

Manejo de un GPS con NIOS II

Jácome Diego, Ponguillo Ronald Ing.
Facultad de Ingeniería en Electricidad y Computación
Escuela Superior Politécnica del Litoral (ESPOL)
Campus Gustavo Galindo, Km 30.5 vía Perimetral
Apartado 09-01-5863. Guayaquil-Ecuador
{djacome, rponguil}@espol.edu.ec

Resumen

El presente trabajo consiste en la implementación de un sistema que maneje un módulo GPS con un FPGA Cyclone II de Altera, para conseguir una aplicación práctica que servirá de base para desarrollos más complejos. Se hace un análisis de la funcionalidad y protocolo de comunicación del módulo GPS marca SANAV que incorpora un puerto serial para trabajar directamente con el protocolo RS232, así se lo conecta directamente a la tarjeta de desarrollo DE2 que, además del FPGA, posee el hardware necesario para manejar el puerto serial y un display lcd en donde se presenta en todo momento el estado del sistema. La información se almacena en una tarjeta de memoria SD. La configuración del hardware en el FPGA se lo hace con la aplicación SOPC Builder en Quartus II. La programación del procesador NIOS II del FPGA esta desarrollada en C en una interfaz de Eclipse. Finalmente se llega a un prototipo que es probado en un vehículo en recorridos dentro de la ciudad y se usa un aplicativo web para convertir la información almacenada en la SD a un archivo KML para visualizar el recorrido con Google Earth.

Palabras Claves: GPS, FPGA, NIOS II, tarjeta de memoria SD.

Abstract

In this paper implements a system that manages a GPS module with an Altera FPGA Cyclone II, to achieve a practical application useful for more complex projects. An analysis of the functionality and communication protocol SANAV GPS module is done. It incorporates a serial port to work directly with the RS232 protocol and connects it to the develop board DE2 that, besides the FPGA, has the necessary hardware to handle the serial port and a LCD display where it displays at the state of the system all of time. The information is stored in SD memory card. The hardware configuration of the FPGA is done with the application SOPC Builder in Quartus II. The NIOS II processor programming of the FPGA is developed in C interface in Eclipse. A prototype is tested in a vehicle routes within the city and a web application is used to convert the information stored on the SD card to a KML file to display the route with Google Earth.

Keywords: GPS, FPGA, NIOS II, SD memory card.

1. Introducción

En los últimos años, la humanidad ha sido testigo de la impresionante velocidad a la que la tecnología va evolucionando, creando cada vez más y mejores dispositivos pensando en resolver problemas o simplemente en el confort de las personas que pueden acceder a dicha tecnología. Esto se evidencia en los dispositivos electrónicos de uso general como teléfonos móviles, tabletas y computadores. En esta línea todos los nuevos desarrollos están basados en tres ejes fundamentales: diseño, rapidez y funcionalidad. Todos basados en procesadores más versátiles, pequeños y rápidos.

Aquí es donde los denominados FPGAs (Field Programmable Gate Array) entran en acción, permitiendo a los usuarios programar tanto el hardware como el software de acuerdo a necesidades

específicas. Las soluciones basadas en esta tecnología son apreciadas por poseer una gran versatilidad a la hora de actualizar soluciones sin tener que cambiar de circuito integrado central, es decir, muy útiles cuando de prototipos se habla.

En este punto, al hacer aplicaciones con este tipo de circuitos integrados, existen muchas posibilidades de configurar el hardware de acuerdo a necesidades muy puntuales.

Hallar una aplicación con un módulo de GPS resulta bastante sencilla, dado lo extendido del uso de los mismo en diversos ámbitos como seguridad, logística y operaciones remotas. En este sentido, el presente trabajo desarrolla una aplicación sencilla pero eficaz para control y logística de recorridos de bienes o personas.

2. Objetivos

2.1 Objetivo General

Aplicar los conocimientos adquiridos para la implementación de un hardware que integre un FPGA Cyclone II de Altera con un módulo receptor de GPS, y lograr una aplicación práctica que incluso servirá de base para desarrollos más complejos.

