



# **ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

## **FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN**

“Implementación de un Sistema de Comunicación por medio de la red GSM en un  
Procesador Embebido Configurable NIOS II”

### **INFORME DE MATERIA DE GRADUACIÓN**

Previa la obtención del Título de:

**INGENIERO EN TELEMÁTICA**

**INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES**

Presentado por:

Diana María Ordóñez Paredes

Andrés Antonio Vélez Naranjo

GUAYAQUIL – ECUADOR

AÑO 2012

# AGRADECIMIENTO

Quiero agradecer a Dios, a mi familia sobre todo a mi mamá por el apoyo incondicional que tuve, a mi papá por el esfuerzo que hizo por mí y por los consejos que recibí , a mi hermano por ser un ejemplo para mí , a mis amigos David Balseca y Darío Calle por la dedicación mutua que teníamos por salir adelante en nuestras carreras y a las personas que conocí y que me proporcionaron su ayuda para salir adelante, al Ing. Ponguillo por confiar en mí en que este proyecto lo íbamos a realizar con éxito, a todos ellos gracias.

**Andrés Vélez Naranjo**

Quiero agradecer a Dios, por iluminarme siempre y nunca desampararme en mi camino, también quiero agradecer mucho a mi mamá, María Piedad, ya que gracias a sus valiosos consejos siempre ha sabido guiarme en todo momento y escucharme en mis peores momentos, a mi papá, Carlos, gracias por su cuidado y protección, y a mis dos hermanos, Víctor y Alfredo que siempre han sido un gran apoyo a lo largo de toda mi educación, a mis amigos por todos los años de habernos conocido y compartido muchas horas de estudio y momentos de diversión, ya sea en las buenas o en las malas.

Un especial agradecimiento a mi profesor de materia, el Ing. Ronald Ponguillo, por haberme brindado su conocimiento, en la realización de este proyecto.

**Diana María Ordóñez Paredes.**

# DEDICATORIAS

Se lo dedico a mi mamá a mi papá por el apoyo incondicional que tuve durante mi carrera universitaria, a mis amigos que conocí en este trayecto ya que sin ellos nada de esto sería posible.

**Andrés Vélez Naranjo**

Este proyecto se lo dedico a mi mamá, mi papá, a mis hermanos, y a todas esas maravillosas personas que he conocido a lo largo de mi vida estudiantil, su apoyo incondicional me ha ayudado a seguir adelante.

**Diana María Ordóñez Paredes.**

# **TRIBUNAL DE SUSTENTACIÓN**

---

Ing. Ronald Ponguillo

PROFESOR DEL SEMINARIO DE GRADUACIÓN

---

Ing. Hugo Villavicencio

PROFESOR DELEGADO POR EL DECANO DE LA FACULTAD

# DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este trabajo, nos corresponde exclusivamente; y el patrimonio intelectual del mismo a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”.

(Reglamento de exámenes y títulos profesionales de la ESPOL)

---

Diana María Ordoñez Paredes

---

Andrés Antonio Vélez Naranjo

# RESUMEN

El presente trabajo “Implementación de un Sistema de Comunicación por medio de la red GSM en un Procesador Embebido Configurable NIOS II” es uno de los temas de la Materia de Graduación “PROCESADORES EMBEBIDOS CONFIGURABLES”. Uno de los principales objetivos es conocer la manera en que los FPGA están día a día mejorando los diseños de sistemas que se implementan en la lógica digital. Esta tecnología es capaz de reemplazar gran cantidad de circuitos digitales en un solo chip, con lo cual permiten que las distintas etapas de desarrollo y aplicación se desarrollen de manera más rápida ahorrando tiempo y costos al momento de la implementación del diseño

La programación la realizamos en lenguaje C utilizando el programa NIOS II. Además del FPGA, trabajamos con las siguientes herramientas: QUARTUS II el cual trabaja con lenguaje HDL, switches magnéticos de apertura de puerta, la tarjeta DE2 de la marca Altera y un módulo GSM de la marca Simcom.

El proyecto se lo ha estructurado en 4 capítulos que los detallamos a continuación:

El capítulo 1, trata de manera general el estado actual de la tecnología, en base al cual se intenta dar una solución óptima proponiendo como solución el presente proyecto.

El capítulo 2, trata el marco teórico; el Procesador Nios II y la Tarjeta DE2 de Altera, así como una introducción a la tecnología GSM, funcionamiento y arquitectura de la red GSM, protocolo de comunicación RS-232, y todos los conceptos claves en base a los cuales se ha trabajado.

El capítulo 3, trata el diseño e implementación del proyecto, el cual se va a orientar a un ambiente de laboratorio, se realiza un diagrama de bloques básico y general para luego ir explicando el funcionamiento detallado de cada uno de los componentes.

En el capítulo 4, se realiza las pruebas y se analiza los resultados en base a los parámetros establecidos, se detalla las ventajas y desventajas del mismo, así como también las recomendaciones en caso de posibles fallas presentadas por un problema físico o de configuración de los dispositivos.



# ÍNDICE GENERAL

RESUMEN

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ABREVIATURAS

INTRODUCCIÓN

1.	GENERALIDADES .....	1
1.1	Alcance y limitaciones del proyecto .....	1
1.2	Objetivos .....	2
1.2.1	Objetivos Generales.....	2
1.2.2	Objetivo Específico .....	2
1.3	Antecedentes .....	3
1.4	Identificación del Problema .....	4
1.5	Descripción breve de la posible solución .....	5

2.	MARCO TEÓRICO.....	6
2.1	Introducción al Microprocesador NIOS II .....	6
2.1.1	Características y Arquitectura .....	7
2.1.2	Programación del NIOS II .....	9
2.2	Tarjeta Altera DE2 .....	10
2.2.1	Comunicación GSM .....	12
2.2.1.1	Introducción Redes GSM.....	12
2.2.1.2	Descripción de los Sistemas Celulares.....	13
2.2.1.3	Arquitectura de una Red GSM.....	13
2.2.1.4	Roaming y Hand-Over .....	15
2.2.1.5	SMS.....	16
2.2.1.6	Funcionamiento del SMSC .....	18
2.3	Switches Magnéticos de Apertura de Puertas .....	19
2.4	Norma RS232 .....	20
2.4.1	Conector RS232D- Conector DB9 de 9 pines.....	21
3.	DISEÑO E IMPLEMENTACIÓN.....	22
3.1	Diseño del Hardware en SOPC Builder .....	24
3.2	Interfaz Paralela de Entrada y Salida PIO_0.....	26

3.2.1 Descripción de Pines Usados para PIO_0 .....	27
3.3 Interfaz Serial RS-232.....	28
3.4 Programación en Lenguaje C usando Software Nios II .....	29
3.5 Bloque de Switches Magnéticos .....	33
3.6 Bloque Convertidor de Voltaje con conector DB9 .....	33
3.7 Módulo SIM 340CZ .....	35
3.7.1 Especificaciones del Módulo GSM.....	36
3.7.2 Descripción de pines utilizados del Módulo GSM .....	37
3.8 Comunicación entre los Switches Magnéticos, FPGA, Módulo GSM y celular.....	37
4. PRUEBAS y Resultados .....	40
4.1 Escenarios .....	40
4.1.1 Escenario A: Cuando no hay señal para la comunicación .....	41
4.1.2 Escenario B: Cuando el chip SIM no tiene saldo para enviar mensajes	42
4.1.3 Escenario C: Comunicación y envío de mensajes exitoso.....	43

## CONCLUSIONES Y RECOMENDACIONES

## ANEXOS

## REFERENCIAS BIBLIOGRÁFICAS

# ÍNDICE DE FIGURAS

Figura 2-1 Sistema Microprocesador basado en NIOS II.....	6
Figura 2-2 Versiones de NIOS II [1] .....	8
Figura 2-3 Diagrama de Bloques de la Arquitectura NIOS II [2].....	9
Figura 2-4 Ejemplo código en C y en Assembler [2] .....	10
Figura 2-5 Tarjeta DE2 ALTERA [3].....	11
Figura 2-6 Esquema de la Comunicación GSM [4] .....	12
Figura 2-7 Arquitectura de una Red GSM [4].....	13
Figura 2-8 Ejemplo de Hand-over [5] .....	16
Figura 2-9 SMS [6] .....	18
Figura 2-10 Switches Magnéticos de Apertura de Puerta [7] .....	20
Figura 2-11 Conector DB9 [8] .....	21
Figura 3-1 Diagrama de Bloques Simplificado .....	22
Figura 3-2 Diagrama de Bloque General .....	23
Figura 3-3 Diseño del sistema en SOPC Builder .....	26

Figura 3-4 Diseño de pio_0 en SOPC Builder.....	27
Figura 3-5 Puerto Serial RS-232.....	28
Figura 3-6 Código Fuente Principal del Main. ....	30
Figura 3-7 Función para el envío de Mensaje.....	31
Figura 3-8 Sentencia Case usada en el Proyecto.....	32
Figura 3-9 Convertidor de Voltajes.....	34
Figura 3-10 Conexión del conector DB9 entre el módulo GSM y tarjetaDE2....	35
Figura 3-11 Módulo GSM [9].....	35
Figura 3-12 Switch Magnético.....	38
Figura 3-13 Esquema de la Comunicación del Proyecto .....	39
Figura 4-1 No hay señal en el Modulo GSM .....	42
Figura 4-2 Chip no tiene saldo para enviar mensaje.....	43
Figura 4-3 Switch Magnético cerrado.....	44
Figura 4-4 Switch Magnético abierto.....	44
Figura 4-5 Primer Led Verde encendido .....	45
Figura 4-6 Mensaje que se muestra en el Access Port.....	46

Figura 4-7 Mensaje del Switch 1 en el Celular .....	46
Figura 4-8 Segundo Led Verde encendido .....	47
Figura 4-9 Mensaje del Segundo Switch.....	48
Figura 4-10 Mensaje del Switch 2 en el celular.....	48
Figura 4-11 Tercer Led Verde encendido .....	49
Figura 4-12 Mensaje del Tercer Switch.....	49
Figura 4-13 Mensaje del Switch 3 en el celular.....	50
Figura 4-14 Cuarto Led Verde encendido .....	50
Figura 4-15 Mensaje del Cuarto Switch .....	51
Figura 4-16 Mensaje del Switch 4 en el celular.....	51

# ÍNDICE DE TABLAS

Tabla 2-1 Descripción de pines conector DB9 .....	21
Tabla 3-1 Descripción de pines en Cyclone II EP2C35F672C6 .....	27
Tabla 3-2 Características Técnicas del Switch Magnético .....	33
Tabla 3-3 Pines Modulo GSM SIM340CZ .....	37
Tabla 4-1 Parámetros de comando AT+CSQ .....	41

# ABREVIATURAS

AuC	Authentication Center
BSC	Base Station Controller
BSS	Base Station Subsystem
BTS	Base Transceiver Station
CAD	Computer-aided design
CLK	Clock
CMOS	Complementary metal–oxide–semiconductor
DCE	Equipo para la Comunicación de Datos
DIP	Dual in-line package
DTE	Equipo Terminal de Datos
FPGA	Field Programmable Gate Array
GIWU	GSM Interworking Unit
GMSC	Gateway Mobile Services Switching Center



GPS	Global Positioning System
GSM	Global System for Mobile Communications
GUI	Graphical User Interfaces
GUIDE	Graphical User Interface Development Environment
HDL	Hardware Description Language
HLR	Home Location Register
ISDN	Integrated Services Digital Network
IMEI	International Mobile Equipment Identity
MS	Mobile Station
MMU	Memory Management Unit
MSC	Mobile Services Switching Center
NSS	Network and Switching Subsystem
OSS	Operation and Support Subsystem
PCB	Printed Circuit Board
PIN	Personal Identification Number
RDSI	Red Digital de Servicios Integrados

RISC	Reduced Instruction Set Computer
RS-232	Recommended Standard 232
RX	Línea de Recepción
SIM	Subscriber Identity Module
SMS	Short Message Service
SMSC	Short Message Service Center
SOC	System On-Chip
SOPC	System on a Programmable Chip
TTL	Transistor-Transistor Logic
TX	Línea de Transmisión
UART	Universal Asynchronous Receiver- Transmitter
VLR	Visitor Location Register

# INTRODUCCIÓN

La tecnología va evolucionando, con ello ha dado muchas mejoras en todo el campo de la electrónica digital. Se han desarrollado miles de sistemas electrónicos basados en comunicaciones con distintas aplicaciones, pero con un solo objetivo, hacer más fáciles las cosas y mejorar la calidad de vida de las personas que los usan.

Dentro de las aplicaciones, están los sistemas de seguridad y de manera específica los sistemas de alarma, que surgieron para resolver la necesidad de alertar y controlar el ingreso de personas no autorizadas en algún lugar determinado, manteniendo así la seguridad de casas, oficinas, empresas, etc.

Una de las tecnologías que podemos usar dentro de la implementación de cualquier diseño lógico, es la FPGA (Arreglo de Puertas Programables por Campos) la cual requiere la configuración o programación por parte del usuario para realizar la operación deseada.

En nuestro proyecto utilizamos dicha tecnología junto con la comunicación GSM para transmitir un mensaje a un celular, de esta manera realizaremos una demostración de las diversas aplicaciones que se pueden implementar en un procesador embebido.

# CAPÍTULO 1

## 1. GENERALIDADES

En este capítulo detallamos de manera general el planteamiento del proyecto, así como sus alcances y limitaciones, explicando la razón de esta implementación y la solución propuesta al problema planteado.

### 1.1 Alcance y limitaciones del proyecto

El proyecto debe cumplir las siguientes características:

- Mediante el módulo GSM únicamente se enviará mensajes hacia un celular.
- Se utilizan cuatro switches magnéticos de apertura de puerta los cuales se conectan a la tarjeta y mediante la programación en lenguaje C debe reconocer que switch es activado para enviar el respectivo mensaje hacia el celular.
- La implementación se la hará solo en ambiente de laboratorio.

## **1.2 Objetivos**

### **1.2.1 Objetivos Generales**

Los objetivos generales para llevar a cabo el trabajo son los siguientes:

Comprender conceptos básicos sobre FPGA, Comunicación GSM, Switches Magnéticos, Sistema de Seguridad.

