

"SISTEMA DE MONITOREO REMOTO DE DISTANCIA RECORRIDA"

Leyria, Santiago.

Felipe, Zabala.

Toledo Victoria.

Tutor: Cuenya, Guillermo Adrian.

Se implementara un sistema electronico mediante el cual se pretende recibir informacion captada de un dispositivo GNSS (Sistema global de navegacion por satelite), procesarla en un microcontrolador y enviarla por un sistema de radio frecuencia a una PC, donde mediante un soft se podra llevar un registro de los datos adquiridos y calculo de distancia recorrida. Una aplicacion de este sistema es el monitoreo remoto de la distancia que recorre un futbolista durante un partido.

Introduccion:

El **Sistema de Posicionamiento Global**, más conocido por sus siglas en inglés, **GPS** (siglas de **Global Positioning System**), es un sistema que permite determinar en toda la tierra la posición de un objeto (una persona, un vehículo). El sistema fue desarrollado, instalado y empleado por el Departamento de Defensa de los Estados Unidos. Para determinar las posiciones en el globo, el sistema GPS se sirve de 24 satélites y utiliza la trilateración.

El GPS funciona mediante una red de 24 satélites en órbita sobre el planeta Tierra, a 20.200km de altura, con trayectorias sincronizadas para cubrir toda la superficie de la Tierra. Cuando se desea determinar la posición, el receptor que se utiliza para ello localiza automáticamente como mínimo tres satélites de la red, de los que recibe unas señales indicando la identificación y la hora del reloj de cada uno de ellos. Con base en estas señales, el aparato sincroniza el reloj del GPS y calcula el tiempo que tardan en llegar las señales al equipo, y de tal modo mide la distancia al satélite mediante el método de trilateración inversa, el cual se basa en determinar la distancia de cada satélite al punto de medición. Conocidas las distancias, se determina fácilmente la propia posición relativa respecto a los satélites. Conociendo además las coordenadas o posición de cada uno de ellos por la señal que emiten, se obtiene la posición absoluta o 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 lleva a bordo cada uno de los satélites.

Teniendo en cuenta la necesidad de poder llevar un control de la distancia que recorre un jugador de futbol en un partido (o algun otro deporte), y poder monitorear, que jugador esta mas apto para continuar en el juego y cual ya esta alcanzando sus limites, o poder llevar estadisticas de rendimiento, se pretende

diseñar un sistemas electronico que me permita llevar a cabo un monitoreo en tiempo real de cada jugador.

Para este proyecto se usaran conocimiento adquiridos de materias de nuestra especialidad (Avionica) y equipos e instrumento que disponemos en nuestro laboratorio de electronica.

Desarrollo:

Una vez tomada la decicion de realizar este proyecto se procedio a buscar informacion en base a materiales y equipos que disponiamos en el laboratorio, luego de realizada la investigacion se dividio el proyecto en dos partes, diseño y armado de harware y diseño de software.

Para el hardware se conto con los siguiente materiales:

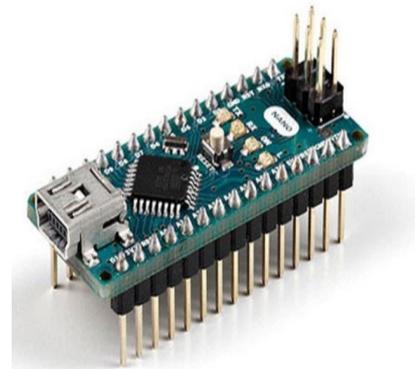
- Arduino uno/nano;
- modulos receptor y trasmisor "3DR Radio Telemetry";
- modulo GPS NEO-6;



GPS NEO-6



3DR Radio Telemetry



Arduino NANO

De la investigación del funcionamiento del GPS surgió el protocolo NMEA 0183 que es una especificación combinada eléctrica y de datos entre aparatos electrónicos marinos y, también, más generalmente, 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.