2.2 Objetivos Específicos

- Investigar la forma de usar el driver de comunicación serial del FPGA Cyclone II, para poder recibir información serial desde un módulo receptor de GPS.
- Interpretar protocolos de comunicación del receptor GPS, así procesar dicha información.
- Almacenar la información del Receptor GPS en una memoria externa.
- Interpretar los datos almacenados para poder ser visualizados en un mapa georeferenciado.

3. Metodología

Este proyecto parte de un receptor GPS serial de marca SANNAV y de una tarjeta de desarrollo DE2 con un FPGA Cyclone II. El almacenamiento de información se lo hace en una tarjeta de memoria SD.

Se aprovecha el hardware que viene incluido en la **tarjeta DE2, específicamente el convertidor de niveles** para poder comunicarse con el receptor GPS a través del protocolo serial RS-232, el conector DB9, la ranura para la tarjeta SD y el LCD de 2x16.

Se configura el hardware en el FPGA, de tal manera que se tenga, además de los módulos básicos, los módulos necesarios para manejar el puerto serial y la memoria SD.

Posteriormente se programa el procesador Nios II para que pueda interpretar la información proveniente, cada segundo, del receptor GPS y almacenar, cada 10 segundos, la información en la tarjeta de memoria SD.

Finalmente la información almacenada en la SD puede ser visualizada con la herramienta gratuita Google Earth, convirtiendo el archivo de texto almacenado al formato KML con una herramienta web.

4. Marco Teórico

4.1 FPGA

FPGA, que en castellano se lo podría traducir como Conjunto de Puertas Programables en Campo, es un tipo de circuito integrado que se utiliza ampliamente en circuitos electrónicos digitales. Es un dispositivo semiconductor que contiene interconexiones y lógica programables. Los componentes lógicos programables pueden ser

configurados de acuerdo a necesidades específicas incluso si el producto ya ha sido instalado en campo, de ahí su nombre. Esta característica hace a los FPGAs muy útiles para ciertas aplicaciones.

Los FPGAs han evolucionado en el tiempo, pasando de usar entradas y salidas con lógica en interconexión programables, a consistir de varias combinaciones de Memorias SRAM configurables, transeptores de alta velocidad, entradas y salidas de alta velocidad, bloques lógicos, que incluyen elementos de memoria, y enrutamiento. Se pueden configurar desde puertas lógicas simples hasta complejas funciones combinatorias.

Con estos antecedentes fácilmente se puede determinar las ventajas de los FPGAs sobre otros circuitos integrados de aplicación específica:

- Menor tiempo para desarrollar prototipos
- Menor tiempo de comercialización
- Capacidad de ser reprogramados en campo
- Menor costo de investigación

Los FPGAs continúan evolucionando combinándolos con mejores procesadores, convirtiéndolos en sistemas en un solo chip. Varios fabricantes como Xilinx, Atmel y Altera fabrican este tipo de tecnología híbrida [1].

La serie de FPGAs Cyclone del fabricante Altera está diseñada para cumplir con un bajo consumo de energía, necesidades de diseño sensibles al costo, permitiendo comercializar productos más rápido, mayor integración, mayor rendimiento. La serie Cyclone va por su quinta generación. En el presente trabajo se usará la segunda generación, denominada Cyclone II, la misma que fue introducida en el 2004, con una tecnología de proceso de 90nm y recomendado para nuevos diseños [2]. El Dispositivo Cyclone II utilizado es el EP2C35 que posee las siguientes características (ver Tabla 1):

Tabla 1. Características de FPGA EP2C35

Característica	EP2C35
Elementos lógicos	33216
Bloques M4K RAM	105
Bits RAM Totales	483840
Multiplicadores Integrados	35
PLLs	4
Pines de E/S máximos	475

Nios II es un núcleo de procesador del fabricante Altera que combinado con un FPGA componen un sistema en un chip (SoC por sus siglas en Inglés). El Procesador Nios II es un procesador de propósito general RISC con las siguientes características:

- Conjunto de instrucciones de 32 bits
- 32 registros de propósito general

- 32 fuentes de interrupción
- Desempeño de hasta 250 DMIPS
- Entorno de desarrollo en C/C++