### **1.2.2 Objetivo Específico**

El objetivo principal de este proyecto es la comunicación por medio de la red GSM hacia un procesador embebido configurable Nios II, para esto desarrollamos un programa en lenguaje C el cual es cargado en la FPGA teniendo así un diseño de un sistema de seguridad que permita al usuario tener el control de un lugar sin necesidad de estar presente, para que por medio de un mensaje que le llegue al celular tomar las debidas precauciones sobre el lugar afectado, todo esto en un ambiente de laboratorio.

Entre estos objetivos, nos enfocáremos en los siguientes:

- Aprender acerca del funcionamiento de la tarjeta de altera DE2, sus características principales y funcionamiento.
- Aprender a usar los programas Quartus II y NIOS II para Eclipse los cuales programarán la tarjeta DE2.

- Conocer un poco más acerca de los Sistemas de Seguridad que existen en el mercado.

### **1.3 Antecedentes**

En los últimos años la inseguridad en nuestro país, ha sido un tema delicado ya que la delincuencia cada vez va en aumento y nuestra sociedad tiene que tomar precauciones para poder frenar la misma. Actualmente las empresas ya sean privadas o públicas así como las viviendas domiciliarias deben tener medidas muy altas de seguridad para el desarrollo de sus actividades.

Con el tiempo la tecnología ha sido tomada muy en cuenta con respecto a la seguridad ya que la misma permite mayor control, fácil uso, información instantánea, detallada y segura de lo que está ocurriendo en algún lugar en específico, ahorrando costos y mejorando la calidad de vida de las personas. Observamos también que la tecnología implementada a la seguridad cada vez va evolucionando, siendo más eficaz. Con estas opciones las personas se preocupan menos de este tema facilitando su normal desarrollo, dicho esto en nuestro proyecto propondremos un método de seguridad utilizando la tecnología FPGA, para poder cumplir las necesidades que nuestra sociedad necesita.

En este proyecto se utiliza una tarjeta DE2 de tecnología FPGA cuatro switches magnéticos, un módulo GSM y un celular.

## **1.4 Identificación del Problema**

Los sistemas de vigilancia, deben estar disponibles las 24 horas del día con el uso de la tecnología se lo puede lograr sin mayor problema. Ya que existen varios sistemas de seguridad como por ejemplo las cámaras de seguridad, las cuales graban en todo momento lo que ocurre en un lugar y si existe algún problema se puede identificar a las personas involucradas en lo sucedido pero este quedaría solo en grabación y luego de algunos estudios realizados en la grabación del video se podrían tomar acciones.

Los usuarios en la actualidad necesitan actuar rápidamente si existe algún inconveniente en un lugar determinado, requieren informarse en todo momento para poder tomar acciones que puedan tratar de enfrentar y contrarrestar el agravio que pueda suceder.

Además el tipo de seguridad como el mencionado anteriormente tiene costos altos y mientras se pueda economizar siempre será una buena opción, el tipo de seguridad que implementaremos será más accesible a los usuarios en comparación a otros sistemas de seguridad, la información se enviará de manera rápida y directamente hacia un celular vía mensajes de texto. Pero muchas empresas han tomado medidas de seguridad y el mercado cada vez está más saturado, pero día a día las ciudades van creciendo y se van construyendo más conjuntos residenciales pero estos necesitan mayor seguridad en cada una de las casas ya que si una no está

habitada, el dueño de la misma debería estar seguro que nadie la invadirá, con este proyecto se puede brindar esta solución dando la información de lo que sucede en dicho lugar a la persona propietaria de la casa.

### **1.5 Descripción breve de la posible solución**

Debido a que en distintos conjuntos residenciales no presentan la seguridad necesaria, nuestra solución es ofrecer, al momento de no estar nadie habitando la misma, un sistema rápido y efectivo donde tendrá la certeza de que todo se encontrará tal y como lo dejó, en condiciones normales, sin ninguna anomalía ni que nadie esté presente en su hogar sustrayendo sus pertenencias, para esto utilizaremos unos switches magnéticos de apertura de puerta que estarán en distintos puntos de acceso al hogar, al momento de ser activados estos switches envían una señal, el cual es un valor lógico alto de voltaje, la tarjeta recepta ese dato y con los dispositivos necesarios que en nuestro caso estará implementado en una tarjeta DE2, reconocerá que switch fue activado para poder enviar el mensaje de texto al dueño de la casa, este mensaje de texto se transmitirá por medio de la red GSM, por medio de un módulo SIM GSM.

La solución propuesta se basa en la programación que tendrá esta tarjeta para que pueda funcionar correctamente en conjunto con los switches magnéticos que se vayan activando.



# CAPÍTULO 2

## 2. MARCO TEÓRICO

En este capítulo se presentan los principales conceptos que utilizamos en este proyecto, para de esta manera familiarizarnos con cada uno de ellos.

### 2.1 Introducción al Microprocesador NIOS II

El fabricante Altera proporciona una infraestructura completa para crear sistemas de microprocesador embebido, según las necesidades del diseñador, por medio de la combinación de una serie de componentes configurables sobre sus FPGAs. Para desarrollar Sistemas embebidos, Altera proporciona un entorno de desarrollo al que denomina SOPC Builder, que permite la configuración a medida del sistema microprocesador Nios II, se muestra en la figura; y que gracias a la herramienta de síntesis Quartus II puede ser implementado directamente sobre una FPGA.

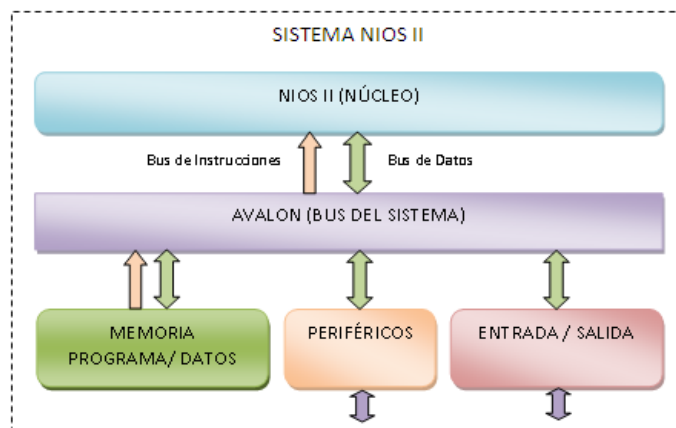


Figura 2-1 Sistema Microprocesador basado en NIOS II

Este sistema microprocesador está compuesto por: el núcleo procesador Nios II, memoria interna de programa y de datos, periféricos integrados e interfaces para memoria externa y/o entrada/salida.

### **2.1.1 Características y Arquitectura**

El Nios II es un núcleo procesador configurable que se puede implementar en alguna de las tres versiones disponibles, ver Figura 2-2, según se busque minimizar el consumo de recursos de la FPGA o maximizar el rendimiento del procesador:

- El Nios II/f (“fast”) es la versión diseñada para alto rendimiento, y que proporciona opciones para aumentar su desempeño, como memorias caché de instrucciones y datos, o una unidad de manejo de memoria (MMU, Memory Management Unit).
- El Nios II/s (“standard”) es la versión que contiene la unidad aritmético lógica (ALU, Arithmetic Logic Unit) y busca combinar rendimiento y consumo de recursos.
- El Nios II/e (“economy”) es la versión que requiere menos recursos de la FPGA, es muy limitada, dado que carece de las operaciones de multiplicación y división.

Core Nios II			
Select a Nios II core:			
	<input type="radio"/> Nios II/e	<input checked="" type="radio"/> Nios II/s	<input type="radio"/> Nios II/f
<b>Nios II</b>	RISC 32-bit	RISC 32-bit <b>Instruction Cache</b> <b>Branch Prediction</b> <b>Hardware Multiply</b> <b>Hardware Divide</b>	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide <b>Barrel Shifter</b> <b>Data Cache</b> <b>Dynamic Branch Prediction</b>
Selector Guide			
Family: Cyclone II			
f <sub>system</sub> : 50,0 MHz			
cpuid: 0			
Performance at 50,0 MHz	Up to 5 DMIPS	Up to 25 DMIPS	Up to 51 DMIPS
Logic Usage	600-700 LEs	1200-1400 LEs	1400-1800 LEs
Memory Usage	Two M4Ks (or equiv.)	Two M4Ks + cache	Three M4Ks + cache

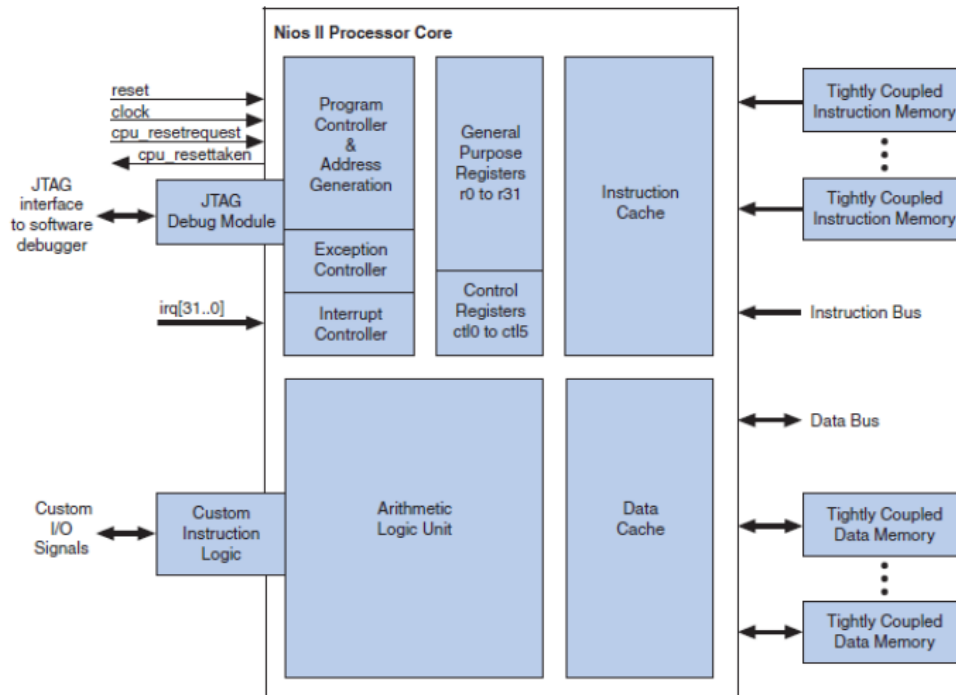
**Figura 2-2 Versiones de NIOS II [1]**

Cada una de estas versiones se completa con una serie de componentes, memoria y periféricos, por medio de su interconexión a través de un bus de sistema al que denominan “Avalon Switch Fabric”, para obtener un sistema Nios II completo en un chip (SOC, System On Chip), según se muestra en la Figura 2-3.

El Nios II es un procesador de 32 bits de propósito general, basado en una arquitectura tipo Harvard, dado que usa buses separados para instrucciones y datos; y cuyas principales características son:

- Tamaño de palabra asignado de 32 bits.
- Juego de instrucciones RISC de 32 bits.
- 32 registros de propósito general de 32 bits (r0 – r31).
- 6 registros de control de 32 bits (ctl0 - ctl5).
- 32 fuentes de interrupción externa.
- Capacidad de direccionamiento de 32 bits.
- Operaciones de multiplicación y división de 32 bits.
- Instrucciones dedicadas para multiplicaciones de 64 y 128 bits.

- Instrucciones para operaciones de coma flotante en precisión simple.
- Acceso a variedad de periféricos integrados e interfaces para manejo de memorias y periféricos.



**Figura 2-3 Diagrama de Bloques de la Arquitectura NIOS II [2]**

### 2.1.2 Programación del NIOS II

Para la programación del Nios II se dispone de una herramienta basado en Eclipse, con el que Altera da soporte para el desarrollo de aplicaciones en C/C++. Nios II también permite que se realice programación en lenguaje ensamblador (“assembler”) como muestra la Figura 2-4.

El carácter configurable del Nios II exige que en todo programa se presente una primera etapa, la cual consiste en definir todos y cada uno de los periféricos del sistema, a los que se les asigna una dirección de memoria.

```

# Refleja la posición de los interruptores
# a través del estado de los indicadores

.include "nios_macros.s"

.equ      pi8, 0x00003000 # interruptores
.equ      po8, 0x00003010 # indicadores

.global _start
_start:
    movia r2, pi8
    movia r3, po8
loop:
    ldbio r4, 0(r2)
    stbio r4, 0(r3)
    br loop

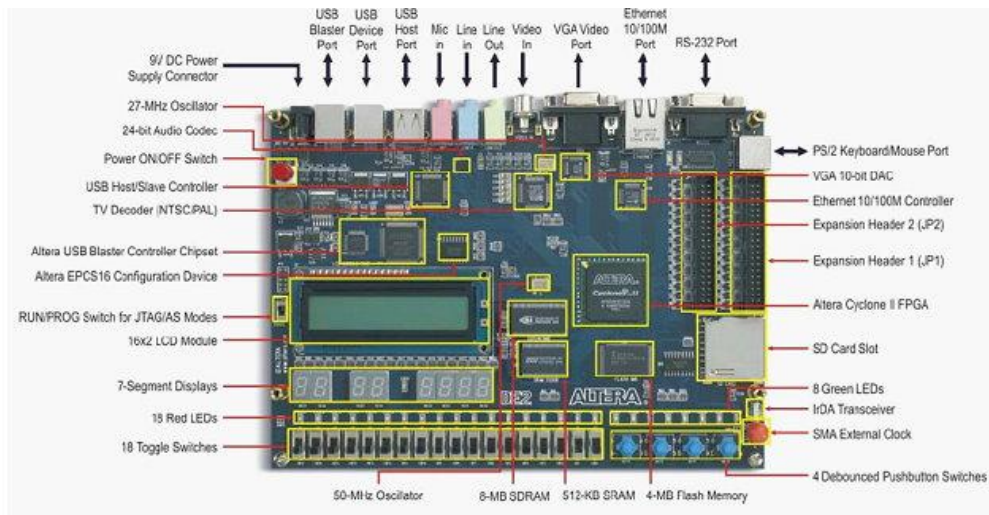
# // Programa equivalente en C
# // Refleja la posición de los interruptores (Switches)
# // a través de los indicadores (LEDs)
#
# #define Switches (volatile char *) 0x00003000
# #define LEDs (char *) 0x00003010
#
# void main() {
#     while (1)
#         *LEDs = *Switches;
# }

```

**Figura 2-4 Ejemplo código en C y en Assembler [2]**

## 2.2 Tarjeta Altera DE2

La tarjeta DE2 de Altera, ver Figura 2-5, permite realizar el aprendizaje de la lógica digital, la organización de computadores, y FPGA's. Utiliza tecnología de última generación tanto en hardware y herramientas CAD para exponer a los estudiantes y profesionales una amplia gama de temas. La tarjeta ofrece un rico conjunto de características que lo hacen apto para su uso en un entorno de laboratorio para los cursos de la universidad, para una gran variedad de diseños y proyectos, así como para el desarrollo de sofisticados sistemas digitales.



