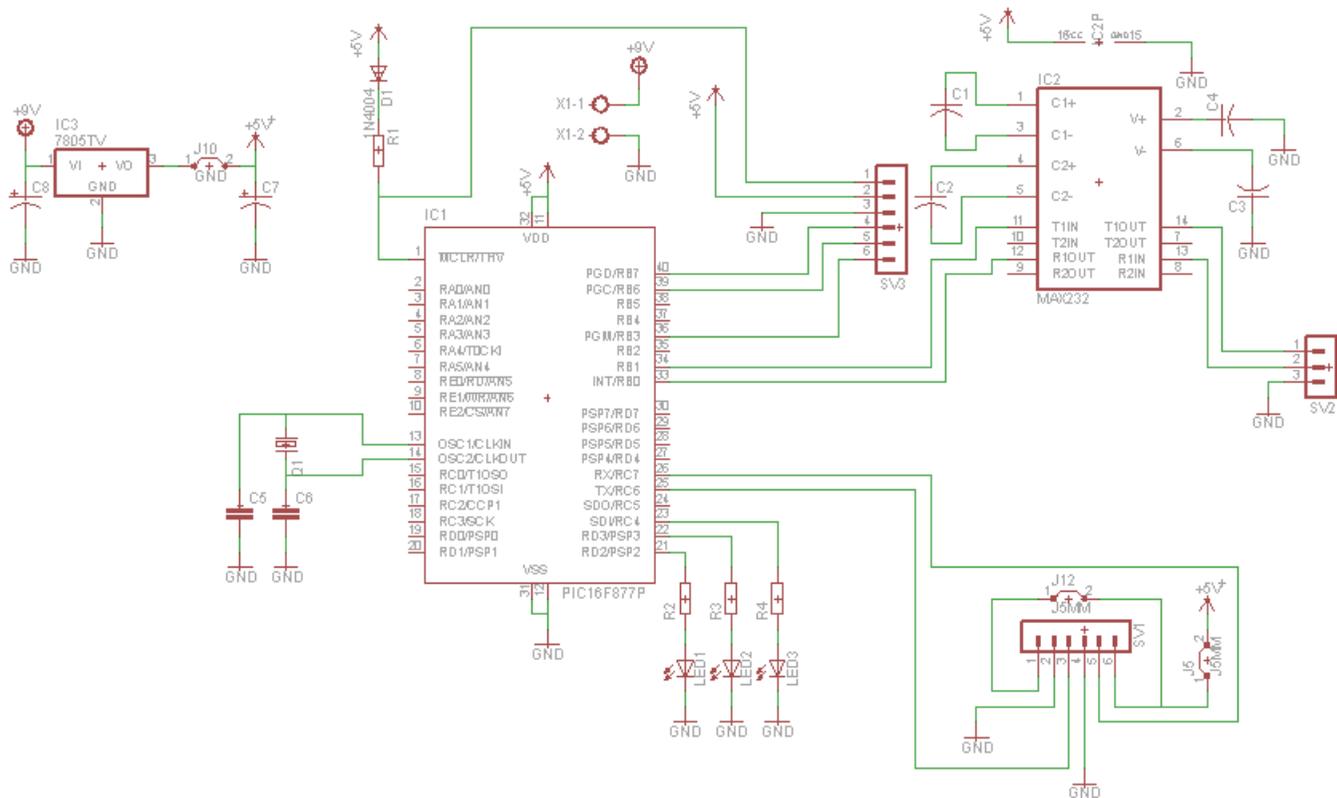
ESQUEMÁTICO 1

RF GPS

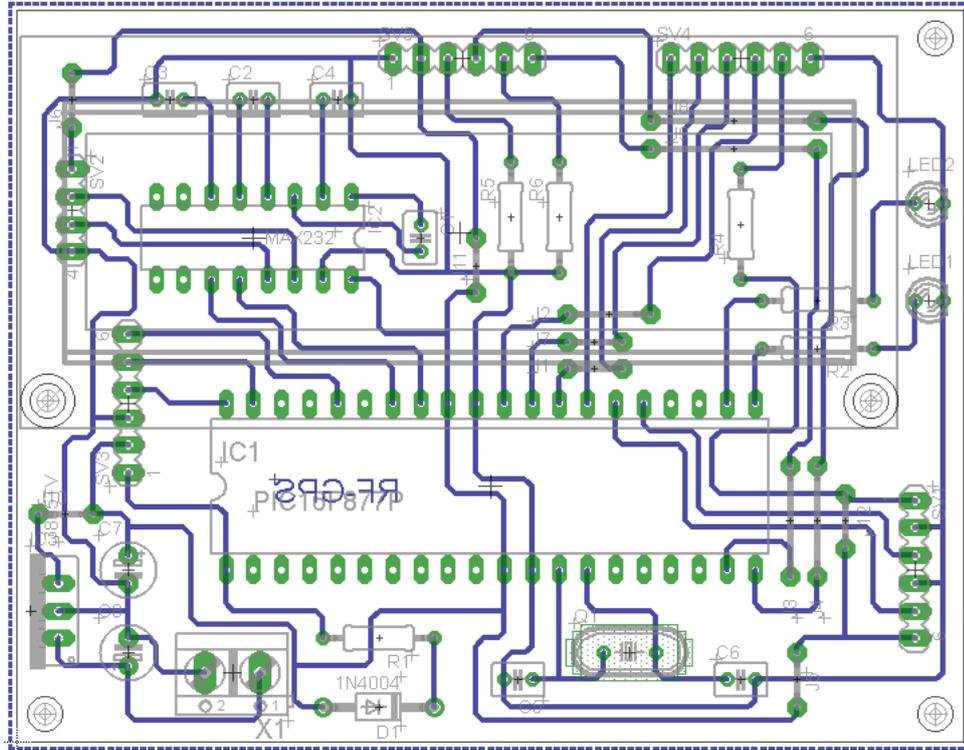


ESQUEMÁTICO 2

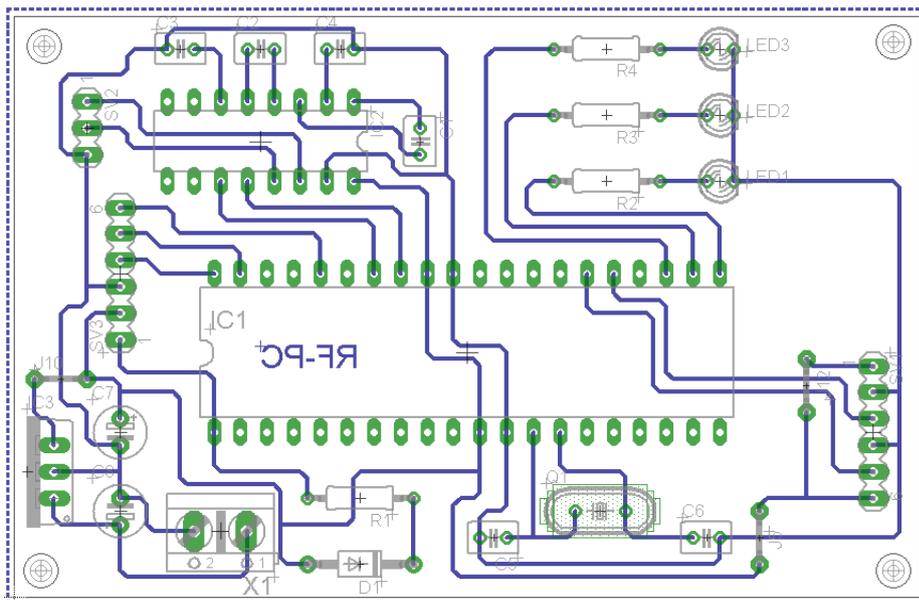
RF PC



PCB 1 RF GPS



PCB 2 RF PC



ANEXO 8

Implementación real del circuito

PROTOTIPO PLACA 1

RF GPS



PROTOTIPO PLACA 2

RF PC



ANEXO 9

Lista de Materiales y Presupuesto

Cant.	Descripción	Precio U	Precio T
1	Tarjeta SmartGPS	\$ 97,00	\$ 97,00
1	PIC18F452 microcontrolador	\$ 9,00	\$ 9,00
1	PIC 16F877A microcontrolador	\$ 4,50	\$ 4,50
2	Modulos HOPE RF 434A	\$ 21,00	\$ 42,00
10	Resistores película de carbón 1/2W	\$ 0,03	\$ 0,30
2	Oscilador de cuarzo de 8 y 20 MHz	\$ 0,80	\$ 0,80
2	Diodo 1N4007	\$ 0,05	\$ 0,10
12	Capacitores cerámicos	\$ 0,06	\$ 0,72
4	Capacitores electrolíticos 35V	\$ 0,10	\$ 0,40
2	LM7805 Regulador de voltaje +5V	\$ 0,25	\$ 0,50
2	Tiras de espadines macho	\$ 0,40	\$ 0,80
2	Tiras de espadines hembra	\$ 1,20	\$ 2,40
1	Tiras de espadines hembra con pines a 90°	\$ 1,00	\$ 1,00
5	Diodos LED 5mm	\$ 0,10	\$ 0,50
2	Borneras	\$ 0,30	\$ 0,60
1	Fabricación PCB en placa de fibra	\$ 11,90	\$11,90
1	Cajas a medida en acrílico transparente	\$ 20,00	\$20,00
1	Pantalla LCD	\$ 10,00	\$ 10,00
		TOTAL	\$ 202,52

BIBLIOGRAFÍA