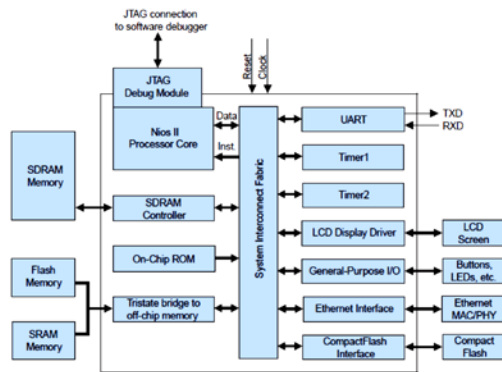


Figura 1. Diagrama de bloques de sistema NIOS II [3]

Un sistema de procesador Nios II es un microcontrolador que incluye un procesador y una combinación de periféricos y memoria en un solo chip. El sistema consiste en un núcleo de procesador Nios II, un conjunto de periféricos integrados, memoria interna, e interfaces para memoria externa, todo implementado en un solo dispositivo de Altera, como se muestra en la Figura 1[3].

4.2 GPS

GPS son las siglas que en castellano se traduce como Sistema de Posicionamiento Global, constituye un sistema de radionavegación por satélite desarrollado y operado por el Departamento de Defensa de EE.UU. El GPS permite a usuarios en tierra, mar, y aire determinar su posición tridimensional, velocidad y tiempo las 24 horas del día, en todo clima, en cualquier parte del mundo con una precisión y exactitud mucho mejor que otros sistemas de radionavegación disponibles en la actualidad o en un futuro previsible. GPS consta de tres segmentos: el espacio, el control y los usuarios.



Figura 2. Sistema de Satélites GPS [4]

El segmento espacial consiste en un mínimo de 24 satélites operacionales (ver figura 3) en seis órbitas circulares a 20.200 km por encima de la tierra en un ángulo de inclinación de 55 grados con un período de 12 horas. Los satélites en órbita están espaciados de modo que en cualquier momento un mínimo de 6 satélites estarán en vista a los usuarios en cualquier parte del mundo. Los satélites transmiten continuamente, en dos frecuencias la civil a 1575,42 MHz y la militar a 1227,60 MHz, datos de posición y tiempo a los usuarios en todo el mundo.

El segmento de control consta de una estación de control principal en tierra, en Colorado Springs, EE.UU., cinco estaciones de monitoreo y tres antenas en tierra ubicados alrededor el mundo. Las estaciones de monitoreo dan seguimiento de todos los satélites de la red GPS a la vista y recogen información enviadas por los satélites. Las estaciones de monitoreo envían la información que se obtiene de cada uno de los satélites a la estación de control principal, que calcula las órbitas de satélite de manera muy precisa. La información se formatea en mensajes de navegación actualizadas para cada satélite. La información actualizada se transmite a cada satélite a través de las antenas de tierra, que también transmiten y reciben señales de control de satélites y de vigilancia. El segmento del usuario consiste de los receptores, procesadores, y las antenas que permiten a los operadores en tierra, mar o aire recibir las emisiones por satélite GPS y calcular su posición exacta, velocidad y tiempo [5].

El concepto de funcionamiento de GPS se basa en el recorrido de los satélites de la red. Los usuarios determinan su posición en la tierra mediante la medición de su distancia del grupo de satélites en el espacio. Los satélites actúan como puntos de referencia precisos [4].

Cada satélite GPS transmite una posición exacta y la señal de tiempo. El receptor del usuario mide el tiempo de retardo que le tomó a la señal en llegar al receptor, que es la medida directa de la aparente distancia para el satélite. Las mediciones recogidas de forma simultánea desde cuatro satélites son procesadas para resolver las tres dimensiones de la posición, velocidad y tiempo del receptor [6].

Muchos receptores GPS pueden enviar datos de la posición a una PC u otro dispositivo usando el protocolo NMEA 0183. Aunque este protocolo se define oficialmente por la National Marine Electronics Association (NMEA), las referencias a este protocolo han sido recopiladas de los registros públicos, permitiendo que las herramientas de código abierto puedan leer el protocolo, sin violar las leyes de propiedad intelectual. Otros protocolos propietarios también existen, como el SiRF y protocolos de MTK. Los receptores pueden interactuar con otros dispositivos que utilizan métodos que incluyen una conexión en serie, USB o Bluetooth.