**Figura 2-5 Tarjeta DE2 de ALTERA [3]**

La tarjeta DE2 se basa en la Familia Lógica Programable FPGA Cyclone II 2C35. Todos los componentes de la tarjeta están conectados a los pines de este chip, permitiendo al usuario controlar todos los aspectos para la operación de la tarjeta.

La tarjeta DE2 contiene:

- Altera Cyclone II 2C35 FPGA.
- Dispositivos de Configuración Serial (EPCS16) para Cyclone II 2C35.
- USB Blaster para configurar la FPGA.
- Memorias de 8Mbyte (1M x 4 x 16) SDRAM.
- 1Mbyte de Memoria Flash (expandible a 4Mbyte).
- Socket para tarjetas SD.
- 4 Botoneras, 18 DPDT switches.
- 9 LEDs Verdes y 18 LEDs Rojos.
- Clocks: 50MHz y 27MHz.
- Línea In/Out, Micrófono (24-bit Audio CODEC).
- 10/100 Controlador Ethernet con socket.
- USB Host/esclavo con USB tipo A y tipo B.

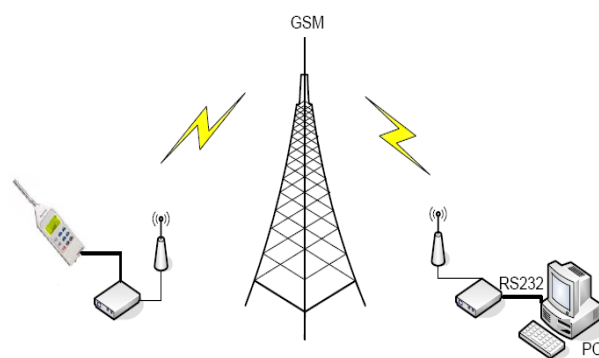
- Interfaz RS-232 con conector de 9-pines.
- Conector PS/2 para el teclado o mouse.
- 2 Cabeceras de Expansión con 40-pines con protección de diodos.

### 2.2.1 Comunicación GSM

A continuación una breve introducción en la comunicación con redes GSM que es la base de este proyecto.

#### 2.2.1.1 Introducción Redes GSM

GSM es un sistema de comunicación basado en el uso de células digitales que se desarrolló para crear un sistema para móviles único que sirviese de estándar para Europa y que fuese compatible con los servicios existentes y futuros sobre ISDN o RDSI. Actualmente GSM es un estándar que se usa a nivel mundial y cada día se van desarrollando todo tipo de dispositivos que integran esta tecnología. A continuación un esquema de la red GSM, ver Figura 2-6.



**Figura 2-6 Esquema de la Comunicación GSM [4]**

Las tres categorías de servicios sobre GSM son: teleservicios que englobaría a los servicios básicos de telefonía; los servicios portadores que son los usados para la transmisión y recepción de datos; y los servicios complementarios

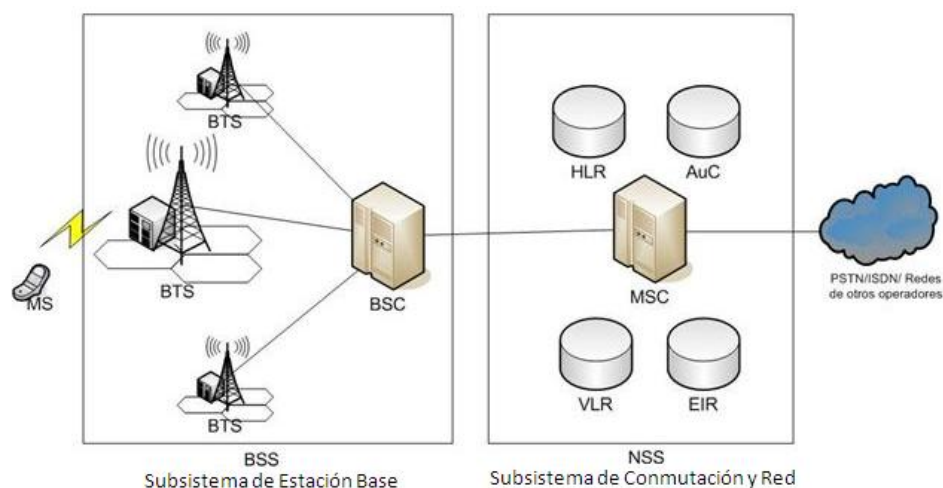
generalmente extensiones de los teleservicios y que proporcionan nuevas características a la red GSM.

### 2.2.1.2 Descripción de los Sistemas Celulares

Como ya hemos visto anteriormente GSM es un sistema basado en células de radio, pero para entender esto tenemos que entrar en otros conceptos. Los sistemas celulares se basan en la división del área de cobertura de un operador en lo que se denomina células, estas células se caracterizan por su tamaño que viene dado por la potencia de transmisión, ya que lo que se persigue en estos casos es que dicha potencia sea lo más baja posible para poder utilizar el mayor número de frecuencias, esto se debe al hecho de que al mayor número de frecuencias libres mayor es el número de usuarios que pueden hacer uso del sistema ya que cada uno puede usar una frecuencia sin interferir en la de otro usuario.

### 2.2.1.3 Arquitectura de una Red GSM

Todas las redes GSM se pueden dividir en cuatro partes fundamentales y bien diferenciadas que a continuación se muestra en la Figura 2-7:



**Figura 2-7 Arquitectura de una Red GSM [4]**



Primero describiremos la Estación Móvil la cual consta a su vez de dos elementos básicos que debemos conocer, por un lado el terminal o equipo móvil y por otro lado el SIM o Subscriber Identity Module. El SIM es una pequeña tarjeta inteligente que sirve para identificar las características de nuestro terminal, el SIM está protegido por un número de cuatro dígitos que recibe el nombre de PIN o Personal Identification Number. Una vez que se introduce el PIN en nuestro terminal, el terminal busca redes GSM que estén disponibles, generalmente la red que tenemos contratada. Una vez que la red valida nuestro terminal, el teléfono queda registrado en la célula que lo validó. Otro componente fundamental es la Estación Base, el cual sirve para conectar a las estaciones móviles con los NSS, además de ser los encargados de la transmisión y recepción. Está formada por dos elementos, el BTS y el BSC; el BTS que consta de transceivers y antenas usadas en cada célula de la red, generalmente su potencia de transmisión determina el tamaño de la célula, y los BSC que se utilizan para controlar a los BTS y tienen como función principal la de estar a cargo de los hand-overs, y los controles de frecuencia de radio de los BTS.

El Subsistema de Conmutación y Red o Network and Switching Subsystem este sistema se encarga de administrar las comunicaciones que se realizan entre los diferentes usuarios de la red, para poder hacer este trabajo la NSS se divide en siete subsistemas diferentes, cada uno con una misión dentro de la red:

- ❖ MSC: Es el componente central del NSS y se encarga de realizar las labores de conmutación dentro de la red, así como de proporcionar conexión con otras redes.
- ❖ GMSC: La misión del GMSC es de servir de mediador entre las redes de telefonía fijas y la red GSM.

- ❖ El HLR es una base de datos que contiene información sobre los usuarios conectados a un determinado MSC. Entre la información que almacena el HLR tenemos fundamentalmente la localización del usuario y los servicios a los que tiene acceso.
- ❖ VLR: contiene toda la información sobre un usuario necesaria para que dicho usuario acceda a los servicios de red. Forma parte del HLR con quien comparte funcionalidad.
- ❖ AuC: Proporciona los parámetros necesarios para la autenticación de usuarios dentro de la red; también se encarga de soportar funciones de encriptación.
- ❖ EIR: También se utiliza para proporcionar seguridad en las redes GSM pero a nivel de equipos válidos. La EIR contiene una base de datos con todos los terminales que son válidos para ser usados en la red. Esta base de datos contiene los IMEI de cada terminal, de manera que si un determinado móvil trata de hacer uso de la red y su IMEI no se encuentra localizado en la base de datos del EIR no puede hacer uso de la red.
- ❖ GIWU: sirve como interfaz de comunicación entre diferentes redes para comunicación de datos.

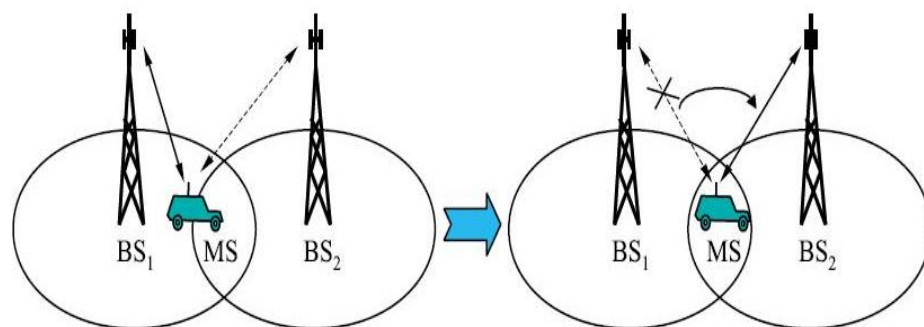
Otra parte fundamental dentro de una arquitectura de red GSM son los Subsistemas de Soporte y Operación los cuales se conectan a diferentes NSS y BSC, para controlar y monitorizar toda la red GSM.

#### **2.2.1.4 Roaming y Hand-Over**

El roaming se produce cuando nos estamos validando dentro de la red GSM y el terminal no es capaz de encontrar la red en la cual somos clientes; esto pasa fundamentalmente cuando salimos de viaje al extranjero, donde existe la red, pero no es la de nuestro operador; en este caso, el roaming consiste

en la utilización de la red que se encuentre disponible y con la que nuestro operador tiene un acuerdo de colaboración. De este modo, podemos seguir conectados con nuestro móvil a la red independientemente de que estemos fuera del alcance de nuestro operador habitual.

El concepto de hand-over consiste en la transición que se produce cuando pasamos del rango de acción de una célula al rango de acción de otra. Esto se produce sobre todo cuando viajamos. El hand-over, por tanto, es el responsable de mantener el servicio de manera constante y de que las transiciones entre una célula y otra sean lo suficientemente pequeñas como para pasar desapercibidas por los usuarios.



**Figura 2-8 Ejemplo de Hand-over [5]**

#### 2.2.1.5 SMS

Short Message Service es un protocolo de comunicación que permite el intercambio de mensajes de textos entre teléfonos móviles. SMS fue diseñado originalmente como un servicio de GSM, pero en la actualidad se encuentra disponible en otras redes móviles incluyendo redes 3G.

El tamaño máximo de un mensaje de texto en SMS es de 160 caracteres de 7 bits, 140 caracteres de 8 bits o de 70 caracteres de 16 bits, este último en idiomas como el chino, coreano, japonés, ruso y arábigo. Existe también el SMS segmentado, el cual es un mensaje que sobrepasó su máximo de

caracteres pero es enviado en múltiples partes. Esto lo logra ya que cada segmento iniciará con una Cabecera de Dato de Usuario o UDH la cual contiene información de su segmento respectivo y la longitud de cada segmento ahora será de 153 caracteres de 7 bits, 134 caracteres de 8 bits y 67 caracteres de 16 bits.

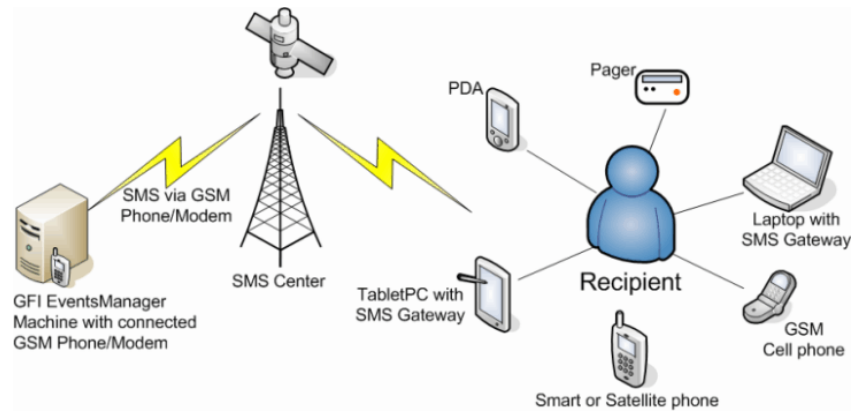
El dispositivo móvil es el encargado de recibir cada segmento y reensamblar el mensaje para poder ser mostrado al usuario como un mensaje de texto largo.

Muchos transceiver y teléfonos móviles tienen soporte para enviar y recibir mensajes de texto usando comandos AT, para estos poder ser ejecutados, el móvil o el transceiver debe ser conectado a una PC a través de una conexión Serial, ya sea por cable o una virtual como es el caso que crean los dispositivos Bluetooth. Por ejemplo [11].

Comandos usados para enviar mensajes a un celular con comandos AT:

- Formato nivel de señal óptimo: AT+CSQ (0 a 31 nivel bueno de señal, 99 no hay señal)
- Formato de mensaje: AT+CMGF="1" (0 tipo binario, 1 tipo texto).
- Envío de mensaje: AT+CMGS="número de móvil".
- Lectura de mensajes: AT+CMGR="0" (0 todos los mensajes).
- El SMS es también usado para comunicaciones M2M, dispositivos, localización vehicular, proyectos de telemetría, etc.

Los mensajes de texto son procesados por un centro de mensajes cortos llamado SMSC; que se encarga de almacenarlos hasta que son enviados; y de conectar con el resto de elementos de la red GSM.



