



**SISTEMA DE MONITOREO REMOTO DE UN
PROCESO INDUSTRIAL EMPLEANDO LA RED DE
TELEFONÍA MÓVIL EN COLOMBIA**

**Autores: Franco David Jaraba Corzo
José David Reyes Torres**

Tutor: Msc. Luis Leonardo Camargo

2012

AGRADECMIENTOS

Agradecemos primero a Dios por habernos permitido realizar este proyecto, brindándonos fuerzas y motivación a terminar esta idea que por mucho tiempo estuvo en nuestras mentes. Y que en muchas situaciones tuvimos desmotivaciones, controversias y discusiones, pero gracias a él todo fue llevándose por el camino correcto y se llegó a la meta.

Agradecemos también a nuestros padres que fueron la principal razón de seguir adelante en nuestras metas, fueron de gran ayuda económica a lo largo de nuestra carrera y estuvieron siempre atentos a cualquier necesidad que se presentara. Con su apoyo y colaboración podemos decir que hoy tenemos conocimientos y que a lo largo de 5 años se estuvieron enriqueciendo con noches en vela y desmotivaciones, pero gracias a ellos teníamos la necesidad de seguir luchando.

Agradecemos a nuestro director de tesis Ing. Luis Camargo por haber puesto a nuestra disposición todos sus conocimientos para el desarrollo del proyecto, también a nuestros compañeros Nazly Terán y Andrés Márquez por habernos abierto los ojos en la programación grafica con LabVIEW, por ayudarnos en la programación necesaria para correcto funcionamiento del proyecto y por habernos brindado su apoyo para la elaboración de este proyecto durante mucho tiempo.

A todos muchas gracias.

Franco Jaraba Corzo
José Reyes Torres

CONTENIDO

	Pág.
RESUMEN.....	ii
ÍNDICE DE CONTENIDO.....	iii
ÍNDICE DE FIGURAS.....	vi
ÍNDICE DE TABLAS.....	ix

RESUMEN

El proyecto consiste en el diseño, desarrollo y evaluación de un sistema de monitoreo remoto para procesos industriales aislados geográficamente, utilizando los servicios de la red del operador de telefonía móvil celular. El sistema se compone de un módulo de adquisición de datos y una aplicación servidora.

El módulo obtiene la información de las variables a supervisar utilizando la interfaz RS232, los datos obtenidos son procesados y ordenados en el módulo para ser enviados por medio del servicio de mensajes cortos (SMS, Short Message Service), a la aplicación servidora.

La aplicación servidora, almacena, gestiona y visualiza la información recibida simulando el proceso industrial; Además, la aplicación tiene la capacidad recibir y responder peticiones a través de SMS, y enviar notificaciones de forma automática a los dispositivos móviles del personal implicado en el proceso según políticas de funcionamiento previamente establecidas.

ÍNDICE DE CONTENIDO

CAPITULO I: IDENTIFICACIÓN DE PROYECTO	1
1.1 TITULO	2
1.2 DEFINICIÓN DEL PROBLEMA	2
1.3 JUSTIFICACIÓN	3
1.4 OBJETIVOS	5
1.5 ALCANCES Y LIMITACIONES	5
CAPITULO II: MARCO DE REFERENCIA	7
2.1 ANTECEDENTES	8
2.2 AMBITO DE APLICACIÓN	9
2.3 BASES TEÓRICAS.....	9
2.3.1 GSM.....	9
2.3.2 EL SERVICIO SMS	15
2.3.3 MODEM GSM.....	23
2.3.4 COMANDOS AT.....	25
2.3.5 GENERALIDADES DEL MODEM SIMCOM SIM300.....	28
2.3.6 ESTANDAR EIA232 (RS232)	41
2.3.7 MICROCONTROLADORES	49
2.3.8 LABVIEW	55
CAPITULO III: MARCO METODOLÓGICO	61
3.1 TIPO DE INVESTIGACIÓN	62
3.2 DISEÑO DE LA INVESTIGACIÓN	62
3.2.1 DESCRIPCIÓN DEL SISTEMA.....	62
3.2.2 ARQUITECTURA DEL SISTEMA	63
3.2.3 GUIA DE DESARROLLO DE ACTIVIDADES	64
3.3 DESARROLLO DEL PROTOTIPO DE PRUEBAS PARA LA CONFIGURACIÓN DEL MODEM GSM.....	65
3.3.1 FUENTE DE ALIMENTACIÓN.....	66
3.3.2 ETAPA DE COMUNICACIÓN	71
3.3.3 ETAPA DE ENCENDIDO Y APAGADO PARA MODULO GSM SIMCOM SIM300	73
3.3.4 INTERFAZ PARA SIM CARD.....	76
3.3.5 CONECTOR MODEM SIMCOM SIM300	76
3.3.6 DAGRAMA ESQUEMATICO GENERAL DE PROTOTIPO DE PRUEBA	78
3.4 DESARROLLO HARDWARE DE LA APLICACIÓN SISDAQ	81
3.4.1 FUENTE DE ALIMENTACIÓN.....	82
3.4.2 ETAPA DE COMUNICACIÓN ENTRE MODEM GSM -MICROCONTROLADOR.	83

3.4.3	ETAPA DE COMUNICACIÓN ENTRE TERMINAL DE DATOS- MICROCONTROLADOR.	83
3.4.4	ETAPA DE CONTROL PARA MODULO GSM SIMCOM SIM300.	84
3.4.5	CIRCUITO DE ENCENDIDO Y APAGADO	86
3.4.6	INTERFAZ PARA SIM CARD7	87
3.4.7	DAGRAMA ESQUEMATICO GENERAL DE LA APLICACIÓN SISDAQ	87
3.5	DESARROLLO FIRMWARE DE LA APLICACIÓN SISDAQ.....	89
3.5.1	CONFIGURACIÓN DEL PIC.....	90
3.5.2	ESPERA DE ENCENDIDO DEL MODEM GSM.....	91
3.5.3	CONFIGURACIÓN DE MODEM GSM.....	92
3.5.4	ESTADO DE ESPERA DEL MENSAJE DE INICIO.....	93
3.5.5	HABILITACIÓN DE INTERRUPCION POR TMR1	94
3.5.6	CICLO NORMAL DEL PROGRAMA	94
3.5.7	INTERRUPCIÓN	96
3.6	DISEÑO DE LA INTERFAZ DE VISUALIZACIÓN Y GESTIONAMIENTO DE DATOS A PETICIONES IVIGED	97
3.6.1	CONFIGURACIÓN DEL PUERTO DE COMUNICACIÓN	98
3.6.2	INTERFAZ DE USUARIO PARA LA CONFIGURACIÓN INICIAL	99
3.6.3	ESCRITURA EN EL PUERTO DE COMUNICACIÓN	103
3.6.4	LECTURA DEL PUERTO DE COMUNICACIÓN.....	105
CAPITULO IV: RESULTADOS		116
4.1	PROTOTIPO DE PRUEBAS PARA CONFIGURACIÓN DE MODEM GSM.....	117
4.1.1	ESPECIFICACIONES DEL PROTOTIPO DE PRUEBAS PARA CONFIGURACIÓN DE MODEM GSM	117
4.1.2	INTERFAZ DE ADQUISICIÓN Y ENVÍO DE DATOS SISDAQ	118
4.1.3	ESPECIFICACIONES DEL MODULO SISDAQ.....	118
4.2	VI DE SIMULACIÓN DE PLANTA EMBOTELLADORA DE ALCOHOL ISOPROPILICO...	119
4.3	INTERFAZ DE CONFIGURACIÓN INICIAL DEL SISTEMA, VISUALIZACIÓN Y GESTIONAMIENTO DE DATOS A PETICIONES IVIGED	120
4.3.1	BLOQUE DE CONFIGURACIÓN DE PARAMETROS	121
4.3.2	BLOQUE DE VISUALIZACIÓN DE DATOS.....	122
CAPITULO V: EVALUACIÓN DEL SISTEMA.....		123
5.1	EVALUACIÓN DEL PROTOTIPO PARA CONFIGURACIÓN DE MODEM GSM	124
5.1.1	COMUNICACIÓN CON HYPERTERMINAL	124
5.1.2	PRUEBAS ELECTRICAS.....	130
5.2	EVALUACIÓN DE LA APLICACION SISDAQ	132
5.2.1	ALGORTIMO PARA SIMULACIÓN DEL TERMINAL DE DATOS.....	132

5.2.2	ALGORITMO PARA PRUEBA DE EVALUACIÓN DE LA APLICACIÓN SISDAQ ...	134
5.2.3	DESARROLLO DE PRUEBAS DE LA APLICACIÓN SISDAQ	135
5.3	EVALUACIÓN DEL FUNCIONAMIENTO DE LA APLICACIÓN IVIGED	138
5.3.1	EVALUACIÓN DEL ENVIO DEL MENSAJE DE TEXTO CON LA CONFIGURACIÓN INICIAL... ..	138
5.3.2	EVALUACIÓN DE LA RECEPCION DE MENSAJES Y VISUALIZACION DE LOS MISMOs.. ..	139
5.3.3	EVALUACIÓN DE LAS RESPUESTAS A LAS SOLICITUDES	141
5.3.4	EVALUACIÓN DE INFORME DIARIO DE LOS TODOS LOS DATOS A UNA HORA ESTIPULADA.....	143
5.4	COMPARACION CON OTRAS TECNOLOGÍAS DE MONITOREO.....	144
CAPITULO VI: CONCLUSIONES Y RECOMENDACIONES		146

ÍNDICE DE FIGURAS

Figura 2-1:Entorno Sistema de automatización, Fuente: (GENIA, 2011).....	9
Figura 2-2: Sistema Celular, Fuente: (Gonzalez Gomez, 2002).....	10
Figura 2-3: Arquitectura de una red GSM, Fuente: (Wikipedia, 2011).....	11
Figura 2-4: Estación Móvil, Fuente: (Tecmoviles, 2011).....	11
Figura 2-5: Servicio SMS. Fuente: (Gonzalez Gomez, 2002).....	16
Figura 2-6: Envío de un SMS entre una MS y una entidad fija, Fuente: (Gonzalez Gomez, 2002).....	16
Figura 2-7: Servicios básicos SM MO, Fuente: (Gonzalez Gomez, 2002).....	17
Figura 2-8: Servicios básicos SM MT, Fuente: (Gonzalez Gomez, 2002).....	17
Figura 2-9: Estructura básica de la red para la transferencia de mensajes cortos, Fuente: (Gonzalez Gomez, 2002).....	18
Figura 2-10: Niveles y servicios para el envío de mensajes cortos, Fuente: (Gonzalez Gomez, 2002).....	19
Figura 2-11: Operaciones en el envío de SMS, Fuente: (Escuela Politecnica Superior de Alcoy, 2012).....	21
Figura 2-12: Operaciones en la recepción de SMS, Fuente: (Escuela Politecnica Superior de Alcoy, 2012) ..	22
Figura 2-13: Tipos de módems, Fuente: (Gonzalez Gomez, 2002).....	23
Figura 2-14: Modem SIMCOM 300, Fuente: (SIMCOM, 2007).....	24
Figura 2-15: Entrada VBAT, Fuente: (SIMCOM, 2007).....	32
Figura 2-16: Tiempo de encendido del sistema, Fuente: (SIMCOM, 2007).....	34
Figura 2-17: Tiempo de apagado del sistema, Fuente: (SIMCOM, 2007).....	35
Figura 2-18: Tiempo de reinicio del sistema, Fuente: (SIMCOM, 2007).....	36
Figura 2-19: RTC alimentado desde capacitor, Fuente: (SIMCOM, 2007).....	37
Figura 2-20: Interfaz del puerto serial, Fuente: (SIMCOM, 2007).....	38
Figura 2-21: Circuito de referencia de interfaz de SIM, Fuente: (SIMCOM, 2007).....	40
Figura 2-22: Consideraciones de diseño para soporte de SIMCARD de 6 pines, Fuente: (SIMCOM, 2007) ..	40
Figura 2-23: Descripción de pines, Fuente: (SIMCOM, 2007) (Arduino, 2011).....	40
Figura 2-24: Descripción de pines conector DB9, Fuente: (Arduino, 2011).....	42
Figura 2-25: Descripción de pines conector DB25, Fuente: (Arduino, 2011).....	43
Figura 2-26: Modem activo con HandShaking, Fuente: (Lammertbies, 2011).....	44
Figura 2-27: Modemnulo, Full Duplex (DB9), Fuente: (Lammertbies, 2011)).....	44
Figura 2-28: Especificaciones eléctricas estándar RS232C, Fuente: (EVA, 2011).....	45
Figura 2-29: Descripción del protocolo RS323C, Fuente: (EVA, 2011).....	46
Figura 2-30: Diagrama de conexión MAX232, Fuente: (Maximic, 2012).....	48
Figura 2-31: Adaptación de niveles RS232 a TTL, (EVA, 2011).....	49
Figura 2-32: Diagrama de bloques arquitectura Jhon Von Neumann, Fuente: (Octoplus, 2012).....	50
Figura 2-33: Diagrama de bloques arquitectura Harvard, Fuente: (Octoplus, 2012).....	51
Figura 2-34: Logo LabVIEW, Fuente: (National Instruments, 2012).....	55
Figura 2-35: Bloques de LabVIEW utilizados Parte 1.....	57
Figura 2-36: Bloques de LabVIEW utilizados Parte 2.....	58
Figura 2-37: Bloques de LabVIEW utilizados Parte 3.....	59
Figura 2-38: Bloques de LabVIEW utilizados.....	60
Figura 3-1: Sistema de monitoreo remoto empleando redes de telefonía móvil celular.....	63
Figura 3-2: Modelo vista controlador.....	64
Figura 3-3: Diagrama de bloques del funcionamiento de prototipo de prueba.....	66
Figura 3-4: Regulador en serie básico, Fuente: (Mata, 2001).....	67
Figura 3-5: Circuito de referencia independiente, Fuente: (Mata, 2001).....	68
Figura 3-6: Circuito de muestreo, Fuente: (Mata, 2001).....	69
Figura 3-7: Amplificador Operacional 741, Fuente: (ECE, 2012).....	69

Figura 3-8: Transistor de control, Fuente: (Mata, 2001).	70
Figura 3-9: Diagrama esquemático de fuente regulada de tensión de 4 VDC.	70
Figura 3-10: Diagrama esquemático circuito de regulación de tensión 5V.	70
Figura 3-11: Diagrama esquemático de conversor de niveles TTL-RS232.	71
Figura 3-13: Temporización de encendido del modulo SIM300, Fuente: (SIMCOM, 2007)	73
Figura 3-12: Diagrama esquemático de circuito conversor de niveles TTL a 4V DC.	73
Figura 3-14: Temporización de apagado del modulo GSM, Fuente: (SIMCOM, 2007)	74
Figura 3-15: Diagrama esquemático de etapa de control.	75
Figura 3-16: Diagrama esquemático de la interfaz para la SIMCARD	76
Figura 3-17: Diagrama esquemático de conexión de conector de 60 pines con conector de 19 pines.	78
Figura 3-18: Diagrama esquemático general de prototipo de prueba parte 1.	79
Figura 3-19: Diagrama esquemático general de prototipo de prueba parte 2.	80
Figura 3-20: Diagrama de bloques del funcionamiento de aplicación.	81
Figura 3-21: Circuito de regulación de tensión a 4V DC	82
Figura 3-22: Etapa de comunicación entre modem GSM - microcontrolador.	83
Figura 3-23: Diagrama esquemático de la etapa de comunicación entre terminal de sats y microcontrolador.	83
Figura 3-24: Diagrama esquemático de circuito de control.	86
Figura 3-25: Diagrama esquemático del circuito de encendido y apagado	86
Figura 3-26: Diagrama esquemático general de la aplicación SISDAQ parte 1.	87
Figura 3-27: Diagrama esquemático general de la aplicación SISDAQ parte2.	88
Figura 3-28: Diagrama de flujo del algoritmo de programación para el microcontrolador.	89
Figura 3-29: Diagrama de flujo del estado configuración del microcontrolador.	90
Figura 3-30: Diagrama de flujo del estado espera de encendido del modem GSM.	91
Figura 3-31: Diagrama de flujo del estado configuración del modem GSM.	92
Figura 3-32: Diagrama de flujo del estado espera del mensaje de inicio.	93
Figura 3-33: Diagrama de flujo del estado habilitación de interrupción de TMR1	94
Figura 3-34: Diagrama de flujo del estado ciclo normal del programa.	95
Figura 3-35: Diagrama de flujo del vector de interrupción.	96
Figura 3-36: Diagrama de estado de la aplicación IVIGED	97
Figura 3-37: Bloque de configuración del puerto serial.	98
Figura 3-38: Parámetros utilizados para la configuración del puerto serial.	98
Figura 3-39: Interfaz de usuario para la configuración inicial.	99
Figura 3-40: Código de asignación de números celulares al array de modelación.	100
Figura 3-41: Asignación de los números celulares.	101
Figura 3-42: Asignación del número de celular de inicio.	102
Figura 3-43: Interfaz de usuario para la configuración inicial del sistema.	102
Figura 3-44: Bloque de escritura en el puerto ante cualquier petición de datos o configuración inicial.	104
Figura 3-45: Código de escritura y lectura en el puerto de comunicación para el envío de mensajes.	104
Figura 3-46: Configuración del puerto serial para su lectura.	105
Figura 3-47: Código para la evaluación del tipo de mensaje recibido.	106
Figura 3-48: Código de identificación del remitente del SMS	107
Figura 3-49: Case de habilitación para la organización de datos.	107
Figura 3-50: Código de organización de datos.	108
Figura 3-51: Bloque de respuestas a las peticiones.	109
Figura 3-52: Código de extracción de información en el array que contiene los datos parte 1	109
Figura 3-53: Código de extracción de información en el array que contiene los datos parte 2	110
Figura 3-54: Interfaz de visualización de los datos.	110

Figura 3-55: Código para la visualización de los datos y respuestas a las solicitudes.....	111
Figura 3-56: Código selector para el tipo de respuesta.....	111
Figura 3-57: Código de selección de entrada en el selector para el tipo respuesta.....	112
Figura 3-58: Código para asignación del número del ingeniero en la línea de comandos enviados al modem.	113
Figura 3-59: Código para las respuestas a las solicitudes y hora de envío diaria de los datos.	113
Figura 3-60: Código para guardar en un documento de texto la información de las variables.....	114
Figura 3-61: Interfaz para localización del documento de texto.....	114
Figura 3-62: Código de configuración inicial, escritura – lectura en el puerto para peticiones y hora de envío de datos.....	115
Figura 3-63: Código de identificación del remitente del mensaje, tipo de mensaje, organización de datos, peticiones y visualización de datos.....	115
Figura 4-1: Prototipo de pruebas para configuración de modem GSM	117
Figura 4-2: Modulo SISDAQ	118
Figura 4-3: Especificaciones del modulo SISDAQ	119
Figura 4-4: Vi de simulación de planta embotelladora de alcohol isopropilico.	120
Figura 4-5: Interfaz de usuario para la configuración inicial del sistema, visualización y gestionamiento de los datos.	120
Figura 4-6: Interfaz de usuario para la configuración inicial.....	121
Figura 4-7: Interfaz de usuario para la visualización de los datos.	122
Figura 5-1: Comando AT	125
Figura 5-2: Comando IPR	125
Figura 5-4: Registro en la red-Comando COPS	126
Figura 5-3: Operadores disponibles-comando COPS.....	126
Figura 5-5: Comando ate.....	127
Figura 5-6: Error en comandos para SMS	127
Figura 5-7: Comando CMGF	128
Figura 5-8: Comando CMGS	128
Figura 5-9: Comando CNMI	129
Figura 5-10: Mensaje recibido.....	129
Figura 5-11: Comando AT&W	130
Figura 5-12: Medición de corriente del modem GSM.....	131
Figura 5-13: Diagrama de bloques de conexión de elementos para evaluación de módulo SISDAQ.....	132
Figura 5-14: Diagrama de flujo de algoritmo simulación de TE	133
Figura 5-15: Diagrama de flujo de algoritmo para aplicación 1 en prueba experimental.....	134
Figura 5-16: Interfaz de visualización de simulador de terminal de datos	135
Figura 5-17: Recepción de mensajes en Nokia PC Suite.....	136
Figura 5-18: Envío de un SMS con la configuración inicial al número de celular de inicio.	138
Figura 5-19: Mensaje recibido con la plantilla de inicio.	139
Figura 5-20: Envío de SMS a la aplicación IVIGED	140
Figura 5-21: Interfaz de visualización de datos.....	140
Figura 5-22: Envío de mensaje de texto con plantilla de petición de datos.....	141
Figura 5-23: Mensaje recibido con la información de los datos solicitados.....	141
Figura 5-24: Mensaje de petición total enviado.	142
Figura 5-25: Mensaje recibido como respuesta a la solicitud total de datos.....	142
Figura 5-26: Hora de envío diaria de todos los datos.	143
Figura 5-27: Mensaje recibido de la aplicación IVIGED con la información de todos los datos.....	143

ÍNDICE DE TABLAS

Tabla 2-1: Cobertura geográfica de operadores de telefónica móvil en Colombia. Fuente: (Contraloría general de la república, 2010).	10
Tabla 2-2: Sintaxis extendida.	26
Tabla 2-3: Descripción de pines del modulo SIMCOM SIM300.....	30
Tabla 2-4: Niveles lógicos de voltaje, Fuente: (SIMCOM, 2007).....	37
Tabla 2-5: Señales de la interfaz de SIM, Fuente: (SIMCOM, 2007)	39
Tabla 3-1: Voltajes soportador por el puerto serial asíncrono del módulo GSM	72
Tabla 3-2: Descripción de pines modem SIMCOM 300.....	77
Tabla 3-3: Características de los PIC16F876A y 16F877A.	85
Tabla 3-4: Tabla de verdad para la selección del tipo de respuesta a las solicitudes.....	112
Tabla 5-1: Respuesta de reguladores ante variaciones de voltaje en la entrada.....	130
Tabla 5-2: Medición de corriente consumo del modem GSM por envío de mensaje.....	131
Tabla 5-3: Tabla de comparación de tiempo de envío y calidad de la información	137
Tabla 5-4: Comparación con otros sistemas de monitoreo.....	145

CAPÍTULO I

1 IDENTIFICACIÓN DE PROYECTO

A continuación se presentan las secciones que componen la identificación del proyecto SISTEMA DE MONITOREO REMOTO DE UN PROCESO INDUSTRIAL EMPLEANDO LA REDES DE TELEFONÍA MÓVIL CELULAR EN COLOMBIA. En este capítulo se describe detalladamente la definición del problema, justificación, objetivos, alcances y limitaciones de proyecto.

1.1 TITULO

SISTEMA DE MONITOREO REMOTO DE UN PROCESO INDUSTRIAL EMPLEANDO LA RED DE TELEFONIA MOVIL EN COLOMBIA

1.2 DEFINICIÓN DEL PROBLEMA

La información es esencial al momento de tomar decisiones. La información oportuna y precisa permite:

- Aprender de las experiencias: al analizar los resultados adquiridos se obtienen nuevos conocimientos que se traducen en nuevas experiencias.
- Identificar las mejores oportunidades para tomar una decisión: al conocer el estado de un sistema y su funcionamiento en cada una de sus partes, se puede decidir con seguridad el mejor momento en que se puedan aplicar nuevos cambios al sistema.
- Evitar situaciones de peligro o de riesgo: al conocer el funcionamiento y el estado de un sistema, se pueden identificar situaciones de riesgo y peligro, que puedan ser evitadas aplicando las respectivas medidas de aseguramiento.

Debido a esto se hace indispensable adaptarse a las nuevas tendencias en las tecnologías de la información y comunicación, porque ofrecen servicios y calidad en el manejo de la información.

Muchas de las industrias colombianas tienen plantas de operación en lugares remotos limitados geográficamente en donde se obtiene la materia prima, como ejemplo de ello tenemos la industria carbonífera, las centrales hidroeléctricas, refinadoras de aceite, refinadoras de petróleo entre otras, esta distancia impide una comunicación óptima entre la planta y una central de monitoreo y por consiguiente un mal manejo en la información.

Debido a la importancia que implica obtener información de manera rápida y oportuna, es necesario integrar este proceso de forma óptima a la empresa y así poder ejercer dominio sobre la información, y unir en un único sistema los recursos y servicios distribuidos en los distintos equipos.

Una solución a este problema es instalar una red dedicada de comunicación entre la empresa y la planta, pero debido a la distancia o aislamiento geográfico que existe entre el proceso industrial y la central de monitoreo de la empresa, trae como consecuencias:

- Alto costo en la implementación y administración de la red.
- Personal exclusivo para el manejo de la misma.

- No sería posible obtener información desde cualquier lugar y momento, sobre los procesos que se llevan a cabo en la planta, salvo solo en la central de monitoreo de la empresa.

Esto lleva a formular la siguiente pregunta:

¿Qué sistema podría obtener información de un proceso industrial en cualquier lugar y momento, que implique bajo costo en la implementación y gestione la información de forma automática?

1.3 JUSTIFICACIÓN

La idea de desarrollar un sistema de monitoreo remoto de un proceso industrial empleando la red de telefonía móvil en Colombia, nace de la necesidad de integrar un proceso aislado geográficamente a un sistema centralizado, con el fin de obtener información vital para el desarrollo del proceso y el gestionamiento del mismo, es decir, un sistema capaz de recibir los datos del proceso, almacenarlos, procesarlos y realizar notificaciones sobre las actividades que se deben realizar para mejorar el proceso industrial.

Es necesario que un sistema de monitoreo permita poder consultar un parámetro en cualquier lugar y momento para estar informado o actualizado del estado de un proceso, además el sistema debe procesar de forma automática la información que es adquirida y una vez ocurra un evento inesperado genere una notificación, informando a un operario de las posibles fallas, lo que permite que el monitoreo sea autónomo y no requiera de ninguna supervisión.

A nivel industrial las empresas cada vez se tornan más dependientes de la tecnología de monitoreo y control de un proceso. Para los ingenieros y técnicos es indispensable contar con alguna herramienta que brinde de forma rápida y efectiva la información de cómo el proceso industrial desarrolla su funcionamiento, sin tener la necesidad de depender de personal que vigile el estado de las variables del proceso. Los jefes de soporte obtendrían la información del proceso desde cualquier lugar en donde exista cobertura de la señal de la red de telefonía móvil.

BENEFICIOS TECNOLÓGICOS

Con la implementación de este sistema se aprovechan todas las ventajas que ha permitido la telefonía móvil celular y el SMS, se optimiza el manejo de la información hasta el punto en que es posible acceder a la información en cualquier momento y lugar.

Debido a que posee una interfaz de comunicación serial RS-232 le otorga versatilidad y flexibilidad ya que no tiene limitaciones en cuanto al tipo de proceso, solo es necesario que el terminal de datos sea compatible con este tipo de tecnologías, como los son la mayoría de computadoras, PLC y microcontroladores.

BENEFICIOS ECONÓMICOS

Al ser un sistema totalmente automatizado requiere de poca mano de obra.

Debido a que la tecnología móvil celular está diseñada para suplir las necesidades de comunicación de cualquier tipo de usuario ya sea una empresa o una persona, se hace notorio el ahorro en la implementación del sistema ante cualquier necesidad.

El consumo de energía es sumamente bajo.

BENEFICIOS SOCIALES

Aumento en la seguridad de las instalaciones y la protección a los trabajadores, debido a que se notifica sobre cualquier alarma.

No se produce ningún tipo de contaminación.

BENEFICIOS INSTITUCIONALES

Para la Universidad del Magdalena, se convierte en una auténtica innovación tecnológica, que le permite fortalecer los antecedentes de aplicación de las nuevas tecnologías inalámbricas.

El desarrollo de este proyecto fortalece la línea de investigación orientada a aplicaciones móviles del Grupo de Investigación en Desarrollo Electrónico y Aplicaciones Móviles (GIDEAM).

Con la realización del estudio y puesta en marcha del proyecto, se adquieren conocimientos científicos y tecnológicos, los cuales serán consignados en la biblioteca para beneficio de los alumnos pertenecientes a la comunidad universitaria.

BENEFICIOS ACADÉMICOS

Permite crear un antecedente en proyectos relacionados con la telefonía móvil celular y aplicaciones en la plataforma de programación grafica LabVIEW, basados en el servicio (SMS) para futuros trabajos en el área de las telecomunicaciones.

1.4 OBJETIVOS

OBJETIVO GENERAL

Diseñar y desarrollar un sistema de monitoreo remoto de un proceso industrial empleando la red de telefonía móvil en Colombia.

OBJETIVOS ESPECÍFICOS

- Recolectar y analizar la información sobre las distintas tecnologías de monitoreo de un proceso industrial que utilizan la red de telefonía móvil celular.
- Obtener la información de las variables del proceso industrial a través del terminal de datos para su análisis.
- Establecer la comunicación entre el modem GSM y el proceso industrial
- Elaborar la aplicación servidora para la gestión de la información obtenida.
- Evaluar el funcionamiento del sistema total y cada de una de sus partes.

1.5 ALCANCES Y LIMITACIONES

El proyecto tiene como alcance desarrollar un sistema de monitoreo remoto de un proceso industrial empleando la red de telefonía móvil en Colombia para integrar procesos industriales aislados geográficamente a una red centralizada.

El monitoreo se limita a las zonas de cobertura que brinda las empresas de telefonía móvil celular para la red GSM.

Debido a que todos los operadores de telefonía móvil ofrecen el servicio de SMS, la escogencia de un determinado operador estará definida por la cobertura que se brinde en el

área donde está ubicada la planta industrial y la comodidad de planes y servicios que se brinde por parte del operador de telefonía móvil celular.

La velocidad con que la información se transmitirá será definida por el tipo de servicio que ofrece el operador.

El tipo de modem escogido para la transmisión de los datos es un modem GSM/GPRS, debido a que las redes que prestan el servicio GSM aunque no ofrecen una alta velocidad de datos, tienen una amplia cobertura frente a las redes 3G y EDGE

La comunicación entre el servidor cliente es realizada a través del servicio SMS, por lo que existe un límite en la cantidad de variables que se pueden monitorear.

Este sistema sólo está orientado a monitorear un proceso industrial, por lo que no realiza ninguna actividad de control del proceso.

Las aplicaciones del servidor del sistema se desarrollaran en su totalidad con software licenciado por la Universidad del Magdalena.

CAPÍTULO II

2 MARCO DE REFERENCIA

En este capítulo se encuentra toda la información relacionada con las bases y fundamentos teóricos para llevar a cabo los objetivos trazados. Se enmarcan aquellos elementos que se utilizaron como soporte principal del estudio realizado.

2.1 ANTECEDENTES

El telecontrol se conoce como el control de operaciones a distancia utilizando la transmisión de información por medio de técnicas de telecomunicación (Acedo, 2007). Esto implica una comunicación entre dos o más dispositivos, muchas de estas aplicaciones se llevan a cabo a través de Internet, redes Ethernet, GSM, radio u otros medios. Por medio de esta comunicación no sólo se puede controlar sino también monitorear un determinado proceso, es decir, en todo lazo de control se tienen datos o variables de entrada, la característica principal del telecontrol es que estas variables de entrada son transmitidas de forma remota para realizar el procesamiento adecuado, y así poder arrojar una señal de control.

Uno de los avances más importantes en telecontrol es implementado hoy día por la empresa Nubitek que ha desarrollado el producto Nubitek SMS Manager, simplemente consiste en la comunicación de las instalaciones de un proceso industrial por medio de mensajes de texto, agregando como periférico a un PLC, un modem GSM (Rivas, 2010), para ello Nubitek ofrece una librería que es compatible al PLC S7-200 de Siemens y que es muy fácil de usar.

La empresa Siemens ofrece el módulo TC35 que permite a la familia de PLC de Siemens enviar y recibir SMS utilizando la red GSM pero no da garantía de que pueda ser utilizado con PLC de marca diferente.

El proyecto “Sistema embebido para la conexión de un PLC Siemens S7-200 a la red GSM” de la Universidad de Sevilla presenta los resultados de la investigación realizada y un prototipo para la conexión del PLC a la red GSM lo cual se logró satisfactoriamente según las pruebas y resultados del proyecto.

El sistema SCADA es una aplicación de software diseñada con la finalidad de controlar y supervisar procesos a distancia, permite la gestión y control de cualquier sistema local y remoto por medio interfaz gráfica que comunica al usuario con el sistema (Penin, 2008).

A partir de las investigaciones expuestas queda en evidencia la necesidad de integrar los aportes de las TICs para mejorar la prestación de servicios y la calidad de la información empleando las ventajas que ofrece la telefonía móvil celular actual.

2.2 AMBITO DE APLICACIÓN

Es evidente que el presente proyecto está destinado a formar parte de aplicaciones de ámbito industrial de pequeño tamaño y que se encuentren aisladas geográficamente, normalmente este tipo de procesos conlleva la siguiente arquitectura



Figura 2-1: Entorno Sistema de automatización, Fuente: (GENIA, 2011)

Todo proceso automatizado comprende una parte de supervisión, una de control y una parte operativa, en la parte de mando se supervisa y coordinan las acciones de la parte operativa, en la parte de control es la encargada de traducir las órdenes del panel de mando en señales eléctricas para que los actuadores realicen la acción asignada y también se encarga de leer las señales eléctricas de los sensores y transductores para informar a la parte de mando sobre el estado del proceso.