La trama del formato NMEA utilizado es:

\$GPRMC,081836,A,3751.65,S,14507.36,E,000.0,3
60.0,130998,011.3,E*62

Donde cada campo es descrito en la tabla 2.

Tabla 2. Campos trama NMEA GPRMC

Parámetro	Ejemplo	Descripción
Identificador	\$GPRMC	Identifica el tipo de Cadena
Hora UTC	081836	Hora UTC en HHmmss
Validez del dato	A	A: válido; V: no válido
Latitud	3751.65	Latitud en GGGmm.mm
Norte / Sur	S	
Longitud	14507.36	Longitud en GGGmm.mm
Este / Oeste	E	
Velocidad Nudos	000.0	Velocidad sobre la tierra
Dirección grados	360.0	Curso de navegación
Fecha UTC	130998	Fecha en DDMMYY
Variación Mag. en grados	011.3	Variación del este que se resta del curso verdadero
Este/Oeste	E	
Checksum	*62	Control de errores

5. Diseño e Implementación

El hardware esta compuesto por el FPGA, el receptor GPS, la Memoria SD y la tarjeta DE2 con los elementos para la comunicación RS232, el slot de memoria SD y el display LCD de 2x16 que se lo usa como auxiliar para verificar el funcionamiento del sistema (figura 4). Primero se debe configurar el hardware del FPGA con el software Quartus II, con la herramienta SOPC Builder se escogen los drivers para interactuar con el hardware externo.

El software del sistema maneja las operaciones del procesador NIOS II, desarrollado en lenguaje C en el entorno de programación Eclipse. Con esta programación se maneja los drivers del hardware configurados en el FPGA.

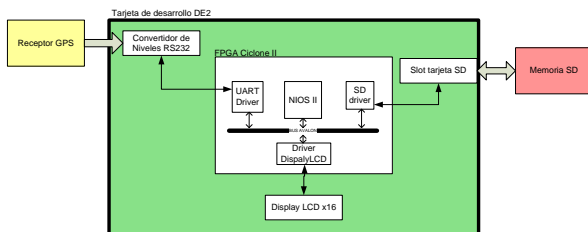


Figura 4. Componentes del sistema

5.1 Componentes Tarjeta DE2

Para el correcto funcionamiento del sistema se escogen los elementos de la tarjeta de desarrollo DE2 necesarios. Son necesarios tres elementos, además del FPGA Cyclone II (ver figura 5).

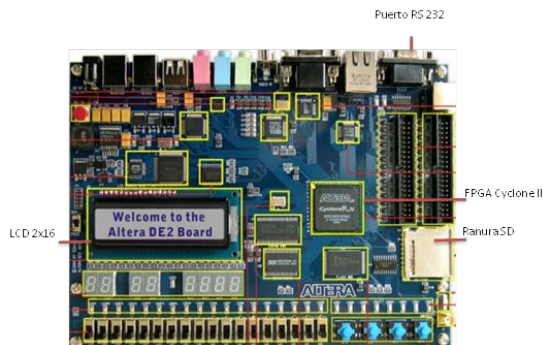


Figura 5. Componentes del sistema [7]

- Interfaz RS232: es la vía de entrada de la información que envía el Receptor GPS.
- Interfaz para tarjeta SD: ranura donde se inserta la tarjeta de Memoria SD que almacena la información de posición procesada.
- LCD 2x16: se presentan los mensajes indicando el estado del sistema y de los componentes conectados a la tarjeta de desarrollo DE2.

5.2 Hardware FPGA

Para configurar el hardware dentro del FPGA se usa el programa Quartus II, que contiene la herramienta SoPC Builder con la que se configura cada uno de los módulos. Los módulos necesarios para este proyecto son (ver figura 6):

- CPU: Procesador NIOS II
- Serial Port: Driver para RS232 UART
- Char_LCD_16x2: Driver para Display de caracteres de 16x2
- AVALON_UP_SD_CARD: Driver para interface de tarjeta.