**Figura 2-9 SMS [6]**

### 2.2.1.6 Funcionamiento del SMSC

Para la correcta gestión de los mensajes SMS se hace necesario introducir en el sistema GSM un nuevo elemento: el SMSC.

El SMSC consulta al HLR cuando sea necesario. El HLR guarda información sobre si el abonado puede enviar y recibir SMS; la dirección del SMSC está almacenada en el terminal como: Centro de Servicios; cualquier persona puede meterse en las características del teléfono móvil, y revisar esa dirección.

El SMSC pregunta al HLR sobre el MSC/VLR donde se encuentra localizado en ese momento el usuario.

El HLR responde al SMSC sobre el MSC en el que se encuentra actualmente el usuario y el SMS es entregado al destinatario.

Si el usuario ha solicitado acuse de recibo del mensaje, este es generado y enviado a través del SMSC al usuario que ha enviado el SMS.

Si el usuario destinatario, que puede encontrarse en cualquier red con la que el operador tenga acuerdo, no está activo cuando se envía el mensaje,

queda pendiente su recepción y se almacena una marca en el HLR para indicar que está pendiente de recibir un SMS. Una vez que dicho usuario vuelve a estar activo; el SMS es entregado y la marca en el HLR es reseteada.

Las funciones del SMSC son:

- Recibir y almacenar los mensajes cortos enviados por los usuarios o por otras fuentes (avisos del operador, buzón de voz, sistemas de publicidad), hasta que puedan ser enviados.
- Verificar los permisos para enviar mensajes, en comunicación con el HLR de origen.
- Verificar si el usuario al que se envía el mensaje está operativo o no, mediante consulta al VLR de destino; si está operativo, el mensaje se envía, y si no se almacena temporalmente en el SMSC.
- Verificar periódicamente el estado de los usuarios que tienen mensajes pendientes.

### **2.3 Switches Magnéticos de Apertura de Puertas**

El sensor magnético actúa como un interruptor (0-1), ideal para controlar puertas y ventanas. Su funcionamiento es básico, cuando los terminales (a y b) se encuentran juntos se produce un cortocircuito enviando un nivel de bajo voltaje (cero lógico); si los terminales se encuentran separados entonces se produce un nivel alto de voltaje (uno lógico) ver Figura 2-10.



**Figura 2-10 Switches Magnéticos de Apertura de Puerta [7]**

## **2.4 Norma RS232**

El protocolo RS-232 es una norma o estándar mundial que rige los parámetros de uno de los modos de comunicación serial. Por medio de este protocolo se estandarizan las velocidades de transferencia de datos, la forma de control que utiliza dicha transferencia, los niveles de voltajes utilizados, el tipo de cable permitido, las distancias entre equipos, los conectores, etc.

Además de las líneas de transmisión (Tx) y recepción (Rx), las comunicaciones seriales poseen otras líneas de control de flujo (Hand-shake), donde su uso es opcional dependiendo del dispositivo a conectar.

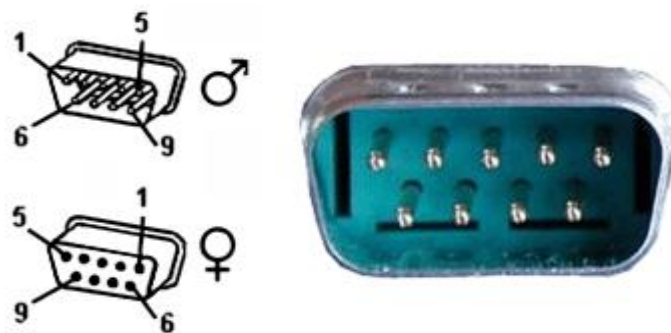
A nivel de software, la configuración principal que se debe dar a una conexión a través de puertos seriales. RS-232 es básicamente la selección de la velocidad en baudios (1200, 2400, 4800, 9200, etc.), la verificación de datos o paridad (paridad par o paridad impar o sin paridad), los bits de parada luego de cada dato (1 ó 2), y la cantidad de bits por dato (7 ó 8), que se utiliza para cada símbolo o carácter enviado.

La Norma RS-232 fue definida para conectar un ordenador a un modem. Además de transmitirse los datos de una forma serie asíncrona para esto son necesarias

una serie de señales adicionales, que se definen en la norma. Las tensiones empleadas están comprendidas entre +15/-15 voltios.

#### 2.4.1 Conector RS232D- Conector DB9 de 9 pines

La Figura 2-11 muestra el conector de 9 pines serie que se encuentra en la mayoría de las PC de hoy en día. A continuación en la tabla se muestra la descripción de todos los pines.



**Figura 2-11 Conector DB9 [8]**

PIN	NOMBRE	DESCRIPCIÓN
1	CD	Detección de Portadora
2	RXD	Recepción de Datos
3	TXD	Transmisión de Datos
4	DTR	Terminal de Datos Preparado
5	GND	Señal de Tierra
6	DSR	Dispositivo Preparado
7	RTS	Petición de Envío
8	CTS	Preparado para Transmitir
9	RI	Indicador de Llamada Entrante

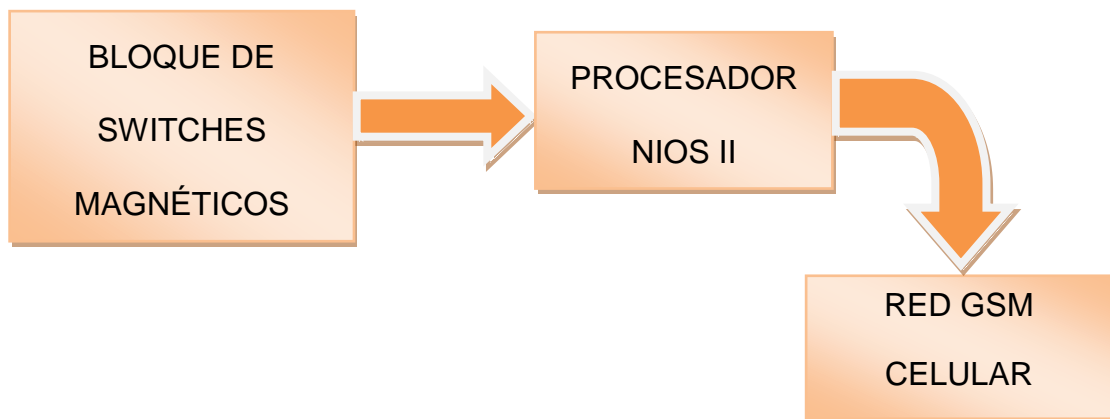
**Tabla 2-1 Descripción de pines conector DB9**



# CAPÍTULO 3

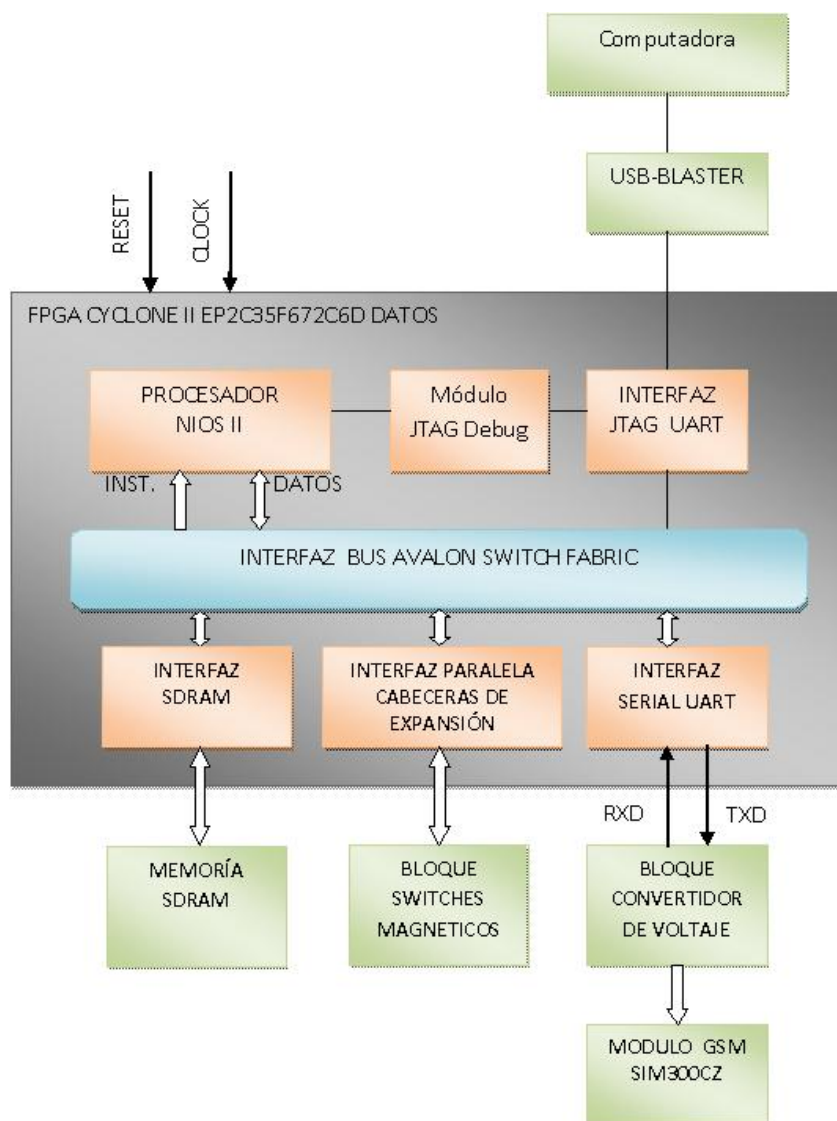
## 3. DISEÑO E IMPLEMENTACIÓN

En este capítulo se procederá con el diseño e implementación del proyecto. Para la explicación se ha elaborado un diagrama de bloques simplificado así como un diagrama de bloques general. En la Figura 3-1 podemos apreciar el diagrama de bloques simplificado, el cual resume el funcionamiento básico del proyecto.



**Figura 3-1 Diagrama de Bloques Simplificado**

En este diagrama se puede observar la interacción entre los switches magnéticos junto con la tarjeta DE2, la cual contiene al procesador Nios II, y a su vez la comunicación con la red celular.



**Figura 3-2 Diagrama de Bloque General**

La Figura 3-2 nos muestra con más detalle el diseño del proyecto cuya finalidad es enviar mensajes de texto vía celular cuando un switch magnético de apertura de puerta se active o en nuestro caso cuando alguien haya invadido la casa. Por tal motivo se desarrolló un código en lenguaje C el cual fue compilado y ejecutado usando el software Nios II para Eclipse, más adelante se explicará con detalle. Esta información se almacena temporalmente en una memoria SDRAM de la tarjeta altera DE2, básicamente el bloque de switches magnéticos se encuentra asignado en una de las cabeceras de expansión de cuarenta pines, que se encuentra en la tarjeta. Después se envía el mensaje a transmitir escrito en comandos AT, a través del puerto serial UART, pasa por el bloque convertidor de voltajes, para nivelar los voltajes de salida para que pueda comunicarse con el modulo GSM SIM 340CZ y finalmente llega el mensaje al celular.

### **3.1 Diseño del Hardware en SOPC Builder**

El Software SOPC Builder propio de Altera, es usado para implementar sistemas que usan el procesador Nios II en dispositivos FPGA, este software se usa en conjunto con Quartus II para implementar sistemas simples y fáciles de usar.

Nios II es un software y un procesador propio de Altera, el cual es definido en lenguaje de descripción de hardware, el mismo que puede ser implementado en dispositivos FPGA de Altera usando Quartus II. Para

implementar un sistema útil de mucho beneficio, es necesario añadir otras unidades funcionales por ejemplo memorias, interfaces de entrada/salida, interfaces de comunicación.

Para facilitar el diseño de tales sistemas, es necesario tener un Software de un diseño asistido por computadora (CAD) para implementar un sistema en un chip programable. SOPC Builder es el software usado para esta tarea.

En la Figura 3-3 mostramos el diseño de nuestro sistema en SOPC Builder, y a continuación describiremos algunas partes importantes de este sistema:

- El Procesador Nios II que es referido como Unidad de Procesador Central (CPU).
- Una interfaz paralela de entrada/salida que llamamos como pio\_0.
- interfaz de comunicación con el computador JTAG UART.
- Interfaz UART, puerto serial RS232, para la comunicación con el módulo GSM de nuestro proyecto.