Pero se puede traspasar la frontera del ámbito industrial al ámbito doméstico en aplicaciones de domótica, debido a la flexibilidad del sistema, donde se puede implementar en la gestión de alarmas técnicas tales como, fuga de gas, incendio e intrusos.

2.3 BASES TEÓRICAS

A continuación se presentan los conceptos estudiados y analizados durante la ejecución del proyecto:

2.3.1 GSM

GSM (Global System for Mobile Communication) es un sistema estándar, libre de regalías, de telefonía móvil digital que ofrece los siguientes servicios:

- Transmisión y recepción de voz.

- Transmisión y recepción de datos.
- Envío y recepción de mensajes cortos (SMS).

El sistema GSM está basado en una arquitectura celular, que permite una mejor utilización del espectro con base en la réplica de las unidades fundamentales o celdas con las que se cubre el área que se quiere comprender. El tamaño de las celdas es un parámetro de diseño que se calcula con base al número medio de usuarios y el porcentaje de utilización de la estación base (Gonzalez Gomez, 2002).

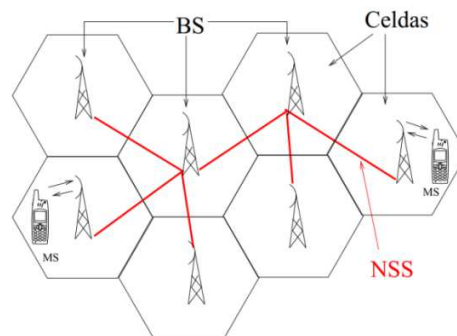


Figura 2-2: Sistema Celular, Fuente: (Gonzalez Gomez, 2002)

¿POR QUE GSM?

En el presente apartado se presenta un análisis de la cobertura de las distintas redes de comunicaciones disponibles en la actualidad por parte de los operadores de telefonía móvil celular MOVISTAR, TIGO y COMCEL en Colombia.

OPERADOR	MUNICIPIOS CON RED GSM	MUNICIPIOS CON RED 3.5G
TIGO	903	63
MOVISTAR	923	67
COMCEL	932	963

Tabla 2-1: Cobertura geográfica de operadores de telefónica móvil en Colombia. Fuente: (Contraloría general de la república, 2010).

Se escoge red GSM a pesar de que ofrece velocidades de datos muy inferiores a las redes 3G y 4G (Tanenbaum, 2003), además tiene una amplia cobertura y su velocidad se ajusta a los requerimientos de la aplicación para el soporte de servicios de mensajería corta SMS.

ARQUITECTURA DE UNA RED GSM

La arquitectura GSM se compone de los siguientes tres sistemas:

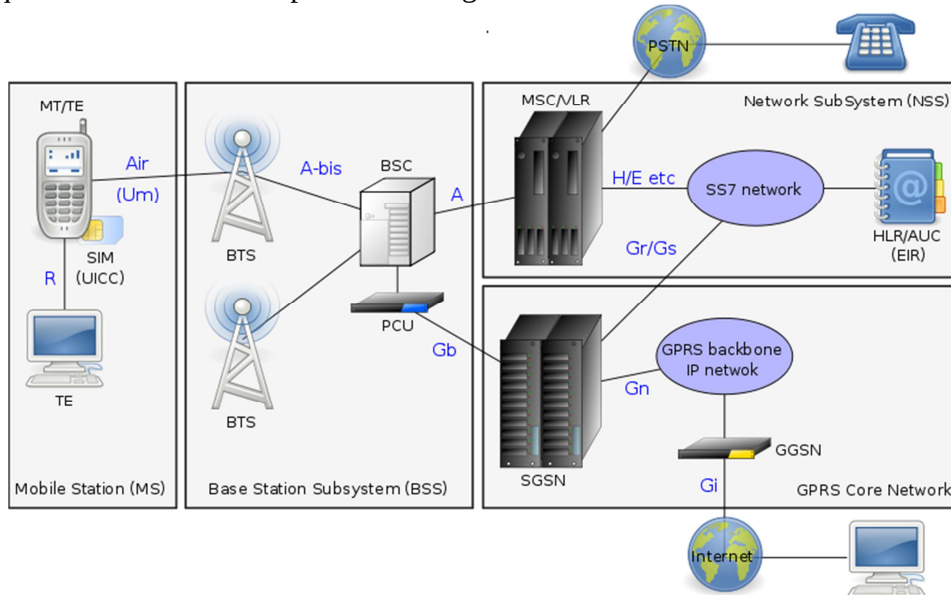


Figura 2-3: Arquitectura de una red GSM, Fuente: (Wikipedia, 2011)

- ESTACION MOVIL



Figura 2-4: Estación Móvil, Fuente: (Tecnoviles, 2011)

La Estación móvil (MS, Mobile Station) es el equipo utilizado por el usuario de GSM para acceder a los servicios proporcionados por la red a través de la interfaz Um (Conzales Castellanos, Jauregui Suarez, & Melchor Mejia, 2002). La MS proporciona la plataforma física para el acceso del equipo móvil, pero es “anónima” y no puede funcionar con la red hasta que se le “personaliza” mediante la inserción de una “tarjeta inteligente” denominada módulo de identidad de abonado, SIM (Subscriber Identity Module). Ver figura 2-4.

La MS debe desempeñar las siguientes funciones:

- Proporcionar una interfaz de comunicación entre los usuarios y la red vía radio.

- Realizar la transmisión/recepción de las informaciones de usuario y de señalización a través de esa interfaz radio.
- Efectuar el inicio de la conexión de red.
- Realizar la sintonización de frecuencias y seguimiento automático de las estaciones base en cuya zona se encuentre.
- Efectuar funciones de procesamiento de la voz: conversión analógico/digital y viceversa.
- Realizar la adaptación de interfaces y velocidades para las señales de datos.

La relación entre MS y este subsistema se realiza a través de una “interfaz aire” llamada también interfaz de radio, que se designa abreviadamente como Um.

- **SUBSISTEMA DE ESTACIÓN BASE**

El subsistema de estación base (BSS, Base Station Subsystem) comprende las funciones de capa física, para la interconexión con las MS a través de una interfaz radio Um. Para esto hace uso de canales lógicos, los cuales son estructuras de datos y protocolos que realizan funciones de intercambio de información necesaria para (Conzales Castellanos, Jauregui Suarez, & Melchor Mejia, 2002):

- Seguimiento / Localización de las MS y aviso a las mismas.
- Establecimiento de las llamadas.
- Mantenimiento de las comunicaciones establecidas.
- Supervisión y control de calidad.
- Facilidades operativas.

El BSS comprende el conjunto de equipos utilizados para proporcionar cobertura radioeléctrica en el área celular. En este sistema se identifican dos unidades funcionales:

- Controladora de base estación, **BSC** (Base Station Controller): Se llevan a cabo la mayoría de las funciones de control. Este es el encargado de la gestión de varios BTS en lo relativo a los recursos de radio: asignación, utilización, liberación de frecuencias, traspasos y funcionamiento con saltos de frecuencias.
- Estación Base, **BTS** (Base Transceiver Station): Constituidas por los equipos transmisores-receptores de radio (Transceptores), los elementos de conexión al sistema radiante (combinadores, multiacopladores, cables coaxiales), las antenas y las instalaciones accesorias.

- SUBSISTEMA DE RED Y CONMUTACIÓN

El Subsistema de Red y Conmutación (NSS, Network and Switching Subsystem): realiza las funciones de conmutación y encaminamiento de las llamadas en el sistema GSM, además de la gestión de las bases de datos con la información relativa a todos los abonados al servicio. El NSS se encarga de establecer la comunicación entre usuarios móviles mediante la conmutación interna de red de un operador o entre usuarios del sistema GSM y usuarios de otras redes de telefonía, (Conzales Castellanos, Jauregui Suarez, & Melchor Mejia, 2002) ya sea de telefonía fija o de telefonía móvil de otros operadores, está compuesta por:

- MSC (Mobile Services Switching Center): Es una central telefónica que realiza las funciones de encaminamiento y conmutación de llamadas para las MS situadas en su demarcación (área central) (Conzales Castellanos, Jauregui Suarez, & Melchor Mejia, 2002). Sus funciones son:
 - o Procedimientos para la localización y registro de abonados y su actualización.
 - o Gestión de llamadas y aviso a la MS de llamadas entrantes.
 - o Procedimientos para el traspaso de llamadas.
 - o Gestión de los protocolos de señalización con el BSC e intercambio de información de señalización con el protocolo MAP del SS7.
 - o Establecimiento de circuitos con conexión entre las MS y las PSTN y otras redes.
 - o Recopilación de datos de tráfico y facturación.
- GMSC (Gateway Mobile Services Switching Center): Es el elemento de interconexión con otras redes.
- BASES DE DATOS
 - o Registro General de Abonados HLR (Home Location Register): Es una base de datos, donde están inscritos todos los clientes de un operador. El HLR contiene toda la información administrativa de cada abonado, junto con los datos de localización del mismo (Conzales Castellanos, Jauregui Suarez, & Melchor Mejia, 2002).

La información que contiene se puede clasificar como sigue:

 - Datos permanentes, como lo son la identificación del abonado (IMSI y MSISDN).

- Datos transitorios, actualizables como lo son la localización de la MS a fin de encaminar las llamadas entrantes al MSC donde está registrada la MS.
- o Registro de abonados itinerantes VLR (Visitor Location Register): Es una base de datos asociada a un MSC que almacena información dinámica sobre los usuarios visitantes en la zona controlada por ese MSC. Cuando un MS itinerante entra a una zona de MSC, este centro notifica esa situación a su VLR asociado. La estación base ejecuta un proceso de inscripción o registro y recibe una dirección de visitante que se traduce en un numero de ruta, MSRN (Mobile Subscriber Roaming Number), que sirve para encaminar las llamadas destinadas a esa MS (Conzales Castellanos, Jauregui Suarez, & Melchor Mejia, 2002).

La información contenida por el VLR es básicamente:

- Identidad permanente del abonado (IMSI).
- Identidad temporal del abonado (TMSI).
- Número telefónico del abonado (MSISDN).
- Numero de encaminamiento (MSRN).
- Identidad de la zona de localización (LAI).
- Datos del abonado del usuario (Servicios, restricciones).

• UNIDADES FUNCIONALES DE SEGURIDAD

Son las encargadas de proteger a los abonados de las intrusiones y al operador de accesos indebidos. Existen dos organismos específicos para la realización de las funciones de seguridad:

- Registro de identidad de equipos EIR (Equipment Identity Register): Es una base de datos que contiene las identidades de los equipos móviles, IMEI (International Mobile Equipment Identity), que identifica los equipos por sus códigos de fabricación y homologación. Cuando un teléfono móvil trata de realizar una llamada, el MSC consulta al EIR la validez del IMEI de ese equipo (Conzales Castellanos, Jauregui Suarez, & Melchor Mejia, 2002).
- Centro de autenticación AUC (Authentication Center): Es una base de datos donde se guardan las identidades (IMSI) de los abonados junto con la clave secreta de identificación de cada usuario. Este está asociado al HLR y proporciona la información necesaria para la validación de los usuarios por parte de la red; para ello tiene almacenado dos algoritmos denominados A3 y A8. El algoritmo A3 se utiliza para la autenticación y el A8 para la generación de la clave de cifrado de la información.

2.3.2 EL SERVICIO SMS

Es un servicio disponible en las Estaciones Móviles que permite el envío de mensajes cortos de texto entre Estaciones Móviles y/o una entidad receptora, que comienza a operar en la red GSM europea en el año 1991, con una cantidad máxima de 160 caracteres. Existen dos tipos de SMS disponibles: **mensajes de difusión** (cellbroadcast), usados para enviar información de control unidireccionales a los terminales móviles que tengan habilitado el servicio dentro de una celda, y **mensajes punto a punto** (point-to-point). Los SMS punto a punto son enviados desde un móvil a otro (Alvarado S., Vinnett P., & Grote H., 2004).

Debido a que la transmisión de los SMS se realiza usando los canales de control de GSM es posible realizar una llamada y recibir un mensaje en forma simultánea, además este servicio se almacena en caso de ser necesario, lo cual permite que si el equipo del destinatario se encuentra apagado, o fuera del área de cobertura, el mensaje se almacene en la red hasta que pueda ser retirado por el destinatario.

SMS está especificado en el estándar técnico GSM 03.40 v7.5.0. Este estándar define los servicios y elementos de servicio, la arquitectura de red, la funcionalidad del centro de servicios, la funcionalidad del MSC (con respecto a los SMS), la funcionalidad del SGSN (con respecto a los SMS), los requisitos de enrutamiento, los protocolos y las capas de protocolo (Institute European Telecommunications Standards - ETSI TS 100 901 V7.3.0 , 1999-11).

El servicio SMS permite la transferencia de un mensaje corto entre una estación móvil y una entidad para la recepción y el envío de mensajes a través de un centro de servicio como se observa en la figura 2-5. El servicio final ofrecido es una comunicación entre la estación móvil y la entidad, ésta última puede ser otra estación móvil o puede estar situada en una red fija (Gonzalez Gomez, 2002).

En el caso de envío de un mensaje entre dos móviles, ambas partes son Estaciones Móviles. Cuando se envía un mensaje para solicitar algún tipo de servicio, (en el caso de conocer el saldo actual) un extremo es una estación móvil y el otro es un servidor que atiende la solicitud.

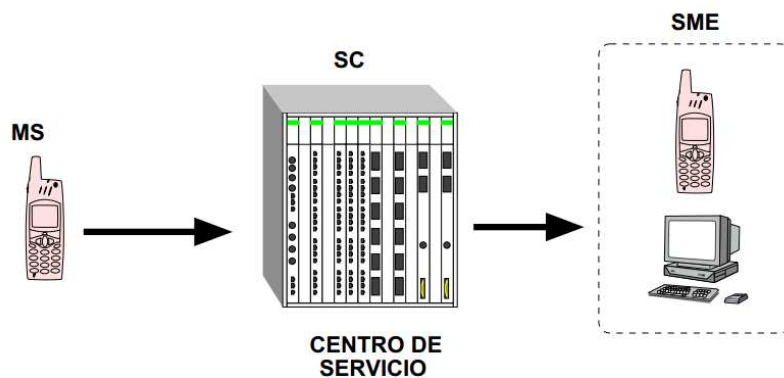


Figura 2-5: Servicio SMS. Fuente: (Gonzalez Gomez, 2002)

En la norma GSM sólo se especifica la parte de comunicaciones entre las Estaciones Móviles y el centro de servicio. La comunicación entre el centro de servicio y las entidades fijas, queda fuera del ámbito de esta norma como se observa en la figura 2-6.

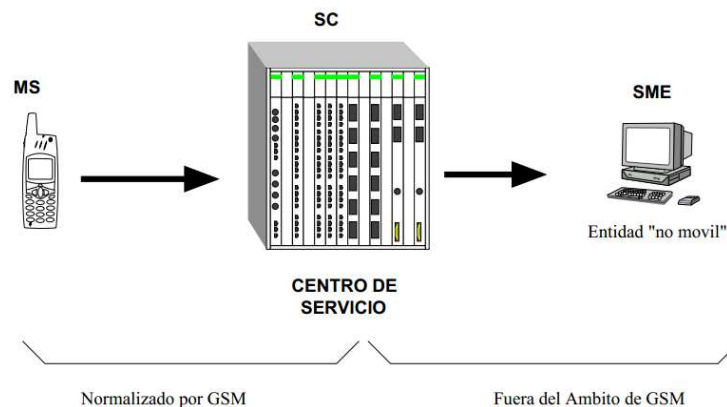


Figura 2-6: Envío de un SMS entre una MS y una entidad fija, Fuente: (Gonzalez Gomez, 2002)

Existen dos tipos de servicios SMS punto a punto como se observa en la figuras 2-7 y 2-8, y que se enuncian a continuación (Gonzalez Gomez, 2002):

- **SM MT (Short Message Mobile Terminated Point-to-Point):** Servicio de entrega de un mensaje desde el centro de servicios hasta una Estación Móvil, obteniéndose un informe sobre lo ocurrido. Son los que se originan en una aplicación y se envían a un terminal móvil, pueden ser enviados de forma masiva o en respuesta a una petición del usuario, en cuyo caso el mensaje suele personalizarse para el usuario.
- **SM MO (Short Message Mobile Originated Point-to-Point):** Servicio de envío de un mensaje desde una Estación Móvil hasta un centro de servicio, obteniéndose un informe sobre lo ocurrido. El usuario envía el mensaje a un número corto, que

previamente ha sido contratado a los operadores móviles por empresas que prestan servicios y contenidos de SMS.

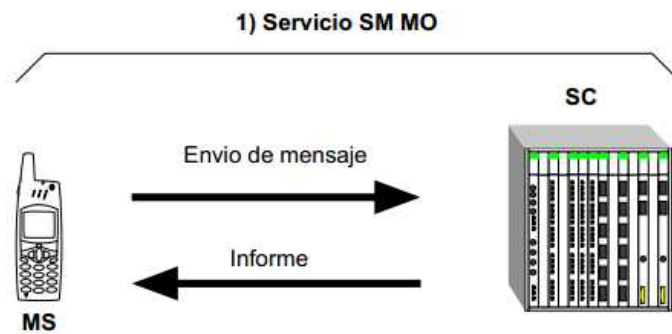


Figura 2-7: Servicios básicos SM MO, Fuente: (Gonzalez Gomez, 2002)

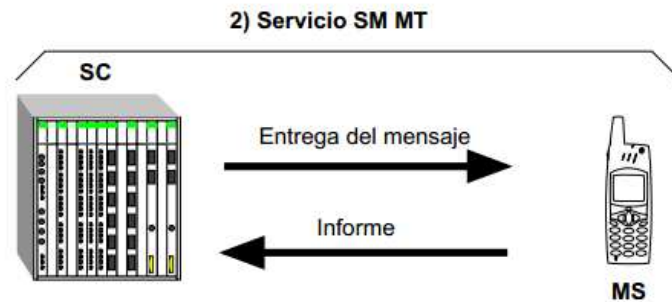


Figura 2-8: Servicios básicos SM MT, Fuente: (Gonzalez Gomez, 2002)

ARQUITECTURA DE RED PARA EL SERVICIO SMS

La estructura básica de la red para el servicio SMS se muestra en la figura 2-9:

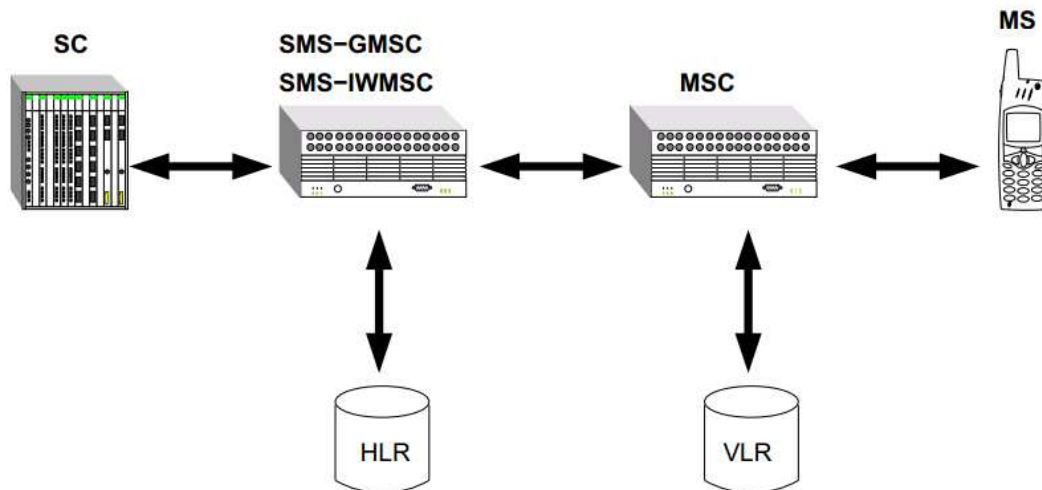


Figura 2-9: Estructura básica de la red para la transferencia de mensajes cortos, Fuente: (Gonzalez Gomez, 2002)

A continuación se describen brevemente cada uno de los componentes:

- **MS (Mobile Station):** Estación móvil.
- **MSC (Mobile Services Switching Center):** Centro de Conmutación.
- **SC (Service Center):** Centro de Servicio.
- **MT (Mobile Terminated):** Terminal Móvil.
- **SME (Short Messaging Entity):** Entidad que puede enviar o recibir mensajes cortos, pudiendo estar localizada en la red fija, una Estación Móvil, u otro Centro de Servicio.
- **SMSC (Short Message Service Center):** Es el responsable de la transmisión y almacenamiento de un mensaje corto, entre la Entidad que puede enviar o recibir mensajes cortos y una Estación Móvil.
- **SMS-GMSC (Short Message Service - Gateway Mobile - Services Switching Center):** Centro de Conmutación que sirve de Pasarela para el servicio de mensajes cortos.
- **SMS-IWMSC (Short Message Service – Inter Working Mobile - Services Switching Center):** Centro de conmutación de interconexión entre PLMN y el Centro de Servicios.

- **HLR (Home Location Register):** Es una base de datos usada para el almacenamiento permanente, gestión de los usuarios y el perfil del Servicio de Mensajería.
- **VLR (Visitor Location Register):** Es una base de datos que contiene información temporal de los usuarios.

Para la descripción detallada de la arquitectura, se utiliza un modelo de capas, en el que cada capa o nivel proporciona un servicio a la capa superior, y este servicio se implementa mediante el protocolo correspondiente. La arquitectura se divide en cuatro niveles que se pueden observar en la figura 2-10, estos niveles son (Gonzalez Gomez, 2002):

- **SM-AL (Short Message Application Layer):** Nivel de aplicación.
- **SM-TL (Short Message Transfer Layer):** Nivel de transferencia. Servicio de transferencia de un mensaje corto entre una Estación Móvil y un Centro de Servicios y obtención de los correspondientes informes sobre el resultado de la transmisión.
- **SM-RL (Short Message Relay Layer):** Nivel de repetición. Proporciona un servicio al nivel de transferencia que le permite enviar TPDU (Transfer Protocol Data Units) a su entidad gemela.
- **SM-LL (Short Message Lower Layers):** Niveles inferiores.

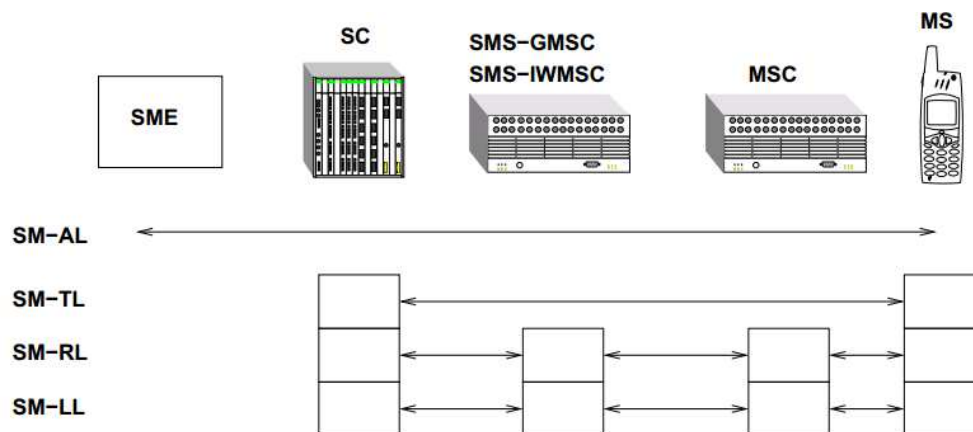


Figura 2-10: Niveles y servicios para el envío de mensajes cortos, Fuente: (Gonzalez Gomez, 2002)

OPERACIONES PARA EL ENVÍO DE SMS

La Parte de Aplicación Móvil (Mobile Application Part, MAP), define las operaciones necesarias para dar soporte al SMS. Ambos estándares, el americano y el europeo han definido el MAP usando los servicios de la parte de aplicaciones de capacidad de

transacciones. El estándar americano publicado por la asociación de la industria de las telecomunicaciones (Telecommunication Industry Association, TIA) define la capa MAP en el estándar IS-41, y el Instituto Europeo de Normas de Telecomunicaciones (European Telecommunication Standards Institute, ETSI) lo identifica como GSM MAP (Tarazona Anteliz, Casadiego, & Alexander, 2008).

Operaciones básicas del MAP necesarias para proporcionar servicio de mensajes cortos punto a punto:

- **Solicitud de Información de Encaminamiento.** En esta operación el Centro de Servicios extrae la información de la base de datos para determinar el servicio de la Estación Móvil tratada. Este proceso se realiza antes de entregar el mensaje, y se lleva a cabo usando los mecanismos establecidos en el estándar IS-41 y GSM respectivamente.
- **Envío del Mensaje Punto a Punto.** Mecanismo importante para transmitir un mensaje corto hacia el Centro de Conmutación que sirve a la Estación Móvil e intenta enviar un mensaje a una Estación Móvil siempre que se encuentre registrada. La operación del envío del mensaje corto proporciona un servicio de envío confirmado. La operación trabaja conjuntamente con el subsistema de la Estación Móvil mientras el mensaje está siendo remitido del Centro de Conmutación hacia la Estación Móvil.
- **Indicación de Espera del Mensaje Corto.** Esta operación se activa cuando el intento de envío por parte del Centro de servicios falla debido a algún incidente temporal. Entonces el Centro de Servicios solicita al HLR que añada una dirección SMSC, para ser informado cuando la Estación Móvil indicada esté accesible.
- **Alerta del Centro de Servicio.** Esta operación hace que el HLR informe al SMSC que la MS es accesible en ese momento, el cual previamente ha intentado enviar un mensaje corto a la MS especificada.

Pasos para el envío de un SMS:

- El mensaje corto es enviado del SMC al SMSC.
- Después de completar su proceso interno, el SMSC pregunta al HLR y recibe de la misma información de encaminamiento del usuario móvil.
- El SMSC envía el mensaje corto hacia el MSC.

- El MSC extrae la información del usuario del VLR. Esta operación puede incluir un procedimiento de autenticación.
- El MSC transfiere el mensaje corto a la estación móvil y devuelve al SMSC el resultado de la operación que se está llevando a cabo.
- Si lo solicita el SMC, el SMSC retorna un informe indicando la salida del mensaje corto.

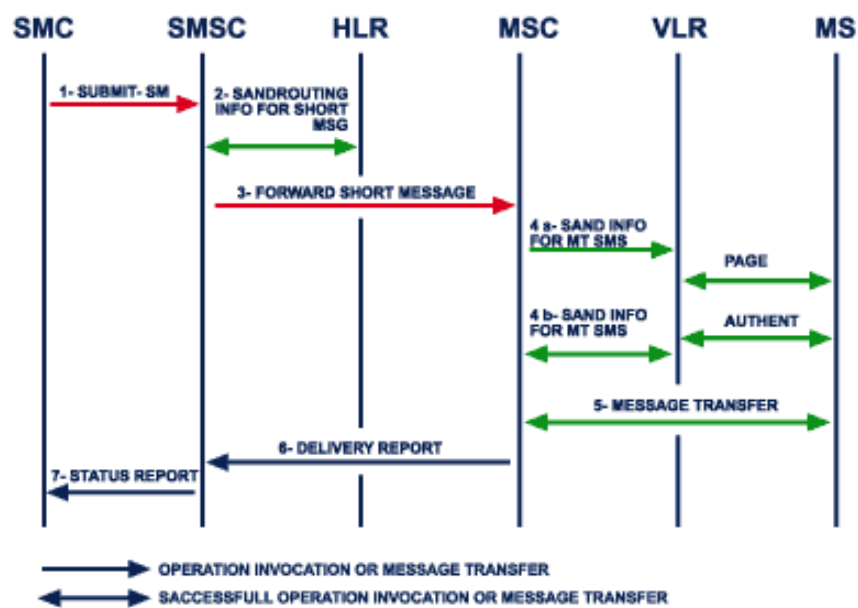


Figura 2-11: Operaciones en el envío de SMS, Fuente: (Escuela Politecnica Superior de Alcoy, 2012)

Pasos para la recepción de un SMS:

- La MS transfiere el mensaje corto al MSC.
- El MSC interroga al VLR para verificar que el mensaje transferido no viola los servicios suplementarios o las restricciones impuestas.
- El MSC envía el mensaje corto al SMSC.
- El SMSC entrega el mensaje corto al SME y reconoce al MSC el éxito del envío.
- El MSC devuelve a la MS el resultado de la operación de envío.

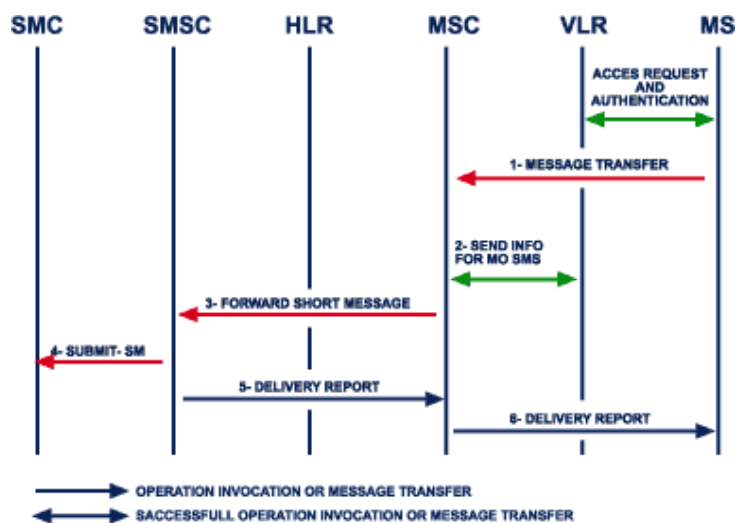


Figura 2-12: Operaciones en la recepción de SMS, Fuente: (Escuela Politecnica Superior de Alcoy, 2012)

PRINCIPALES APLICACIONES

Las principales aplicaciones basadas en el SMS son, los mensajes de persona a persona que utilizan los usuarios de la telefonía móvil celular para comunicarse entre ellos, el servicio de notificar al usuario de la telefonía móvil celular los nuevos mensajes de voz o de texto por medio de un SMS. Actualmente se ha realizado una unión entre el servicio de correo electrónico y SMS para que el usuario pueda recibir mensajes de alerta cada vez que tiene un mensaje de correo electrónico nuevo.

El servicio SMS es utilizado para solicitar el envío de información sobre alguna consulta por medio del terminal móvil hacia un servidor, ejemplo de esto es la consulta de saldo para el operador TIGO al marcar *10#. Recientemente se ha utilizado para solicitar también descargas de melodías y para participar o realizar votaciones en concursos de televisión.

El Servicio de Mensajes Cortos, puede usarse para administrar equipos en ambientes de supervisión remota. Esta aplicación proporciona información sobre el estado de los equipos.

ACCESO A LOS SERVICIOS SMS

Para acceder a los servicios de SMS es necesario el uso de software y hardware que lo permitan, las formas más comunes en la actualidad son:

- Algunos teléfonos se pueden conectar directamente a un PC y mediante un software propietario por ejemplo Nokia PC Suite, Sony Ericsson PC Suite, se puede acceder a los datos de móvil (agenda, tarjeta SIM, entre otros), así como enviar y recibir mensajes SMS.
- Utilizando un MÓDEM GSM, es un tipo especializado de módem, que acepta una tarjeta SIM, y opera a través de una suscripción a un operador de telefonía móvil celular, al igual que un teléfono móvil. Desde la perspectiva del operador de telefonía móvil celular, un modem GSM se parece a un teléfono móvil.
- Utilizando una plataforma del SMS, estas plataformas son servicios que ofrecen algunas páginas web e incluso algunos operadores de telefonía móvil celular, que permiten el envío y recepción de SMS desde la web.

En el proyecto se utiliza un modem GSM y un teléfono móvil como métodos de acceso al servicio. Por lo que se concentra en el estudio de los modem GSM.

2.3.3 MODEM GSM

Es un tipo especializado de módem, que acepta una tarjeta SIM, y opera a través de una suscripción a un operador de telefonía móvil, al igual que un teléfono móvil.

Mediante un MÓDEM GSM se puede conectar cualquier sistema digital a la red GSM, no sólo para enviar mensajes SMS sino también para transmitir datos. Existen dos tipos de MODEMS, según la aplicación que se quiera realizar. Ver figura 2-13.

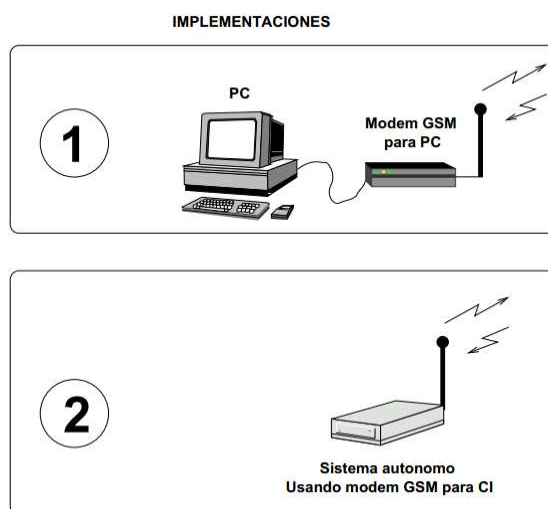


Figura 2-13: Tipos de módems, Fuente: (Gonzalez Gomez, 2002)

Para el desarrollo de la aplicación se escogió el modem SIMCOM SIM300, ver Figura 2-14.