Use	Conn.	Module Name	Description
<input checked="" type="checkbox"/>		CPU	Nios II Processor
<input checked="" type="checkbox"/>		instruction_master	Avalon Memory Mapped Master
<input checked="" type="checkbox"/>		data_master	Avalon Memory Mapped Master
<input checked="" type="checkbox"/>		bug_debug_module	Avalon Memory Mapped Slave
<input checked="" type="checkbox"/>		SRAM	SRAM Controller
<input checked="" type="checkbox"/>		SRAM	SRAM/SSRAM Controller
<input checked="" type="checkbox"/>		Red_LEDs	Parallel Port
<input checked="" type="checkbox"/>		Green_LEDs	Parallel Port
<input checked="" type="checkbox"/>		Slider_Switches	Parallel Port
<input checked="" type="checkbox"/>		PushButtons	Parallel Port
<input checked="" type="checkbox"/>		Expansion_JP2	Parallel Port
<input checked="" type="checkbox"/>		PS2_Port	PS2 Controller
<input checked="" type="checkbox"/>		JTAG_UART	JTAG UART
<input checked="" type="checkbox"/>		Serial_Port	RS232 UART
<input checked="" type="checkbox"/>		avalon_rs232_slave	Avalon Memory Mapped Slave
<input checked="" type="checkbox"/>		Interval_Timer	Interval Timer
<input checked="" type="checkbox"/>		sysid	System ID Peripheral
<input checked="" type="checkbox"/>		External_Clocks	Clock Signals for DE-Series Board Pe...
<input checked="" type="checkbox"/>		Char_LCD_16x2	16x2 Character Display
<input checked="" type="checkbox"/>		parallel_port_0	Parallel Port
<input checked="" type="checkbox"/>		avalon_parallel_port_slave	Avalon Memory Mapped Slave
<input checked="" type="checkbox"/>		pico_0	PIC (Parallel IO)
<input checked="" type="checkbox"/>		s1	Avalon Memory Mapped Slave
<input checked="" type="checkbox"/>		HEX7_HEX4	Parallel Port
<input checked="" type="checkbox"/>		avalon_parallel_port_slave	Avalon Memory Mapped Slave
<input checked="" type="checkbox"/>		Altera_up_sd_card_avalon_interface_0	SD Card Interface
<input checked="" type="checkbox"/>		avalon_sdcard_slave	Avalon Memory Mapped Slave

Figura 6. Componentes seleccionados en SOPC Builder

5.3 Módulo GPS

Se escoge un módulo del proveedor San Jose Navigation, del que ya se conocen productos de muy buen desempeño. Este proveedor ofrece el receptor GM-158 (ver figura 7), ideal para integraciones rápidas que cuenta con un conector serial para comunicación y un conector ps2 para alimentación de 5V [8]. Para efectos prácticos el conector PS2 se lo reemplazó por una pequeña fuente de 12V.



Figura 7. Módulo GM-158 [8]

Para que el receptor funcione se debe energizar, conectar a un puerto serial con velocidad de 9600 baud N-7-1, además de poner la antena con vista directa al cielo. El tiempo que le toma sincronizar depende de las condiciones de cobertura, que en condiciones óptimas de cobertura y habiendo estado apagado, toma aproximadamente 50 segundos sincronizar. El Protocolo de envío de datos se muestra a continuación:

```
$PMTK010,001*2E
```

```
$POLYN,TIME,RESTART_OCCURRED  
$POLYN,EPH,0,00000000  
$POLYN,ALM,0,00000000  
$GPRMC,235946.005,V,8960.0000,N,00000.0000,E,0.00,0.00,050180,,,N*77  
$GPRMC,235947.004,V,8960.0000,N,00000.0000,E,0.00,0.00,050180,,,N*77  
$GPRMC,235948.004,V,8960.0000,N,00000.0000,E,0.00,0.00,050180,,,N*78  
$GPRMC,194856.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*77  
$GPRMC,194857.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*76  
$GPRMC,194858.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*79  
$GPRMC,194859.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*78  
$GPRMC,194900.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*75  
$GPRMC,194901.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*74  
$GPRMC,194902.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*77  
$GPRMC,194903.159,V,8960.0000,N,00000.0000,E,0.00,0.00,180612,,,N*76  
$GPRMC,194904.153,V,0202.6747,S,07954.9338,W,0.16,0.00,180612,,,N*78  
$GPRMC,194906.000,V,0202.6748,S,07954.9278,W,0.22,0.00,180612,,,N*70  
$GPRMC,194907.000,V,0202.6740,S,07954.9336,W,0.04,0.00,180612,,,N*76  
$GPRMC,194908.000,A,0202.6750,S,07954.9327,W,0.04,0.00,180612,,,A*60  
$GPRMC,194909.000,A,0202.6752,S,07954.9326,W,0.11,0.00,180612,,,A*66  
$GPRMC,194910.000,A,0202.6757,S,07954.9325,W,0.21,0.00,180612,,,A*6B  
$GPRMC,194911.000,A,0202.6759,S,07954.9329,W,0.29,0.00,180612,,,A*60
```