Los datos recibidos del GPS deberían llegar como una trama en la cual cada parate contiene una codificación de la información que requerimos. A continuación se muestran las cabeceras de la trama y sus significados:

Option	Description
GGA	Time, position and fix type data.
GLL	Latitude, longitude, UTC time of position fix and status.
GSA	GPS receiver operating mode, satellites used in the position solution, and DOP values.
GSV	The number of GPS satellites in view satellite ID numbers, elevation, azimuth, and SNR values.
MSS	Signal-to-noise ratio, signal strength, frequency, and bit rate from a radio-beacon receiver.
RMC	Time, date, position, course and speed data.
VTG	Course and speed information relative to the ground.
ZDA	PPS timing message (synchronized to PPS).
150	OK to send message.

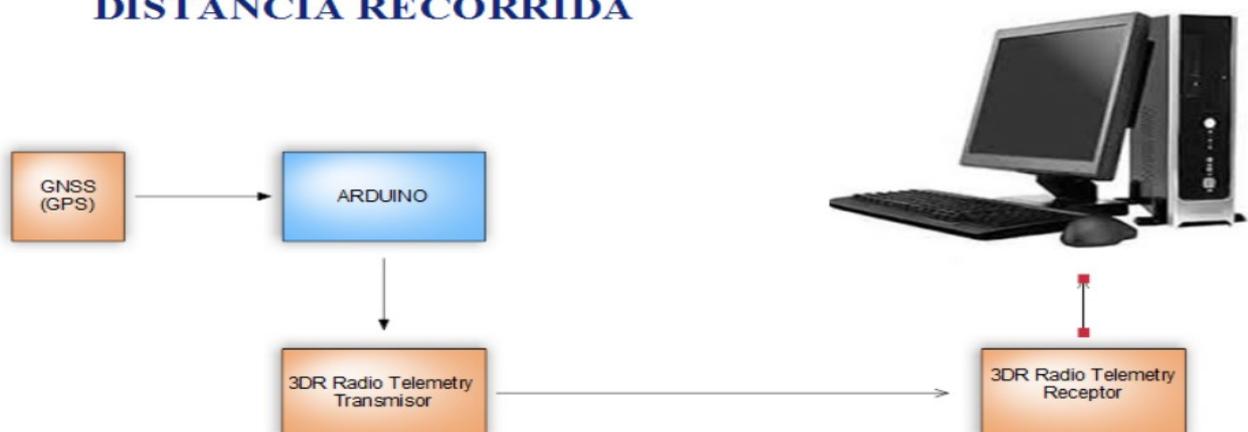
Un ejemplo de trama:

\$GPGGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M, , , ,0000*18

Name	Example	Units	Description
Message ID	\$GPGGA		GGA protocol header
UTC Time	161229.487		hhmmss.sss
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
Position Fix Indicator	1		See Table 1-4
Satellites Used	07		Range 0 to 12
HDOP	1.0		Horizontal Dilution of Precision
MSL Altitude	9.0	meters	
Units	M	meters	
Geoid Separation		meters	
Units	M	meters	
Age of Diff. Corr.		second	Null fields when DGPS is not used
Diff. Ref. Station ID	0000		
Checksum	*18		
<CR> <LF>			End of message termination

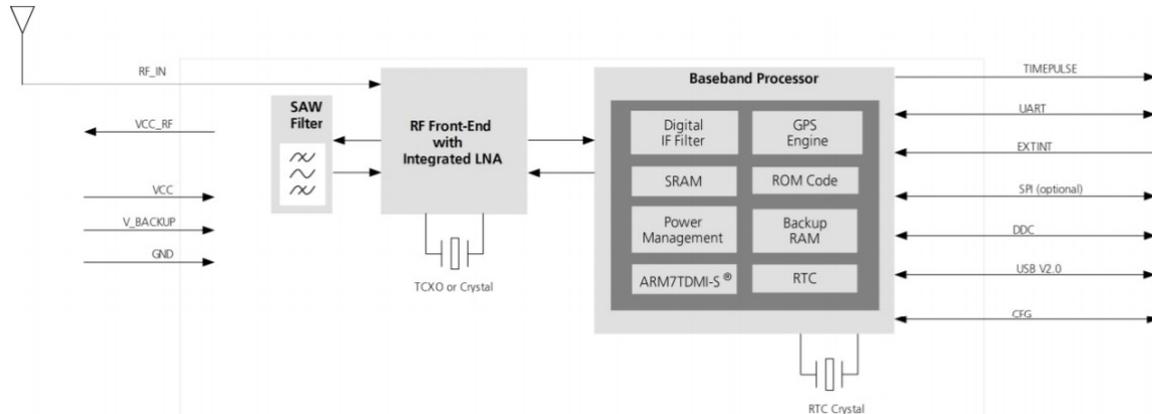
Luego se procedio a realizar la conexion de los modulos de telemetria y GPS al arduino para trabajar en su programacion. A continuacion se muestra el esquema de cuadros que describe la conexion:

SISTEMA DE MONITOREO REMOTO DE DISTANCIA RECORRIDA



Se detallan aspectos técnicos de cada dispositivo utilizado:

1. Diagrama de bloques del GPS extraído del datasheet:

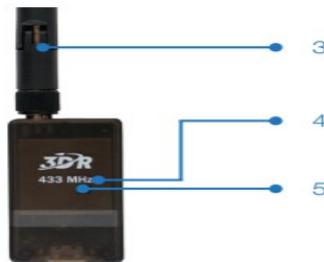


2. Descripción de los módulos de telemetría:

RADIO DESCRIPTION



- 1 Micro-USB port
- 2 6-wire cable connector
- 3 Antenna
- 4 Frequency
- 5 LED indicator



SPECIFICATIONS

Processing

100 mW maximum output power (adjustable)
 -117 dBm receive sensitivity
 Based on HopeRF's HM-TRP module
 RP-SMA connector
 2-way full-duplex communication through adaptive TDM
 UART interface
 Transparent serial link
 MAVLink protocol framing
 Frequency Hopping Spread Spectrum (FHSS)
 Configurable duty cycle
 Error correction corrects up to 25% of bit errors
 Open-source SIK firmware
 Configurable through Mission Planner & APM Planner

Features

Interchangeable air and ground modules
 915 or 433 MHz
 Micro-USB port
 6-position DF13 connector

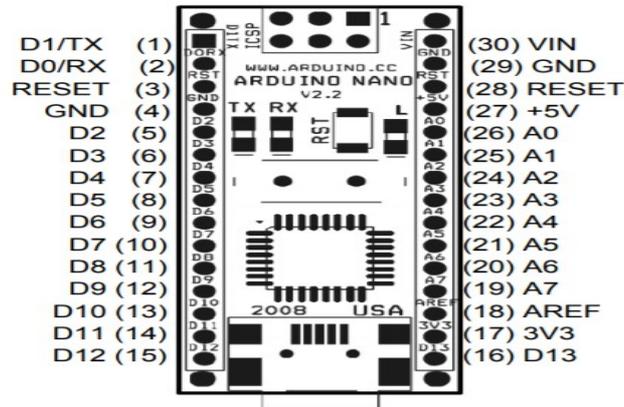
Dimensions

26.7 cm x 55.5 cm x 13.3 cm
 (without antenna)

Power

Supply voltage: 3.7-6 VDC (from USB or DF13)
 Transmit current: 100 mA at 30 dBm
 Receive current: 25 mA
 Serial interface: 3.3 V UART

3. Descripcion pines Arduino NANO:



Pin No.	Name	Type	Description
1-2, 5-16	D0-D13	I/O	Digital input/output port 0 to 13
3, 28	RESET	Input	Reset (active low)
4, 29	GND	PWR	Supply ground
17	3V3	Output	+3.3V output (from FTDI)
18	AREF	Input	ADC reference
19-26	A7-A0	Input	Analog input channel 0 to 7
27	+5V	Output or Input	+5V output (from on-board regulator) or +5V (input from external power supply)
30	VIN	PWR	Supply voltage

En cuanto a la programación, se debió realizar dos programas, uno en lenguaje C++ para arduino y otro en C# para la pc, que recibiría los datos y sería la interfaz con el usuario.

En la programación del arduino se utilizaron las siguientes librerías:

```
#SoftwareSerial.h
#TinyGPS.h
```

Las cuales nos facilitan el uso de puertos serie de la placa como también la conexión e interpretación de tramas de datos del GPS.

Basicamente, este programa recibe la trama que le envía el GPS, la interpreta y la reenvía por puerto serie al módulo transmisor.