Figura 2-14: Modem SIMCOM 300, Fuente: (SIMCOM, 2007)

MODEMS PARA CIRCUITO IMPRESO

Son de reducido tamaño y perfectamente apantallados que están preparados para ser incorporados dentro de un circuito impreso y que permiten desarrollar un hardware específico y que no depende de un PC, este tipo de modem es el que se utiliza en el presente proyecto.

MODEMS PARA PC

Tienen un tamaño también bastante reducido, y disponen de un conector DB9 hembra para conectarse al PC a través de un cable serie estándar. Son muy útiles para permitir que desde cualquier ordenador de una intranet se puedan enviar mensajes SMS.

INTERFAZ CON MODEM GSM

Los módems GSM no sólo se comportan de forma muy parecida a un modem normal, permitiendo el intercambio de datos con otro modem y utilizando los comandos AT originales, sino que incluyen muchas más características. Son como pequeños teléfonos móviles, que incluyen su propia tarjeta SIM para poder funcionar y por tanto permiten gestionar la base de datos de teléfonos, la lista de los mensajes SMS recibidos, enviar mensajes SMS, configurar diversos parámetros.

Para establecer comunicación continua con un modem GSM y utilizar los servicios de la red GSM es necesario comprender el lenguaje de Comandos AT, el cual es prácticamente un estándar abierto de comandos para configurar y parametrizar los módems, y es necesario

el desarrollo de una interfaz RS-232 adaptada a los niveles de tensión sugeridos por el modem GSM.

2.3.4 COMANDOS AT

Es un lenguaje desarrollado por la compañía HAYES COMMUNICATIONS que prácticamente se convirtió en estándar abierto de comandos para configurar y parametrizar módems. Los caracteres «at», que preceden a todos los comandos, significan «atención», e hicieron que se conociera también a este conjunto de comandos como comandos AT (Lázaro Laporta & Miralles Aguiñiga, 2005).

Los comandos AT con cadenas ASCII que comienzan por los caracteres AT y terminan con un retorno de carro. Cada vez que el módem recibe un comando, lo procesa y devuelve un resultado, que normalmente es una cadena ASCII salvo que hayamos indicado lo contrario. Al estar la comunicación en ASCII, pondremos utilizar un terminal de comunicaciones desde un ordenador para acceder al módem, bien para configurarlo, bien para hacer pruebas o bien para establecer una comunicación con otro módem (Lázaro Laporta & Miralles Aguiñiga, 2005).

SINTAXIS DE LOS COMANDOS AT

El prefijo AT debe ser cargado en el inicio de cada línea de comando, al terminar una línea de comando escribir enter <CR>, los comandos usualmente son seguidos por una respuesta que tiene la siguiente forma "<CR><LF><response><CR><LF>", donde (<CR><LF>) indica salto de línea (SIMCOM, 2005). Los tipos de sintaxis son:

- **Sintaxis Básica**

Este tipo de comando tiene la forma "AT<x><n>" o "AT&<x><n>" donde <x> es el comando y <n> es o son los argumentos del comando, un ejemplo de esto es ATE<n> que le dice al DTE si los caracteres recibidos pueden ser devueltos al DTE de acuerdo con el valor de <n>, <n> es opcional y por defecto estará si falta (SIMCOM, 2005).

- **Sintaxis de parámetros**

Estos comandos tienen la forma "ATS<n>=<m>", donde <n> es índice del registro S a cargar y <m> es el valor asignarle, <m> es opcional si falta es asignado un valor por defecto (SIMCOM, 2005).

- **Sintaxis Extendida**

Estos comandos pueden operar en diferentes modos como se muestra en la siguiente tabla.

Comando de Prueba	AT+<x>=?	El equipo móvil retorna una lista de parámetros y valores de rangos establecidos en el comando AT escrito
Comando de Lectura	AT+<x>?	Este comando retorna el valor actual cargado o cargados en el parámetro
Comando de Escritura	AT+<x>=<...>	Carga valores definidos por el usuario en el parámetro
Comandos de Ejecución	AT+<x>	Los comandos de ejecución leen los parámetros no variables afectados por procesos internos en el modem

Tabla 2-2: Sintaxis extendida.

COMBINACIÓN DE COMANDOS AT EN UNA MISMA LÍNEA

Se pueden varios comandos en una misma línea, en este caso no es necesario escribir el prefijo “AT” o “at” antes de todos los comandos, en este caso solo es necesario escribir “AT” o “at” en el inicio de cada línea de comando, separar los comandos con punto y coma en una misma línea, usar esto como delimitador (SIMCOM, 2005).

COMANDOS DE EQUIPOS DE RESPUESTA AUTOMÁTICA (ESTÁNDARITU-T V.25TER)

- ATH: Desconecta la conexión existente.
- ATI: Muestra la información de identificación del producto.
- ATZ: Carga los parámetros actuales como perfil definido por el usuario.
- AT&F: Carga los parámetros de fábrica.
- AT&W: Almacena los parámetros definidos en el perfil del usuario.
- AT+ILRR: Carga la tasa de baudios para la transmisión.
- AT+ICF: Configura la forma de elaboración de caracteres de control.

COMANDOS GENERALES (ESTÁNDAR GSM07.07)

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

COMANDOS DEL SERVICIO DE RED (ESTÁNDAR GSM07.07)

- AT+CSQ: Obtener calidad de la señal.
- AT+COPS: Selección de un operador.
- AT+CREG: Registrarse en una red.
- AT+WOPN: Leer nombre del operador3.

COMANDOS DE SEGURIDAD (ESTÁNDAR GSM07.07)

- AT+CPIN: Introducir el PIN
- AT+CPINC: Obtener el número de reintentos que quedan
- AT+CPWD: Cambiar password.

COMANDOS PARA LA AGENDA DE TELÉFONOS (ESTÁNDAR GSM07.07)

- AT+CPBR: Leer todas las entradas
- AT+CPBF: Encontrar una entrada
- AT+CPBW: Almacenar una entrada
- AT+CPBS: Buscar una entradas.

COMANDOS PARA SMS (ESTÁNDAR GSM07.05)

- AT+CPMS: Seleccionar lugar de almacenamiento de los SMS
- AT+CMGF: Seleccionar formato de los mensajes SMS Modo texto o Modo PDU
- AT+CMGR: Leer un mensaje SMS almacenado
- AT+CMGL: Listar los mensajes almacenados
- AT+CMGS: Enviar mensaje SMS
- AT+CMGW: Almacenar mensaje en memoria
- AT+CMSS: Enviar mensaje almacenado
- AT+CSCA: Establecer el Centro de mensajes a usar
- AT+ WMSC: Modificar el estado de un mensaje

2.3.5 GENERALIDADES DEL MODEM SIMCOM SIM300

El modem seleccionado está diseñado para una fácil integración con otros componentes y las redes GSM existentes. Se comunica por una interfaz serial regida por el estándar RS-232 y emplea un juego de comandos AT que le permiten monitorizar la red y avisar de condiciones que puedan ser relevantes para la gestión de una red conformada por varios terminales, presenta la siguientes características (SIMCOM, 2007):

- Tri-Banda GSM/GPRS 900/1800/1900 MHz.
- Sujeto al estándar GSM fase 2/2+.
Clase 4 (2W 900 MHz)
Clase 1 (1W 1800/1900MHz)
- Dimensiones: 40mm x 33mm x 2.85mm.
- Weight: 8 g.
- Control vía comandos AT.
- Rango de voltaje de alimentación 3.4V-4.5V.
- Bajo poder de consumo. Max 2A en ráfaga de transmisión.
- Rango de operación de temperatura:-20°C hasta +55°C.

DESCRIPCIÓN DE PINES

En la siguiente tabla se realiza la descripción de pines del módulo SIMCON SIM 300.

Nombre del PIN	I/O	Descripción	Características en Dc
Fuente de alimentación			
VBAT		5 de los pines del conector DIP están dedicados para la conexión de la fuente de voltaje, se debe colocar una sola fuente de voltaje VBAT=3.4V....4.5V. Debe ser capaz de suministrar suficiente corriente en una ráfaga de transmisión que por lo general se eleva a 2A en algunos casos puede ser de 3A durante 0.1ms.	Vmax= 4.5V Vmin=3.4V Vnorm=4.0V
VRTC	I/O	Corriente de entrada para RTC cuando la batería no se suministra para el sistema. Salida de corriente para batería de respaldo cuando la batería principal está presente y la batería de respaldo de baja tensión del Estado.	Vmax=2.0V Vmin=1.2V Vnorm=1.8V Inorm= 20uA
VDD_EXT	O	Suministro de 2.93V para circuito externo. Por medio de este pin el usuario puede juzgar el estado del sistema, si esta encendido o apagado. Cuando el voltaje de bajo el sistema está apagado de lo contrario esta encendido.	Vmax=3.0V Vmin=2.75V Vnorm=2.93V Imax=60mA
CHG_IN	I	Tensión de entrada para el circuito de carga, haciendo que el sistema detecte el cargador.	Vmax=5.25V Vmin=1.1 * VBAT Vnorm=5.1V
GND		Tierra digital	
Encender o Apagar			
PWRKEY	I	Tensión de entrada para encender el sistema. Se debe llevar a PWRKEY a un bajo nivel de voltaje para que el usuario apague o encienda el sistema. Porque el sistema necesita un margen de tiempo para confirmar el software	VILmax=0.3*VBAT VIHmin=0.7*VBAT VImax=VBAT
Interfaces de Audio			
MIC1P MIC1N	I	Positivo y negativo de la entrada de la banda de voz	
MIC2P MIC2N	I	Auxiliar-Positivo y negativo de la entrada de la banda de voz	
SPK1P SPK1N	O	Positivo y negativo de la salida de la banda de voz	
SPK2P SPK2N	O	Auxiliar-Positivo y negativo de la salida de la banda de voz	
BUZZER	O	Salida para zumbador	
AGND		Tierra Análoga	
Entradas y salidas de propósito general			
KBC0~KBC4	O	La salida de propósito de general (GPO) pueden ser configuradas por comandos AT para colocar la salida en alto o bajo voltaje. Todos los GPOs	VILmin=0V VILmax=0.3 *VDD_EXT VIHmin=0.7*VDD_EXT
KBR0~KBR4	I		

SPI_DATA	I/O	están inicialmente en un estado bajo sin ningún tipo de ajuste por medio de comandos AT	VIHmax= VDD_EXT+0.3 VOLmin=GND VOLmax=0.2V VOHmin= VDD_EXT-0.2 VOHmax= VDD_EXT
SPI_CLK	O		
SPI_CS	O		
SPI_D/C	O		
SPI_RST	O		
Network LED	O		
STATUS	O	Indicar el estado de trabajo	
GPIO5	I/O	Puerto normal de entrada o salida	
GPIO32	I/O	Puerto normal de entrada o salida	
Interface serial 1			
DTR	I	Data Terminal Ready	VILmin=0V VILmax=0.3*VDD_EXT VIHmin=0.7*VDD_EXT VIHmax= VDD_EXT+0.3 VOLmin=GND VOLmax=0.2V VOHmin= VDD_EXT-0.2 VOHmax= VDD_EXT
RXD	I	Receive Data	
TXD	O	Transmit Data	
RTS	I	Request to Send	
CTS	O	Clear to Send	
RI	O	Ring Indicator	
DCD	O	Data Carrierdetection	
Interface serial 2			
DBGTX	O	Interface serial para comunicación y depuración	
DBGRX	I		
SIM Interface			
SIM_VDD	O	Voltaje de alimentación para SIM card	El voltaje puede ser seleccionado para interface de 1.8V o 2.85V
SIM_I/O	I/O	Datos de salida de la SIM	VILmin=0V VILmax=0.3*SIM_VDD VIHmin=0.7*SIM_VDD VIHmax= SIM_VDD+0.3 VOLmin=GND VOLmax=0.2V VOHmin= SIM_VDD-0.2 VOHmax= SIM_VDD
SIM_CLK	O	SIM clock	
SIM_PRESENCE	I	Detección de SIM card	
SIM_RST	O	SIM Reset	
AUXADC			
ADC1	I	Convertor análogo digital de propósito general	Rango de valor de voltaje de entrada. 0V hasta 2.4V

Tabla 2-3: Descripción de pines del módulo SIMCOM SIM300

MODOS DE OPERACIÓN

Los modos de operación son los siguientes (SIMCOM, 2007):

- **MODO DE OPERACIÓN NORMAL**

- Función GSM/GPRS sleep

El modulo automáticamente se colocara en modo de descanso si el DTR es argado en un nivel alto, si no hay actividad en aire o audio y no hay interrupción por parte del hardware por ejemplo interrupción en el GPIO o datos en el puerto serial. En este modo la corriente de consumo se reduce a un bajo nivel. Durante el modo de descanso el modulo puede recibir mensajes y SMS.

- Función GSM IDLE

El software está activo. El modulo está registrado en la red GSM y está listo para enviar y recibir.

- Función GSM TALK

La conexión CSD (Circuit Switched Data) va entre dos subscriptores. En este caso, el consumo de potencia depende de las configuraciones de la red.

- Función GPRS IDLE

El modulo está listo para transferir datos GPRS, pero no hay datos recibidos o enviados en este caso la potencia de consumo depende de las configuraciones de red y de GPRS.

- Función GPRS DATA

Hay datos GPRS en transferencia (PPP o TCP o UDP), en este caso la potencia de consumo está relacionada con las configuraciones de red.

- **MODO APAGADO**

El apagado normal se realiza mediante el comando “AT+CPOWD=1” o usando el PWERKEY. La administración de potencia se realiza por medio de ASIC (Application Specific Integrated Circuit) esta desconecta la alimentación desde banda base del módulo y solo la alimentación para RTC es mantenida.

El software no está activado. Las interfaces seriales no son accesibles. El voltaje de operación conectado a VBAT se mantiene aplicado.

- **MODO DE FUNCIONALIDAD MÍNIMA**

Usando el comando “AT+CFUN” se puede cargar el modo de funcionalidad mínima sin remover la fuente de alimentación, en este caso la parte RF del módulo no trabajara o la SIMCARD no será accesible la parte RF del módulo y la SIMARD se cerrará totalmente. Las interfaces seria siguen siendo accesibles. La potencia de consumo es realmente baja.

- **MODO ALARMA.**

Está restringido el lanzamiento de la función alerta de RTC mientras que el modulo este apagado. El modulo no estará en la red GSM y solo parte de los comandos AT pueden ser validados.

- **MODO GHOST**

Modo de carga durante operación normal. Comienza a cargar mientras el modulo este en operación normal.

ALIMENTACIÓN

El modulo se debe alimentar desde una sola fuente de voltaje $V_{BAT} = 3.4V \dots 4.5V$. En algunos casos la corriente de consumo puede alcanzar picos de 2A y puede haber caída de voltaje en una ráfaga de transmisión. Para la entrada VBAT es recomendable un bypass capacitivo para reducir picos de voltaje, ver figura 2-15.

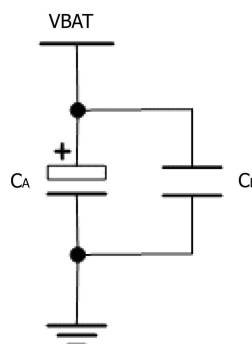


Figura 2-15: Entrada VBAT, Fuente: (SIMCOM, 2007)

Donde CA es un capacitor de tantalio electrolítico de 100uf y CB es un capacitor cerámico de entre 1uf y 10uf.

La tensión nunca debe caer por debajo de 3.4V en una ráfaga de transmisión donde la corriente de consumo puede elevarse a dos amperios, si esto sucede el modulo puede apagarse (SIMCOM, 2007).

Se puede monitorear la fuente de voltaje con el comando “AT+CBC” que incluye tres parámetros: estado de la carga, porcentaje de batería y voltaje en mV, es decir el valor entre VBAT y GND. La corriente será medida continuamente dependiendo del modo de operación.

ESCENARIOS DE ENCENDIDO Y APAGADO

El modulo puede ser encendido de varias maneras:

- Por el pin PWRKEY, una vez iniciado el modo de operación normal.
- Por interrupción en el RTC, una vez iniciado el modo alarma.

Notas:

Solo enviar comandos AT después de que el modulo este encendido y se reciba el código no solicitado RDY por el puerto serial 1

Si configuro una tasa de baudios fija el modulo enviara el código de resultado RDY indicando que está listo para operar, este código de resultado no aparece cuando se cuándo el autobauding está activado.

- **ENCENDIDO DE MODULO CON EL PWRKEY**

Se puede encender el modulo conduciendo al PWRKEY a un bajo nivel de voltaje por un periodo de tiempo como se muestra en la figura 2-16.

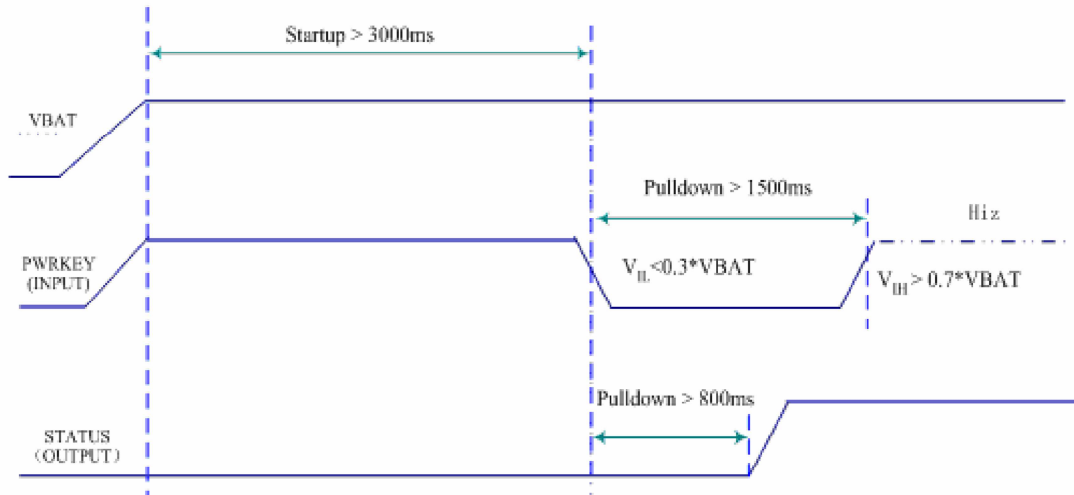


Figura 2-16: Tiempo de encendido del sistema, Fuente: (SIMCOM, 2007)

Cuando el procedimiento este completo el modulo enviara el código de resultado RDY si el modulo no está en autobauding. Y el pin STATUS se colocara en nivel alto con 2.83 V (SIMCOM, 2007).

- **APAGADO DEL MODULO**

Se pueden utilizar los siguientes procedimientos para apagar el módulo SIM340C:

- Procedimiento normal de apagado utilizando el pin PWRKEY.
- Procedimiento normal de apagado utilizando los comandos AT.
- Apagado automático por caída de voltaje detectada.
- Apagado automático por detección de exceso de temperatura.

- **PROCEDIMIENTO NORMAL DE APAGADO UTILIZANDO EL PIN PWRKEY**

Se puede apagar el modulo llevando el PWRKEY a un bajo voltaje por un periodo de tiempo. Este procedimiento llevará a al módulo a desconectarse de la red, a que el software entre en un estado seguro y grabar datos antes de apagarse.

Antes de apagarse completamente el modulo enviara el código de resultado de salida:

- **NORMAL POWER DOWN**

Después de este momento ningún comando AT puede ser ejecutado y solo el RTC está activado. El apagado solo puede ser indicado por el pin VDD-EXT que toma un bajo nivel de voltaje en este modo. Ver figura 2-17.

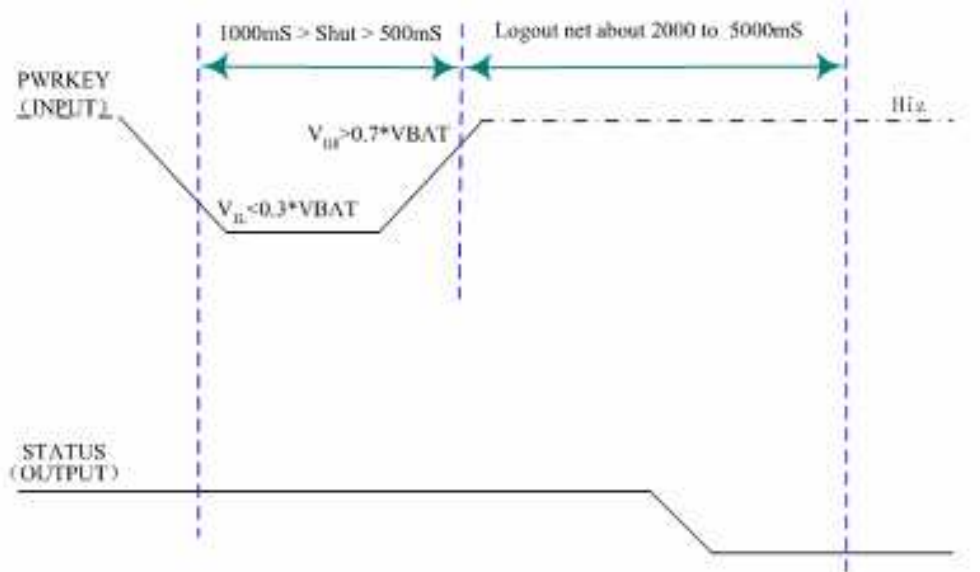


Figura 2-17: Tiempo de apagado del sistema, Fuente: (SIMCOM, 2007)

- PROCEDIMIENTO NORMAL DE APAGADO UTILIZANDO COMANDOS AT

Con el comando AT “AT+CPOWD=1” se puede apagar el modulo. Este procedimiento llevara a él modulo a desconectarse de la red, a que el software entre en un estado seguro y grabar datos antes de apagarse.

Antes de apagarse completamente el modulo enviara el código de resultado de salida:

- **NORMAL POWER DOWN**

Después de este momento ningún comando AT puede ser ejecutado y solo el RTC está activado. El apagado solo puede ser indicado por el pin VDD-EXT que toma un bajo nivel de voltaje en este modo.

- APAGADO AUTOMATICO POR CAIDA DE VOLTAJE DETECTADA

El software constantemente monitoreará el valor del voltaje en VBAT, si la medida de voltaje no es más que 3.5 V enviara el código URC (código no solicitado) “**POWER LOW WARNING**”, si la medida de voltaje no es más que 3.4 V se enviara el URC “**POWER LOW DOWN**”. Después de este momento ningún comando AT puede ser ejecutado y solo el RTC está activado. El apagado solo puede ser indicado por el pin VDD-EXT que toma un bajo nivel de voltaje en este modo.

REINICIAR EL MODULO CON PWRKEY

Se puede reiniciar el modulo llevando el PWRKEY a un nivel bajo por un periodo de tiempo luego poner en un nivel alto, una vez se detecta que el pin STATUS cambio a un nivel bajo, se inicia la temporización de un delay de 500ms, luego transcurrido este tiempo se lleva a un nivel bajo PWRKEY y el pin STATUS nuevamente tomara un nivel alto. Ver figura 2-18.

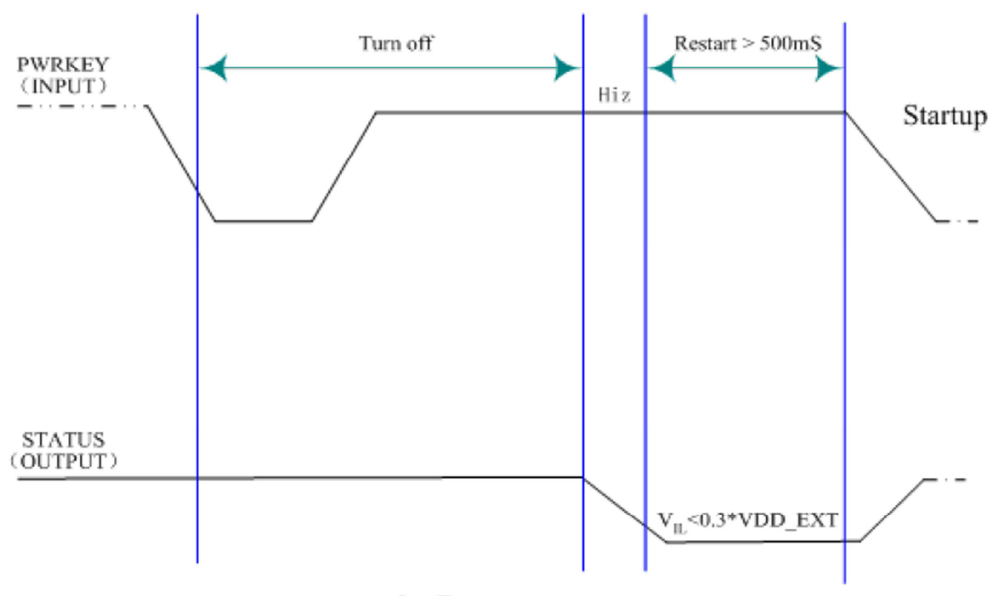


Figura 2-18: Tiempo de reinicio del sistema, Fuente: (SIMCOM, 2007)

RTC BACKUP

La alimentación RTC (Real time Clock) puede ser suministrada por una batería externa a través del PIN 14 del conector DIP. Hay una resistencia de 10K que ha sido integrada limitando el paso de la corriente.

El pin VRTC no pudo ser diseñado para ser un pin NC (not conected) (no conectado) en el circuito. Se debe conectar el pin a una batería o a un capacitor de alta capacitancia. Ver figura 2-19.

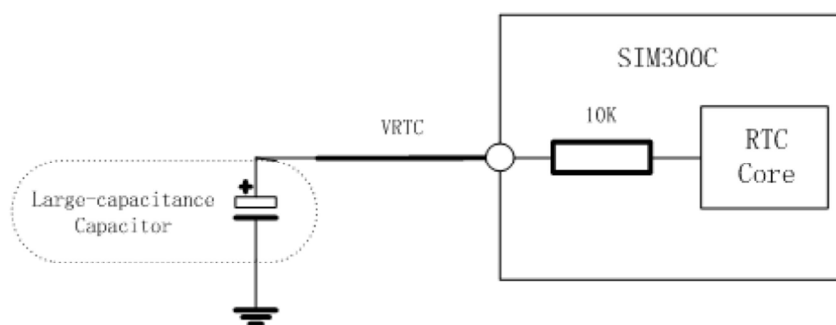


Figura 2-19: RTC alimentado desde capacitor, Fuente: (SIMCOM, 2007)

INTERFAZ SERIAL

El modulo proporciona un puerto serial asíncrono, está diseñado como un DCE (Data Communication Equipment) siguiendo la tradicional conexión DCE-DTE (Data Terminal Equipment). El autobauding va desde 1200 bps hasta 115200 bps, ver tabla 2-4.

Parameter	Min	Max	Unit
Logic low input	0	$0.3 \cdot VDD_EXT$	V
Logic high input	$0.7 \cdot VDD_EXT$	$VDD_EXT + 0.3$	V
Logic low output	GND	0.2	V
Logic high output	$VDD_EXT - 0.2$	VDD_EXT	V

Tabla 2-4: Niveles lógicos de voltaje, Fuente: (SIMCOM, 2007)

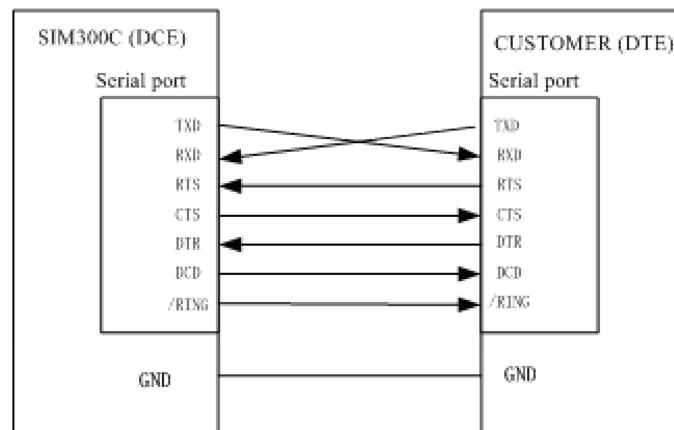


Figura 2-20: Interfaz del puerto serial, Fuente: (SIMCOM, 2007)

Características de puerto serial

- 7 líneas para interfaz del puerto serial
- Contiene líneas de datos RX y TX, líneas de estado RTS and CTS, línea de control DTR, DCD and RING;
- El puerto serial soporta las siguiente velocidades en baudios: 300,1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 y por defecto 115200 bps
- El autobauding soporta las siguientes velocidades de comunicación: 1200, 2400, 4800, 9600, 19200, 38400, 57600 y 115200bps.

En la figura 2-20 se muestra la conexión de la interfaz para una conexión entre DCE-DTE

El autobauding permite detectar automáticamente la velocidad de bits configurada en el host, para aprovechar las ventajas del modo autobauding se deben tener en cuenta los siguientes requerimientos:

- **Sincronización entre DTE y DCE**

Para permitir la sincronización de la velocidad de baudios simplemente se emite una cadena "AT" o "at".

Esto es necesario:

- Después de tener activado el autobauding.

- Cuando se ponga en marcha el modulo mientras el autobauding este habilitado, se recomienda esperar de 3 a 5 segundos antes de mandar el primer comando at.

- **Restricciones de operación en autobauding**

- La interfaz serial tiene que ser operada con 8bits, no paridad y un bit de parada (configuración de fábrica).
- Los código de resultado no solicitados como “RDY”, “+CFUN: 1” y “+CPIN: READY” no son indicados cuando.

INTERFAZ DE LA SIMCARD

Por medio de los comando AT se puede obtener información dentro la SIMCARD.

La interfaz de la SIM soporta la funcionalidad para la fase 1 de GSM y también soporta las especificaciones de funcionalidad de la fase GSM 2+ para una SIM rápida de 64Kbps, también soporta SIMCARDS de 1.8V y 3V.

La interfaz de la SIMCARD es alimentada desde un regulador interno que tiene un voltaje nominal 2.8V. Todos los pines reinician como salidas en bajo voltaje. Los niveles lógicos son descritos en la tabla 2-5.

Pin	Signal	Description
25	SIM_VDD	SIM Card Power output automatic output on SIM mode, one is $2.85 \pm 0.1V$, another is $1.8 \pm 0.1V$. Current is about 10mA.
29	SIM_I/O	SIM Card data I/O
31	SIM_CLK	SIM Card Clock
27	SIM_RST	SIM Card Reset
33	SIM_PRESENCE	SIM Card Presence

Tabla 2-5: Señales de la interfaz de SIM, Fuente: (SIMCOM, 2007)

Como dispositivo de protección ante descargas electrostáticas se recomienda usar el SMF05.

Si no se utiliza la función de detección de SIMCARD, se puede dejar el pin SIM_PRESENCE no conectado o conectado a GND. La figura 2-21 muestra la configuración para soportes de SIMCARD de 6 pines y se soporta en las figuras 2-22 y 2-23.

2.3.6 ESTANDAR EIA232 (RS232)

Esta norma se originó con el fin de especificar la interconexión de un equipo terminal de datos (DTE), por ejemplo un ordenador, una impresora, o un terminal de entrada, y un equipo de terminación de un circuito de datos (DCE), por ejemplo un modem o un ordenador. Este estándar se basa en comunicación asíncrona es decir que los datos pueden ser transmitidos en cualquier momento por lo que deben tomarse precauciones para sincronizar la transmisión y recepción (Manuel, 2001).

La comunicación serial es un protocolo muy común para comunicación entre dispositivos que se incluye de manera estándar en prácticamente cualquier computadora. La mayoría de las computadoras incluyen dos puertos seriales RS-232. La comunicación serial es también un protocolo común utilizado por varios dispositivos para instrumentación; existen varios dispositivos compatibles con GPIB que incluyen un puerto RS-232. Además, la comunicación serial puede ser utilizada para adquisición de datos si se usa en conjunto con un dispositivo remoto de muestreo (National Instruments, 2006).

El puerto serial envía y recibe bytes de información un bit a la vez. Aun y cuando esto es más lento que la comunicación en paralelo, que permite la transmisión de un byte completo por vez, este método de comunicación es más sencillo y puede alcanzar mayores distancias. Por ejemplo, la especificación *IEEE 488* para la comunicación en paralelo determina que el largo del cable para el equipo no puede ser mayor a 20 metros, con no más de 2 metros entre cualesquier dos dispositivos; por el otro lado, utilizando comunicación serial el largo del cable puede llegar a los 1200 metros (National Instruments, 2006).

El RS-232 define especificaciones mecánicas, eléctricas, funcionales y de procedimientos típicos de un protocolo orientado al enlace físico punto a punto.

ESPECIFICACIONES MECÁNICAS

El conector normalmente empleado en los interfaces RS-232 es un conector DB-25, aunque es normal encontrar la versión de 9 pines DB-9 de forma más difundida. El estándar define que el conector hembra se situará en los DCE y el macho en el DTE. Aunque es fácil encontrar excepciones. También es frecuente que muchas interfaces sólo incorporen parte de los circuitos descritos en la especificación. Ver figura 2-24 y 2-25.