5.4 Tarjeta de memoria SD

Para el almacenamiento de información se escogió a la marca Sandisk en la versión microSD (ver figura 8) con un adaptador para ser insertado en la interfaz disponible para el efecto en la tarjeta de desarrollo DE2. Esta marca es una de las populares por su precio

y calidad, siendo muy estables para aplicaciones en condiciones de alta temperatura como es el caso dentro de un vehículo. La versión micro SD se la escogió por estar más al alcance al momento de desarrollar el presente proyecto. El almacenamiento se lo hará cada 10 segundos por lo que no es muy importante la velocidad de grabación y únicamente se almacenan los datos válidos.



Figura 8. Tarjeta de memoria micro SD [8]

5.4 Programación del Procesador Nios II

En el entorno de Eclipse se puede fácilmente programar en lenguaje C las tareas que debe hacer el procesador. Se crea un nuevo proyecto basándose la plantilla "hello world", seleccionando el archivo nios_system.sopcinfo, generado en SOPC Builder, que describe la configuración del hardware dentro del FPGA y así agrega las librerías con las funciones a utilizar.

Además se agregan dos librerías, la 259macros.h y addressmap.h. Las dos se las utilizan en el manejo del LCD.

Una vez se tienen listas las librerías, se revisan los ejemplos de uso en la documentación disponible para partir de programaciones básicas. Es útil que el lenguaje sea C, ya que es conocido y la sintaxis resulta bastante familiar.

Lo primero que se hace en el código del programa es verificar que, tanto el módulo GPS como la tarjeta SD, estén correctamente conectados y funcionando. Luego se verifica si los datos recibidos del módulo GPS son válidos o no, para, finalmente, almacenarlo en un archivo de texto en la tarjeta de memoria SD. Se repite el ciclo infinitamente hasta que alguno de los componentes externos, módulo GPS o Tarjeta SD, sean desconectados. Al detectar que ya no se reciben datos seriales, se cierra el archivo de texto en la memoria SD y se termina el programa en el procesador. El archivo de texto en la SD, es utilizado para generar el recorrido en un mapa.

El hilo de ejecución de la programación en el procesador Nios II se lo muestra en un diagrama de flujo (ver figura 9).

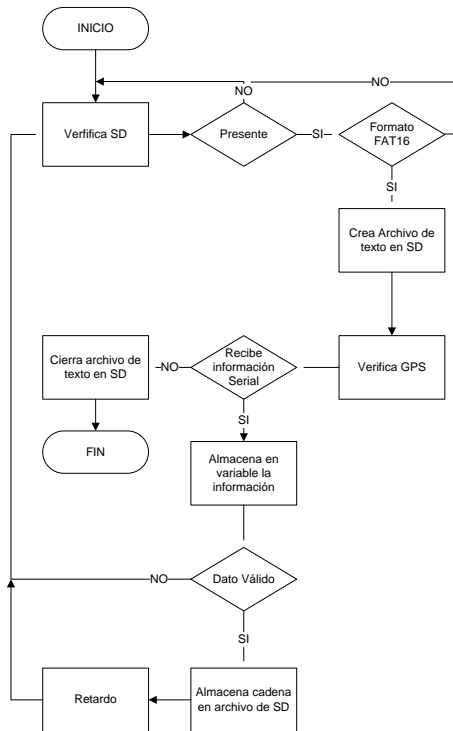


Figura 9. Flujo de programa en NIOS II

Al momento de encender el vehículo, se energiza el sistema, y se espera a que el módulo se sincronice y comience a almacenar la información recibida cada 10 segundos. Esta información es procesada y almacenada en un archivo de texto (ver figura 10). El tiempo del recorrido fue de aproximadamente 15 minutos por el norte de la ciudad de Guayaquil.