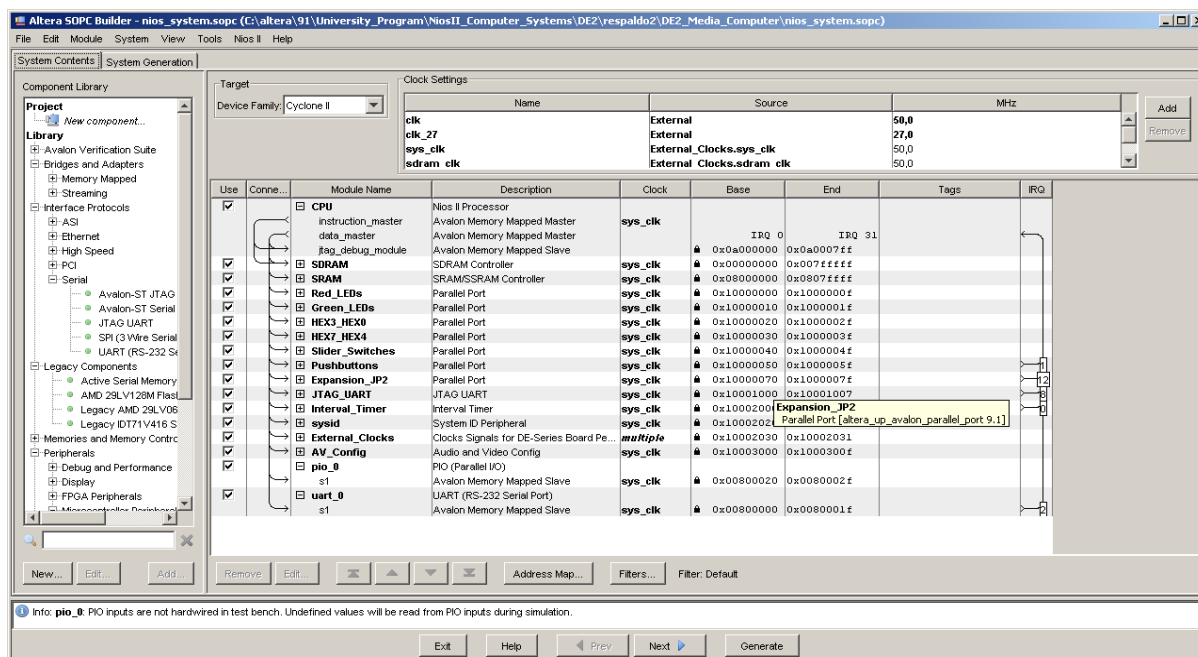
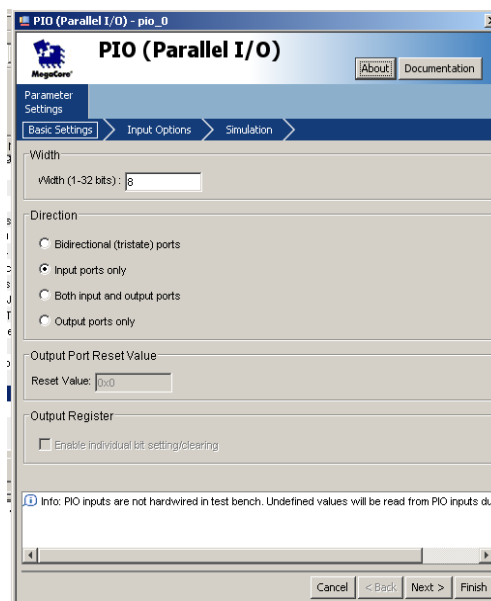


Figura 3-3 Diseño del sistema en SOPC Builder

### 3.2 Interfaz Paralela de Entrada y Salida PIO\_0

Este tipo de interfaz se creó para ser usado como entrada para los switches magnéticos, normalmente este tipo de tarjetas DE2 viene con dos interfaces paralela de 40 pines llamado Expansion\_JP1 y Expansion\_JP2, pero en este caso usaremos solo uno de ellos, aunque nuestra interfaz pio\_0 consta en total de 8 pines los cuales serán usados por los switches magnéticos, sin embargo para efectos de simulación del proyecto se realizó solo con cuatro switches magnéticos que identifica cuatro pines. La Figura 3-4 muestra el diseño de la interfaz paralela de ocho pines en SOPC Builder.



**Figura 3-4 Diseño de pio\_0 en SOPC Builder**

### 3.2.1 Descripción de Pines Usados para PIO\_0

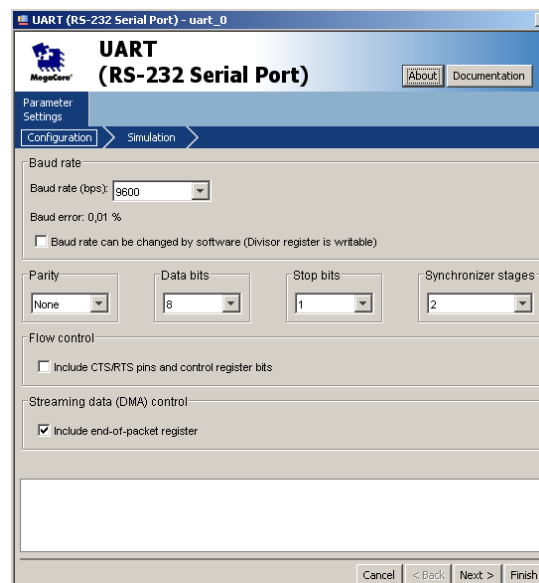
A continuación mostramos la descripción de asignación de pines y las señales que usamos.

Nombre de la Señal	Dirección	Locación PINES	Estándar I/O	Descripción
GPIO_SENSOR[7]	Entrada	PIN_J20	3.3-V LVTTTL	Estos pines son las señales de entrada de los sensores magnéticos que se utilizarán para la implementación del proyecto
GPIO_SENSOR[6]	Entrada	PIN_J21	3.3-V LVTTTL	
GPIO_SENSOR[5]	Entrada	PIN_F23	3.3-V LVTTTL	
GPIO_SENSOR[4]	Entrada	PIN_F24	3.3-V LVTTTL	
GPIO_SENSOR[3]	Entrada	PIN_E25	3.3-V LVTTTL	
GPIO_SENSOR[2]	Entrada	PIN_E26	3.3-V LVTTTL	
GPIO_SENSOR[1]	Entrada	PIN_J22	3.3-V LVTTTL	
GPIO_SENSOR[0]	Entrada	PIN_D25	3.3-V LVTTTL	

**Tabla 3-1 Descripción de pines en Cyclone II EP2C35F672C6**

### 3.3 Interfaz Serial RS-232

Se le llama serial, porque permite el envío de datos, uno detrás de otro, mientras que un paralelo se dedica a enviar los datos de manera simultánea, este tipo de puerto es necesario para la comunicación y transmisión de datos con el modulo GSM que se usa en el proyecto, así mismo hay que añadirlo en el diseño del hardware del proyecto en SOPC Builder, y en la Figura 3-5 se muestra la configuración que se usó para el correcto funcionamiento del puerto serial RS-232.



**Figura 3-5 Puerto Serial RS-232**

### **3.4 Programación en Lenguaje C usando Software Nios II**

El programa descrito a continuación fue realizado en lenguaje C, posteriormente compilado y ejecutado en software Nios II para Eclipse. Se muestra en la Figura 3-6 una fracción de las partes más importantes del código fuente del proyecto de graduación.

El siguiente código descrito a continuación muestra la máscara que se usa en cada switch, la idea general es que se pueden utilizar más de cuatro switches en el proyecto en caso de que se refiera a la residencia de algún usuario, por esta razón se utiliza ocho bits de los cuales solo usamos los cuatro bits más significativos. También se describe que led verde debe encenderse en caso de que algún switch magnético se active, así como la función en donde se transmite el mensaje vía comandos AT al celular.



```

#include <stdlib.h>
#include <ctype.h>
#include "gsm.h"
#include <stdio.h>
unsigned int getGPIO();
int main( int argc, char *argv[] )
{
    unsigned int csw;
    while(1){
        csw = getGPIO();
        if(csw & 0x80){
            IOWR(GREEN_LEDS_BASE,0,1);
            mensajes_texto(1);
        }else if(csw & 0x40){
            IOWR(GREEN_LEDS_BASE,0,2);
            mensajes_texto(2);
        }else if(csw & 0x20){
            IOWR(GREEN_LEDS_BASE,0,3);
            mensajes_texto(3);
        }else if(csw & 0x10){
            IOWR(GREEN_LEDS_BASE,0,4);
            mensajes_texto(4);
        }else {
            IOWR(GREEN_LEDS_BASE,0,0);
        }
    }
    return 0;
}
unsigned int getGPIO()
{
    return (*pGPIO0);
}

```

**Figura 3-6 Código Fuente Principal del Main.**

La Figura 36 muestra el código fuente principal que contiene el main, y las librerías usadas; así como unas funciones propias de la librería system.h, y algunas funciones que se crearon para el funcionamiento del proyecto.

```

void mensajes_texto(unsigned int door){

unsigned char *comando="AT\r";
unsigned char *modoasc="AT+CMGF=1\r";
unsigned char mtexto[]="AT+CMGS=";
unsigned char *num1 = "089866287";
unsigned char *num2 = "096918689";
unsigned char *mens1= "\rpuerta 1 abierta!!!";
unsigned char *mens2= "\rpuerta 2 abierta!!!";
unsigned char *mens3= "\rpuerta 3 abierta!!!";
unsigned char *mens4= "\rpuerta 4 abierta!!!";

unsigned char *mensaje1, *mensaje2 ;
unsigned int count1,count2,count3,count4;
    mensaje1=malloc(sizeof(char) *22);
    mensaje2=malloc(sizeof(char) *22);
    strcpy(mensaje1, mtexto);
    char ch = '\n';
    int len,a;
    len = strlen(mensaje1);
    mensaje1[len] = ch;
    mensaje1[len +1] =(char) NULL ;
    strcat(mensaje1,num2);
    a=strlen(mensaje1);
    mensaje1[a] = ch;
    mensaje1[a +1] =(char) NULL ;

```

**Figura 3-7 Función para el envío de Mensaje**

El siguiente código describe la función principal que envía el mensaje a algún número de celular en particular. Se muestra como se construye la estructura de los comandos AT en forma de cadena que es el formato usado en código C.

```

switch(door){
case 1:
    transmitir_uart(comando);
    delay(1000000);
    count1=0;
    while(count1 < 1){
        transmitir_uart(modosasc);
        delay(1000000);
        transmitir_uart(mensaje1);
        transmitir_uart(mens1);
        IOWR(UART_O_BASE, 1, 0x1A);
        delay(1000000);
        transmitir_uart(mensaje2);
        transmitir_uart(mens1);
        IOWR(UART_O_BASE, 1, 0x1A);
        delay(1000000);
        count1++;
    }
    delay(1000000);
    break;
case 2:
    transmitir_uart(comando);
    delay(1000000);
    count2=0;
    while(count2 <2){
        transmitir_uart(modosasc);
        delay(1000000);
        transmitir_uart(mensaje1);
        transmitir_uart(mens2);
        IOWR(UART_O_BASE, 1, 0x1A);
        delay(1000000);
        transmitir_uart(mensaje2);
        transmitir_uart(mens2);
        IOWR(UART_O_BASE, 1, 0x1A);
        delay(1000000);
        count2++;
    }
    delay(1000000);
    break;
}

```

**Figura 3-8 Sentencia Case usada en el Proyecto.**

Esta es la segunda parte de la función descrita anteriormente en la Figura 3-7 en la cual mediante el uso de una sentencia switch case, podemos transmitir, según sea el switch activado, el mensaje correspondiente al celular del destinatario. Hay muchas maneras de realizar la programación, incluso se puede usar un *if else* para reemplazar al switch case, ver Figura 3-8.

### 3.5 Bloque de Switches Magnéticos

Este tipo de sensores actúa como un switch (0-1), y es de mucha utilidad en puertas y ventanas, su funcionamiento es parecido a una botonera en lógica negativa, es decir cuando las terminales de los sensores están unidos se produce un cortocircuito enviando un nivel bajo de voltaje, es decir un cero lógico, lo cual indicaría que las puertas y ventanas están cerradas, pero por otro lado cuando las terminales del switch están separados entonces se produce un nivel alto de voltaje, es decir un uno lógico, y esta característica indicaría que las ventanas y puertas están abiertas y alguien ingreso al lugar donde estaban colocados este tipo de switches.

Interruptores	Normalmente Cerrado	Normalmente Abierto
Entre en contacto con la energía	10W	3W
Corriente de interruptores:	0.5 A	0.2 A
Voltaje del principio:	100V	30V
Resistencia inicial	200 $\Omega$	-
Distancia de la acción:	18-23 mm	20-25 mm

**Tabla 3-2 Características Técnicas del Switch Magnético**

### 3.6 Bloque Convertidor de Voltaje con conector DB9

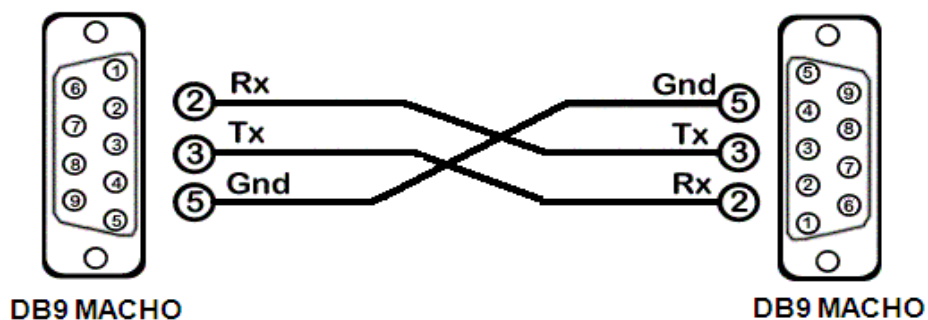
Para realizar la comunicación entre la tarjeta DE2 de Altera y el módulo GSM SIM340CZ, necesitamos que funcionen en el mismo nivel de voltaje, por este motivo utilizamos un convertidor de voltajes, el cual transforma la señal RS-232 en TTL. En RS232 un 1 lógico equivale a -15V y un 0 lógico

equivale a +15V. Como solo se tiene una fuente de alimentación de +5V para el módulo GSM y el convertidor, este valor se utiliza como nivel lógico positivo (el cero lógico será representado por 0V), se muestra en la Figura 3-9 el convertidor usado en el proyecto.



**Figura 3-9 Convertidor de Voltajes**

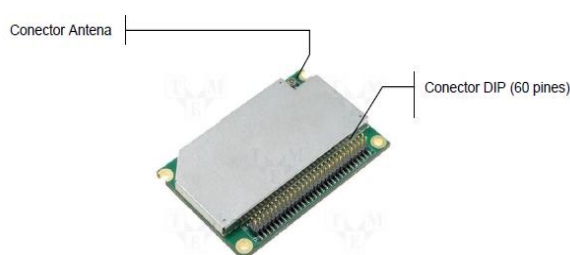
Pero hay q tener en cuenta que un lado de la comunicación debe ser DTE y el otro lado DCE, como lo muestra la Figura 3-10.



**Figura 3-10 Conexión del conector DB9 entre el módulo GSM y tarjetaDE2**

### 3.7 Módulo SIM 340CZ

Módulo SIM340cz GSM/GPRS de cuatro bandas (850/900/1800/1900MHz) con rendimiento para voz, SMS, datos y fax a bajo consumo de potencia. Este módulo se encargará de enviar los mensajes de texto al teléfono móvil comandadas por el procesador Nios II, una vez que la seguridad haya sido violada. Los comandos enviados son de tipo AT dentro de una comunicación UART entre el procesador y el módulo. Ver Figura 3-11.



**Figura 3-11 Módulo GSM [9]**

### 3.7.1 Especificaciones del Módulo GSM

- La fuente de alimentación es de 3.4V a 4.5V.
- Proporciona conectividad GPRS multi-slot clase 8 / clase 10.
- Capacidad y soporte de esquemas de codificación GPRS CS-1, CS-2, CS-3 y CS-4.
- La interfaz física del módulo contiene un conector DIP de 60 pines, la cual es toda la interfaz de hardware entre el módulo y el board del usuario.
- Temperatura Normal de operación es -20°C a +55°C.
- EL módulo integra el protocolo TCP/IP.
- Soporta transmisión de mensajes SMS en CSD y GPRS.
- La interfaz de la tarjeta SIM soporta 1.8V y 3V.
- Posee una antena externa con un conector de 50 OHM.
- Posee una interfaz de comunicación serial, que puede ser usado para CSD FAX, servicios GPRS y para enviar comandos AT para controlar al módulo.