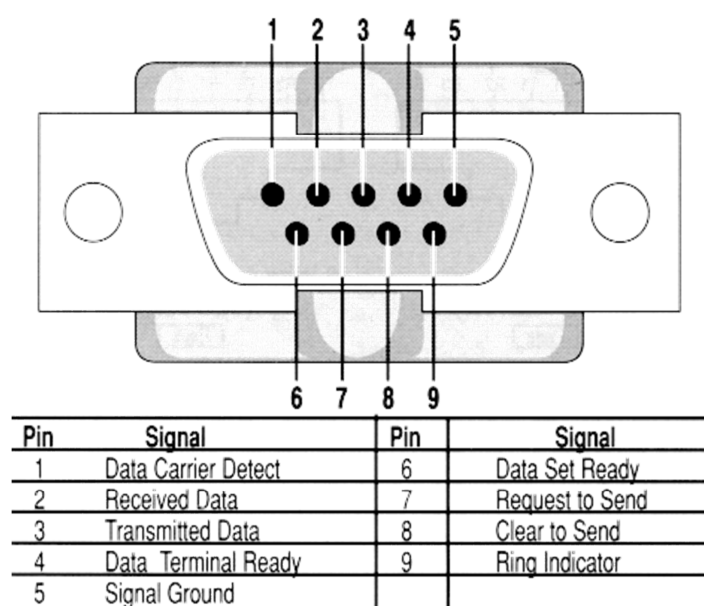
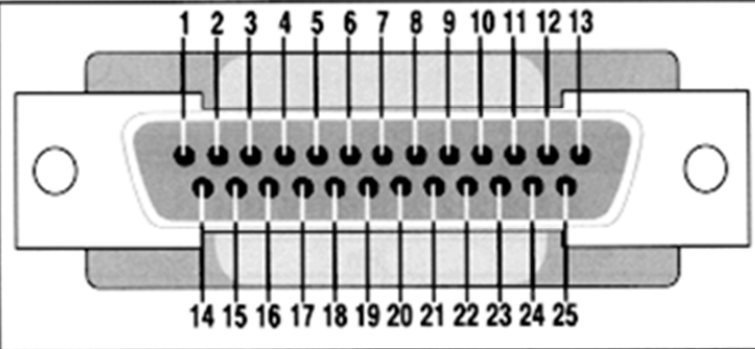


Figura 2-24: Descripción de pines conector DB9, Fuente: (Arduino, 2011)



Pin	Description	EIA CKT	From DCE	To DCE
1	Frame Ground	AA		
2	Transmitted Data	BA		D (Data)
3	Received Data	BB	D	
4	Request to Send	CA		C (Control)
5	Clear to Send	CB	C	
6	Data Set Ready	CC	C	
7	Signal Gnd/Common Return	AB		
8	Rcvd. Line Signal Detector	CF	C	
11	Undefined			
12	Secondary Rcvd. Line Sig. Detector	SCF	C	
13	Secondary Clear to Send	SCB	C	
14	Secondary Transmitted Data	SBA		D
15	Transmitter Sig. Element Timing	DB	T (Timing)	
16	Secondary Received Data	SBB	D	
17	Receiver Sig. Element Timing	DD	T	
18	Undefined			
19	Secondary Request to Send	SCA		C
20	Data Terminal Ready	CD		C
21	Sig. Quality Detector	CG		C
22	Ring Indicator	CE	C	
23	Data Sig. Rate Selector (DCE)	CI	C	
23	Data Sig. Rate Selector (DTE)	CH		C
24	Transmitter Sig. Element Timing	DA		T
25	Undefined			

Figura 2-25: Descripción de pines conector DB25, Fuente: (Arduino, 2011)

Para conseguir establecer la comunicación serial entre dos equipos se puede emplear los siguientes tipos de alambrado:

- **MODEM ACTIVO CON HANDSHAKING, FULL DUPLEX**

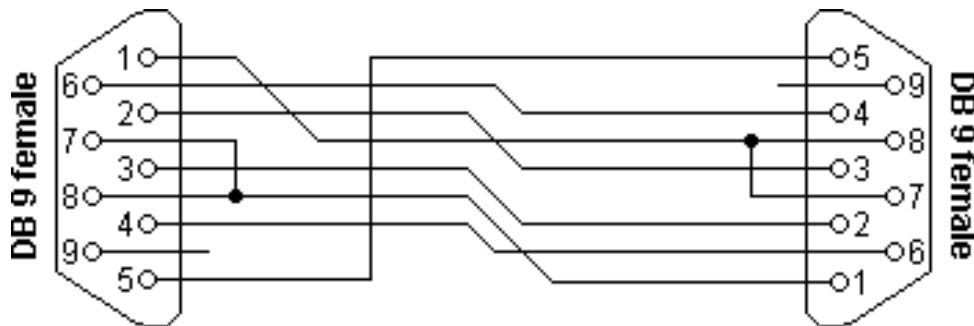


Figura 2-26: Modem activo con Hand Shaking, Fuente: (Lammertbies, 2011)

- **MODEM NULO, FULL DUPLEX**

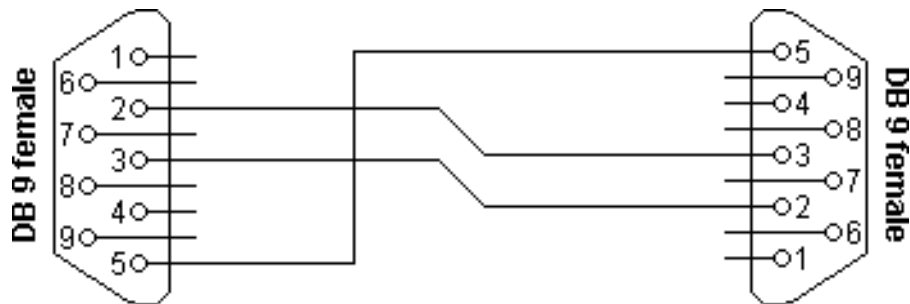


Figura 2-27: Modem nulo, Full Duplex (DB9), Fuente: (Lammertbies, 2011))

El método de comunicación usado por RS-232 requiere de una conexión muy simple, utilizando sólo tres líneas: TX, RX, y GND. Sin embargo, para que los datos puedan ser transmitidos correctamente ambos extremos deben estar sincronizados a la misma velocidad. Aun cuando este método es más que suficiente para la mayoría de las aplicaciones, es limitado en su respuesta a posibles problemas que puedan surgir durante la comunicación; por ejemplo, si el receptor se comienza a sobrecargar de información. Es en estos casos cuando el intercambio de pulsos de sincronización, o handshaking, es útil.

Debido a la practicidad de este tipo conexión y la eficiencia con la que logran llegar los datos se escoge este tipo de conexión para el diseño de la interfaz serial.

ESPECIFICACIONES ELÉCTRICAS

La interfaz eléctrica utiliza una conexión eléctrica asimétrica con circuitos no equilibrados, todos referenciados a tierra. Ver figura 2-28.

Los estados lógicos son definidos por los siguientes niveles de voltaje:

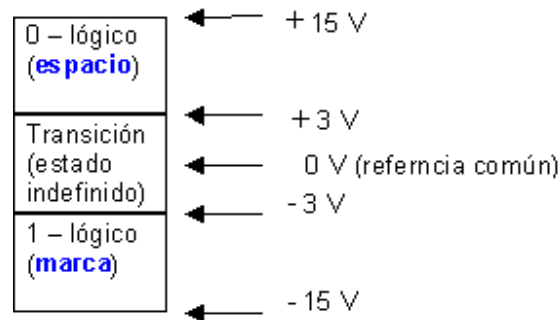


Figura 2-28: Especificaciones eléctricas estándar RS232C, Fuente: (EVA, 2011)

La convención RS-232-C especifica que, con respecto a masa, cualquier voltaje más negativo que -3 V es interpretado como un "1", y cualquier voltaje más positivo que +3V es interpretado como un "0". La comunicación requiere que el transmisor y el receptor acuerden un protocolo a utilizar. El protocolo RS-232-C consiste en (National Instruments, 2006):

- Frecuencia de reloj (Baudrate): La velocidad de transmisión se mide en bits por segundo. Es necesario que el transmisor y el receptor tengan sus relojes sincronizados a la misma velocidad.
- Marca (Markingstate): Período de tiempo durante el cual no se transmite información. El transmisor mantiene la línea en un estado alto (1 lógico) denominado "marca". Se emplea para sincronizar.
- Bit de inicio (Start bit): Es un bit "0" que indica que la transmisión va a empezar. Este estado bajo se denomina "spacingstate" o "espacio".
- Bits de datos (Data bits): La cadena de bits correspondiente al dato puede estar compuesta por 5, 6, 7 u 8 bits y representan al carácter transmitido. El bit menos significativo (LSB – Least Significant Bit) es el primero en ser transmitido y el bit más significativo (MSB – Most Significant Bit) el último.

- Bit de paridad (Parity bit): Es opcional. Se transmite tras los bits de datos y se usa para detección de errores.
- Bits de stop (Stop bit): Uno o más bits a nivel alto ("1" lógico). Garantizan que el receptor tendrá tiempo de procesar la información antes de que le llegue el próximo dato. Ver figura 2-29.

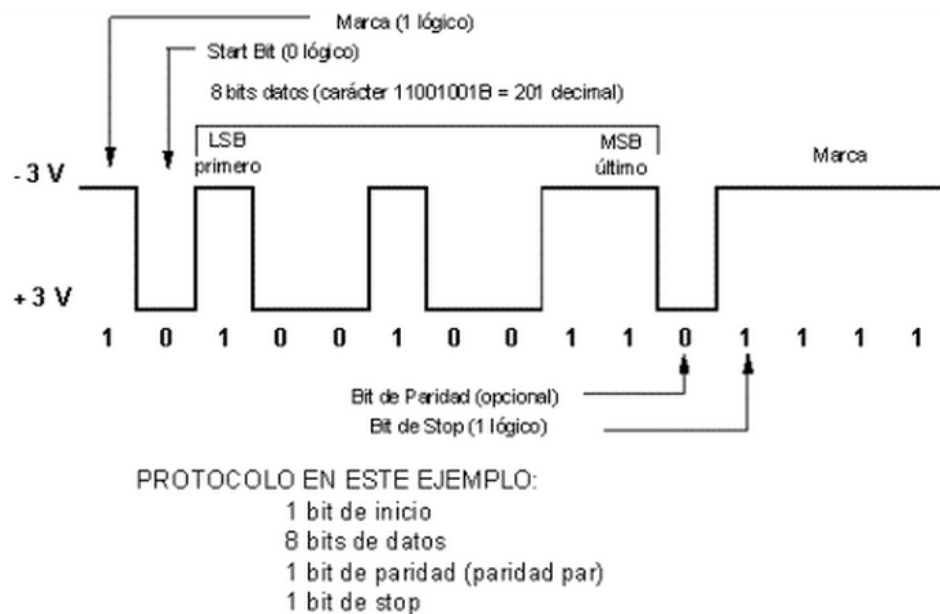


Figura 2-29: Descripción del protocolo RS323C, Fuente: (EVA, 2011)

ESPECIFICACIONES FUNCIONALES

- SG (SignalGround) o GND: Esta línea debe estar conectada al chasis del PC y desde ahí hacer tierra.
- DTR (Data Terminal Ready): Esta señal prepara al modem para conectarse a una línea de comunicación y mantener la conexión establecida. Una vez que modem está conectado a la línea, este pin debe estar encendido para mantenerse la conexión, si DTR es apagado, provoca una desconexión de la línea, interrumpiendo el enlace de datos en progreso. DTR también puede mantenerse encendido para efectuar la transmisión de datos por el pin TXD. Básicamente DTR es el pin de control maestro del modem.
- DSR (Data Set Ready): Es la línea que indica que el modem está preparado. Usualmente el DSR está todo el tiempo encendido porque este valor es un indicador que el MODEM esta encendido y listo.

- **RTS (Request To Send):** Es la línea que dice al modem que el PC quiere enviar datos. El estándar RS-232 dice que RTS condiciona al modem para la transmisión, en realidad esta es solo una función que sirve de interruptor a un modem half duplex para transmitir o recibir. Mientras un modem half dúplex está recibiendo, el DTE guarda el RTS apagado (cero), Cuando le toca al DTE el turno de transmitir, este informa al modem que desea transmitir poniendo el RTS a (uno). El DTE no puede comenzar inmediatamente con el envío de datos al modem porque éste, no puede cambiar inmediatamente el modo de transmisión a recepción. Después de poner a uno RTS el DTE monitorea el pin CTS el cual es apagado por el modem que está en modo de recepción, cuando el modem está listo para transmitir, este enciende el CTS en función al DTE que está listo para el envío de datos. Este RTS-CTS (handshaking), también es permitido en sentido contrario, es decir cuando el transmisor vuelve a recibir. Como en una comunicación full dúplex hay dos canales, no se necesita el RTS-CTS, de esta forma un modem full duplex pone permanentemente el pin CTS conectado al pin DCD (Data Carrier Detect).
- **CTS (Clear To Send):** Es la línea que indica que el modem está preparado para recibir datos desde el PC.
- **DCD (Data Carrier Detect):** Es la línea que indica que el modem tiene de verdad conexión remota. Este pin es también llamado “Receiver Time Signal Detect”, es encendido cuando el modem recibe una señal remota y se mantiene encendido durante el enlace. En conexiones de modem half duplex el DCD es encendido solamente por el MODEM que está en recepción.
- **TXD (Transmit Data):** Es la línea de transmisión de datos serie al modem. El TXD no puede transmitir datos a menos que los siguientes circuitos hayan sido encendidos: DTR, DSR, RTS (listo para enviar), CTS (listo para recibir).
- **RXD (Receive Data):** Es la línea de recepción de datos serie desde el MODEM.
- **RI (Ring Indicator):** Es la línea que indica que el MODEM ha detectado la señal de “llamada” (se pone en uno).
- **RTXC (Transmit/Receive Clock):** Reloj común para transmisiones sincrónicas (solo existe en algunos PC's).

COMPATIBILIDAD CON TTL

El MAX232 es un circuito integrado de Maxim que convierte las señales de un puerto serie RS-232 a señales compatibles con los niveles TTL de circuitos lógicos. El MAX232 sirve como interfaz de transmisión y recepción para las señales RX, TX, CTS y RTS (Angulo Usategui, Angulo Martínez, & Etxebarria Ruiz, 2007).

El circuito integrado tiene salidas para manejar niveles de voltaje del RS-232 (aprox. ± 7.5 V) que las produce a partir de un voltaje de alimentación de + 5 V utilizando multiplicadores de voltaje internamente en el MAX232 con la adición de condensadores externos. Esto es de mucha utilidad para la implementación de puertos serie RS-232 en dispositivos que tengan una alimentación simple de + 5 V. Ver figura 2-30.

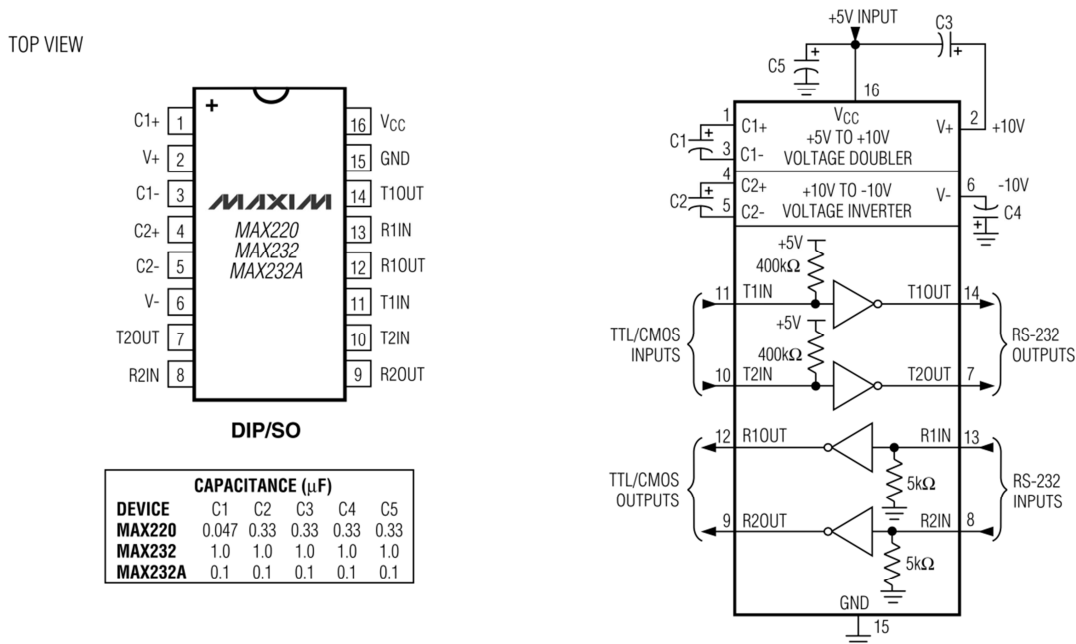


Figura 2-30: Diagrama de conexión MAX232, Fuente: (Maximic, 2012)

DESCRIPCION DE PINES DEL IC MAX232

C1+: Conexión positiva del condensador C1 del doblador de voltaje de +5V a +10V.

C1-: Conexión negativa del condensador C1 del doblador de voltaje de +5V a +10V.

C2+: Conexión positiva del condensador C2 del inversor de voltaje de +10V a -10V.

C2-: Conexión negativa del condensador C2 del inversor de voltaje de +10V a -10V.

V-: Conexión de salida del voltaje de -10V.

V+: Conexión de salida del voltaje de +10V.

T1in, T2in, R1out, R2out: Conexiones a niveles de voltaje de TTL o CMOS.

T1out, T2out, R1in, R2in: Conexiones a niveles de voltaje del protocolo RS-232.

VCC: Alimentación positiva del MAX232.

GND: Alimentación negativa del MAX232.

Con la implementación del MAX232 se logra la conversión de niveles de voltaje RS232 a niveles TTL como se observa en la Figura 2-31.

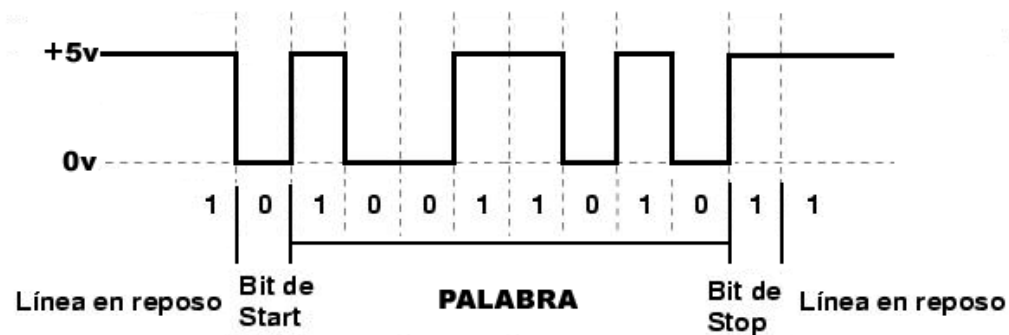


Figura 2-31: Adaptación de niveles RS232 a TTL, (EVA, 2011)

2.3.7 MICROCONTROLADORES

Un microcontrolador es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria.

Es un circuito integrado programable que contiene todos los componentes de un computador, se emplea para controlar el funcionamiento de una tarea determinada y, debido a su reducido tamaño, suele ir incorporado en el propio dispositivo al que gobierna (Angulo Usategui, Angulo Martínez, & Etxebarria Ruiz, 2007).

El microcontrolador es un computador dedicado. En su memoria sólo reside un programa destinado a gobernar una aplicación determinada; sus líneas de entrada/salida soportan el conexionado de los sensores y actuadores del dispositivo a controlar, y todos los recursos complementarios disponibles tienen como única finalidad atender sus requerimientos. Una vez programado y configurado el microcontrolador solamente sirve para gobernar la tarea asignada.

ARQUITECTURA Y PARTES DE UN MICROCONTROLADOR.

Básicamente existen dos arquitecturas de computadoras, y por supuesto, están presentes en el mundo de los microcontroladores: Von Neumann y Harvard. Ambas se diferencian en la forma de conexión de la memoria al procesador y en los buses que cada una necesita, y siguen estando presentes porque cada una ofrece ventajas frente a la otra (Angulo Usategui, Angulo Martínez, & Etxebarria Ruiz, 2007).

- **ARQUITECTURA DE JHON VON NEUMANN**

Su principal característica radica en que en el dispositivo computarizado solo existe un canal para acceder a la memoria FLASH y a la RAM desde la CPU, es decir, el bus de datos, de dirección y de control es compartido entre estas dos memorias lo que limita el acceso a ellas, pues a pesar de ser memorias de diferente tipo solo se puede leer o escribir una a una. Ver figura 2-36.

Como ventajas podemos nombrar el ahorro de pines o señales que deben salir de la CPU a la memoria RAM y FLASH, haciendo más sencilla la fabricación del microcontrolador, además si pensamos en la board de un computador, son menos señales las que salen desde el procesador hasta la memoria RAM y el disco duro. Esta es una de las varias razones por las que es tan usada esta arquitectura en la fabricación de procesadores para computadoras.

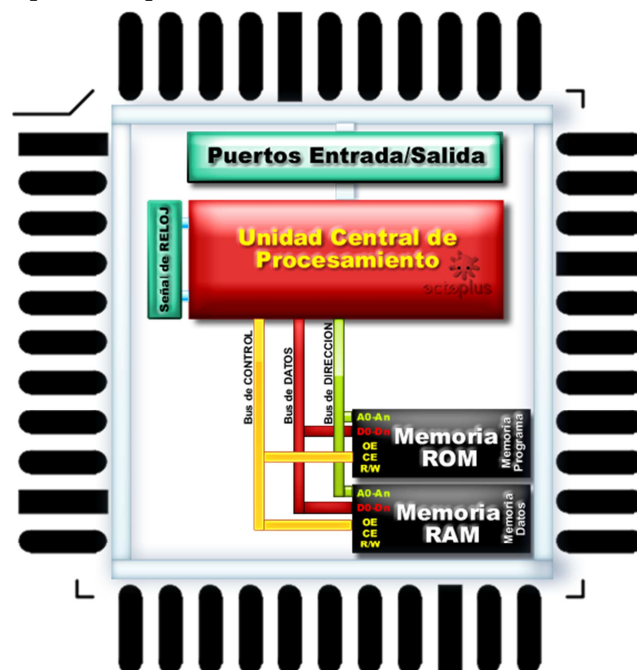


Figura 2-32: Diagrama de bloques arquitectura Jhon Von Neumann, Fuente: (Octoplus, 2012)

- **ARQUITECTURA HARVARD**

A diferencia del modelo anterior cada memoria tiene su canal de comunicación con la unidad central de procesamiento, separando de esta manera los buses de comunicación entre la memoria de programa y de datos, se puede lograr acceder en el mismo instante de tiempo a la memoria RAM y ROM. Ver figura 2-37.

Con lo anterior es fácil deducir que la velocidad de operación de los microcontroladores es muy superior comparada con los que usan Von Neumann, además el proceso de búsqueda y ejecución de cada instrucción se puede segmentar logrando de cierta manera doble operación por cada ciclo de reloj (Angulo Usategui, Angulo Martínez, & Etxebarria Ruiz, 2007).

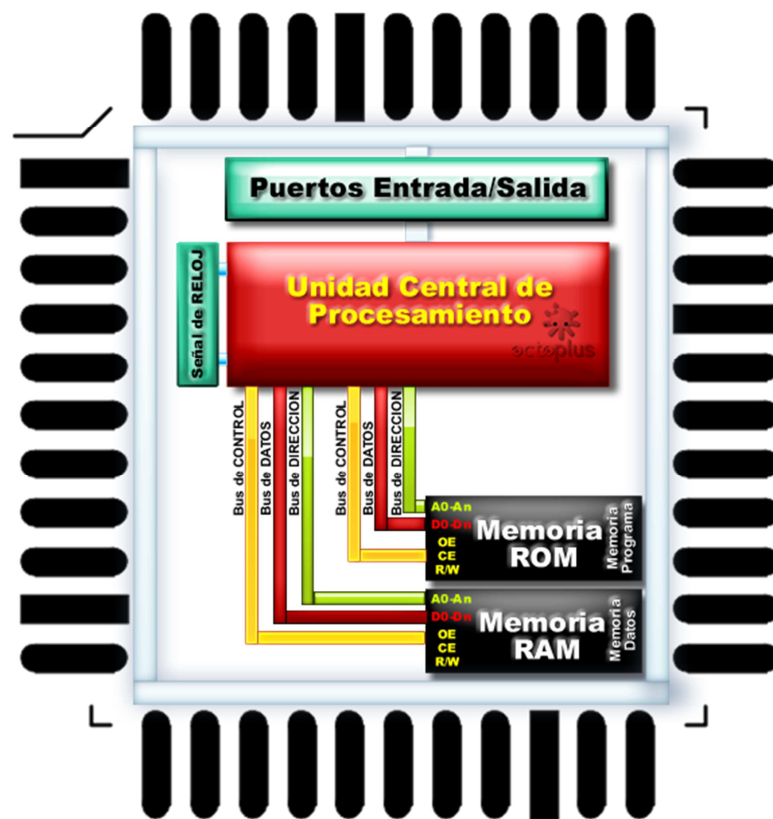


Figura 2-33: Diagrama de bloques arquitectura Harvard, Fuente: (Octoplus, 2012)

PARTES DE UN MICROCONTROLADOR

Sin importar el tipo de arquitectura que tenga un microprocesador las partes principales de un microcontrolador son:

- **PROCESADOR**

Es el elemento más importante del microcontrolador y determina sus principales características, tanto a nivel hardware como software. Se encarga de direccionar la memoria de instrucciones, recibir el código OP de la instrucción en curso, su decodificación y la ejecución de la operación que implica la instrucción, así como la búsqueda de los operandos y el almacenamiento del resultado. Existen tres orientaciones en cuanto a la arquitectura y funcionalidad de los procesadores actuales (Angulo Usategui, Angulo Martínez, & Etxebarria Ruiz, 2007).

- **ARQUITECTURA CISC**

Un gran número de procesadores usados en los microcontroladores están basados en la filosofía CISC (Computadores de Juego de Instrucciones Complejo). Disponen de más de 80 instrucciones máquina en su repertorio, algunas de las cuales son muy sofisticadas y potentes, requiriendo muchos ciclos para su ejecución. Una ventaja de los procesadores CISC es que ofrecen al programador instrucciones complejas que actúan como macros, es decir, que si las tuviésemos que implementar con instrucciones básicas, acabaríamos con dolor de cabeza.

- **ARQUITECTURA RISC**

Tanto la industria de los computadores comerciales como la de los microcontroladores están decantándose hacia la filosofía RISC (Computadores de Juego de Instrucciones Reducido). En estos procesadores el repertorio de instrucciones máquina es muy reducido y las instrucciones son simples y, generalmente, se ejecutan en un ciclo. La sencillez y rapidez de las instrucciones permiten optimizar el hardware y el software del procesador.

- **ARQUITECTURA SISC**

En los microcontroladores destinados a aplicaciones muy concretas, el juego de instrucciones, además de ser reducido, es específico, o sea, las instrucciones se adaptan a las necesidades de la aplicación prevista. Esta filosofía se ha bautizado con el nombre de SISC (Computadores de Juego de Instrucciones Específico).

- **MEMORIA NO VOLÁTIL PARA CONTENER EL PROGRAMA**

El microcontrolador está diseñado para que en su memoria de programa se almacenen todas las instrucciones del programa de control. No hay posibilidad de utilizar memorias externas de ampliación.

Como el programa a ejecutar siempre es el mismo, debe estar grabado de forma permanente. Los tipos de memoria adecuados para soportar esta función admiten cinco versiones diferentes (Angulo Usategui, Angulo Martínez, & Etxebarria Ruiz, 2007):

- **ROM CON MÁSCARA**

En este tipo de memoria el programa se graba en el chip durante el proceso de su fabricación mediante el uso de «máscaras». Los altos costes de diseño e instrumental sólo aconsejan usar este tipo de memoria cuando se precisan series muy grandes.

- **EPROM**

La grabación de esta memoria se realiza mediante un dispositivo físico gobernado desde un computador personal, que recibe el nombre de grabador. En la superficie de la cápsula del microcontrolador existe una ventana de cristal por la que se puede someter al chip de la memoria a rayos ultravioletas para producir su borrado y emplearla nuevamente. Es interesante la memoria EPROM en la fase de diseño y depuración de los programas, pero su coste unitario es elevado.

- **OTP (Programable una vez)**

Este modelo de memoria sólo se puede grabar una vez por parte del usuario, utilizando el mismo procedimiento que con la memoria EPROM. Posteriormente no se puede borrar. Su bajo precio y la sencillez de la grabación aconsejan este tipo de memoria para prototipos finales y series de producción cortas.

- **EEPROM**

El contenido de la EEPROM se puede cambiar durante el funcionamiento (similar a la RAM), pero se queda permanentemente guardado después de la pérdida de la fuente de alimentación (similar a la ROM). Por lo tanto, la EEPROM se utiliza con frecuencia para almacenar los valores creados durante el funcionamiento, que tienen que estar permanentemente guardados.

- **FLASH**

Se trata de una memoria no volátil, de bajo consumo, que se puede escribir y borrar en circuito al igual que las EEPROM, pero suelen disponer de mayor capacidad que estas últimas. El borrado sólo es posible con bloques completos y no se puede realizar sobre posiciones concretas. En las FLASH se garantizan 1.000 ciclos de escritura-borrado.

Por sus mejores prestaciones está sustituyendo a la memoria EEPROM para contener instrucciones. De esta forma Microchip comercializa dos microcontroladores prácticamente iguales, que sólo se diferencian en que la memoria de programa de uno de ellos es tipo EEPROM y la del otro tipo Flash.

• **MEMORIA DE LECTURA Y ESCRITURA PARA GUARDAR LOS DATOS**

Los datos que manejan los programas varían continuamente, y esto exige que la memoria que los contiene deba ser de lectura y escritura, por lo que la memoria RAM estática (SRAM) es la más adecuada, aunque sea volátil (Angulo Usategui, Angulo Martínez, & Etxebarria Ruiz, 2007).

• **LÍNEAS DE EIS PARA LOS CONTROLADORES DE PERIFÉRICOS**

A excepción de dos patitas destinadas a recibir la alimentación, otras dos para el cristal de cuarzo, que regula la frecuencia de trabajo, y una más para provocar el Reset, las restantes patitas de un microcontrolador sirven para soportar su comunicación con los periféricos ex-ternos que controla.

Las líneas de E/S que se adaptan con los periféricos manejan información en paralelo y se agrupan en conjuntos de ocho, que reciben el nombre de Puertas. Hay modelos con líneas que soportan la comunicación en serie; otros disponen de conjuntos de líneas que implementan puertas de comunicación para diversos protocolos, como el I2C, el USB, etc.

• **RECURSOS AUXILIARES**

- **Circuito de reloj:** Encargado de generar los impulsos que sincronizan el funcionamiento de todo el sistema.
- **Temporizadores:** Se encargan de controlar los tiempos.
- **Perro Guardián (watchdog):** Destinado a provocar una re inicialización cuando el programa queda bloqueado o se encuentra en un bucle infinito.

- **Conversores AD y DA:** Para poder recibir y enviar señales analógicas.
- **Comparadores analógicos:** Para verificar el valor de una señal analógica.
- **Protección ante fallos de la alimentación:** Para evitar daños.
- **Estado de reposo o de bajo consumo:** En el que el sistema queda congelado y el consumo de energía se reduce al mínimo a medida que la evolucionan las técnicas de integración y los sistemas digitales cada vez más se podrán adjuntar nuevos periféricos a los microcontroladores.

2.3.8 LABVIEW

LabVIEW es un extenso entorno de desarrollo que brinda a científicos e ingenieros integración con hardware sin precedentes y amplia compatibilidad. LabVIEW ayuda a resolver problemas, acelera su productividad y le da la seguridad para innovar continuamente para crear y desplegar sistemas de medidas y control. LabVIEW es un revolucionario ambiente de desarrollo gráfico con funciones integradas para realizar adquisición de datos, control de instrumentos, análisis de mediciones y presentaciones de datos. LabVIEW le da la flexibilidad de un poderoso ambiente de programación sin la complejidad de los ambientes tradicionales (National Instruments, 2012).



Figura 2-34: Logo LabVIEW, Fuente: (National Instruments, 2012)

ÁREAS DE APLICACIÓN

Las áreas de aplicación van desde realizar medidas simples de temperatura hasta controlar el acelerador de partículas más grande del mundo, los ingenieros y científicos utilizan la plataforma de diseño gráfico de sistemas NI LabVIEW para resolver una amplia variedad de retos en aplicaciones (National Instruments, 2012).

Se puede dividir en 5 áreas funcionales:

- **Adquirir datos y procesar señales:** LabVIEW es un software de diseño de sistemas que acelera la productividad al automatizar varias medidas desde una variedad de sensores. Con una estrecha integración de hardware, se puede conectar a más de 200 dispositivos de adquisición de datos y de terceros, y con las incomparables características para analizar, visualizar y compartir datos, se puede ahorrar tiempo al transformar sus resultados en decisiones (National Instruments, 2012).
- **Automatizar Sistemas de Pruebas y Validación:** LabVIEW es un entorno intuitivo de programación gráfica que ayuda a desarrollar rápidamente un potente software de pruebas. Con soporte para miles de instrumentos y tecnologías como multinúcleo y arreglos de compuertas programable en campo (FPGAs), se puede desarrollar sistemas de pruebas automatizadas de alto rendimiento (National Instruments, 2012).
- **NI LabVIEW para Enseñanza Académica:** NI LabVIEW es un entorno de programación gráfica usado en universidades en todo el mundo para ofrecer aprendizaje práctico, para mejorar las aplicaciones de investigación y para impulsar el desarrollo de la próxima generación de innovadores. Con la naturaleza intuitiva del diseño de sistemas gráficos, los educadores e investigadores pueden diseñar, generar prototipos y desplegar sus aplicaciones (National Instruments, 2012).
- **Control de instrumentos:** NI LabVIEW es un entorno de programación gráfica que permite fácilmente controlar y adquirir datos desde cualquier instrumento en cualquier bus. Se puede automatizar medidas desde varios dispositivos, analizar datos conforme son adquiridos y crear reportes personalizados, todo en cuestión de minutos (National Instruments).
- **Diseño de sistemas Embebidos:** LabVIEW es un software de diseño de sistemas que es usado por ingenieros y científicos para diseñar, generar prototipos y desplegar de manera eficiente aplicaciones embebidas de monitoreo y control. Combina cientos de bibliotecas pre escritas, estrecha integración con hardware comercial y una variedad de enfoques de programación incluyendo desarrollo gráfico, scripts de archivos .m y conectividad a código C y HDL existente. Ya sea para diseñar dispositivos médicos o

robots complejos, puede reducir el tiempo y los costos totales del monitoreo y control embebidos con LabVIEW (National Instruments, 2012).