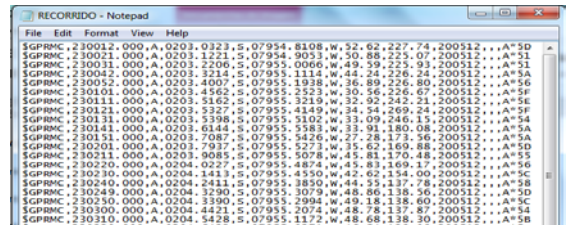


Figura 11. Archivo de texto almacenado en SD

Durante esta prueba el archivo de texto, con los datos válidos de posición identificados por la letra A de la cadena recibida, ocupa un espacio de 7 KB en la tarjeta de memoria en 15 minutos de recorrido, por lo que se podría estimar que la tarjeta de 2GB utilizada albergue hasta 70000 horas de información.

6. Pruebas y Resultados

6.1 Almacenamiento de información

La instalación del sistema se la hizo con un inversor de voltaje que se conecta en la cigarrillera de un vehículo, que transforma los 12 VDC del sistema eléctrico a 110VAC para alimentar la tarjeta de prueba y el receptor GPS (ver figura 10).



Figura 10. Prototipo en vehículo

6.2 Visualización en Google Earth

Bajamos la información contenida en la memoria SD en un computador, y los cargamos en el programa en línea en el enlace http://www.gpsvisualizer.com/map_input?form=googleearth, donde se escoge la opción Google Earth KML.

Se carga una nueva página donde se escogen las siguientes opciones:

- Tipo de archivo de salida: .kml
- Desviación de Hora: -5
- Adicionar etiquetas de tiempo: Si
- Dibujar como Puntos de ruta: Si
- Etiquetas de Puntos de ruta: Etiquetas en puntos de ruta + Marcas + puntos de trayectoria.
- Se escoge el archivo en la ruta almacenada, que contiene la información en modo texto de los datos de posición en formato NMEA.
- Se presiona el botón Crear archivo KML

Con estas opciones el archivo KML resultante de la figura 21, una vez ejecutado, mostrará en la interface de Google Earth la trayectoria en rojo con puntos numerados donde al hacer clic en cada uno de los puntos se vera información de hora, fecha y velocidad (ver figura 12).