- [1]. Wikipedia; Sistema de Posicionamiento Global;
http://es.wikipedia.org/wiki/Sistema_de_posicionamiento_global; **Fecha de consulta:** 19/agosto/2010.
- [2]. U-blox; Data Sheet LEA-5X;
<http://www.u-blox.com/en/download-center.html> ; **Fecha de consulta:** 20/agosto/2010.
- [3]. Wikipedia; Radiofrecuencia;
<http://es.wikipedia.org/wiki/Radiofrecuencia>; **Fecha de consulta:** 20/agosto/2010.
- [4]. Ciberconta; APLICACIONES de la Radiofrecuencia;
<http://www.ciberconta.unizar.es/LECCION/INTRODUC/436.HTM>; **Fecha de consulta:** 20/agosto/2010.
- [5]. Microchip, Data Sheet PIC16F877A;
<http://ww1.microchip.com/downloads/en/devicedoc/39582b.pdf>; **Fecha de consulta:** 22/agosto/2010.
- [6]. Microchip, Data Sheet PIC18F4520;
<http://ww1.microchip.com/downloads/en/DeviceDoc/39631a.pdf> **Fecha de consulta:** 22/agosto/2010.
- [7]. GPS – NMEA sentence information;
<http://aprs.gids.nl/nmea/> ; **Fecha de consulta:** 23/agosto/2010.
- [8]. Arduino; Modulo transmisor receptor FSK;
<http://www.open.com.au/mikem/arduino/HopeRF/HopeRF.pdf> ; **Fecha de consulta:** 30/agosto/2010.
- [9]. GARMIN, Manual de operación de la tarjeta GPS 18 LVC;
http://www8.garmin.com/manuals/425_TechnicalSpecification.pdf ; **Fecha de consulta:** 22/septiembre/2010.