VISA

Visa es un estándar para configurar, programar y solucionar problemas de sistemas de instrumentación comprendiendo GPIB, VIX, PXI, Serial, Ethernet, y/o interfaz USB. VISA provee la interfaz de programación entre el hardware y entornos de desarrollo como LabView. NI-VISA es la implementación de National Instruments para el estándar VISA I/O. NI-VISA incluye librerías de software, utilidades interactivas como NI Spy y el control interactivo VISA, y programas de configuración a través de Medición y Automatización para todas las necesidades de desarrollo (National Instruments, 2012).

FUNCIONES O BLOQUES UTILIZADOS EN EL PROYECTO

Para la elaboración del SISTEMA DE MONITOREO REMOTO DE UN PROCESO INDUSTRIAL EMPLEANDO LA RED DE TELEFONIA MOVIL EN COLOMBIA se emplearon los siguientes bloques, estructuras o funciones: Ver figuras 2-35, 2-36, 2-37 y 2-38.

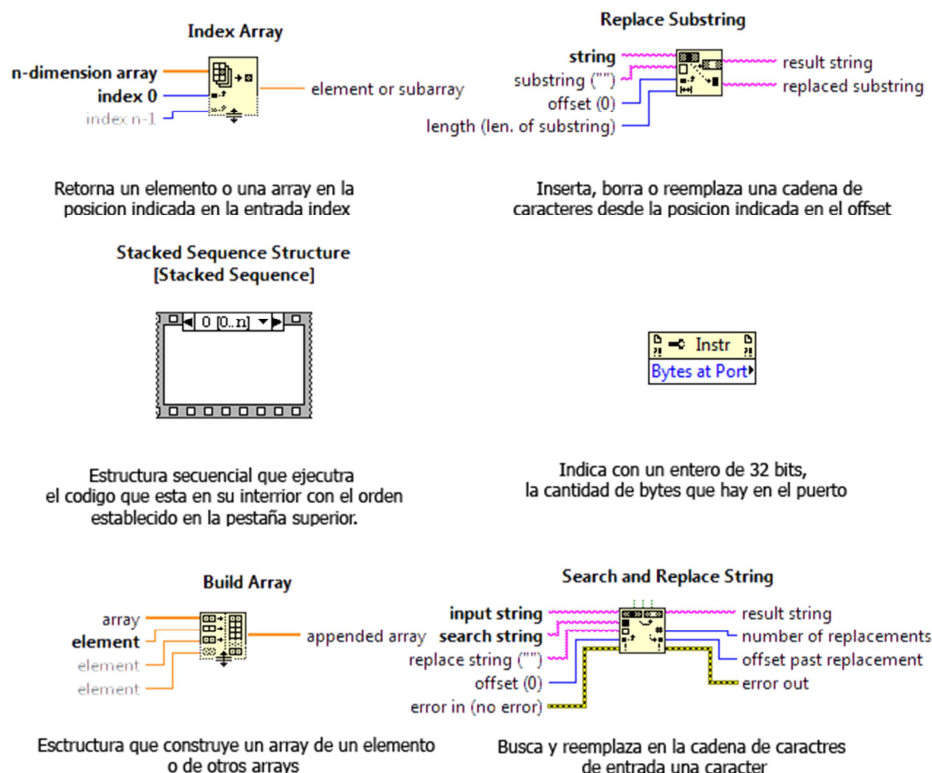
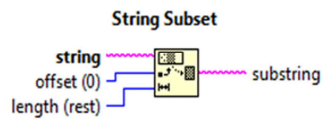
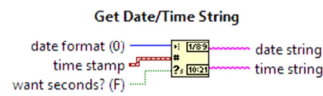


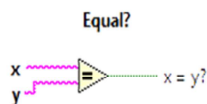
Figura 2-35: Bloques de LabVIEW utilizados Parte 1



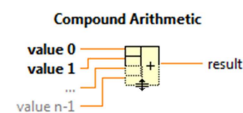
Retorna la cadena de valores desde la posicion indicada con la longitud indicada.



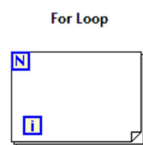
Obtiene la fecha o la hora actual del sistema



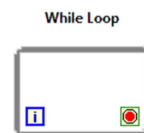
Estructura comparativa entre dos cadenas de caracteres. La respuesta es booleana, TRUE y FALSA



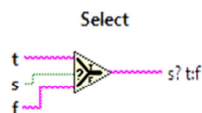
Estructura que tiene especificaciones aritmeticas de uno o mas numeros, array, cluster o entrada booleana.



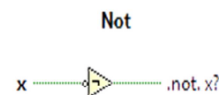
Estructura FOR que tiene n iteracciones. Donde i es la n-esima iteracion.



Estructura While, donde i es la n-esima iteracion y la condicion de parada es una seña booleana TRUE en el cuadro rojo.



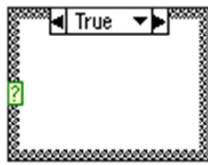
Estructura selectiva que selecciona entre dos entradas, una para la condicion TRUE (t) y otra para la condicionn FALSE (f) en la entrada s.



Esctructura que niega la entrada booleana.

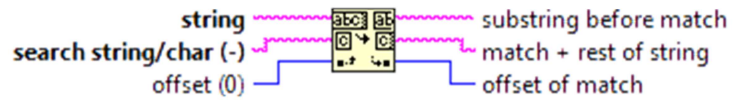
Figura 2-36: Bloques de LabVIEW utilizados Parte 2

Case Structure



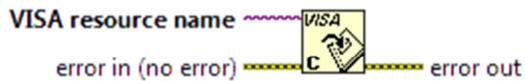
Estructura condicional parecida al SWITCH, IF y TRY. Su función es ejecutar un código dependiendo de la condición de entrada.

Search/Split String



Divide una cadena de caracteres en dos sub-cadenas de caracteres. Esto lo hace con la búsqueda de un carácter.

VISA Close



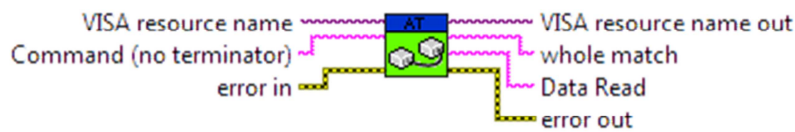
Cierra la comunicación con el puerto. VISA resource name es el ID del puerto de comunicación.

Decimal String To Number



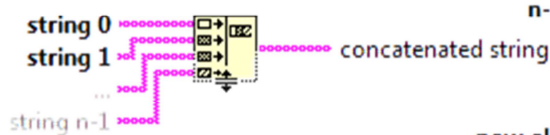
Convierte la cadena de caracteres numérica, en datos enteros de 32 bits

SubVI GSM lectura-escritura.vi



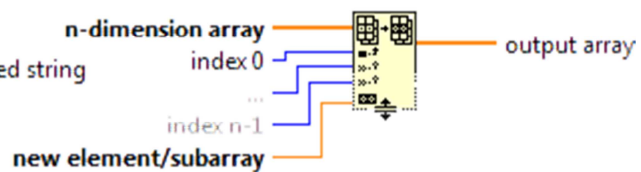
Bloque de lectura-escritura en el puerto, para el envío de mensajes y lectura de respuesta del modem.

Concatenate Strings



Concatenación de cadenas de caracteres en una sola cadena.

Replace Array Subset



Reemplaza un elemento o una subarray en una array en el punto especificado en la entrada index.

Figura 2-37: Bloques de LabVIEW utilizados Parte 3

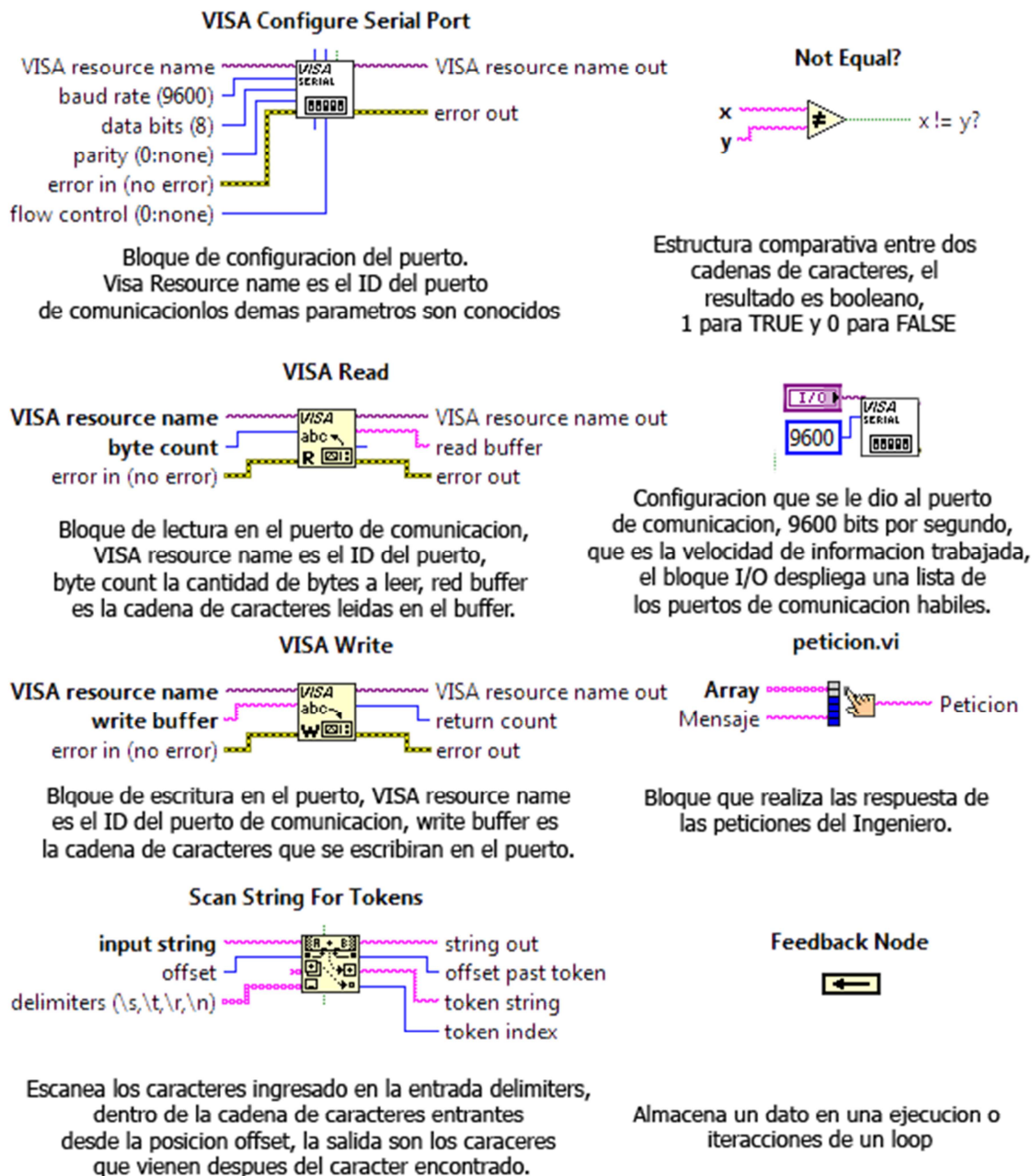


Figura 2-38: Bloques de LabVIEW utilizados

CAPÍTULO III

3 MARCO METODOLÓGICO

En este capítulo se presenta el marco metodológico del proyecto, donde se determina el tipo de investigación a realizar, diseño de la investigación y descripción de la realización del proyecto en cada una de las siguientes etapas:

- PROTOTIPO PARA LA CONFIGURACIÓN DE MODEM GSM.
- APLICACIÓN SISDAQ
- APLICACIÓN IVIGED

3.1 TIPO DE INVESTIGACIÓN

El tipo de investigación empleada es una investigación de tipo proyecto factible la cual tiene como resultado un prototipo del sistema que soluciona el problema planteado. La investigación se apoya en un estudio tipo descriptivo y en una revisión documental.

En este capítulo se enuncia, la forma como se realiza la documentación, la implementación y las pruebas necesarias para asegurar el funcionamiento adecuado del sistema de monitoreo remoto de un proceso industrial empleando la red de telefonía móvil de Colombia.

3.2 DISEÑO DE LA INVESTIGACIÓN

A continuación se determinan los pasos seguidos para abordar cada uno de los objetivos propuestos.

3.2.1 DESCRIPCIÓN DEL SISTEMA

En esta etapa se describe el software y los componentes de hardware necesarios para el desarrollo del proyecto que se desea implementar. La definición textual del sistema se describe a continuación.

Se requiere un sistema que sea capaz de obtener información de las variables involucradas en un proceso industrial aislado, que implique bajo costo en la implementación y gestione la información de forma automática para obtener acceso a ella en cualquier lugar y momento.

Para ello se realizaron dos aplicaciones, la aplicación SISDAQ y la aplicación IVIGED, SISDAQ se encarga de comunicarse con un terminal de datos, que en este caso actúa como fuente de información, para ello realiza una petición al terminal de datos que debe ser correspondida, de esta forma se logra la captura de información de las variables del proceso industrial. Después de los capturados los datos SISDAQ envía la información a la aplicación IVIGED.

IVIGED es un módulo administrador que se encarga del procesamiento (almacenamiento, organización y distribución) de la información que es suministrada por la etapa anterior. Para esto, IVIGED cuenta con una aplicación software que opera un modem GSM para la recepción de datos, peticiones y envío de notificaciones, es decir, que el usuario tiene la facultad de hacer peticiones sobre las variables, por medio de mensajes de texto, con esto se le brinda acceso a la información desde cualquier lugar y momento. Ver figura 3-1.

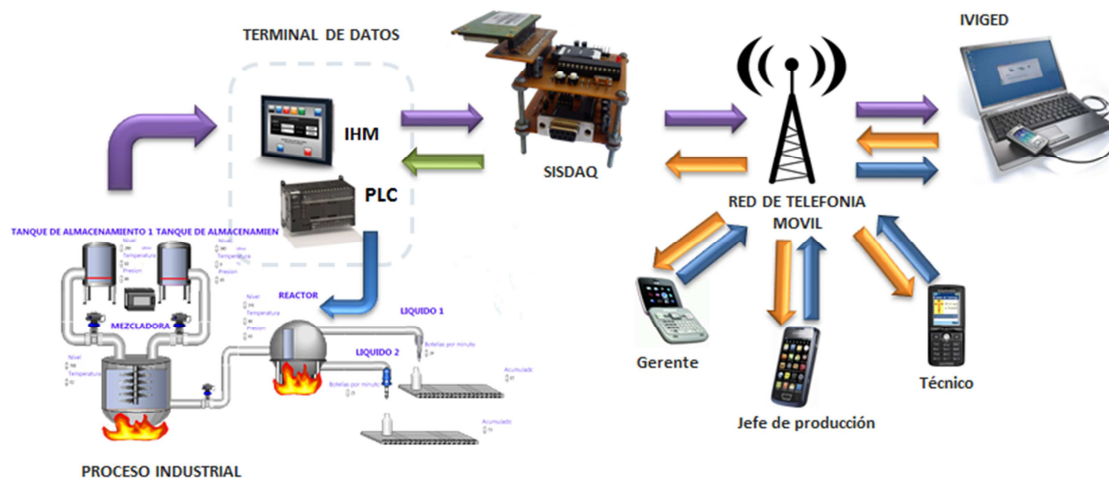


Figura 3-1: Sistema de monitoreo remoto empleando redes de telefonía móvil celular

3.2.2 ARQUITECTURA DEL SISTEMA

El modelado de capas de un sistema permite mejorar, reparar y remplazar partes de una aplicación de forma sencilla y sin que afecte a las demás partes, para el desarrollo del proyecto se escogió el modelo vista controlador. Este separa los datos de una aplicación en tres componentes distintos que son interfaz de usuario, lógica y control, con el fin de mejorar la reusabilidad de la aplicación al desacoplar la vista del modelo. Los elementos del patrón son los siguientes:

- El modelo: se encarga de acceder a la capa de almacenamiento de datos y definir la funcionalidad del sistema; como las políticas de envío de notificación con los valores de las variables monitoreadas.
- El control: recibe los eventos de entrada, por ejemplo un cambio en un variable del proceso monitoreado, y establece las reglas de la gestión de eventos es decir que si ya ocurrió un evento que tipo de acción debe ejecutar el sistema.
- La vista: recibe los datos del modelo y los muestra al usuario en un SMS o desde la interfaz de la aplicación servidoras.

En la siguiente figura se observa un diagrama de bloques del patrón de arquitectura de diseño de la aplicación.

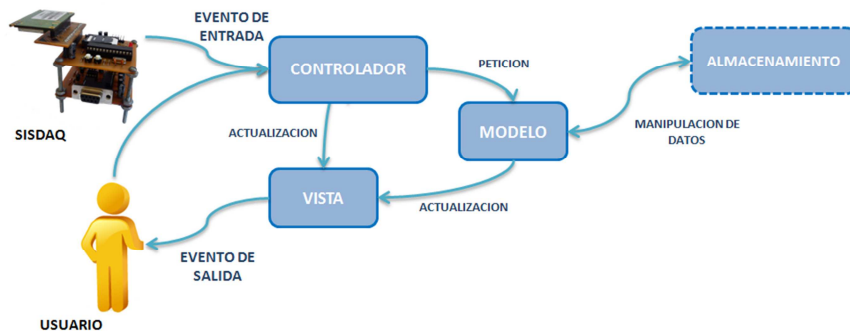


Figura 3-2: Modelo vista controlador

3.2.3 GUIA DE DESARROLLO DE ACTIVIDADES

En esta etapa se establecen las actividades necesarias para garantizar la consecución de los objetivos propuestos en el proyecto, con el fin de realizar una aplicación que preste un servicio efectivo y que sea agradable a los usuarios. A continuación se presentan las actividades desarrolladas:

- Para diseñar un prototipo de aplicación hardware, que permita realizar pruebas experimentales para la configuración adecuada de un modem GSM, la configuración de la interfaz serial entre modem GSM – microcontrolador y microcontrolador-terminal de datos, se realizan las siguientes actividades:
 - Diseñar los circuitos, para la fuente de alimentación e interfaces de comunicación con un terminal de datos y periféricos.
 - Analizar las especificaciones eléctricas, mecánicas de cada componente.
 - Fabricar PCB.
- Para obtener la información de las variables del proceso industrial a través del terminal de datos por medio de una interfaz serial RS232 entre el terminal de datos y un microcontrolador se realizan las siguientes actividades:
 - Diseñar la interfaz serial ajustada a niveles de voltaje RS232 y TTL.
 - Diseñar un circuito para suministrar energía al microcontrolador.
 - Compilar, verificar y ejecutar un algoritmo o programa en el microcontrolador para lograr la comunicación con el terminal de datos.
 - Desarrollar una sintaxis compuesta por comandos AT, que sea común tanto para el terminal de datos como para el microcontrolador, para que el terminal de datos responda de forma adecuada y entregue la información que se le pide.
 - Fabricar PCB.

- Para establecer la comunicación serial entre el modem GSM y el microcontrolador, basándose en la tradicional arquitectura (DTE-DCE), para el envío de la información a través de mensajes de texto, se realizan las siguientes actividades:
 - Diseñar una interfaz serial ajustada a niveles de voltaje TTL y al nivel especificado por el modem GSM.
 - Compilar, verificar y ejecutar un algoritmo o programa en el microcontrolador para lograr la comunicación con el terminal de datos a través de comandos AT, enviándolos a través de la interface serial, además que ordene los datos capturados y los envíe a través de mensajes de texto, este algoritmo también debe responder a las solicitudes del servidor.
 - Fabricar PCB.
- Para elaborar una aplicación servidora para la gestión de la información obtenida del terminal de datos, que se encargue de, comunicarse con el teléfono móvil conectado a la PC, y capturar los datos recibidos en los mensajes de texto, procesar los datos y enviar las notificaciones respectivas a los usuarios respectivos, se realizan las siguientes actividades:
 - Establecer la conexión del teléfono celular y la aplicación servidora por medio de LabVIEW.
 - Desarrollar un algoritmo capaz de obtener y organizar los datos recibidos en los mensajes de texto.
 - Diseñar un algoritmo para procesamiento de los datos.
 - Definir los parámetros y variables de la aplicación servidora, que genere automáticamente el contenido de los mensajes de texto, dependiendo de las políticas de administración.
 - Estructurar el contenido de los mensajes SMS que se envían como respuesta a las consultas realizadas por los usuarios.

3.3 DESARROLLO DEL PROTOTIPO DE PRUEBAS PARA LA CONFIGURACIÓN DEL MODEM GSM

El prototipo de prueba es una interfaz que permite la comunicación serial entre el modem GSM SIMCOM SIM300 y el puerto serie de un terminal de datos.

Está compuesta por 5 partes:

- Fuente de alimentación: Su función es suministrar energía a todos los dispositivos implicados en la aplicación.

- Etapa de comunicación: Su función es la comunicación entre el modem y un terminal de datos.
- Etapa de encendido y apagado para modulo: Se encarga del encendido y apagado del modem GSM.
- Interfaz para SIM CARD: su función es conectar la SIMCARD al modem GSM
- Conector para modem GSM: su función es pasar de un conector de 60 pines a un conector de 19 pines para conectar en la PCB.

El funcionamiento se describe mediante el siguiente diagrama de bloques mostrado en la figura 3-3.

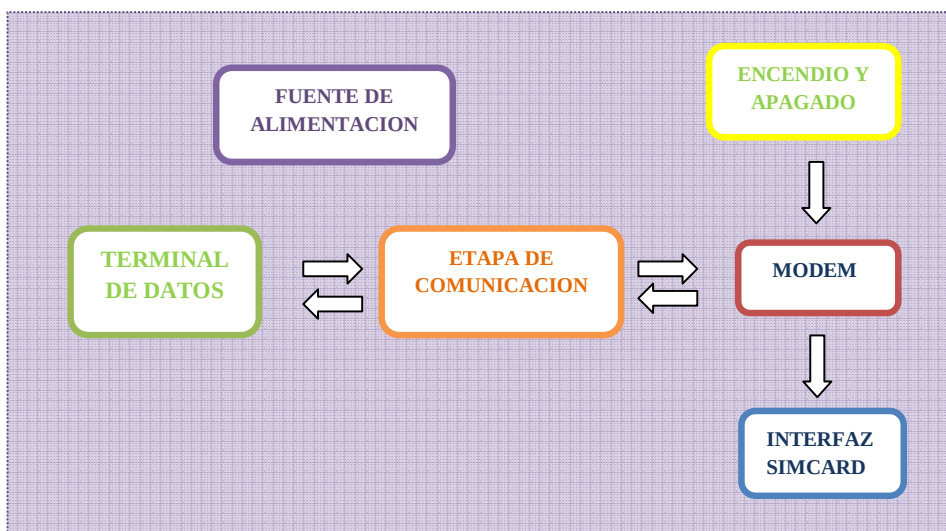


Figura 3-3: Diagrama de bloques del funcionamiento de prototipo de prueba

3.3.1 FUENTE DE ALIMENTACIÓN

REQUERIMIENTOS

- Fuente de tensión DC regulada a 4V que suministre corriente suficiente para soportar una carga que puede llegar a consumir picos de corriente de hasta 2A en un lapso de tiempo de 0.1ms.
- Fuente de tensión DC regulada a 5V que para suministrar alimentación a integrados TTL de bajo consumo de corriente.

CIRCUITO DE REGULACIÓN DE TENSIÓN DC DE 4V

Para el diseño de esta fuente se elige un regulador en serie como se muestra en la figura 3-4.

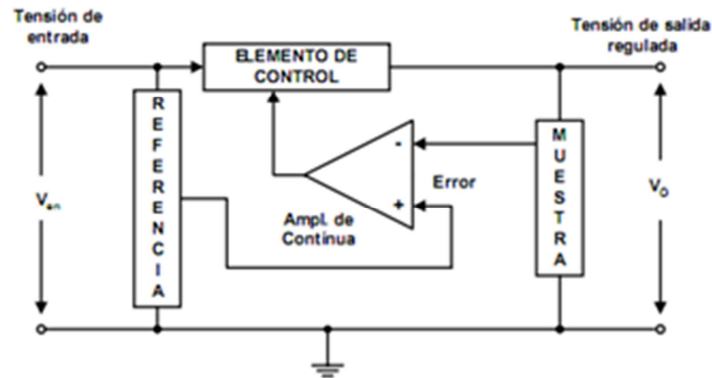


Figura 3-4: Regulador en serie básico, Fuente: (Mata, 2001)

Este es un regulador en serie básico que se divide en 4 partes:

Elemento de Referencia: Es un nivel de tensión estable ante la tensión de entrada, corriente de carga y temperatura de ambiente de operación (Mata, 2001).

Elemento de muestreo: El elemento de muestreo tomará el voltaje de salida y lo acondicionará para entregar una fracción equivalente al nivel de la tensión de referencia, para un voltaje de salida especificado. Cuando se producen variaciones en el voltaje de salida, se producirá un cambio en el nivel de la tensión realimentada en un valor que podrá ser mayor o menor que la tensión de referencia. Esta diferencia o voltaje de error conducirá directamente al elemento regulador para compensar la diferencia detectada, corrigiendo de esta manera la tensión de salida para llevarla al voltaje deseado (Mata, 2001).

Elemento de comparación: Este elemento de comparación aparte de monitorear la diferencia entre la tensión de referencia y el voltaje de salida muestreado, producirá la amplificación necesaria para la señal de error. Por esta razón, es normal que se denomine al elemento de comparación como amplificador de error. La salida de error amplificada producirá sobre el circuito de control del regulador la variación paramétrica correspondiente para retornar al voltaje prescrito de salida (Mata, 2001).

Elemento de control: En este tipo de regulador generalmente el elemento de control es un transistor que se comporta como una resistencia variable, cualquier variación de la tensión de entrada implica una variación en la resistencia en serie, siempre y cuando la corriente en la carga no varíé simultáneamente, por lo que en todo instante la caída de tensión en la

resistencia en serie por la que circula la corriente de carga I_L será la diferencia del voltaje de entrada menos la tensión fija seleccionada para la salida.

ELEMENTO DE REFERENCIA

El elemento de referencia está conformado por el siguiente circuito, denominado circuito referencia independiente de las variaciones de entrada. Ver figura 3-5.

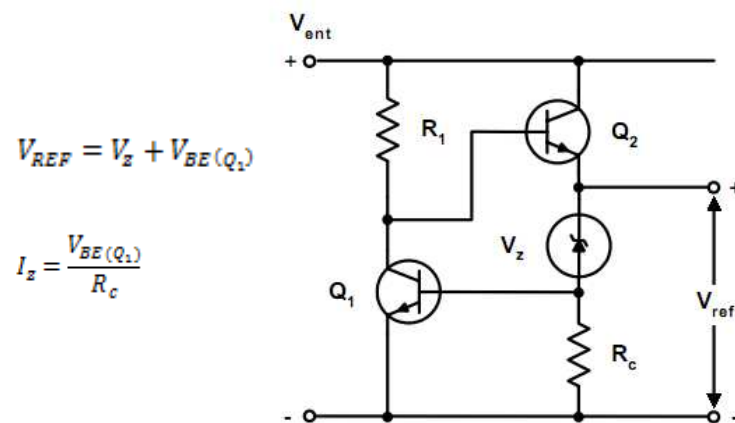


Figura 3-5: Circuito de referencia independiente, Fuente: (Mata, 2001)

Debido a que la fuente de tensión que se requiere es de 4 voltios, se escogió $V_z = 3.3 V$, también se escogió el diodo zener 1N4728A entonces $V_{REF} = 3.3V + 0.7V = 4V$ de la hoja de datos del diodo se tiene que $I_{zmax} = 100mA$ entonces $I_{zmin} = 0.1I_{zmax}$ $I_{zmin} = 10mA$ para disminuir los efectos de la temperatura se calcula la resistencia más alta posible en este caso $R_c = \frac{0.7}{10mA} = 70\Omega$, el valor comercial más cercano es 68Ω .

ELEMENTO DE MUESTREO

El elemento de muestreo está conformado por un divisor de voltaje como se muestra en la figura 3-6, se toma una muestra del valor de voltaje de salida para ser comparado con la referencia y obtener una señal de error.

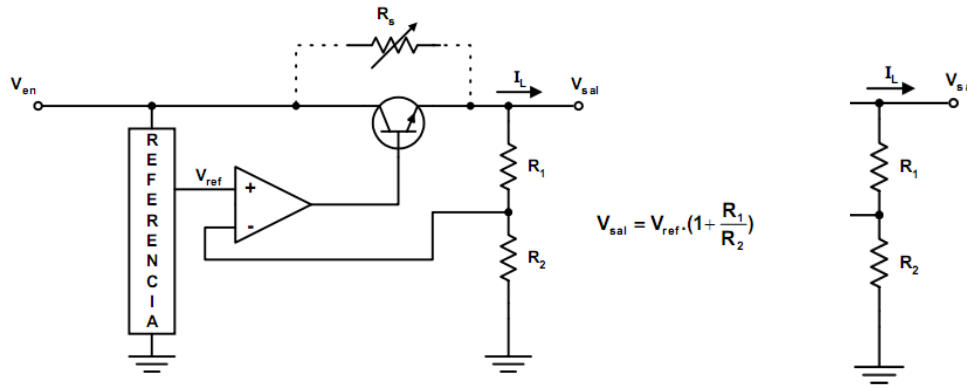


Figura 3-6: Circuito de muestreo, Fuente: (Mata, 2001)

La ganancia de voltaje de respecto al voltaje de referencia es $V_{sal} = V_{ref} * (1 + \frac{R_1}{R_2})$, se escogen los valores de tal forma que el cociente sea cercano a cero para que $V_{sal} \cong V_{ref}$ por lo que se $R_1 = 220$ y $R_2 = 330K$.

ELEMENTO DE COMPARACIÓN

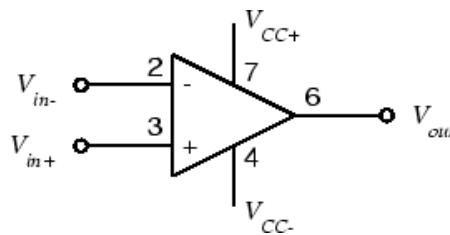


Figura 3-7: Amplificador Operacional 741, Fuente: (ECE, 2012)

El elemento de comparación es un amplificador operacional en este caso el UA741, es un amplificador operacional de propósito general, es el encargado de hacer la diferencia y amplificar el error.

ELEMENTO DE CONTROL

Como elemento de control se escoge al transistor 2N3055, que se caracteriza por ser un transistor de potencia y por su alta ganancia de corriente, soporta una corriente de colector de 15 A, que es un valor suficiente para suplir las demandas de corriente de la carga. Ver Figura 3-8.

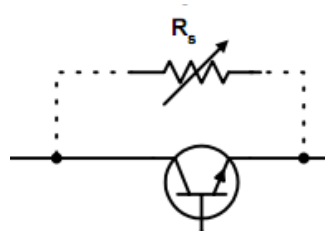


Figura 3-8: Transistor de control, Fuente: (Mata, 2001).

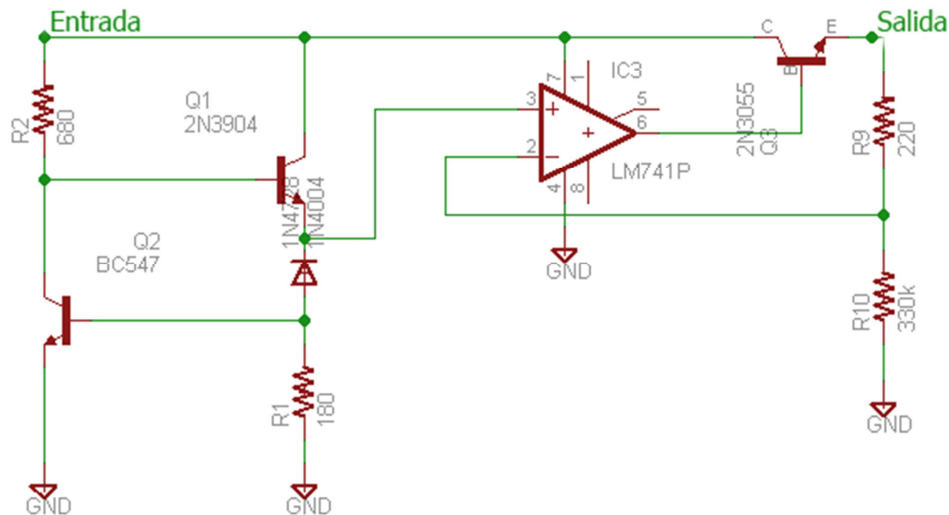


Figura 3-9: Diagrama esquemático de fuente regulada de tensión de 4 VDC.