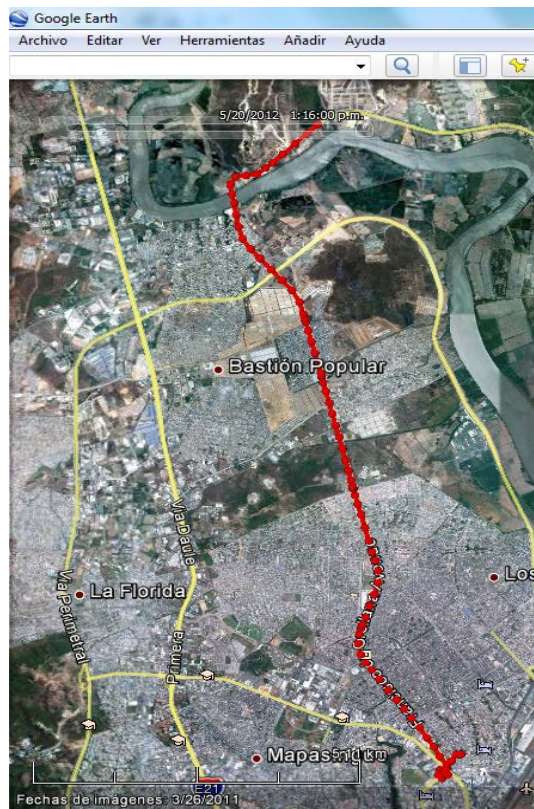


Figura 12. Recorrido en Google Earth

7. Conclusiones

1. La tarjeta DE2 facilita mucho el desarrollo de cualquier prototipo donde el núcleo del procesador es el NIOS II y el hardware adicional de la tarjeta sea suficiente para el mismo. Sin embargo, información con respecto a cierto hardware está muy limitada en la página del fabricante para programar en lenguaje C.
2. Se pudo constatar la facilidad de configurar un nuevo hardware en un FPGA, aun luego de haber implementado el proyecto en la primera versión. Para aumentar el LCD de 16x2, luego de las primeras configuraciones del hardware en el FPGA, simplemente se configuró el módulo para controlar el LCD y se aumentaron las líneas de programación en el procesador para manejarlo, sin tener que cambiar de circuito integrado central, como hubiese sido el caso si se trabaja con un microcontrolador y ya no se tuviesen mas puertos de comunicación.
3. Se concluye además que es posible leer y procesar datos provenientes de un receptor GPS. La información de esta tecnología es bastante amplia en la red, así como existe una

importante cantidad de fabricantes de este tipo de dispositivos que utilizan este servicio que actualmente es gratuito.

4. Las aplicaciones para el sistema GPS son muchas, y van creciendo a medida que los módulos son cada vez más pequeños, llegando fácilmente a pensar que en pocos años se tendería a perder la privacidad al saber la ubicación de todo a nuestro alrededor.
5. Las tarjetas SD son una solución económica y popular para cualquier tipo de aplicación donde se requiera almacenar temporalmente información, debiendo tomar en cuenta las condiciones de uso como temperatura, salinidad y humedad.
6. Se verificó además las limitaciones de la tecnología GPS, como el hecho de no siempre tener vista al cielo y los márgenes de error al no contar con buena cobertura.

8. Agradecimientos

Agradezco a las personas que nunca perdieron las esperanzas que terminaría este proyecto en especial mi familia y el Ing. Ronald Ponguillo.

9. Referencias

- [1] Wikipedia, Field Programmable Gate Array, http://es.wikipedia.org/wiki/Field_Programmable_Gate_Array, 15 de Junio 2012.
- [2] Altera Corporation, Cyclone II Device Handbook, http://www.altera.com/literature/hb/cyc2/cyc2_cii5v1.pdf, 18 de febrero de 2008.
- [3] Altera Corporation, Nios II Processor Reference Handbook, <http://www.altera.com/literature/lit-index.html>, 3 de Mayo de 2011.
- [4] TECNoproject, ¿Qué es GPS?, <http://www.tecnoprojectltda.com/QUEESGPS.htm>, 15 de junio de 2012.
- [5] Centro de Navegación de la Guardia Costera de EE.UU., General Information on GPS, <http://www.navcen.uscg.gov/?pageName=GPSmain>, 15 de junio de 2012.
- [6] WIKIPEDIA, Global Positioning System http://en.wikipedia.org/wiki/Global_Positioning_System, 15 de junio de 2012.
- [7] Altera Corporation, Altera University Program Media Computer Manual, <http://www.altera.com/literature/lit-index.html>, 28 de Julio de 2010.
- [8] San Jose Navigation, Mini GPS Locator, http://www.sanav.com/gps_receivers/gps_locators/TK-158.htm, 15 de junio de 2012.
- [9] Sistema Automático de Información de Posición de Holanda, NMEA Format, <http://aprs.gids.nl/nmea>, 15 de junio de 2012.

- [10] SD Group, SD Specifications: Part 1 Physical Layer Simplified Specification, https://www.sdcard.org/downloads/pls/simplified_specs/Part_1_Physical_Layer_Simplified_Specification_Ver_3.01_Final_100518.pdf, Mayo 18 de 2010.
- [11] SD Group, SD Choices, <https://www.sdcard.org/consumers/choices/>, 15 de junio de 2012.
- [12] Altera Corporation, Altera University Program Secure Data Card IP Core, <http://www.altera.com/literature/lit-index.html>, 12 de Junio de 2009.
- [13] Altera Corporation, Quartus II 7.1 Handbook, http://www.altera.com/literature/hb/qts/quartusii_handbook.pdf, 25 de abril de 2007.