Dentro de la programación de arduino también se realizó el cálculo de distancia recorrida utilizando longitud y latitud, para lo

cual se empleo la formula de HAVERSINE, la cual se aplica de la siguiente manera:

$$R = \text{radio de la Tierra}$$

$$\Delta\text{lat} = \text{lat2} - \text{lat1}$$

$$\Delta\text{long} = \text{long2} - \text{long1}$$

$$a = \sin^2(\Delta\text{lat}/2) + \cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \sin^2(\Delta\text{long}/2)$$

$$c = 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

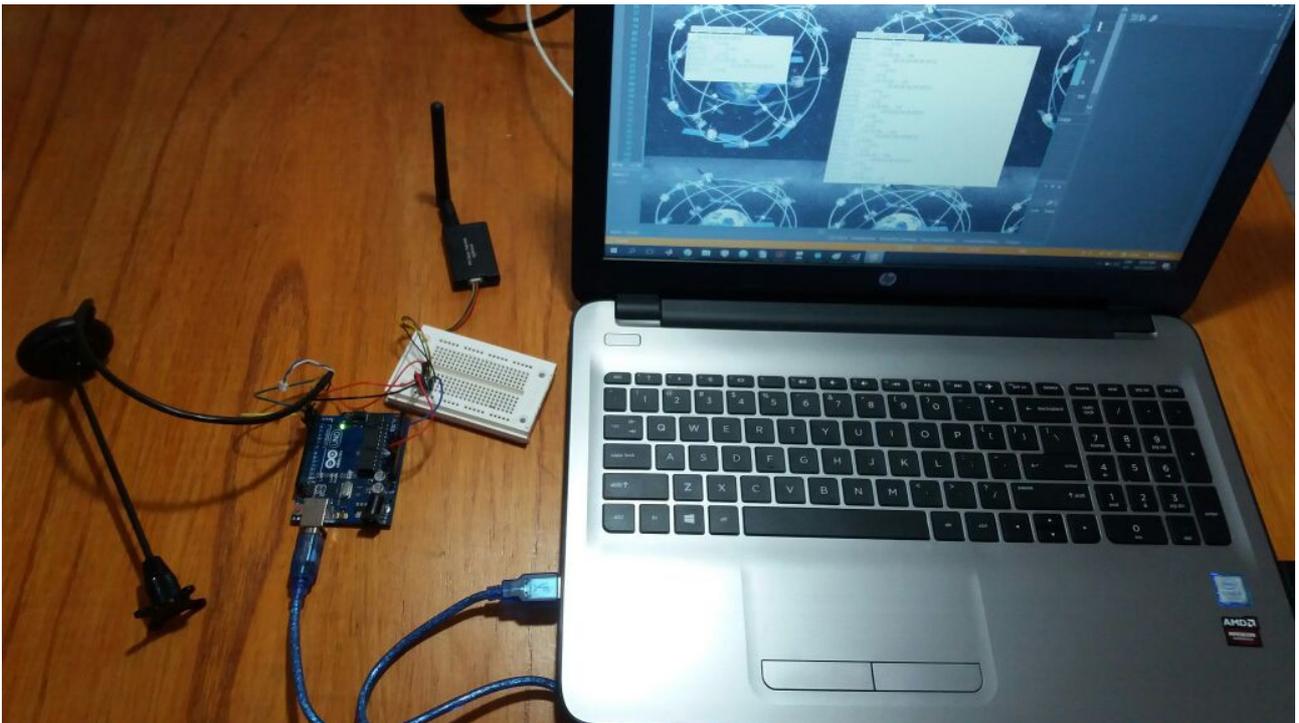
$$d = R \cdot c$$

Siendo "d" la distancia recorrida.