Para aumentar la corriente en la base del transistor se colocaron tres amplificadores operacionales en paralelo como fuentes de voltaje en paralelo.

CIRCUITO DE REGULACIÓN DE TENSIÓN DC DE 5V

Para el circuito de regulación de tensión DC a 5V se utilizó el circuito integrado LM7805 que es un regulador integrado a 5V, con la configuración típica de regulador con cómo se muestra en la hoja de datos del integrado, ver figura 3-10.

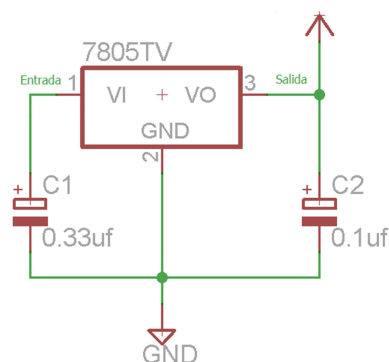


Figura 3-10: Diagrama esquemático circuito de regulación de tensión 5V

3.3.2 ETAPA DE COMUNICACIÓN

A continuación se describe la etapa de comunicación entre el modem y el TE, se divide en dos partes:

- Conversor de niveles TTL_RS232
- Conversor de niveles TTL_ 4V DC

CONVERSÓR DE NIVELES TTL-RS232

Para establecer la comunicación serial entre modem GSM y el puerto serial de la PC es necesario utilizar un conversor de señales de puerto serie RS-232 a niveles lógicos TTL como lo es el MAX232 un circuito ampliamente utilizado en este tipo de aplicaciones. Cuando un circuito integrado MAX232 recibe un nivel TTL lo convierte, cambia un nivel lógico TTL de 0 a un nivel comprendido entre +3 y +15 V, y cambia un nivel lógico TTL 1 a un nivel comprendido entre -3 a -15 V, y viceversa. Ver figura 3-11.

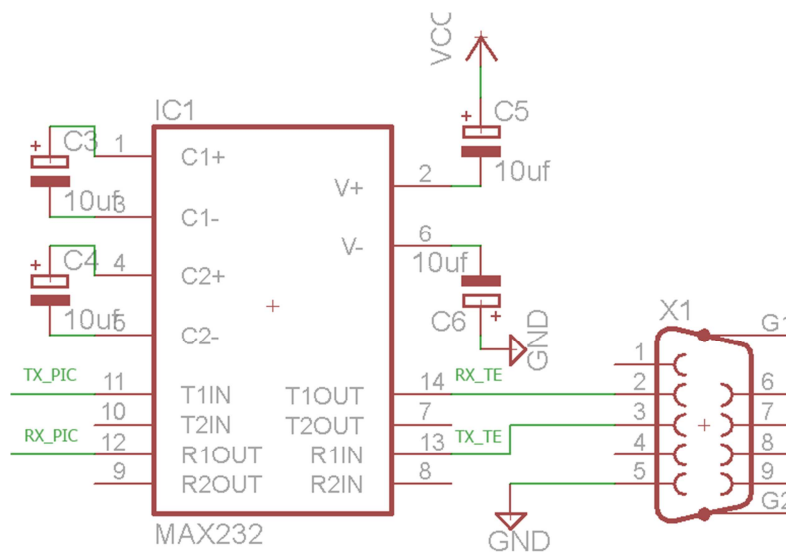


Figura 3-11: Diagrama esquemático de conversor de niveles TTL-RS232

CONVERSÓR DE NIVELES TTL-4V DC

La siguiente tabla describe los valores lógicos de voltaje soportados por el puerto serial asíncrono del módulo SIMCOM SIM300.

PARÁMETRO	MIN	MAX	UNIDAD
Entrada lógica baja	0	0,3*VDD-EXT	V
Entrada lógica alta	0,7*VDD-EXT	VDD-EXT+0.2	V
Salida lógica baja	GND	0.2	V
Salida lógica alta	VDD-EXT-0.2	VDD-EXT	V

Tabla 3-1: Voltajes soportador por el puerto serial asíncrono del módulo GSM

Donde VDD-EXT=2,93V.

Para convertir la señal TTL para los niveles de voltaje adecuado se elaboró un circuito transmisor y un circuito receptor, cuyo diseño se describe a continuación:

Calculo del circuito para el receptor

Para reducir el nivel de tensión se empleó un divisor de voltaje, como se muestra en la figura 3-12. $V_{Rec} = \frac{R_3 * V_{TTL}}{R_3 + R_4}$ se escogió $V_{Rec} = 2.93$ entonces

$$\frac{V_{Rec}}{V_{TTL}} = \frac{R_3}{R_3 + R_4}$$

$$\frac{2.93}{5} = \frac{R_3}{R_3 + R_4}$$

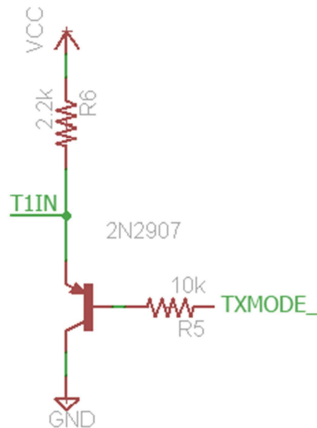
$$R_4 = \frac{R_3}{0.586} - R_3$$

Si $R_3 = 10K$ entoces $R_4 = 7K \cong 6.8K$.

Calculo del circuito para el transmisor

Para el circuito transmisor se elevó la tensión hasta 5V DC, por medio del corte y saturación de un transistor PNP se logra la conversión, T1IN se encuentra en un nivel alto hasta que TXMODE = 0 aquí el transistor se satura y T1IN = 0.

Circuito transmisor



Circuito Receptor

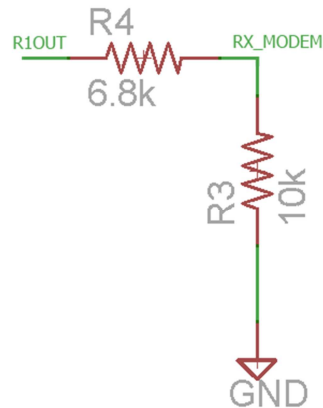


Figura 3-12: Diagrama esquemático de circuito conversor de niveles TTL a 4V DC.

3.3.3 ETAPA DE ENCENDIDO Y APAGADO PARA MODULO GSM SIMCOM SIM300

El módulo SIMCOM SIM300 presenta varios escenarios de encendido y apagado, pero la forma recomendada es usando el pin PWR_KEY con el procedimiento que se muestra en la siguiente figura.

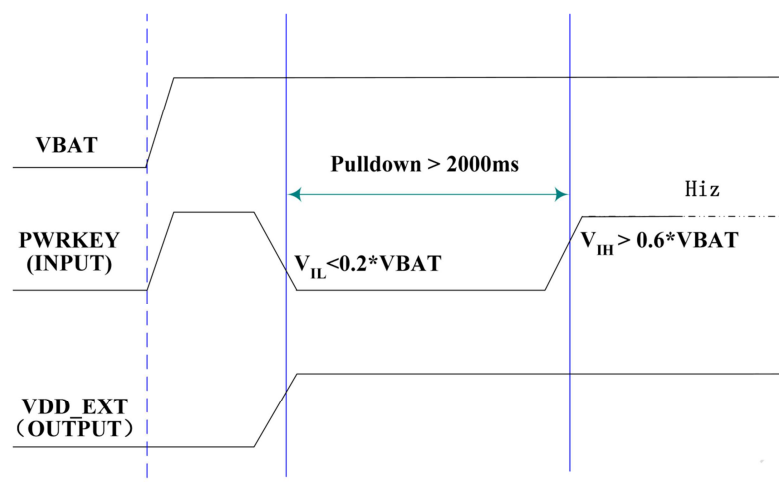


Figura 3-13: Temporización de encendido del módulo SIM300, Fuente: (SIMCOM, 2007)

Conduciendo a un nivel bajo durante un periodo de tiempo mayor que 1.5 segundos se logra el encendido del módulo luego de este tiempo se lleva nuevamente a un nivel alto. Ver figura 3-14.

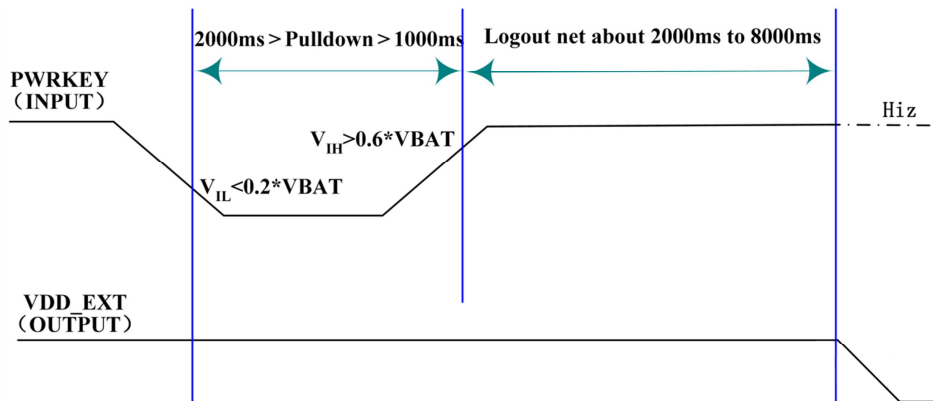


Figura 3-14: Temporización de apagado del módulo GSM, Fuente: (SIMCOM, 2007)

De forma similar al encendido el apagado del módulo se logra llevando el PWR_KEY a un nivel bajo durante un periodo de tiempo entre 1 y 0.5 segundos luego de este tiempo es necesario llevar el PWR_KEY a un nivel alto.

Por medio del PIC16F84A se logra suministrar la temporización necesaria para el encendido y apagado del módulo. De acuerdo a la hoja de datos del módulo, el valor alto del pin PWR_KEY y STATUS es igual al voltaje de alimentación, es decir 4V DC y el nivel bajo es GND, los niveles de voltajes lógicos del PIC son TTL, por lo que es necesario la conversión de TTL a 4V DC y viceversa, para este caso se usó la misma solución que para el circuito del transmisor del módulo.

De forma similar que en el circuito transmisor se utiliza un transistor PNP en corte o saturación pero con la fuente de alimentación VBAT del modem GSM como se observa en la figura 3-15.

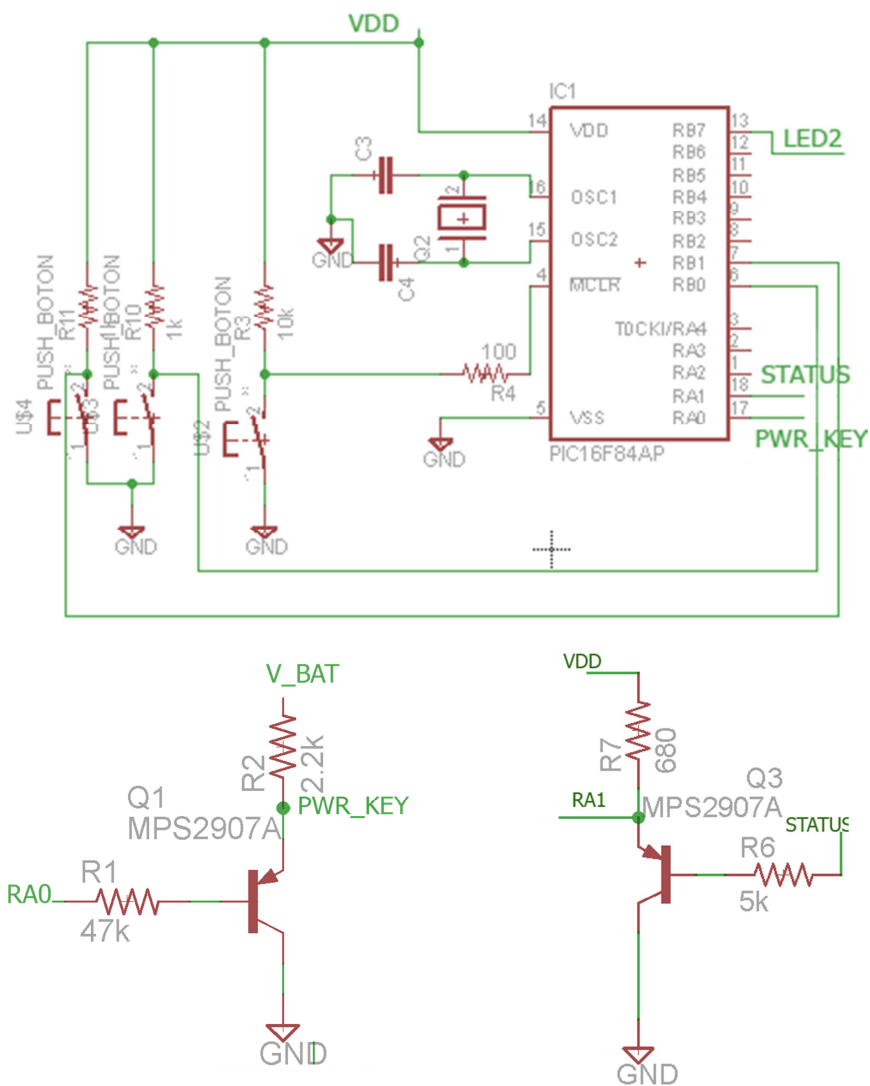


Figura 3-15: Diagrama esquemático de etapa de control.

3.3.4 INTERFAZ PARA SIM CARD

La interfaz para la SIMCARD soporta las especificaciones de GSM fase número 1 y fase número 2 para una velocidad máxima de 64Kbps, la interface de la SIM es alimentada por un regulador interno que tiene un voltaje nominal de 2.8V, por lo que la conexión de la SIMCARD al modem GSM es directa y no se necesita de ningún componente electrónico. Ver figura 3-16.

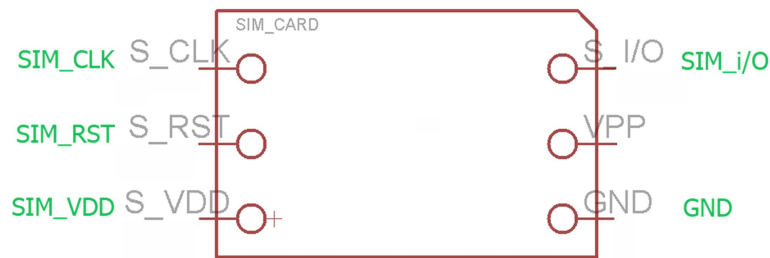


Figura 3-16: Diagrama esquemático de la interfaz para la SIMCARD

3.3.5 CONECTOR MODEM SIMCOM SIM300

De acuerdo a la descripción de pines del modem GSM de la tabla 6 modem SIM300 tiene 60 pines cada uno con su respectiva función, para el desarrollo de la aplicación no es necesario utilizar todos los pines del dispositivo. De acuerdo a las funciones que cumplen se seleccionaron los pines que se muestran en la tabla 6.

PIN NO.	PIN NAME	I/O	PIN NO.	PIN NAME	I/O
2	GND		1	VBAT	I
4	GND		3	VBAT	I
6	GND		5	VBAT	I
8	GND		7	VBAT	I
10	GND		9	VBAT	I
12	ADC1	I	11	CHG_IN	I
14	VRTC	I	13	NC	I
16	Network LED	O	15	VDD_EXT	O
18	KBC0	O	17	PWRKEY	I
20	KBC1	O	19	STATUS	O
22	KBC2	O	21	GPIO5	I/O
24	KBC3	O	23	BUZZER	O
26	KBC4	O	25	SIM_VDD	O
28	KBR0	I	27	SIM_RST	O
30	KBR1	I	29	SIM_I/O	I/O
32	KBR2	I	31	SIM_CLK	O
34	KBR3	I	33	SIM_PRESENT	I
36	KBR4	I	35	GPIO32	I/O
38	SPI_EN	O	37	DCD	O
40	SPI_CLK	O	39	DTR	I
42	SPI_DO	I/O	41	RXD	I
44	SPI_AO	O	43	TXD	O
46	SPI_RESET	O	45	RTS	I
48	DBG RX	I	47	CTS	O
50	DBG TX	O	49	RI	O
52	AGND		51	AGND	
54	MIC1P	I	53	SPK1P	O
56	MIC1N	I	55	SPK1N	O
58	MIC2P	I	57	SPK2P	O
60	MIC2N	I	59	SPK2N	O

Tabla 3-2: Descripción de pines modem SIMCOM 300.

El software Eagle PCB Layout no contiene en su librería de componentes un conector compatible con modulo por lo que fue necesario crear una librería a partir de las características mecánicas de este, ver figura 3-17. De acuerdo a la figura los pines útiles para la aplicación son 19 por este motivo se utilizó un conector convencional de 19 pines como conector del modem GSM para el PCB.

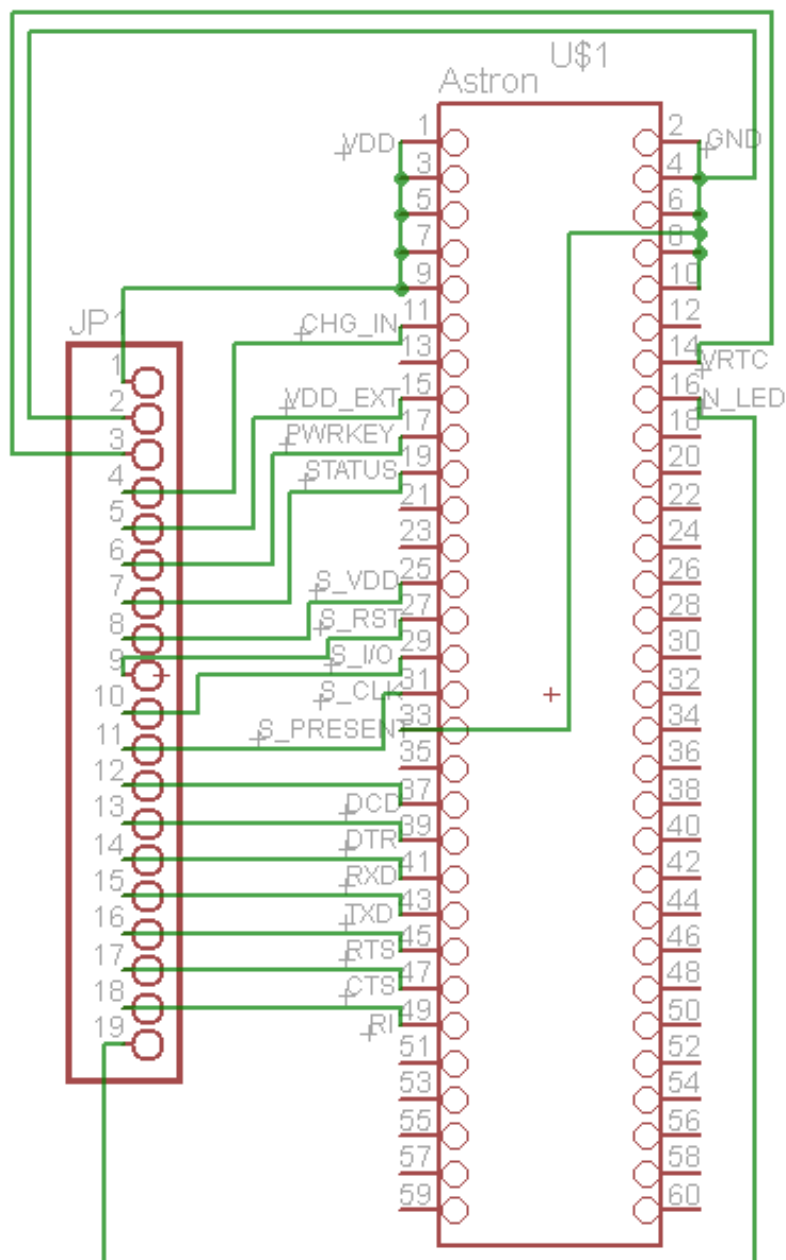


Figura 3-17: Diagrama esquemático de conexión de conector de 60 pines con conector de 19 pines.

3.3.6 DIAGRAMA ESQUEMATICO GENERAL DE PROTOTIPO DE PRUEBA

Para la elaboración de la aplicación del prototipo de prueba se dividió el circuito general en dos PCB's, el primero contiene la etapa de fuente de alimentación y una etapa de comunicación, mientras que en el segundo PCB se encuentra la etapa de control, interfaz para SIMCARD y etapa de conector modem SIMCOM, ver figuras 3-18 y 3-19.

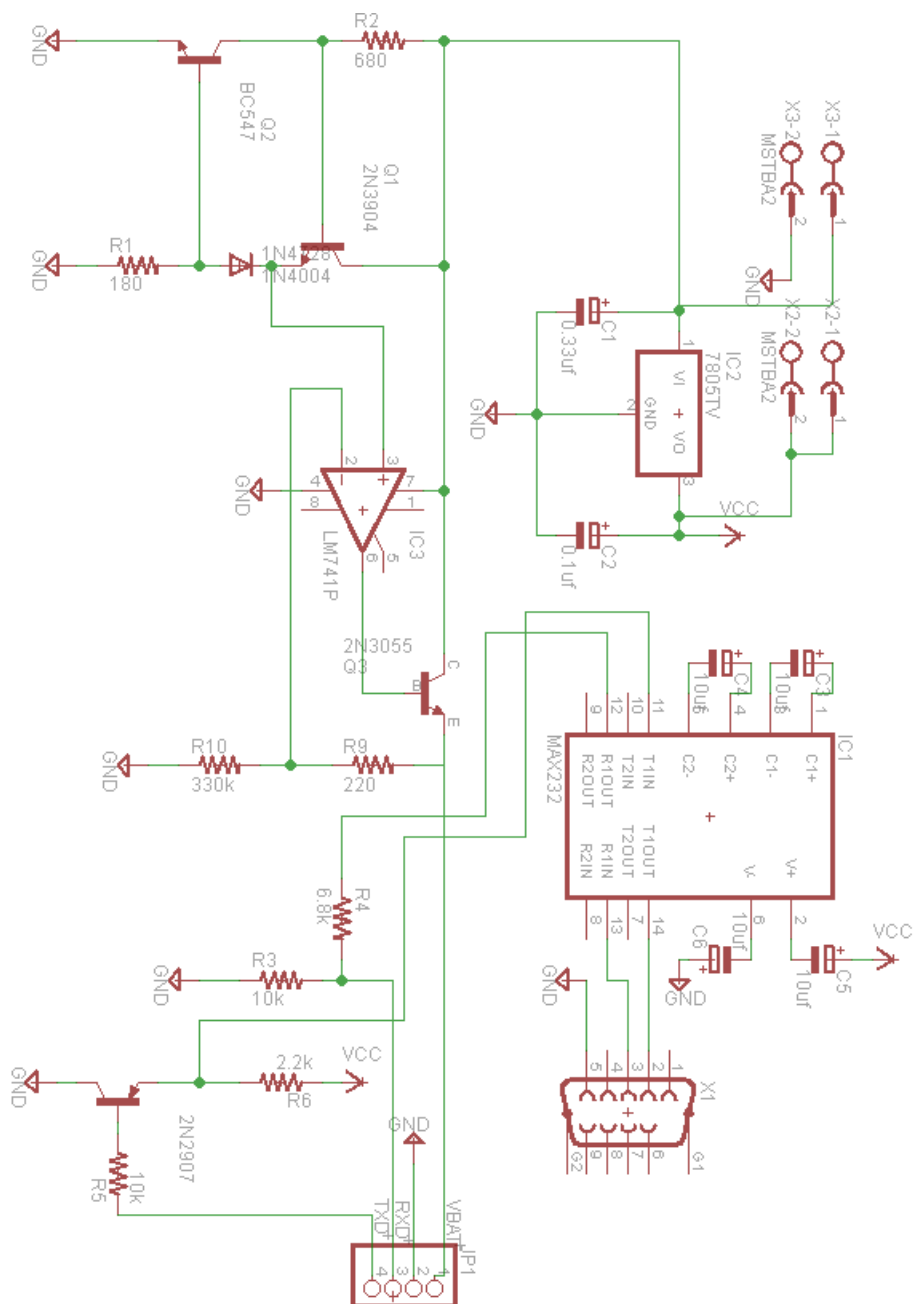


Figura 3-18: Diagrama esquemático general de prototipo de prueba parte 1.

Figura 3-19: Diagrama esquemático general de prototipo de prueba parte 2.

3.4 DESARROLLO HARDWARE DE LA APLICACIÓN SISDAQ

La aplicación SISDAQ es un módulo que administra la comunicación entre dos interfaces seriales, la interfaz serial PC-microcontrolador y un modem GSM-Microcontrolador a nivel hardware está compuesta por:

- Fuente de alimentación: Su función es suministrar energía a todos los dispositivos implicados en la aplicación.
- Etapa de comunicación entre modem GSM-microcontrolador.
- Etapa de comunicación entre terminal de datos-microcontrolador.
- Etapa de control para módulo GSM SIMCOM SIM300: Se encarga de las configuraciones del modem GSM, la captura de datos, la elaboración de mensajes de texto y el envío a la aplicación.
- Interfaz para SIM CARD: su función es conectar la SIMCARD al modem GSM.
- Conector MODEM SIMCOM SIM300: su función es pasar de un conector de 60 pines a un conector de 19 pines para conectar en la PCB y así comunicar al modem GSM con los demás dispositivos en el PCB.

Su funcionamiento se describe mediante el diagrama de bloques mostrado la figura 3-20.

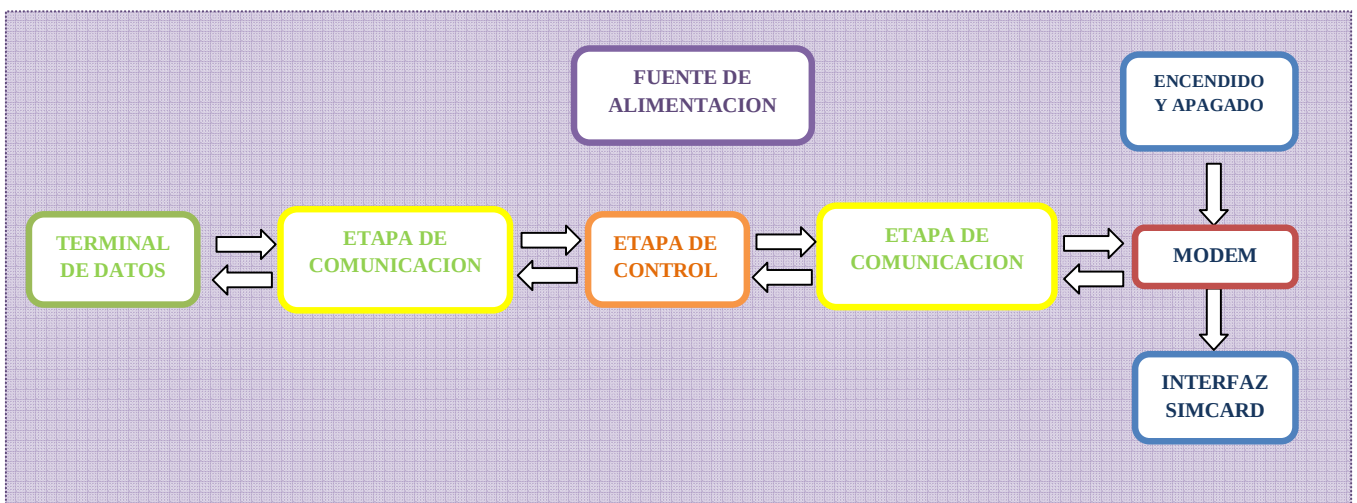


Figura 3-20: Diagrama de bloques del funcionamiento de aplicación

3.4.1 FUENTE DE ALIMENTACIÓN

REQUERIMIENTOS

- Fuente de tensión DC regulada a 4V que suministre corriente suficiente para soportar una carga que puede llegar a consumir picos de corriente de hasta 2A en un lapso de tiempo de 0.1ms.
- Fuente de tensión DC regulada a 5V que para suministrar alimentación a integrados TTL de bajo consumo de corriente.

CIRCUITO DE REGULACIÓN DE TENSIÓN DC DE 4V

Para la elaboración de esta aplicación se reutilizo el diseño de la fuente del prototipo de prueba descrita en la sección 3.3.1, con la diferencia en que el elemento de control es un transistor TIP41C y el elemento comparador está conformado por 4 operacionales en paralelo para conseguir un aumento en la corriente de base y por ende en la corriente que circula por el colector del transistor por lo que simplemente se adjunta el diagrama esquemático del circuito. Ver figura 3-21.

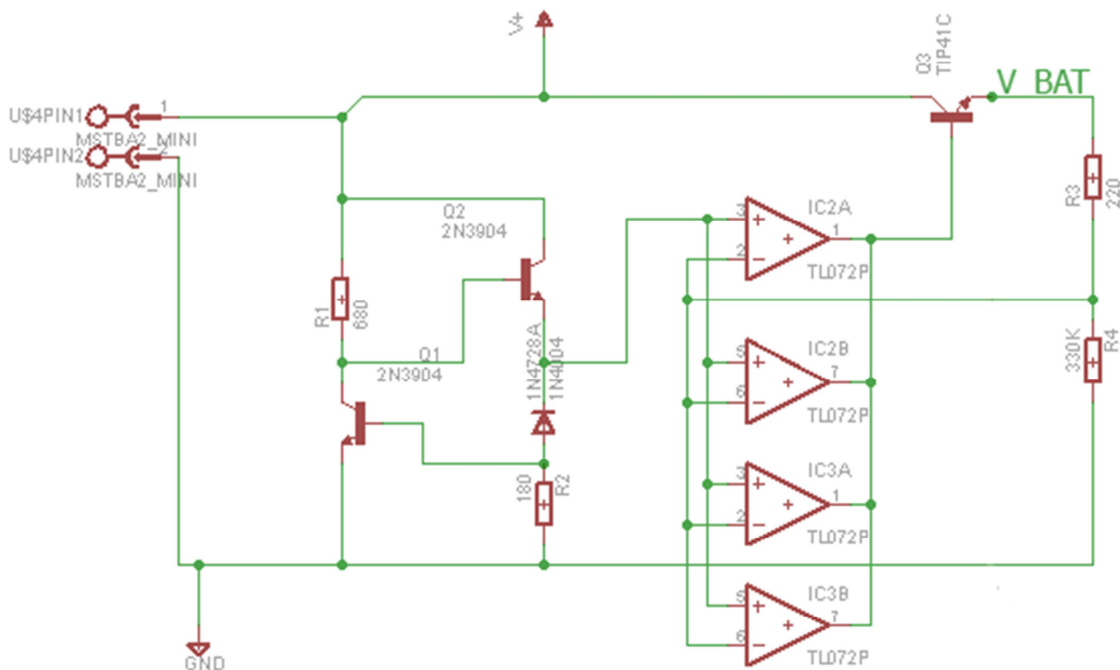


Figura 3-21: Circuito de regulación de tensión a 4V DC

3.4.4 ETAPA DE CONTROL PARA MODULO GSM SIMCOM SIM300.

ELECCIÓN DEL MICROCONTROLADOR

Para el desarrollo de este proyecto se ha elegido la familia de microcontroladores PIC que son fabricados por MICROCHIP por las siguientes razones:

- Fácil adaptabilidad: cuando se aprende a manejar uno de ellos, y se conoce su arquitectura y repertorio de instrucciones, es muy fácil emplear otro modelo.
- Gran distribución en el mercado: es fácil encontrar algunos PIC en las tiendas de electrónica, por lo que en caso de que se averíe un microcontrolador se puede encontrar fácil su remplazo.
- Herramientas de desarrollo gratuitas.
- Abundante información técnica gratuita oficial y de terceros (ambas por Internet).
- Distintos modelos adecuados para cada necesidad.

REQUERIMIENTOS DE LA APLICACIÓN

A continuación se presentan los requerimientos de hardware para el desarrollo de la aplicación:

- Módulo UART: para el desarrollo de la aplicación es necesario que el microcontrolador tenga la capacidad de establecer comunicación serial con dos dispositivos destinando una línea de comunicación para cada uno por separado, Microchip ofrece microcontroladores que están provistos de dos UART pero lastimosamente son difíciles de encontrar en el mercado, por lo que se exige que el microcontrolador tenga como mínimo un UART.
- Memoria de programa: para este tipo de aplicaciones se escoge un tamaño mínimo de 4Kbytes que es la cantidad de memoria que tiene un PLC S7-200 usado comúnmente en aplicaciones industriales.
- Memoria de datos: debido a que el proyecto se enfoca en el procesamiento de SMS, el mayor dato a tratar tiene un tamaño de 160 caracteres que es el número de caracteres máximo que contiene un SMS, un carácter tiene 8 bits por lo que un mensaje de texto completo ocupa un espacio de 1280 bits o 160 bytes pero es necesario dejar espacio suficiente para el procesamiento de otras variables por lo que se exige un PIC con tamaño de memoria no menor a 200bytes.