### 3.7.2 Descripción de pines utilizados del Módulo GSM

A continuación se muestra la tabla con las señales usadas en el presente proyecto.

PIN	I/O	DESCRIPCIÓN	CARACTERÍSTICAS DC
VBAT		5 pines del conector DIP son dedicados para conectar la fuente de voltaje, esta fuente es de VBAT=3.4V...4.5V, debe ser capaz de proveer la suficiente corriente en la transmisión la cual es típicamente de 2A, estos 5 pines son para el voltaje de entrada	Vmax=4.5V Vmin=3.4V Vnormal=4.0V
GND		Tierra Digital	
PWRKEY	I	Voltaje de entrada para encender por medio de un pulso, PWRKEY funciona a un voltaje bajo para que el usuario encienda o apague el sistema. Se usa para encender o apagar el sistema, el usuario presiona el botón por un momento para encender o apagar el sistema, ya que el sistema necesita un margen de tiempo para normalizar el software	VILmax=0.3*VBAT VIHmax=0.7*VBAT Vimax=VBAT
RXD	I	Receptor de Datos	
TXD	O	Transmisor de Datos	
DTR	I	Terminal de Datos Preparado	

**Tabla 3-3 Pines Modulo GSM SIM340CZ**

### 3.8 Comunicación entre los Switches Magnéticos, FPGA, Módulo GSM y celular

En el presente proyecto la comunicación se realiza de la siguiente forma:



Los cuatro switches magnéticos que vamos a utilizar están configurados en lógica negativa, en su estado normal se encuentran unidos como observamos en la siguiente Figura 3-12.



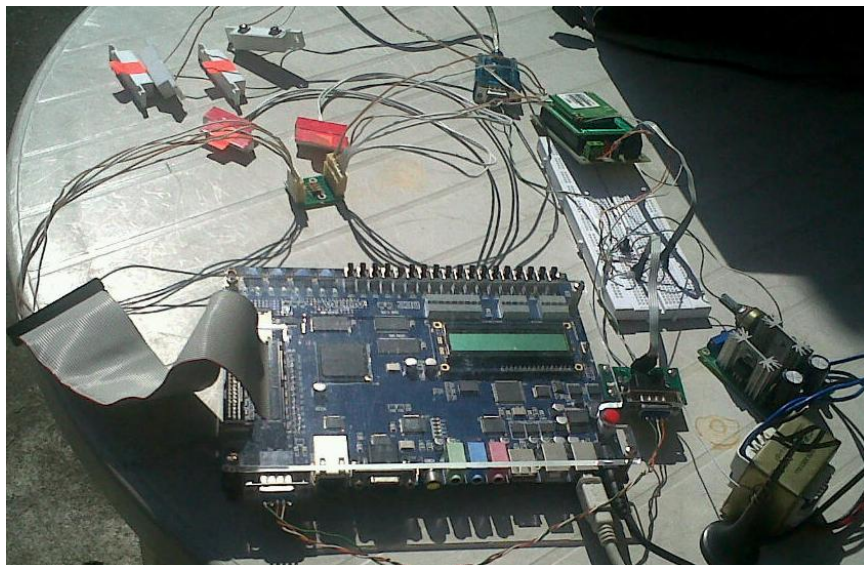
**Figura 3-12 Switch Magnético**

Estos switches están conectados a la tarjeta DE2 mediante un bus de datos los cuales se conectan hacia unos pines previamente configurados (PIO\_0 de la tarjeta).

Al momento en que se abre un switch magnético, la tarjeta recibe una señal la cual será un valor alto de voltaje y dependiendo de qué switch se haya abierto, la señal llega a su respectivo pin los cuales como anteriormente dijimos están configurados para recibir esta señal, la tarjeta reconoce que pin se activó, para efectos de nuestro proyecto programamos la tarjeta para que los leds se enciendan dependiendo de qué sensor se activó, por ejemplo, si se activó el switch uno, la tarjeta encenderá los leds representando el número uno en binario.

Una vez que se haya reconocido que switch se activó, el siguiente paso en la programación es la de enviar los respectivos comandos AT al módulo GSM para que se envíe el mensaje de texto. Estos comandos AT son enviados a través del protocolo de comunicación RS232 entre la tarjeta y el módulo GSM, los cuales son conectados por medio de un convertidor de 12v a 5v, ya que estos trabajan en esos voltajes respectivamente, el módulo GSM recibe los comandos, en este caso el respectivo mensaje que debe enviar dependiendo de qué switch se activó. Los mensajes son enviados a través del módulo por medio de la red GSM los cuales pueden ser de las operadoras CLARO, MOVISTAR o ALEGRO dependiendo de qué chip se ingresó en el módulo.

El mensaje llega al celular indicando que puerta se activó.



**Figura 3-13 Esquema de la Comunicación del Proyecto**

# CAPÍTULO 4

## 4. PRUEBAS Y RESULTADOS

En este capítulo se muestra las pruebas realizadas, incluyendo la presentación por pantalla en la computadora mediante el Access Port, en el cual observamos la comunicación que realiza la tarjeta DE2 con el módulo GSM, veremos paso a paso los comandos enviados al módulo GSM para el envío del mensaje de texto hacia un celular; por este motivo se han incluido tres escenarios importantes para entender el funcionamiento del proyecto.

### 4.1 Escenarios

Para el diseño del presente proyecto se establecen los siguientes escenarios, los cuales tienen como objetivo explicar de mejor manera como se envían los mensajes de texto entre el bloque de switches magnéticos de apertura de puerta y el celular:

- Escenario A: No hay señal para la comunicación.
- Escenario B: El chip SIM no tiene saldo para enviar mensajes.
- Escenario C: Comunicación y envío de mensajes exitosos.

### 4.1.1 Escenario A: No hay señal para la comunicación

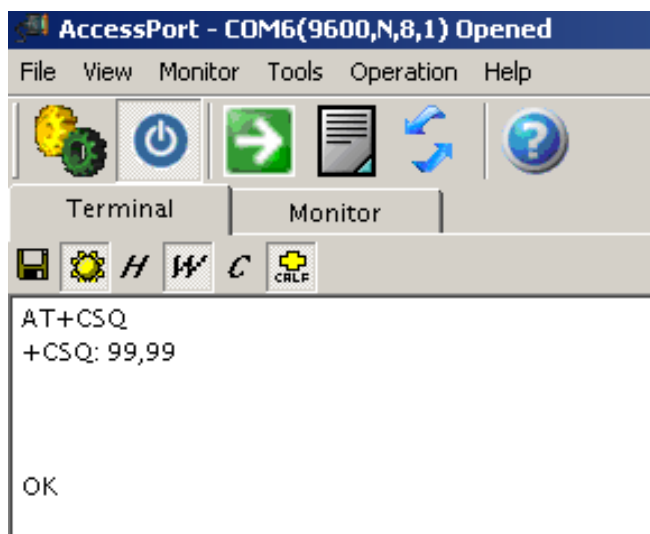
Una vez instalado todos el sistema para la transmisión del mensaje de texto a un número celular, es importante tener en cuenta el nivel de señal óptimo para que la comunicación pueda llevarse a cabo para este caso lo verificamos con el siguiente comando que reporta la calidad de la señal.

AT+CSQ <Enter>

Al ejecutar este comando, ver Figura 4-1, retorna los siguientes parámetros: la indicación de fuerza señalada recibida <rsssi> y la tasa de error de bit <ber> desde el equipo móvil. A continuación se muestra la tabla con los valores de parámetros:

Parámetros	
<b>&lt;rsssi&gt;</b>	
0	-113 dBm o menor (mínimo)
1	-111 dBm
2-30	-109 a -53 dBm
31	-51 dBm (máximo)
99	no detectable, no hay señal
<b>&lt;ber&gt;</b>	
0...7	valores RXQUAL
99	no detectable

**Tabla 4-1 Parámetros de comando AT+CSQ**



**Figura 4-1 No hay señal en el Modulo GSM**

#### **4.1.2 Escenario B: El chip SIM no tiene saldo para enviar mensajes**

En este escenario nos encontramos con un problema muy común cuando no podemos enviar mensaje, la falta de saldo, normalmente se conoce como que la tarjeta SIM no está habilitada para enviar mensajes, el comando usado para enviar mensajes es:

1. Poner sistema SMS del modem en modo ASCII:

```
AT+CMGF=1
```

2. Envío de mensaje de texto.

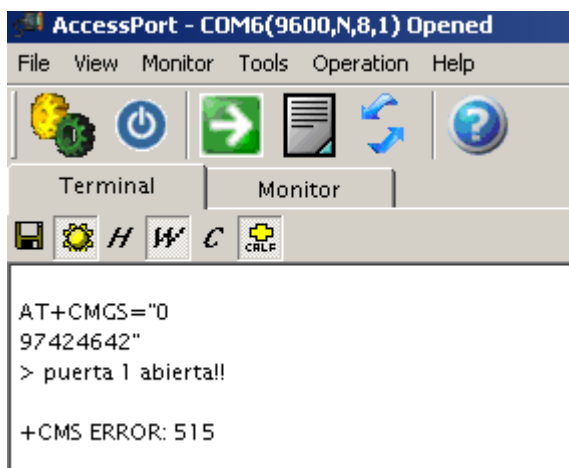
```
AT+CMGS="número telefónico" <Enter>
```

```
<Mensaje a enviar>
```

```
Finalizar y enviar SMS con CTL-Z (0x1A).
```

Mensajes: ERROR 193 + ERROR 515 si la SIM no está habilitada para SMS.

Mensaje: +CMGS: 89 o mayor valor significa Mensaje enviado.



**Figura 4-2 Chip no tiene saldo para enviar mensaje**

#### **4.1.3 Escenario C: Comunicación y envío de mensajes exitoso**

En nuestro proyecto tendremos cuatro puertas, cada una tendrá un switch magnético, a continuación observamos un ejemplo cuando se abre la primera puerta:

El switch magnético se encuentra en su estado inactivo, en esta posición el switch de la tarjeta DE2 no recibirá ninguna señal.



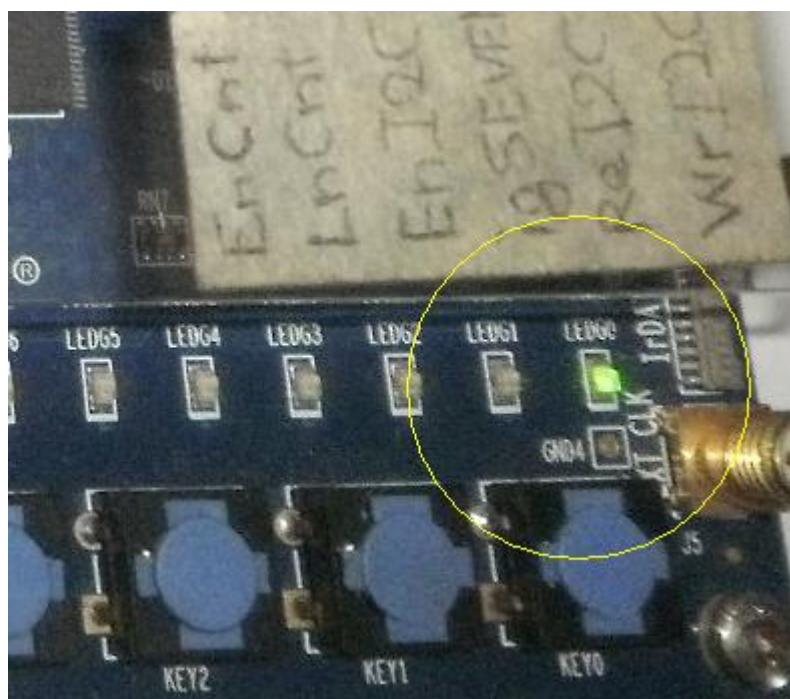
**Figura 4-3 Switch Magnético cerrado**

El momento en que se abrió la primera puerta; el switch se separa tal como vemos en la Figura 4-4.



**Figura 4-4 Switch Magnético abierto**

Como nuestro ejemplo indica que se abre primera puerta, la tarjeta muestra en los leds el número uno en binario, de esta manera nos aseguramos que se encendió el respectivo switch.



**Figura 4-5 Primer Led Verde encendido**

La tarjeta reconoció que switch se activó, la programación envía el respectivo comando AT al módulo por medio del puerto seriales RS232, conectados entre la tarjeta y el módulo. A continuación por medio del Access Port, observamos en la pantalla de nuestra computadora la comunicación entre la tarjeta y el módulo GSM.



```

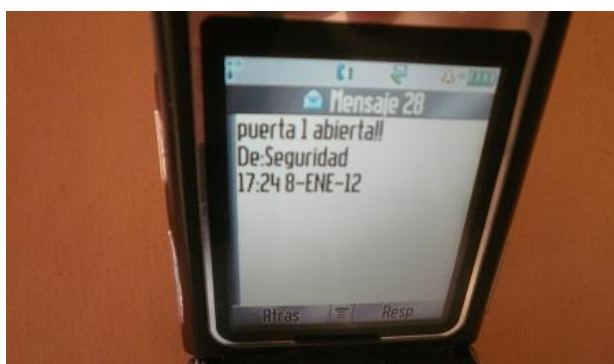
AccessPort - COM6(9600,N,8,1) Opened
File View Monitor Tools Operation Help

Terminal Monitor

AT
OK
AT+CMGF=1
OK
AT+CMGS="097424642"
> puerta 1 abierta!!
+CMGS: 232
OK
AT+CMGF=1
OK
AT+CMGS="097424642"
> puerta 1 abierta!!
+CMGS: 233
OK
  
```

**Figura 4-6 Mensaje que se muestra en el Access Port**

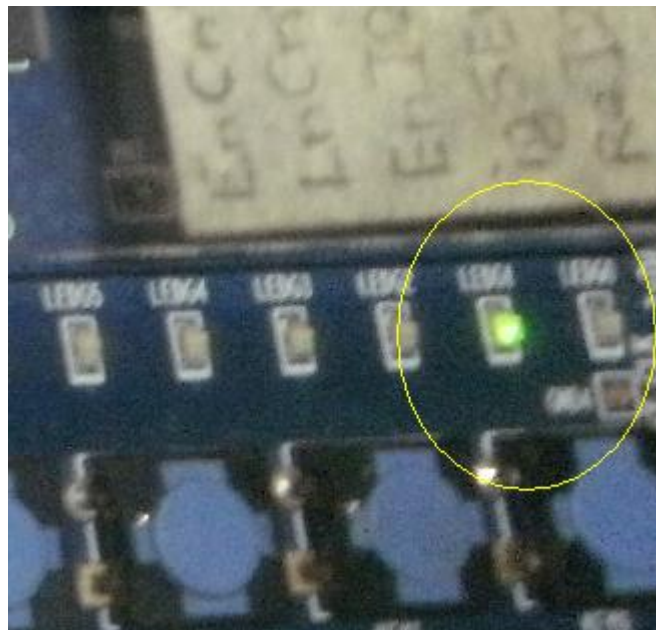
El mensaje llega al celular indicando que puerta se activó.