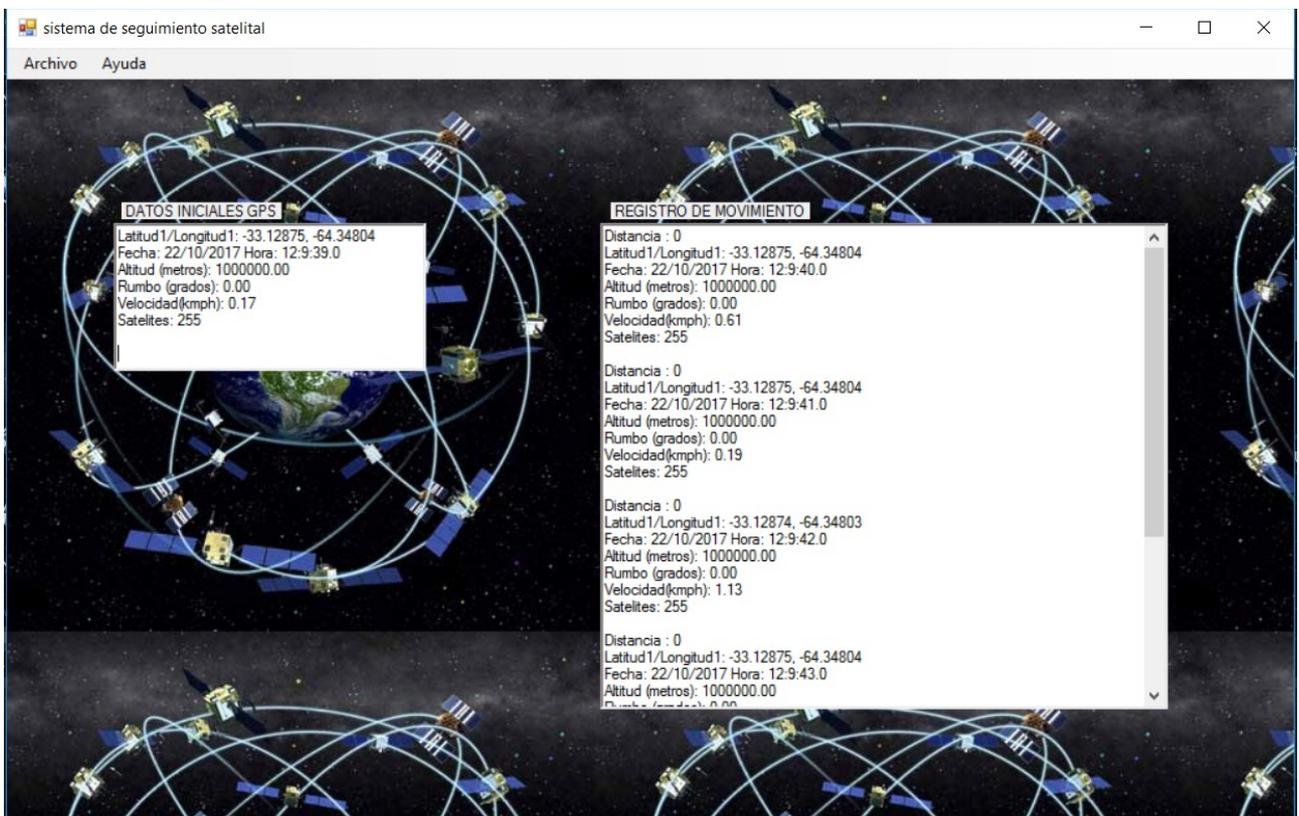
Por otro lado, la programacion de la pc que actuara como estacion monitora, programada en C#, solo estara escuchando el puerto donde se conecta el modulo receptor de telemetria y al llegar una cadena de datos la almacenara en una variable para luego mostrarla por un "richtextbox".

Tambien se agrego la posibilidad de guardar la informacion recibida en un archivo de texto para poder llevar registros de la informacion recibida.

Imágenes del sistema durante las pruebas:



Interfaz programada en C# recibiendo datos:



Conclusion:

Se logro realizar un sistema capaz de recibir tramas de datos captadas por un receptor gps, interpretarlas en una placa arduino y enviarlas a una pc ubicada a una distancia de mas de 100m.

El soft en la PC permite no solo ver la informacion en tiempo real sino tambien guardarla para poder analizarla con detalle, lo cual podria ser muy veneficioso para llevar estadisticas de, por ejemplo, rendimiento deportivo de un jugador de futbol.

Esta aplicacion se podria expandir de manera que permita monitorear mas de un jugador a la vez utilizando un identificador (id) para cada trama en los arduinos, y asi saber de que jugador provienen los datos.