Los siguientes microcontroladores cumplen con las características citadas, además ofrecen la ventaja de que son comúnmente vendidos en las tiendas electrónicas:

CARACTERÍSTICAS	PIC16F876A	PIC16F877A
Frecuencia de operación	20 MHz	20 MHz
Memoria de programa (Flash)	14.3Kbytes	14.3 Kbytes
Memoria de datos	368 bytes	368 bytes
Memoria de datos EEPROM	256 bytes	256 bytes
Fuentes de interrupción	13	14
Puertos de entradas y salida	A,B,C	A,B,C,D,E
Interfaz de comunicación serial	MSSP, USART	MSSP, USART

Tabla 3-3: Características de los PIC16F876A y 16F877A.

Con base en los requerimientos de la aplicación y en los datos mostrados en la tabla 7 se escoge el PIC16F876A, puesto que se ajusta a las características necesarias y a diferencia del PIC16F877A es de menor tamaño con solo 28 pero tiene la misma cantidad de memoria de datos, memoria de programa y memoria EEPROM.

El control del modem GSM es ejercido por medio de un microcontrolador PIC16F876A, a través de comandos AT este realiza las debidas configuraciones para el funcionamiento del modem, le indica cuando enviar un mensaje, leer los mensajes recibidos y detectar el encendido del modem.

En la figura 3-24 se observa se observa el circuito de control para el modem GSM. Por medio del pin VDD.EXT el microcontrolador detecta el encendido del modem, este es el punto de inicio para el algoritmo que controla al modem a través de la etapa de comunicación.

3.4.6 INTERFAZ PARA SIM CARD7

Para el desarrollo de esta etapa se reutilizo el diseño descrito en el prototipo de prueba. Ver sección 3.3.4.

3.4.7 DAGRAMA ESQUEMATICO GENERAL DE LA APLICACIÓN SISDAQ

Para la elaboración de la aplicación del prototipo de prueba se dividió el circuito general en dos PCB's, el primero contiene la etapa de fuente de alimentación y la etapa de comunicación entre terminal de datos y microcontrolador, mientras que en el segundo PCB se encuentra la etapa de control, la etapa de comunicación entre microcontrolador y modem, interfaz para SIMCARD y el conector para modem SIMCOM, ver figura 3-26.

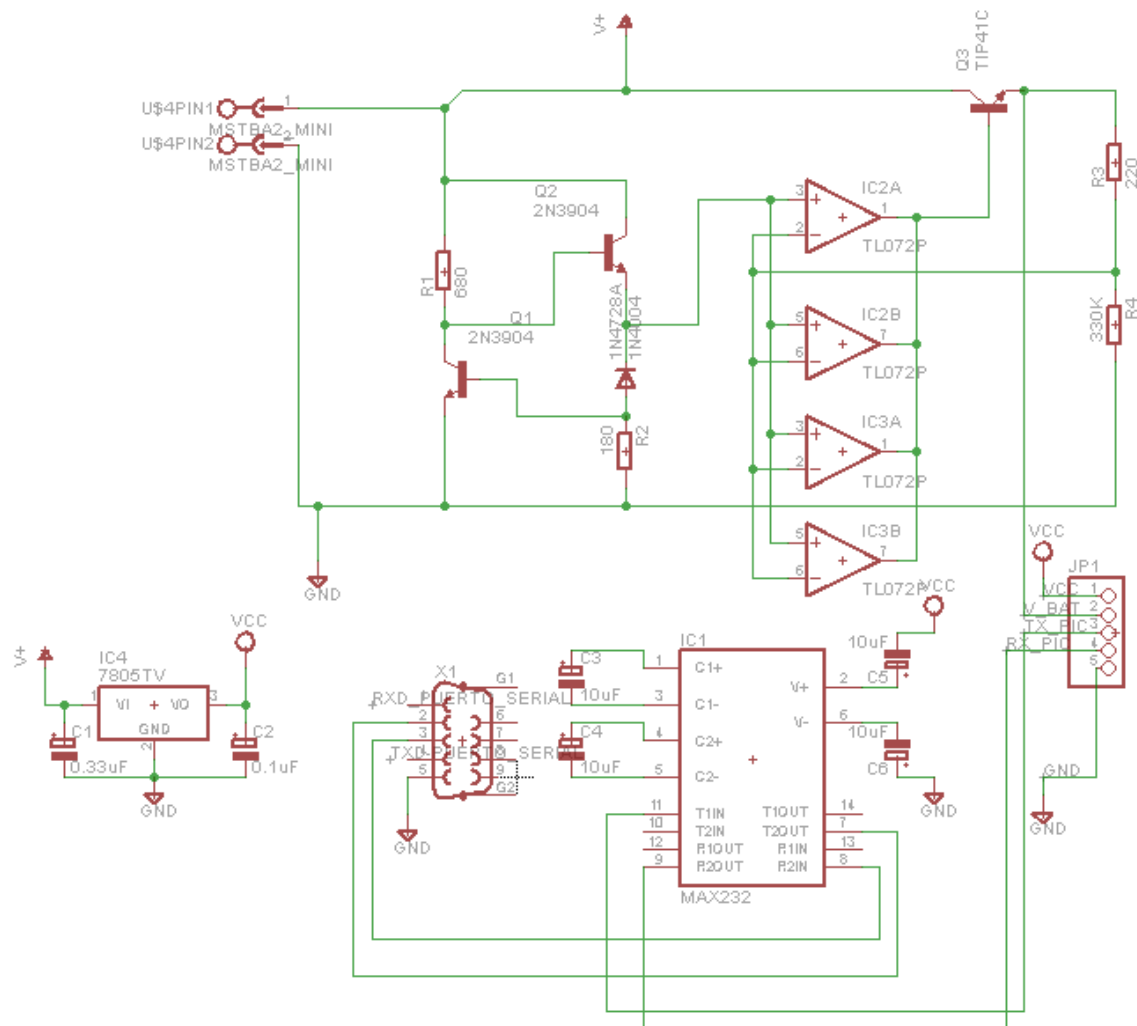


Figura 3-26: Diagrama esquemático general de la aplicación SISDAQ parte 1.

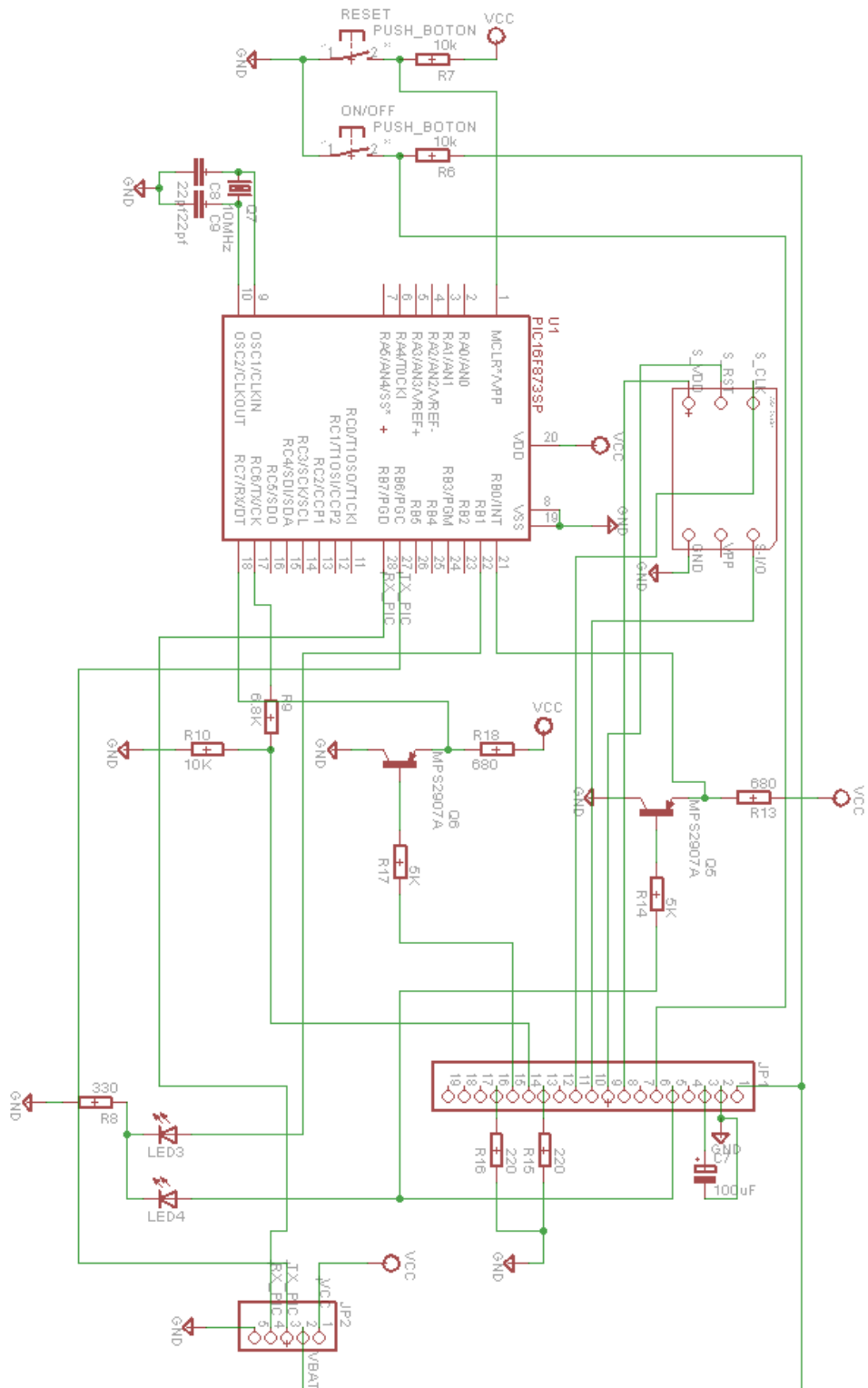


Figura 3-27: Diagrama esquemático general de la aplicación SISDAQ parte2

3.5 DESARROLLO FIRMWARE DE LA APLICACIÓN SISDAQ

En el siguiente apartado se describe con detalle el algoritmo implementado en el microcontrolador PIC16F876A, ver figura 3-28, se abordaran cada uno de los estados principales los cuales a su vez están compuestos por subalgoritmos, para ver el código completo compilado en MikroC ir al anexo.



Figura 3-28: Diagrama de flujo del algoritmo de programación para el microcontrolador

3.5.1 CONFIGURACIÓN DEL PIC

Una vez el microcontrolador es encendido, la primera tarea a realizar es la configuración de pines de entrada y salida, configuración del módulo UART, la configuración del UART por software y la configuración del TMR0 como temporizador, ver figura 3-29.

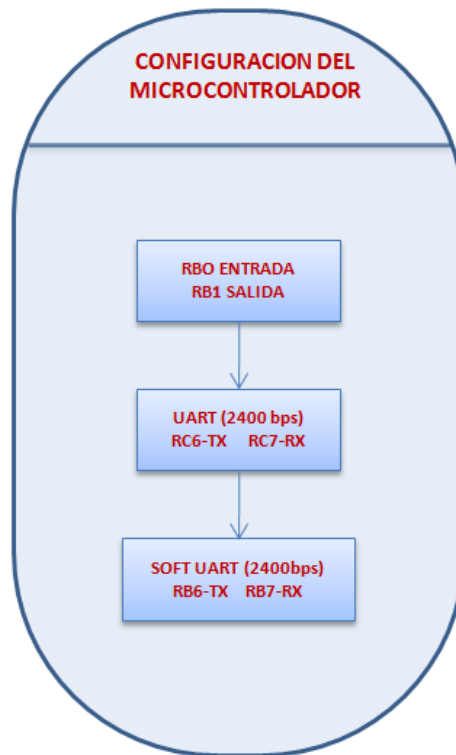


Figura 3-29: Diagrama de flujo del estado configuración del microcontrolador

El pin RB0 se configura como entrada al microcontrolador y le indica cuando el modem GSM esta encendido.

El pin RB1 es una salida, este controla el apagado y encendido de un led que cumple con las siguientes funciones:

- Indicar que el modem GSM está apagado, en este caso el led parpadea a una frecuencia 1Hz.
- Indicar el momento en que la aplicación 1 está en espera del mensaje de inicio, en este caso el parpadea a una frecuencia 5Hz durante 10 segundos y cuando termina de parpadear ya la aplicación está preparada para recibir el mensaje
- Indicar error en la conexión, cuando el modem GSM o el TE no responden cuando el microcontrolador envía algún dato, este realiza una temporización de espera si

transcurrido este tiempo no responden el microcontrolador entra en estado de error y el led parpadea a una frecuencia de 0.33Hz en donde 66,6% del ciclo el led está apagado y el resto encendido.

Unas de las características físicas de este microcontrolador es que posee un módulo UART, para el desarrollo de la aplicación se configuro a una velocidad de 2400 bps, por defecto el pin RC6 es el transmisor y el pin RC7 es el receptor.

Además de establecer comunicación serial con el modem, el microcontrolador también debe establecer comunicación serial con el TE por ello se elabora una subrutina que simula un UART, también se configuró con una velocidad de 2400bps, se asignó como transmisor el pin RB6 como receptor el pin RB7.

3.5.2 ESPERA DE ENCENDIDO DEL MODEM GSM

El encendido del modem GSM se realiza a través de un pulsador, cuando se enciende el modem GSM el pin VDD_EXT del modem se pone en alto, este valor es censado por el pin RB0 del microcontrolador. Mientras el modem GSM no se encienda el led3 de la PCB parpadea a una frecuencia de 1Hz.

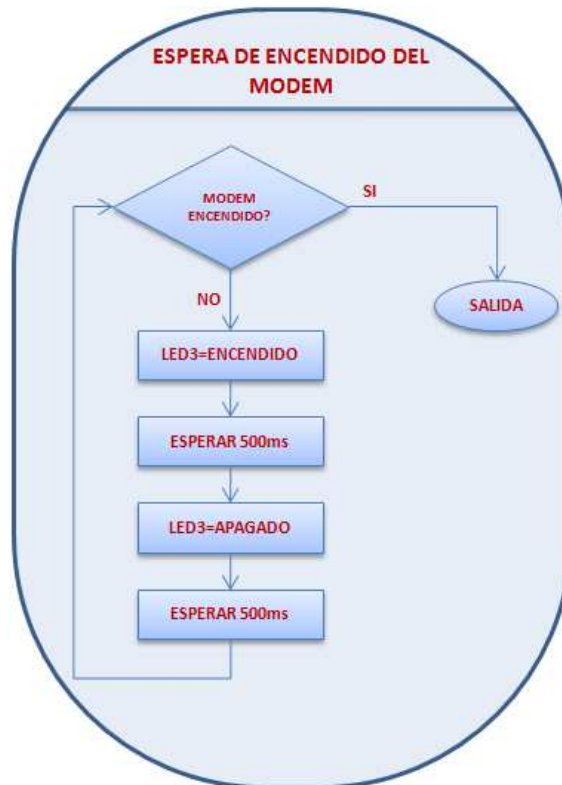


Figura 3-30: Diagrama de flujo del estado espera de encendido del modem GSM

3.5.3 CONFIGURACIÓN DE MODEM GSM

El diagrama flujo para la configuración del modem se muestra en la figura 3-31, en este se observan las subrutinas que van implicadas en el proceso. La subrutina “Enviar y Recibir Comando” se encarga de enviar un comando AT y verificar su respuesta, esta utiliza la subrutina “Recibir” que tiene como función recibir la respuesta y compararla para verificar su valides, utiliza el temporizador TMR0 para recibir los datos en primera instancia se activa el temporizador y cada vez que llega un dato se reinicia la temporización, cuando la temporización haya terminado indica que ya no hay más datos por recibir y se procede a comparar los datos capturados. La bandera FLAG se activa al terminar la temporización, pero también se utiliza para indicar la valides de dato, el algoritmo tiene un contador que indica las veces en que el dato recibido no es válido si ese número es igual a 8, el algoritmo entra en el estado de error y se bloquea el led3 parpadea a una frecuencia de 0.33Hz. Esto indica que el modem no está respondiendo.

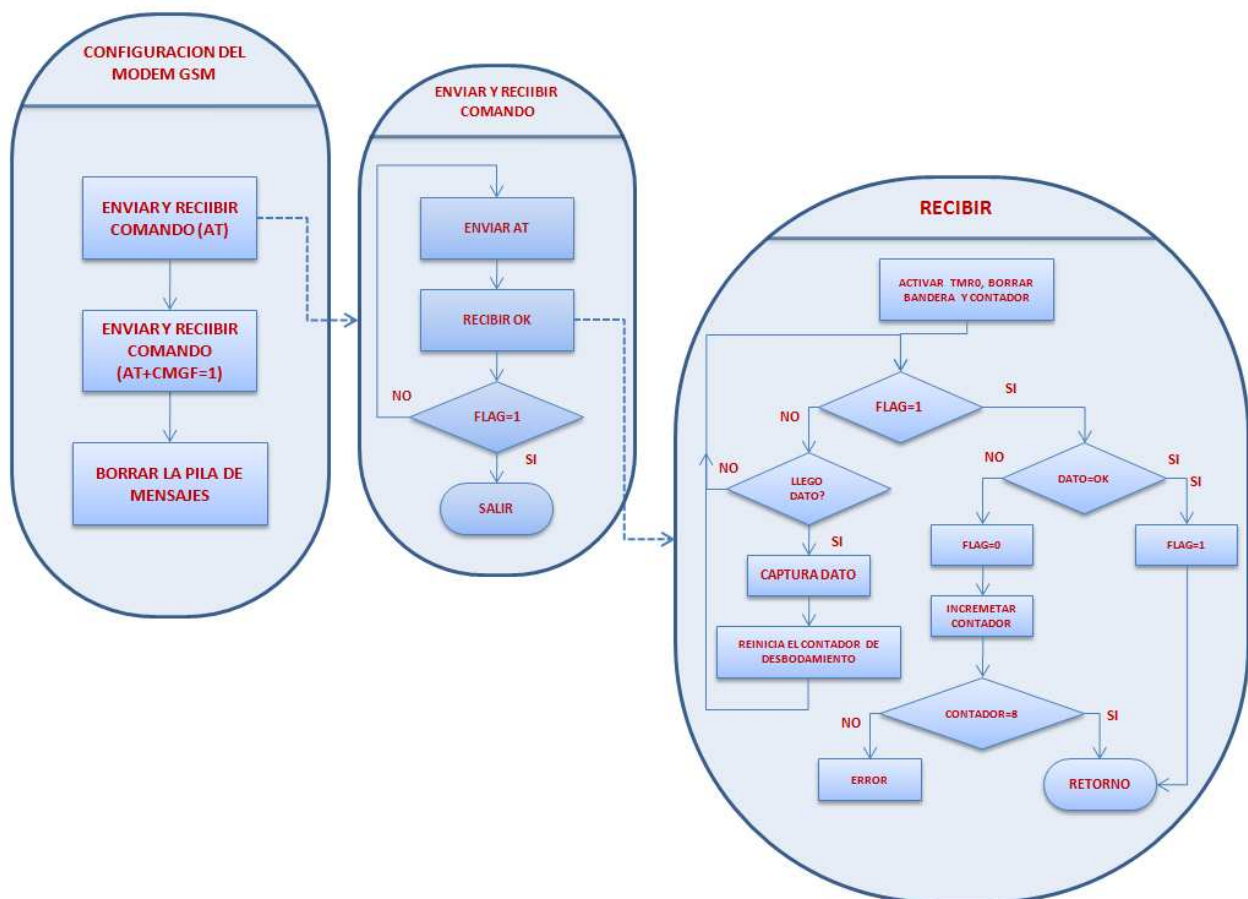


Figura 3-31: Diagrama de flujo del estado configuración del modem GSM.

3.5.4 ESTADO DE ESPERA DEL MENSAJE DE INICIO

En este estado el microcontrolador se encuentra en espera del mensaje de inicio por parte de la aplicación 2, este mensaje debe contener la siguiente estructura “INICIO,XXX,YYYYYYYYYYY”, la palabra INICIO en letra mayúscula seguido de tres caracteres (xxx) que son la cantidad en minutos con la que se desean recibir los mensajes de datos y por ultimo 10 caracteres (yyyyyyyyyyy) que corresponden a los números a quienes se enviaran mensajes de alarma mínimo 2 máximo 3, todo va separado por comas. Ver figura 3-32.

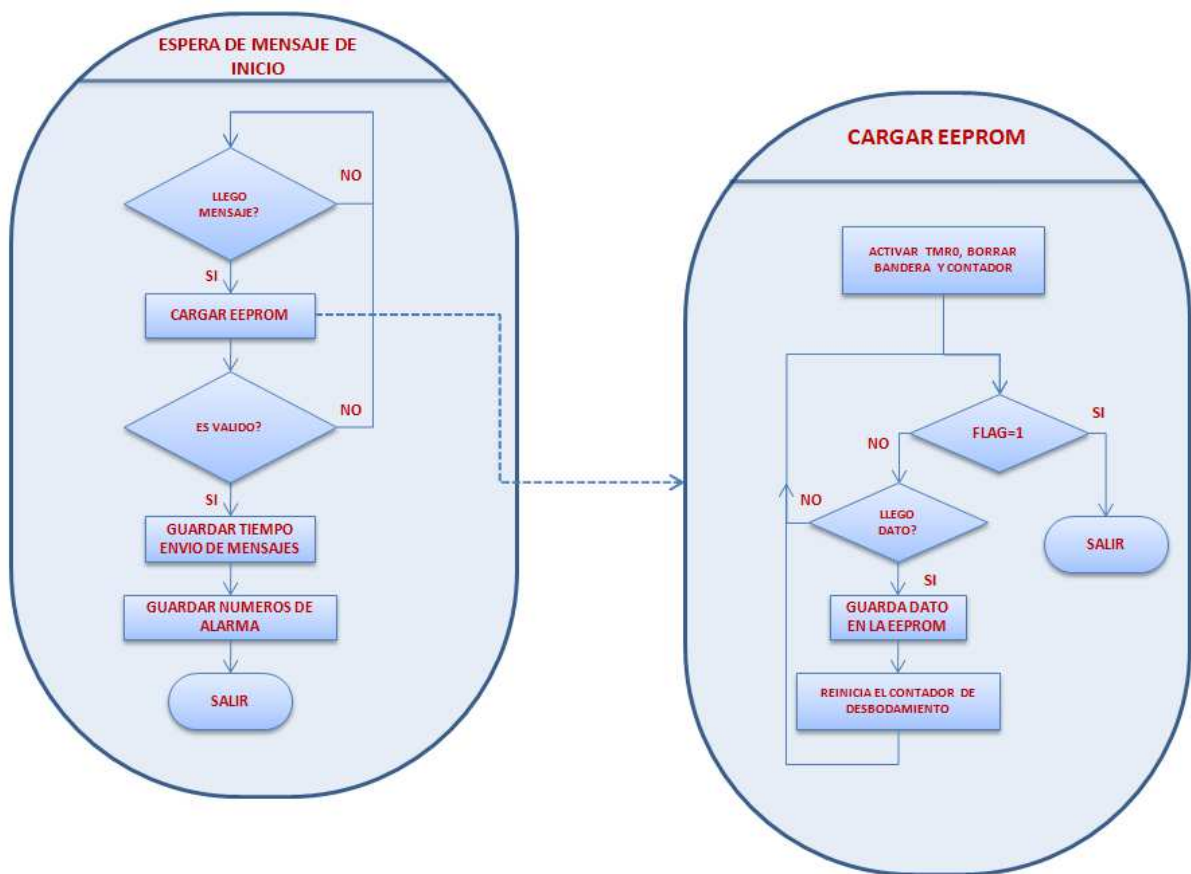


Figura 3-32: Diagrama de flujo del estado espera del mensaje de inicio.

3.5.5 HABILITACIÓN DE INTERRUPTOR POR TMR1

En este estado se realiza la configuración de los registros INTCON, PIR1, PIE1 y T1CON para realizar una temporización con el TMR1, al terminar la temporización este se desborda y se produce la interrupción del flujo del programa, ver figura 3-33.



Figura 3-33: Diagrama de flujo del estado habilitación de interrupción de TMR1

3.5.6 CICLO NORMAL DEL PROGRAMA

En este estado el programa espera por una cadena de caracteres de TE puede ser de dos tipos:

- Notificación de alarma: El microcontrolador recibe una cadena de caracteres con la siguiente estructura “ALARMAxxxxxx” con la palabra ALARMA en mayúscula seguida de caracteres que describen el evento (xxxxxx), el número de caracteres debe ser tal que no sobrepase la capacidad del mensaje por ejemplo, “ALARMA TA1-001”, cabe resaltar que la alarma no es un tipo de información solicitada por lo que el microcontrolador le da prioridad a la detección de una alarma.
- Datos de las variables: Para el correcto funcionamiento del sistema se implementó una comunicación basada en semáforos, donde la cadena “XOFF” indica que el microcontrolador no puede recibir una alarma por parte del TE a partir ese momento, ya que se dispone a realizar una petición de dato para enviar un mensaje de texto con la información enviada por parte del TE.

La cadena de caracteres “XONN” le indica al terminal de datos que está disponible nuevamente para recibir cualquier alarma.

En el diagrama de flujo de la figura 3-34 se muestra el ciclo normal del programa, después de pasar por los estados anteriores el programa entra en este ciclo repetitivo donde la única forma de salir es entrar en un estado de error. El primer estado es la activación del temporizador TMR1, la temporización depende del tiempo de envío de mensajes este valor se obtiene en el mensaje de inicio, este valor se convierte a un dato numérico y le indica al TMR1 hasta donde realizará la temporización, luego entra en un ciclo que donde la prioridad es esperar una notificación de alarma por parte de TE, por otra parte también se compara si se cumplió la temporización del TMR1, cuando esto sucede se activa la bandera FLAG2 y el programa realiza una petición de información de la variables al TE, en este momento se hace uso del semáforo.

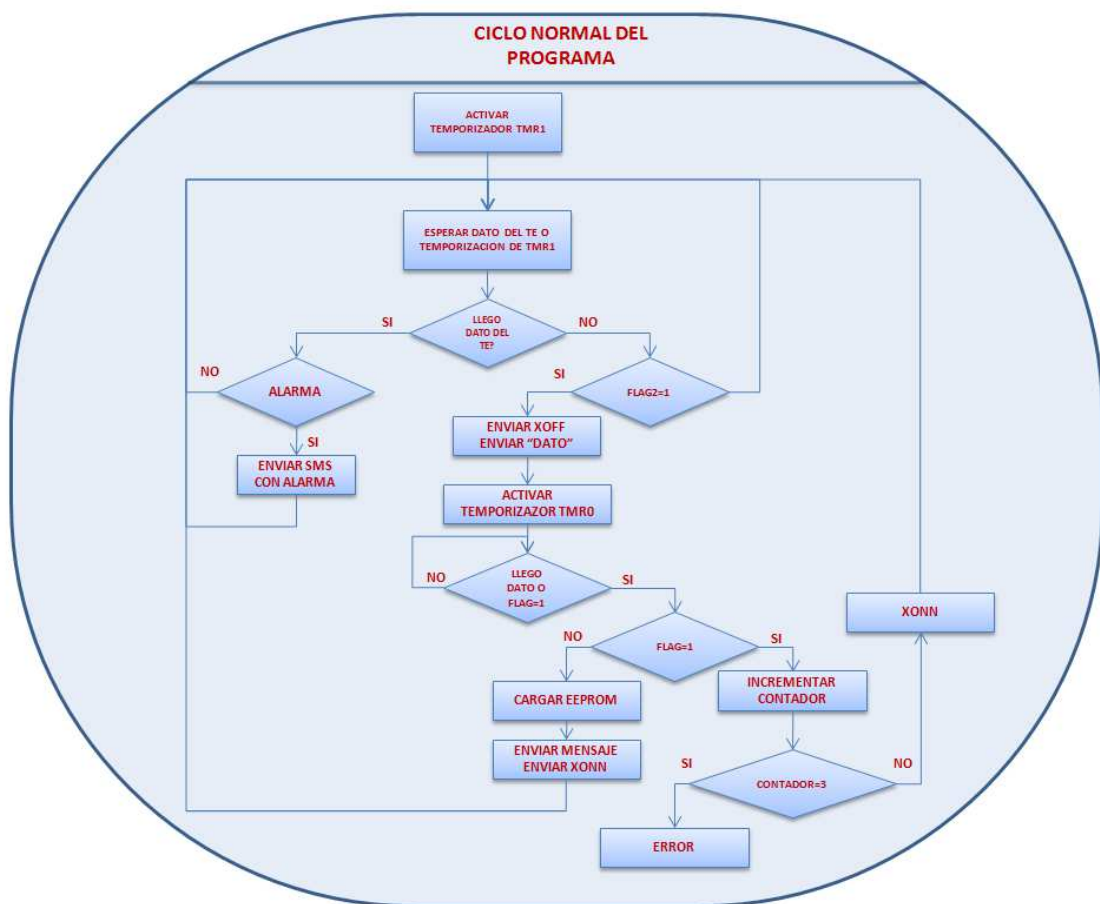


Figura 3-34: Diagrama de flujo del estado ciclo normal del programa.

3.5.7 INTERRUPCIÓN

El algoritmo de programación de la aplicación SISDAQ utiliza los temporizadores TMR0 y TMR1 cada uno con su respectiva función como se observa en la figura 3-35.

El TMR0 tiene como función indicar que ha finalizado la recepción de datos una vez la bandera FLAG este activada.

El TMR1 tiene como función la captura de las mediciones de las variables que se encuentran en el terminal de datos, empaquetarlas en un mensaje de texto y enviarlas a la aplicación IVIGED. Esta temporización puede ser indicada por el usuario y dará la resolución con la que se perciben los datos en la aplicación IVIGED. Es decir entre menor sea este tiempo, más rápido se enviaran actualizaciones de los datos, si por el contrario este tiempo es mayor las actualizaciones demoraran más tiempo en llegar.

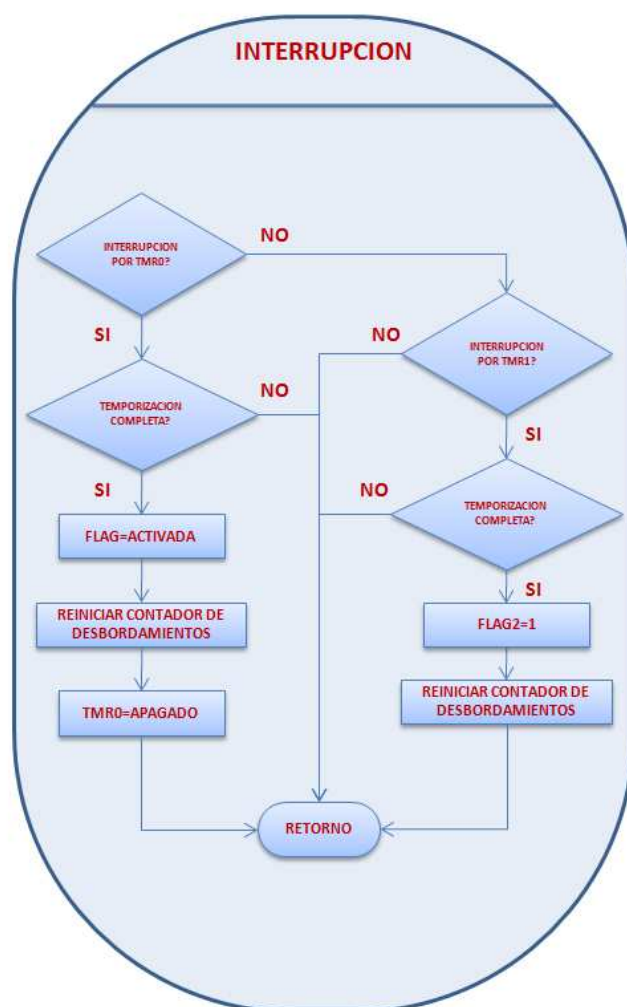


Figura 3-35: Diagrama de flujo del vector de interrupción.

3.6 DISEÑO DE LA INTERFAZ DE VISUALIZACIÓN Y GESTIONAMIENTO DE DATOS A PETICIONES IVIGED

La interfaz de visualización y gestionamiento de datos a peticiones es la aplicación servidora del Sistema de monitoreo remoto, en ella se encuentra la interfaz de usuario en donde se captura la información necesaria para realizar la configuración inicial del sistema, recibe información de las variables y notifica sobre el estado de las mismas.

El diagrama de bloque que representa el funcionamiento de la interfaz es el mostrado en la figura 3-36. Se debe entender que cada bloque representa una función dentro de la interfaz, es decir, se pueden realizar dos o más funciones en paralelo.