**Figura 4-7 Mensaje del Switch 1 en el Celular**

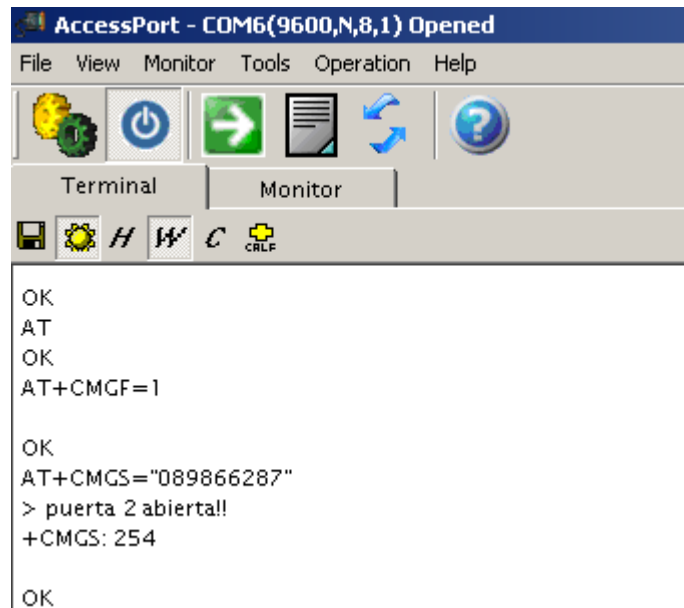
A continuación un ejemplo cuando se abre la segunda puerta:

El switch se abre, la señal es enviada y la tarjeta enciende sus leds, representando en binario el número dos.



**Figura 4-8 Segundo Led Verde encendido**

La programación envía el respectivo comando realizando la siguiente comunicación entre la tarjeta y el módulo.



**Figura 4-9 Mensaje del Segundo Switch**

El mensaje llega al celular:



**Figura 4-10 Mensaje del Switch 2 en el celular**

A continuación un ejemplo cuando se abre la puerta tres:



**Figura 4-11 Tercer Led Verde encendido**

El switch se abre, este manda la señal y la tarjeta enciende sus leds, representando en binario el número tres.

La programación envía el respectivo comando realizándose la siguiente comunicación entre la tarjeta y el módulo.

```

AccessPort - COM6(9600,N,8,1) Opened
File View Monitor Tools Operation Help
Terminal Monitor
OK
AT
OK
AT+CMGF=1

OK
AT+CMGS="089866287"
> puerta 3 abierta!!
+CMGS: 254

OK
  
```

**Figura 4-12 Mensaje del Tercer Switch**

El mensaje llega al celular.



**Figura 4-13 Mensaje del Switch 3 en el celular**

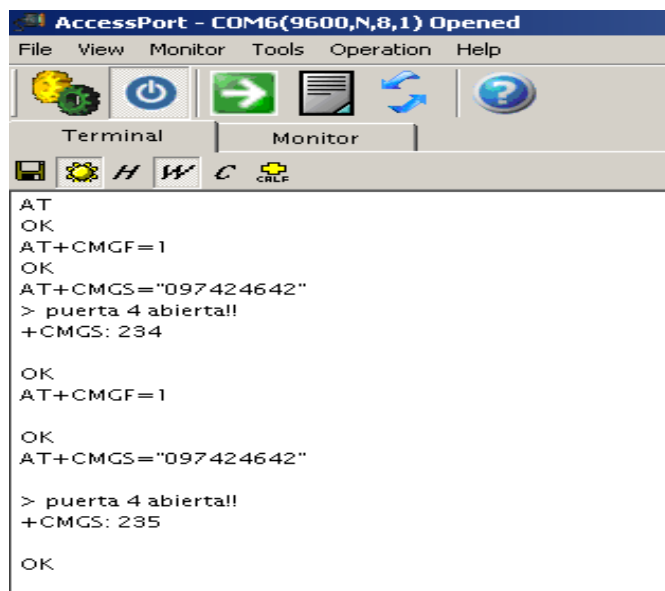
Por último la demostración de la puerta cuatro:

El switch se abre, esta manda la señal y la tarjeta enciende sus leds, representando en binario el número cuatro.



**Figura 4-14 Cuarto Led Verde encendido**

La programación envía el respectivo comando realizándose la siguiente comunicación entre la tarjeta y el módulo.



```
AccessPort - COM6(9600,N,8,1) Opened
File View Monitor Tools Operation Help
Terminal Monitor
AT
OK
AT+CMGF=1
OK
AT+CMGS="097424642"
> puerta 4 abierta!!
+CMGS: 234

OK
AT+CMGF=1
OK
AT+CMGS="097424642"
> puerta 4 abierta!!
+CMGS: 235

OK
```

**Figura 4-15 Mensaje del Cuarto Switch**

Llega el mensaje al celular:



**Figura 4-16 Mensaje del Switch 4 en el celular**

**CONCLUSIONES Y  
RECOMENDACIONES**

## CONCLUSIONES

1. En nuestro proyecto de graduación pudimos demostrar una de las innumerables aplicaciones en las que se usa tecnología inalámbrica junto con la tecnología FPGA, mediante un módulo que se comunica con una tarjeta de altera, además de un estudio preliminar en el que se demuestra que se puede llegar a implementar un proyecto de tal magnitud para uso comercial en nuestro país, convirtiéndose de esta manera en un generador de propuestas.
2. Se uso el lenguaje C para programar nuestro proyecto y verificamos que un lenguaje de alto nivel puede usarse para implementaciones a nivel de hardware, lo cual es una ventaja, debido a su facilidad de entendimiento y de desarrollo.
3. Los switches magnéticos de apertura de puerta fueron de gran utilidad para la realización de este proyecto; sin embargo se pueden utilizar otros tipos de dispositivos externos como es el caso de los sensores de movimiento, de calor o sonido ya que la tarjeta solo necesita una señal lógica de voltaje sea un alto o un bajo, para responder ante alguna variante en un sistema.
4. La red GSM se la puede usar para varias aplicaciones en este caso a nivel de seguridad, mediante un modulo GSM que recibe comandos AT específicos.



5. Podemos concluir que es posible realizar una comunicación entre dispositivos utilizando la comunicación GSM, lo que se necesita es un módulo GSM que entienda comandos AT, para ser transmitidos a través de la tarjeta DE2 que hace las veces de controlador de nuestro proyecto.

## RECOMENDACIONES

1. La tarjeta SIM a utilizar debe tener saldo disponible o paquete de mensajes de texto activo, para que el sistema pueda enviar de manera correcta el sms a los números almacenados en la consola.
2. Verificar la hoja técnica de cualquier dispositivo externo a usar en cualquier proyecto, como el caso de nuestro modulo GSM, para que ningún voltaje que sobrepase el especificado logre causar algún daño.
3. Se recomienda usar un programa de comunicación como es el caso del Access Port para poder visualizar la comunicación entre el módulo y la tarjeta a la hora de realizar el proyecto, de esta forma observaríamos errores de programación o electrónicos.
4. En lenguaje de alto nivel hay que tener precaución al momento de usar comandos de control, porque son más usados a nivel de hardware, y estos no pueden ser bien interpretados al momento de ejecutar el programa.

# **ANEXOS**

# ANEXO A



## Using the Expansion Headers

The DE2 Board provides users two 40-pin expansion headers. Each header provides DC +5V (VCC5), DC +3.3V (VCC33), two GND pins for users to build their own daughter cards using the DE2 expansion ports.

Figure 3.10 shows the related schematics. The pin assignment of the associated interface is shown in Table 3.8.

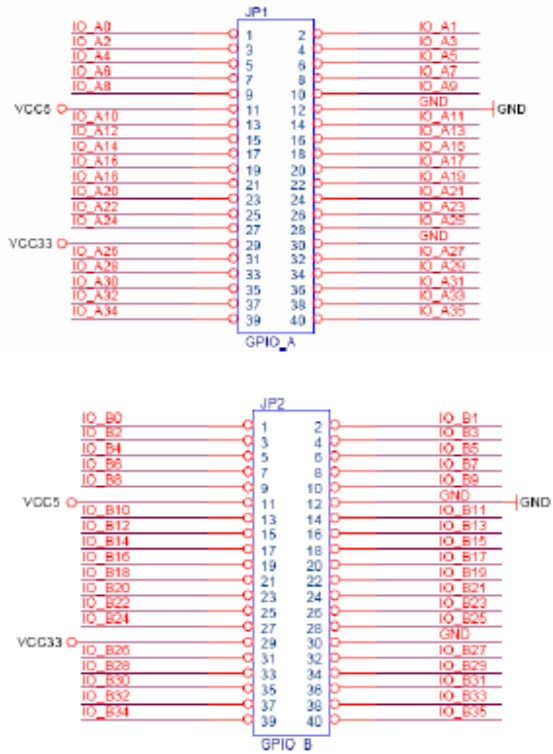


Figure 3.10 Two 40-pin Expansion Headers

## ANEXO B

### Using the Serial Ports(RS232)

The DE2 Board uses the standard 9-pin D-SUB connector for RS-232 communications between PC and the board. The transceiver chip used is MAX232. For detailed information on how to use the chip, users can refer to the spec under **C:\DE2\Datasheet\RS232**. Figure 3.11 shows the related schematics. The pin assignment of the associated interface is shown in Table 3.9.

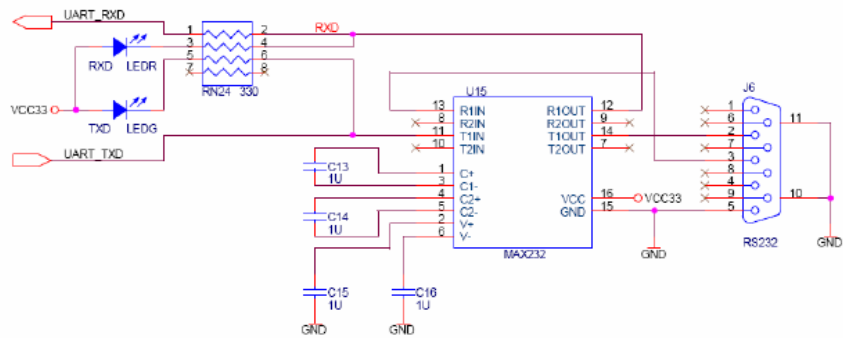


Figure 3.11 MAX232 chip for RS-232 communication

Signal Name	FPGA Pin No.	Description
UART_RXD	PIN_C25	UART Receiver
UART_TXD	PIN_B25	UART Transmitter

Table 3.9 Pin Assignment for Serial Ports (RS232)

## ANEXO C

### Hoja de especificaciones y comandos AT del Modulo SIM340C

Link1: <http://www.scribd.com/doc/40229924/SIM340C-Hardware-Manual>

Link2: <http://www.scribd.com/doc/40230515/SIM340C-AT-command>

#### **COMANDOS AT MÁS UTILIZADOS**

##### *1 Comandos generales*

- a) AT+CGMI: Identificación del fabricante
- b) AT+CGSN: Obtener número de serie
- c) AT+CIMI: Obtener el IMSI.
- d) AT+CPAS: Leer estado del modem

##### *2. Comandos del servicio de red*

- a) AT+CSQ: Obtener calidad de la señal
- b) AT+COPS: Selección de un operador
- c) AT+CREG: Registrarse en una red
- d) AT+WOPN: Leer nombre del operador

##### *3. Comandos de seguridad:*

- a) AT+CPIN: Introducir el PIN
- b) AT+CPINC: Obtener el número de reintentos que quedan
- c) AT+CPWD: Cambiar password

##### *4. Comandos para la agenda de teléfonos*

- a) AT+CPBR: Leer todas las entradas

- b) AT+CPBF: Encontrar una entrada
- c) AT+CPBW: Almacenar una entrada
- d) AT+CPBS: Buscar una entrada

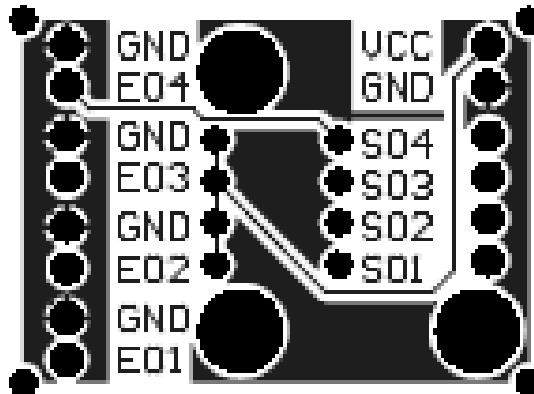
#### *5. Comandos para SMS*

- a) AT+CPMS: Seleccionar lugar de almacenamiento de los SMS
- b) AT+CMGF: Seleccionar formato de los mensajes SMS
- c) AT+CMGR: Leer un mensaje SMS almacenado
- d) AT+CMGL: Listar los mensajes almacenados
- e) AT+CMGS: Enviar mensaje SMS
- f) AT+CMGW: Almacenar mensaje en memoria
- g) AT+CMSS: Enviar mensaje almacenado
- h) AT+CSCA: Establecer el Centro de mensajes a usar
- i) AT+WMSC: Modificar el estado de un mensaje.

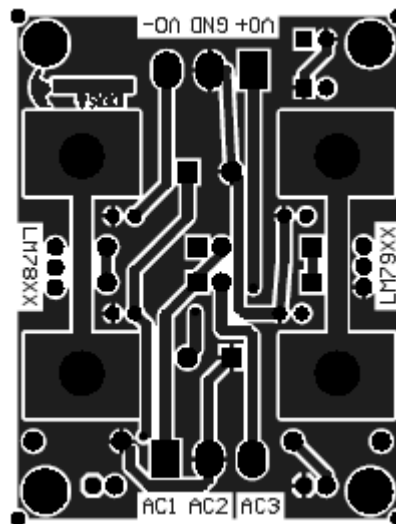
## **ANEXO D**

### **Esquemáticos y PCB del Switch Magnético de Apertura de Puerta**

A continuación se muestran los diseños de los PCB que se van utilizar en nuestro proyecto.



**Placa de Switches Magnéticos**



**Placa de Fuente 5V**

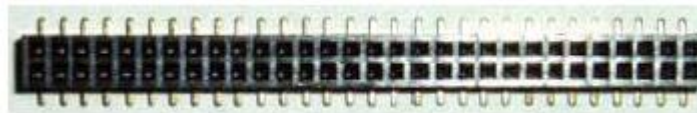
## **ANEXO E**

### **Interfaz PCB para el Módulo GSM**

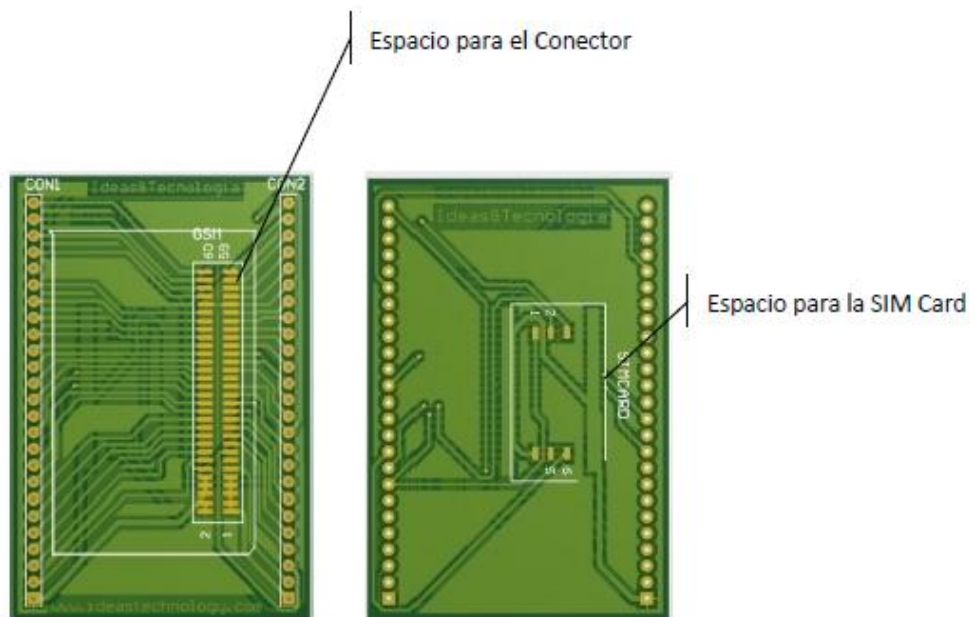
Las características del módulo se dan a continuación:



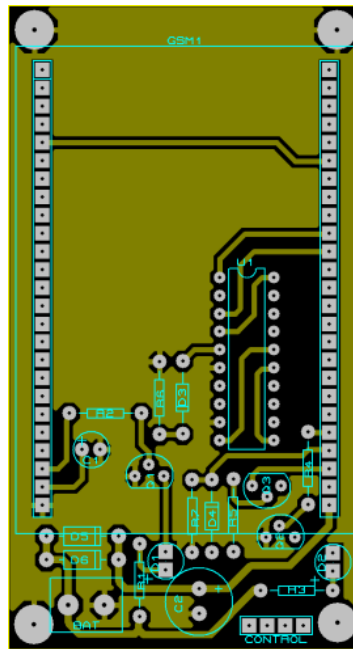
- Modelo IpC-IT-01
- Conector ASTRON 1590060-093
- Comunicación con la SIM Card



**Conector ASTRON 1590060-093**



**Circuito Impreso Módulo GSM (Superior y Posterior)**



PCB del Módulo SIM340CZ

## ANEXO F

### Código Completo del Proyecto

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "system.h"
#include "altera_avalon_pio_regs.h"
#include "gsm.h"
```

```
/*Esta función es importante ya que es la función, que
construye la estructura básica del mensaje a transmitir al
celular cuando un sensor está abierto, además contiene los
comandos AT usados en la red GSM para comunicarse con el
celular*/
```

```

void mensajes_texto(unsigned int door){
unsigned char *comando="AT\r";
unsigned char *modoasc="AT+CMGF=1\r";
unsigned char mtexto[]="AT+CMGS=";
unsigned char *num1 = "089866287";
unsigned char *num2 = "096918689";
unsigned char *mens1= "\rpuerta 1 abierta!!!";
unsigned char *mens2= "\rpuerta 2 abierta!!!";
unsigned char *mens3= "\rpuerta 3 abierta!!!";
unsigned char *mens4= "\rpuerta 4 abierta!!!";
unsigned char *mensaje1, *mensaje2 ;
unsigned int count1,count2,count3,count4;

    mensaje1=malloc(sizeof(char)*22);
    mensaje2=malloc(sizeof(char)*22);
    strcpy(mensaje1, mtexto);
    char ch = '\n';
    int len,a;
    len = strlen(mensaje1);
    mensaje1[len] = ch;
    mensaje1[len +1] =(char) NULL ;
    strcat(mensaje1,num2);
    a=strlen(mensaje1);
    mensaje1[a] = ch;
    mensaje1[a +1] =(char) NULL ;
    //-----
    strcpy(mensaje2, mtexto);
    char c = '\n';
    int leng,ai;
    leng = strlen(mensaje2);
    mensaje2[leng] = c;
    mensaje2[leng +1] =(char) NULL ;
    strcat(mensaje2,num1);
    ai=strlen(mensaje2);
    mensaje2[ai] = c;
    mensaje2[ai +1] =(char) NULL ;
    //-----

```

```

switch(door) {
case 1:
    transmitir_uart(comando);
    delay(1000000);
    count1=0;
    while(count1 < 1){
        transmitir_uart(modoasc);
        delay(1000000);
        transmitir_uart(mensajel);
        transmitir_uart(mens1);
        IOWR(UART_0_BASE, 1, 0x1A);
        delay(1000000);
        transmitir_uart(mensaje2);
        transmitir_uart(mens1);
        IOWR(UART_0_BASE, 1, 0x1A);
        delay(1000000);
        count1++;
    }
    delay(1000000);
    break;
case 2:
    transmitir_uart(comando);
    delay(1000000);
    count2=0;
    while(count2 <2){
        transmitir_uart(modoasc);
        delay(1000000);
        transmitir_uart(mensajel);
        transmitir_uart(mens2);
        IOWR(UART_0_BASE, 1, 0x1A);
        delay(1000000);
        transmitir_uart(mensaje2);
        transmitir_uart(mens2);
        IOWR(UART_0_BASE, 1, 0x1A);
        delay(1000000);
        count2++;
    }
}

```

```

    }
    delay(1000000);
    break;
case 3:
    transmitir_uart(comando);
    delay(1000000);
    count3=0;
    while(count3 <2){
        transmitir_uart(modoasc);
        delay(1000000);
        transmitir_uart(mensajel);
        transmitir_uart(mens3);
        IOWR(UART_0_BASE, 1, 0x1A);
        delay(1000000);
        transmitir_uart(mensaje2);
        transmitir_uart(mens3);
        IOWR(UART_0_BASE, 1, 0x1A);
        delay(1000000);
        count3++;
    }
    delay(1000000);
    break;
case 4:
    transmitir_uart(comando);
    delay(1000000);
    count4=0;
    while(count4 <2){
        transmitir_uart(modoasc);
        delay(1000000);
        transmitir_uart(mensajel);
        transmitir_uart(mens4);
        IOWR(UART_0_BASE, 1, 0x1A);
        delay(1000000);
        transmitir_uart(mensaje2);
        transmitir_uart(mens4);
        IOWR(UART_0_BASE, 1, 0x1A);
    }

```

```

        delay(1000000);
        count4++;
    }
    delay(1000000);
    break;

    default: printf("\r NINGUN MENSAJE ENVIADO");

}

}

void delay(unsigned int k)
{
    unsigned int i;
    for(i = 0; i <= k; i++){
        }
    }

/* Esta función transmite el comando AT al modulo con el
mensaje que queremos que se muestre en el celular.*/

void transmitir_uart(unsigned char mensaje[])
{
/* uart_status es la variable usada para leer el dato desde
el UART status register.*/
    int uart_status;
    int index;
    IOWR(UART_0_BASE, 2, 0);

    for (index = 0; mensaje[index] != '\0'; index++)
    {
        while (!(uart_status = IORD(UART_0_BASE, 2) &
0x40));
        IOWR(UART_0_BASE, 1, mensaje[index]);

```

```
    }  
}
```

## MAIN PRINCIPAL

```
#include "259macros.h"  
#include "altera_avalon_pio_regs.h"  
#include "system.h"  
#include "alt_types.h"  
#include <io.h>  
#include <stdlib.h>  
#include <string.h>  
#include <stdlib.h>  
#include <ctype.h>  
#include "gsm.h"  
#include <stdio.h>  
  
unsigned int getGPIO();  
  
/*****  
El programa principal del proyecto, mediante un lazo while  
hacemos interactuar la tarjeta DE2 con los switches magnéticos y  
el módulo GSM, se realiza mediante el uso de máscaras, las  
cuales indican la posición de cada switch magnético el mensaje  
se transmitirá de acuerdo a que switch es abierto  
*****/  
  
int main( int argc, char *argv[] )  
{  
    unsigned int csw;  
    while(1){  
        csw = getGPIO();  
        if(csw & 0x80){  
            IOWR(GREEN_LEDS_BASE,0,1);  
            mensajes_texto(1);  
        }else if(csw & 0x40){  
            IOWR(GREEN_LEDS_BASE,0,2);  
            mensajes_texto(2);  
        }  
    }  
}
```

```
    }else if(csw &0x20) {
        IOWR(GREEN_LEDS_BASE,0,3);
        mensajes_texto(3);
    }else if(csw & 0x10) {
        IOWR(GREEN_LEDS_BASE,0,4);
        mensajes_texto(4);
    }else {
        IOWR(GREEN_LEDS_BASE,0,0);
    }
}
return 0;
}
unsigned int getGPIO()
{
return (*pGPIO0);
}
```



## REFERENCIAS BIBLIOGRÁFICAS

[1] Altera, "Procesador Nios II".

<http://www.altera.com/devices/processor/nios2/ni2-index.html>

Fecha de Consulta: Julio 2011.

[2] Jorge Rodríguez Araújo, "Estudio del Procesador Nios II".

<http://es.scribd.com/doc/28358833/Estudio-del-microprocesador-Nios-II>

Fecha de Consulta: Junio 2011.

[3] Altera, "Tarjeta DE2".

[http://www.altera.com/education/univ/materials/boards/de2/unv-de2-board.html?GSA\\_pos=1&WT.oss\\_r=1&WT.oss=de2](http://www.altera.com/education/univ/materials/boards/de2/unv-de2-board.html?GSA_pos=1&WT.oss_r=1&WT.oss=de2)

Fecha de Consulta: Noviembre 2011.

[4] Universidad de Chile, "Red GSM".

<http://www.uchile.cl/>

Fecha de Consulta: Septiembre 2011.

[5] Wikipedia, "Hand over".

[http://es.wikipedia.org/wiki/Archivo:Hard\\_handover.jpg](http://es.wikipedia.org/wiki/Archivo:Hard_handover.jpg)

Fecha de Consulta: Octubre 2011.

[6] Microcaos, "Todo sobre GSM".

<http://www.microcaos.net/informatica/gsm-todo-sobre-el-gsm/>

Fecha de Consulta: Septiembre 2011.

[7] Spanish, "Switches Magnéticos de Apertura de Puerta".

[http://spanish.dooraccesscontroller.com/buygarage\\_door\\_contact\\_switch.html](http://spanish.dooraccesscontroller.com/buygarage_door_contact_switch.html)

Fecha de Consulta: Junio 2011.

[8] Wanadoo, "Comunicación Serial"

<http://perso.wanadoo.es/pictob/comserie.htm>

Fecha de Consulta: Octubre 2011.

[9] SIMCOM Ltd, "SIM340C Hardware Interface Description".

<http://www.scribd.com/doc/40229924/SIM340C-Hardware-Manual>

Fecha de Consulta: Octubre 2011.

[10] Altera, "DE2 Reference Manual".

<http://www.scribd.com/doc/39591828/DE2-Reference-Manual>

Fecha de Consulta: Agosto 2011.

- [11] Zonabot, "Comandos AT".

<http://www.zonabot.com/electronica/2-comunicaciones/18-comandos-at.html>

Fecha de Consulta: Octubre 2011.

- [12] Altera, "Manual de Introducción al Procesador Nios II".

[ftp://ftp.altera.com/up/pub/Tutorials/DE2/Computer\\_Organization/tut\\_nios\\_2\\_introduction.pdf](ftp://ftp.altera.com/up/pub/Tutorials/DE2/Computer_Organization/tut_nios_2_introduction.pdf)

Fecha de Consulta: Junio 2011.

- [13] Altera, "Manuel SOPC Builder User Guide".

[http://www.ee.ryerson.ca/~courses/coe718/Data-Sheets/Nios-Lab/ug\\_sopcbuilder.pdf](http://www.ee.ryerson.ca/~courses/coe718/Data-Sheets/Nios-Lab/ug_sopcbuilder.pdf)

Fecha de Consulta: Diciembre 2011.

- [14] Wordpress, "Mensajes Cortos SMS"

<http://movilfacil.wordpress.com/2011/03/11/mensajes-cortos-sms/>

Fecha de Consulta: Febrero 2012.