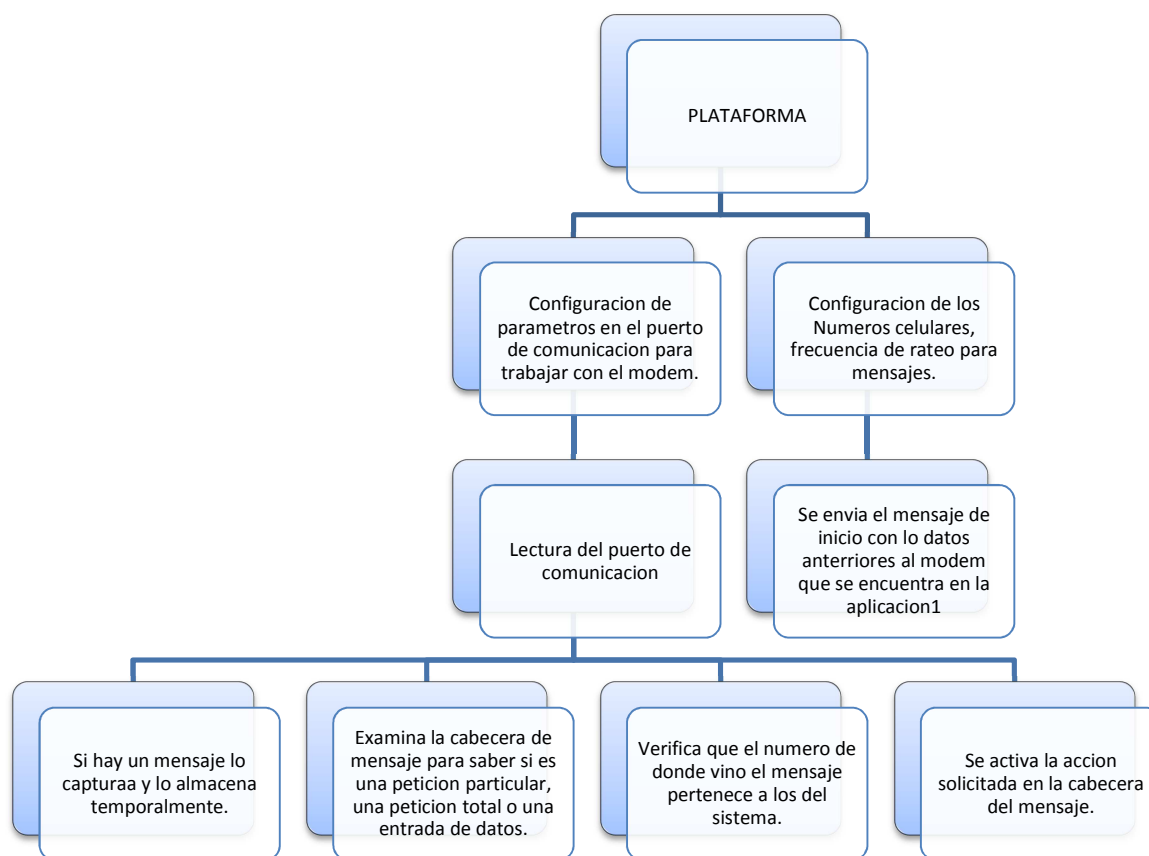


Figura 3-36: Diagrama de estado de la aplicación IVIGED

PLATAFORMA

Para el primer bloque, se encuentra la PLATAFORMA que es el bloque en general del proyecto. El cual se obtiene al crear un Vi en la plataforma de programación grafica LabVIEW, luego de haberla creado se comenzó con la Configuración de los parámetros en el puerto de comunicación para trabajar con el modem. Para poder entender cómo se hizo la plataforma es necesario repasar el funcionamiento de los bloques utilizados en la sección 2.11.3.

3.6.1 CONFIGURACIÓN DEL PUERTO DE COMUNICACIÓN

La configuración del puerto de comunicación se realizó con el bloque de VISA Configure Serial Port como se muestra en la figura 3-37, el cual contiene los siguientes parámetros:

- Identificación del puerto de comunicación que utiliza el modem.
- Velocidad de información que soporta el modem.
- Bits por dato que genera el modem.
- Paridad
- Error de entrada.
- Control de flujo.

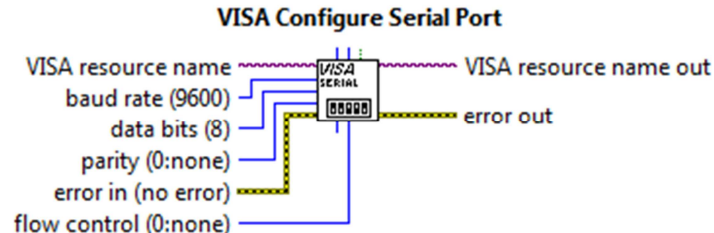


Figura 3-37: Bloque de configuración del puerto serial.

Para el proyecto se utilizó la configuración que se muestra en la figura 3-38, el bloque I/O de VISA es un identificador lógico único que sirve para comunicarse con un recurso, manteniendo una sesión en la que se puedan realizar varias operaciones, y especifica la fuente o la identificación del puerto que se utilizará. Los parámetros que están en blanco toman por defecto el valor mostrado en la figura 3-37.

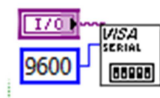


Figura 3-38: Parámetros utilizados para la configuración del puerto serial.

Luego se realizó la configuración para encontrar la manera de obtener los números celulares, frecuencia de muestreo para mensajes, número de celular de inicio que se definen como:

- 2 números de celulares para notificar acerca de alguna alarma en el proceso industrial.
- 1 número de celular del ingeniero que es el único capaz de solicitar datos del proceso industrial.
- Periodo en min para el envío de mensajes desde la aplicación SISDAQ hasta la aplicación IVIGED.
- 1 número de celular de inicio que es el asociado al modem conectado a la aplicación SISDAQ. Esto para dar inicio al monitoreo con los números de configuración mencionados anteriormente.

3.6.2 INTERFAZ DE USUARIO PARA LA CONFIGURACIÓN INICIAL

IVIGED le brinda a la aplicación SISDAQ la información de los números de celular de alarma, el número del ingeniero, la frecuencia de envío de mensajes, el puerto de comunicación con el que se trabajará y el número de celular de inicio por medio de la interfaz mostrada en la figura 3-39.



Figura 3-39: Interfaz de usuario para la configuración inicial.

Estos datos son almacenados como cadenas de caracteres y pasan a ser los valores del array de modelación para la configuración inicial como lo muestra la figura 3-40.

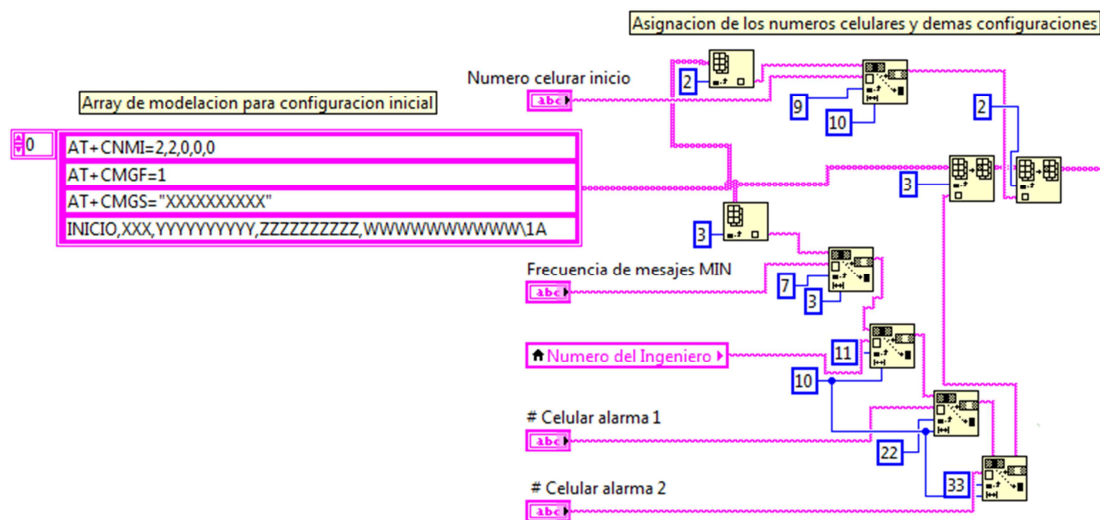


Figura 3-40: Código de asignación de números celulares al array de modelación.

• Array de modelación para configuración inicial

Se creó un array que modelara los comandos AT necesarios para la configuración inicial de la aplicación SISDAQ. En este se encuentra la secuencia de comandos que debe recibir el modem para enviar un mensaje de texto. Los comandos AT necesarios:

Posición 0 del arreglo:

AT+CNMI=2,2,0,0,0.

Posición 1 del arreglo:

AT+CMGF=1.

Posición 2 del arreglo:

AT+CMGS="XXXXXXXXXX".

Posición 3 del arreglo:

La ultima cadena de caracteres es el modelo de inicio del programa para la aplicación SISDAQ, la estructura comienza por la palabra INICIO, seguido tres caracteres que indican la frecuencia de mensajes de los datos, es decir, un muestreo periódico de los datos que están en el proceso industrial, luego los números de teléfonos móviles, el del Ingeniero y los números a quienes se les notificará acerca de alguna anomalía en el proceso industrial. La cadena de caracteres resultante es:

INICIO,[Freq. Men],[Cel. Ing],[Cel. Alarma 1],[Cel. Alarma 2]\1A. Véase la figura 3-40.

Todos estos son modelados en un arreglo de 4 posiciones. Que van desde la posición 0 que contiene “AT+CNMI=2,2,0,0,0” hasta la posición 3 que contiene “INICIO,XXX,YYYYYYYYYY,ZZZZZZZZZZ,WWWWWWWWW\1A”. Los caracteres \1A le indican al modem el fin de línea del texto a enviar por SMS.

En cada uno de los bloques, se trabajó con el array para ir modificando el arreglo respecto a los valores necesarios, por ejemplo en la figura 3-41, se tomó la cadena de caracteres de la posición 3 para ir modificando secuencialmente los valores. Inicialmente se asignó la frecuencia de mensajes, esto lo hace tomando el carácter de la posición 7 de la cadena de caracteres tres, y 3 valores siguientes a él, incluyéndolo. A ese valor le asigna la frecuencia establecida por el usuario con la interfaz mostrada en la figura 3-41. Así también para el celular de alarma 1 y el celular de alarma 2.

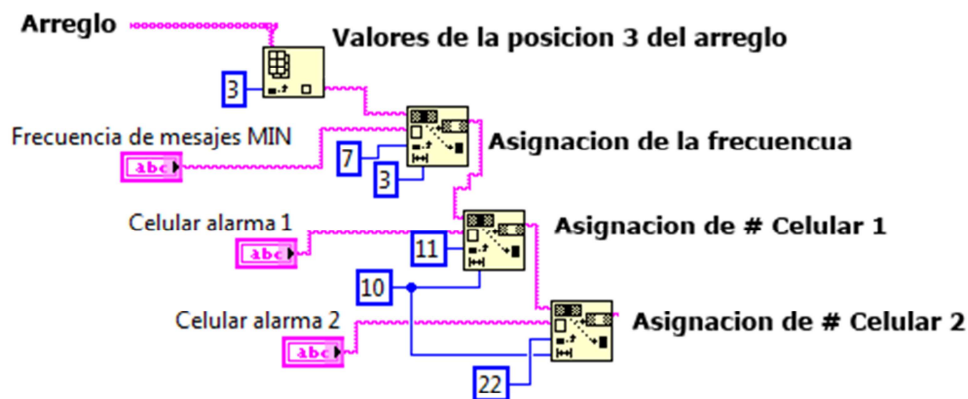


Figura 3-41: Asignación de los números celulares

Así mismo para el número de celular de inicio. Véase la figura 3-42.

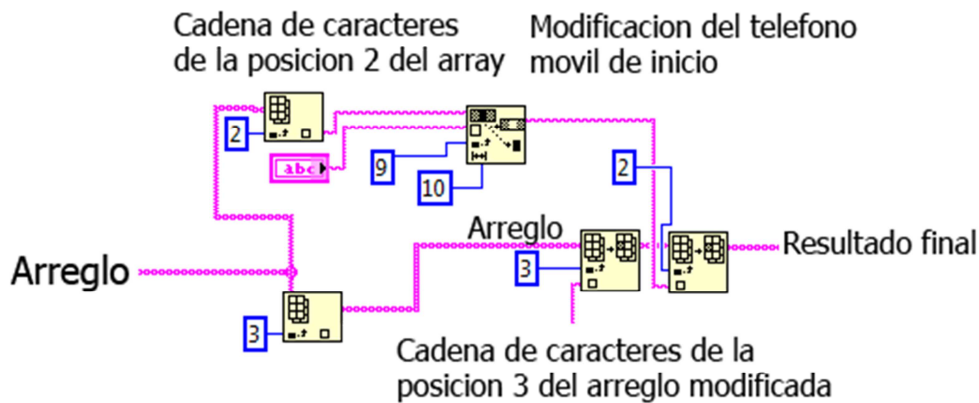


Figura 3-42: Asignación del número de celular de inicio.

La interfaz de modificación del campo “Numero de celular de inicio” se muestra en la figura 3-43. El cual contiene una lista de valores para la frecuencia de mensajes en minutos que van desde 005 hasta 060 minutos. Se hace referencia que la frecuencia de mensajes es el tiempo de envío que hay entre mensajes, esto define la frecuencia de monitoreo de las variables. Las celdas para ingresar la información de los números de teléfonos móviles para el Ingeniero y números telefónicos de alarma son libres y deben contener 10 dígitos.



Figura 3-43: Interfaz de usuario para la configuración inicial del sistema.

En la interfaz se implementaron:

- 3 Indicadores booleanos que notifican acerca del tipo de mensaje recibido, si es un dato que viene del modem de la aplicación SISDAQ, o si es un mensaje de petición

particular de alguna variable o una petición total de los datos por parte del ingeniero.

- 2 Botones, INICIO para el envío del mensaje de configuración inicial. STOP para la detención de la escritura, lectura del puerto de comunicación y el programa.
- Un indicador para la hora actual y un control para ingresar la hora diaria de envío de los todos los datos. Esto como medida de notificación diaria de todos los datos a la hora estipulada.
- Una pantalla negra que muestra la respuesta del modem cuando se envía el mensaje de inicio.
- De las otras celdas se explica en la configuración de los números celulares, frecuencia de muestreo para mensajes, numero de celular de inicio.
- Y la última es la identificación del puerto de comunicación. Que muestra una lista desplegable con todos los puertos de comunicación hábiles y sin uso.

3.6.3 ESCRITURA EN EL PUERTO DE COMUNICACIÓN

Después de tener la información necesaria para la configuración inicial se pasó a escribir en el puerto de comunicación. Para eso se implementaron dos estructuras Case que condicionan la operación de algún código en su interior, uno para el botón de inicio, el cual da la orden de escribir en el puerto y el otro case para la condición de No error en el puerto, esto como medida de protección y sincronización en el puerto de comunicación, un bloque de configuración del puerto de comunicación indicándole la identificación del puerto y la velocidad de información, luego un ciclo FOR para digitar secuencialmente las líneas de comandos establecidas en el array de configuración inicial, y un SubVi que escribiera en el puerto y leyera las respuestas del modem ante los comandos digitados. El indicador Read String monitorea lo que hay en el puerto de comunicación. Se utiliza un Shift Register para guardar los valores por cada iteración e ir concatenándolos con los nuevos valores obtenidos en la lectura del puerto. Por último se cierra el puerto de comunicación. El código es el mostrado en la figura 3-44.

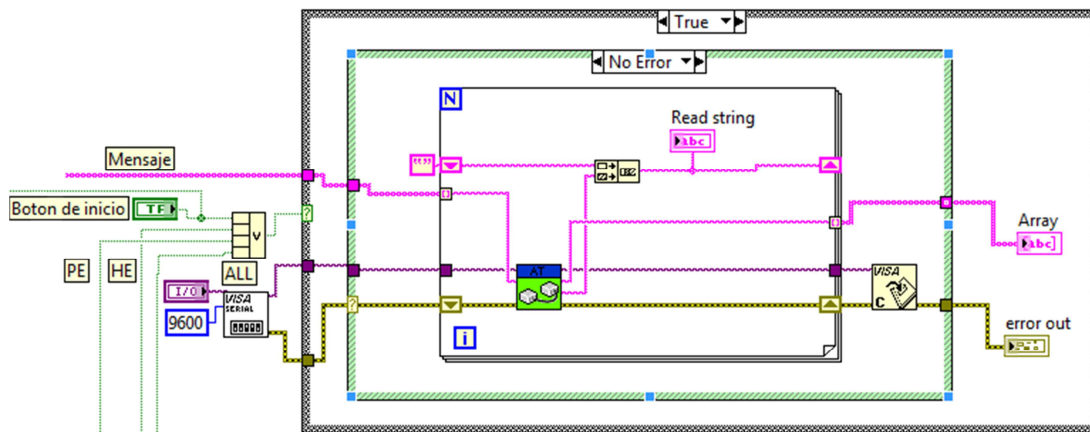


Figura 3-44: Bloque de escritura en el puerto ante cualquier petición de datos o configuración inicial.

El código anterior es el único utilizado para escribir en el puerto de comunicación. Es decir, por el indicador “Mensaje” viaja toda la información que será escrita en el puerto de comunicación. El bloque de color azul con verde es el SubVi del código utilizado para escribir y leer en el puerto de comunicación. La figura 3-45 muestra el código de este SubVi.

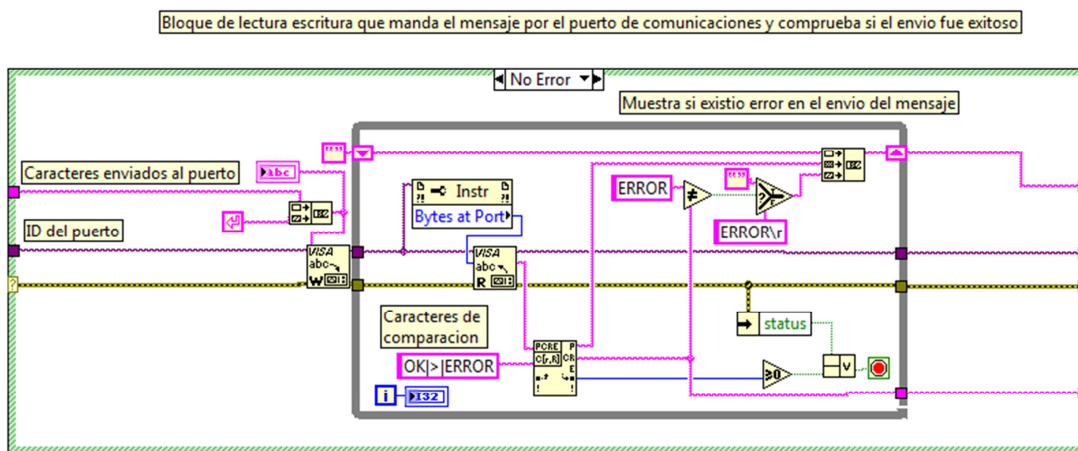


Figura 3-45: Código de escritura y lectura en el puerto de comunicación para el envío de mensajes.

Está compuesto de la siguiente forma:

- Un Case que iniciará siempre y cuando no exista algún tipo de error de escritura o lectura en el puerto de comunicación.
- La primera etapa escribe en el puerto los comandos que vienen del arreglo de vectores. Es decir, coloca la primera posición del vector seguido de retorno de carro, luego la siguiente posición seguido del retorno de carro. Así sucesivamente hasta completar todas las posiciones.

- Se realiza un escaneo continuo del puerto, para cerciorarse de la respuesta correcta por parte del modem. Las condiciones para evaluar son las siguientes:
 - OK: Ejecución correcta del comando enviado.
 - ERROR: Error en la ejecución del comando.
 - >: Respuesta del modem para dar inicio al contenido del mensaje a enviar.

El indicador “Mensaje” en la figura 3-45 hace referencia a la cadena de caracteres que lleva la configuración que se le dará a la aplicación SISDAQ.

3.6.4 LECTURA DEL PUERTO DE COMUNICACIÓN

Luego de enviar el mensaje de configuración inicial, IVIGED queda esperando los datos que vienen del SISDAQ. Entonces, se fue necesario implementar una etapa de lectura en el puerto de comunicación, para eso se implementó el bloque de lectura en el puerto de comunicación, en donde se configuró la velocidad de información del puerto que en este caso fue de 9600 bits por segundo, el ID del puerto de comunicación y la cantidad de bytes que se van a leer en cada trama, luego de esto se pasa a la lectura del puerto de comunicaciones. Véase la figura 3-46.

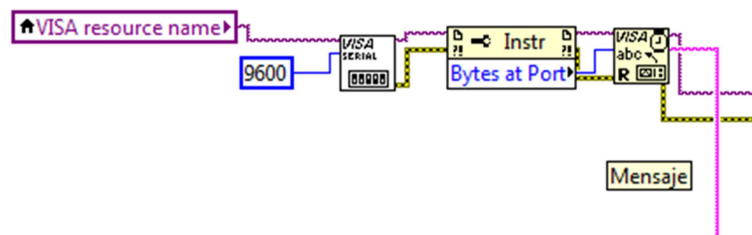


Figura 3-46: Configuración del puerto serial para su lectura.

El indicador “Mensaje” hace referencia a los caracteres leídos en el puerto de comunicación. Dentro de los mensajes recibidos se crearon plantillas para reconocer el tipo de mensaje que es recibido. Las plantillas contienen caracteres y símbolos que indican el tipo de mensaje como lo son:

- #ALL: Para notificación de todos los datos que se encuentran en la base de datos. La condición se evalúa con la cabecera de la trama, cerciorándose que exista el string #ALL.
- #PE: Para notificación de algún dato en particular que se encuentre en la base de datos. La condición se evalúa con la cabecera de la trama, cerciorándose que exista el string #PE.
- #IN: En esta condición se evalúa si en la cadena de caracteres enviada en el mensaje existe los caracteres #IN, al igual que se hizo en los casos anteriores.

Entonces para reconocer el tipo de mensaje se programaron secuencialmente los bloques de LabVIEW que tiene como función buscar un carácter dentro de una cadena y retornar la cadena de caracteres que está después del carácter buscado, el bloque que toma los valores de cabecera del mensaje para reconocer el tipo de mensaje comparándolo con los caracteres mencionados anteriormente. Entonces se realizó la captura del mensaje para pasar a la evaluación de las 4 condiciones como lo muestra la figura 3-47, las condiciones son:

- Si el número recibido pertenece al número permitido de iniciar el sistema.
- Si el mensaje recibido es una entrada de datos.
- Si el mensaje recibido es una petición de algún dato en particular.
- Si el mensaje recibido es una petición de todos los datos.

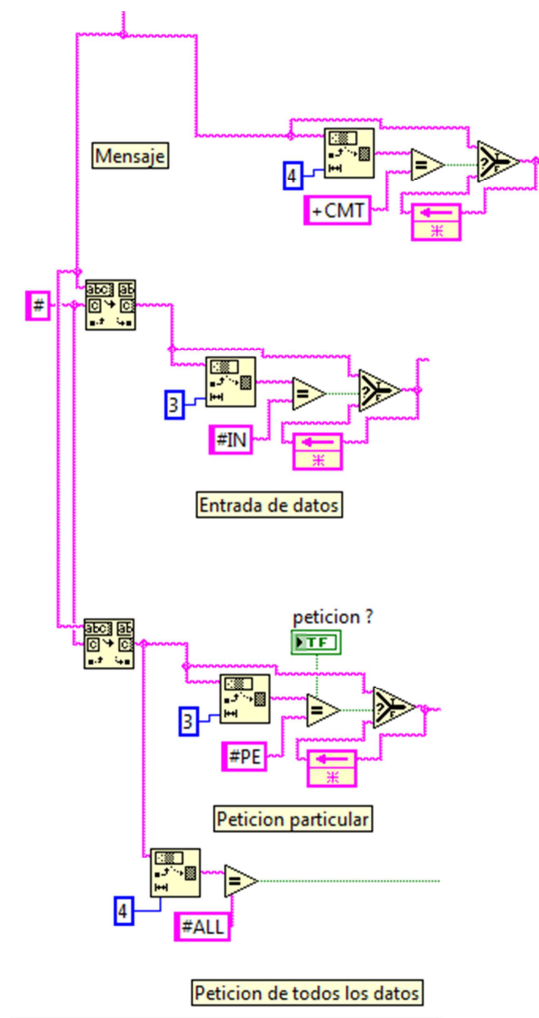


Figura 3-47: Código para la evaluación del tipo de mensaje recibido.

Para la primera condición el sistema se tiene en cuenta la configuración inicial que se hace en paralelo a la lectura del puerto de comunicación. Con el botón de inicio se configuran los usuarios del sistema que son: 2 números de celular para notificaciones de alarmas, un número de celular para el Ingeniero encargado del proceso, un número de celular de inicio del sistema. En el mensaje recibido se evalúan los primeros cuatro caracteres que indican la llegada de un mensaje (+CMT), luego pasa a evaluarse en toda la cadena de caracteres la búsqueda del carácter (“), para luego pasar a la obtención de los 10 caracteres siguientes, que es el número de celular remitente del mensaje. Se realiza la comparación, si es el número de celular del Ingeniero (Introducida con anterioridad) o si es el número de celular de inicio. Ver figura 3-48.

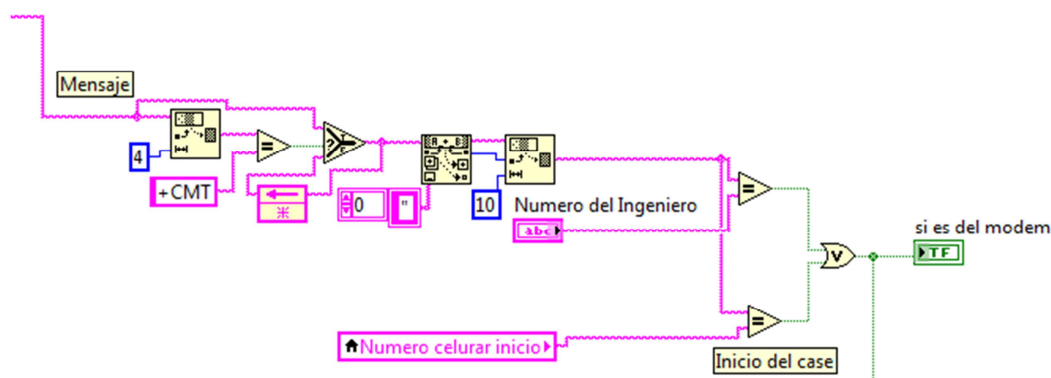


Figura 3-48: Código de identificación del remitente del SMS

Todo esto con el fin de dar inicio a un Case en donde se almacenan los datos de entrada. Véase la figura 3-49.

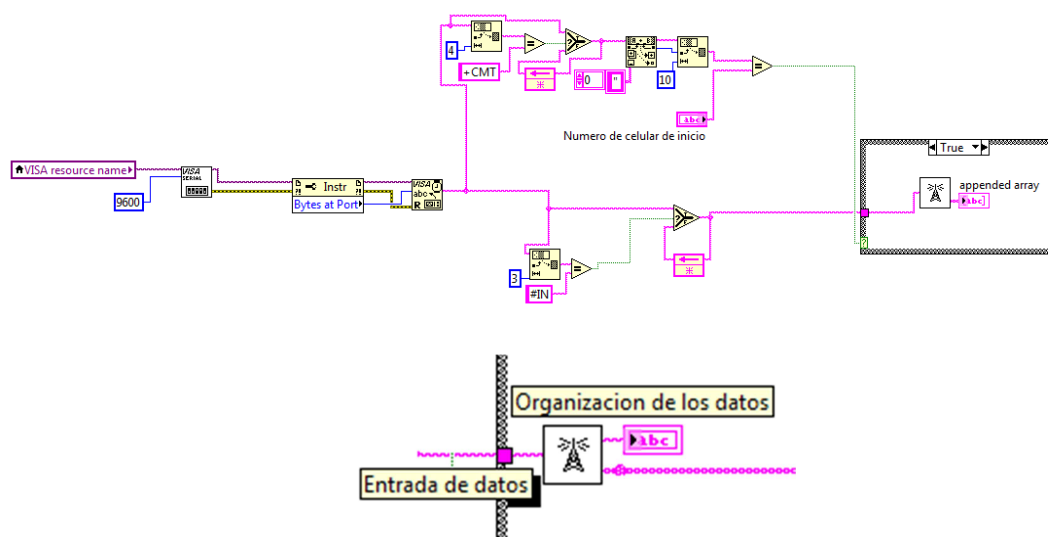


Figura 3-49: Case de habilitación para la organización de datos.

En la figura anterior el selector en complemento con el feedback node mantienen el dato en la salida del selector si la condición de entrada es falsa. Si pasa a verdadera se actualiza con el nuevo dato y se mantiene hasta nueva orden. Todos los datos vienen en una cadena de caracteres, uno tras otro, el SubVi implementado en la figura anterior, organiza los datos en un array para tener mejor dominio sobre las variables. Teniendo en cuenta que los datos vienen representados de la siguiente manera:

- #IN,T00-156,T01-453,T02-800...

En donde #IN es la indicación que es un mensaje tipo dato de entrada que viene de la aplicación SISDAQ, TXX es la posición de registro del dato, y los otros tres dígitos es el dato. En el arreglo de vectores, se organizaron los datos para un mejor manejo de los mismos. Esto se realizó tomando 7 caracteres por cada trama, y se guardan en el arreglo de vectores. Los datos que vienen del mensaje, vienen en orden desde la aplicación SISDAQ, es decir, del primer valor hasta el último del dato. Es por esto que la organización de los datos se hizo en el mismo orden. Se tomó desde el carácter 4 hasta 7 posiciones en adelante para el primer dato, desde el carácter 12 hasta 7 posiciones en adelante y así sucesivamente para todos los datos como lo muestra la figura 3-50. Se estableció que como para un mensaje de texto corto (SMS), la cantidad máxima de caracteres es de 160 por mensaje, entonces existirán 19 variables monitoreándose, esto debido a la plantilla establecida para el monitoreo.

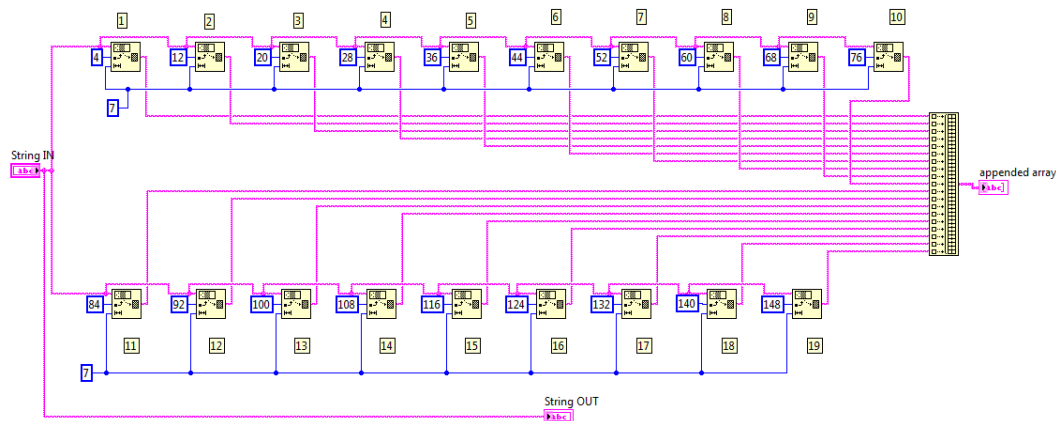


Figura 3-50: Código de organización de datos.

Hasta este punto solo se ha resuelto la solicitud para la entrada de datos e identificación del remitente del mensaje para dar inicio al caso de organización de los datos. Para las otras 2 condiciones de peticiones de datos, tanto particulares como totales, se estableció una plantilla la cual consta de los siguientes caracteres:

- #PE,TXY,TZH.

Donde X, Y, Z, H son números enteros que indican la posición del dato guardado, o dicho de otra manera el dato a interesar. Para reconocer cuales son las variables de interés por el Ingeniero, se implementó un SubVi que extrajera los números de las posiciones de los datos a interesar. El bloque que responderá a las solicitudes es el mostrado en la figura 3-51.

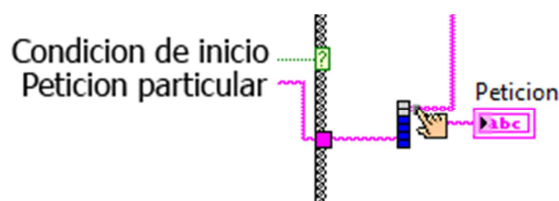


Figura 3-51: Bloque de respuestas a las peticiones.

La condición de inicio es la comparación que se hace en los bloques de la figura 3-48, para cerciorarse que el número provenga del Ingeniero o si es el número del modem asociado a la aplicación SISDAQ. El SubVi toma la cadena de caracteres que viene en el mensaje y va adquiriendo los valores de X, Y, Z, H, con el fin de reconocer los campos requeridos por el Ingeniero. Se estableció que máximo se pueden solicitar 10 datos en el mismo mensaje. El código que realiza la función de este SubVi es el mostrado en la figura 3-52.

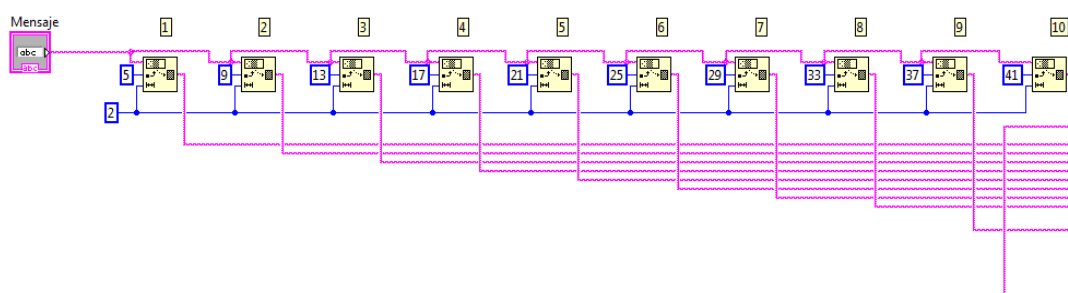


Figura 3-52: Código de extracción de información en el array que contiene los datos parte 1

Se utilizaron bloques de extracción de valores según la posición y longitud indicada, entonces como solo se requiere los valores de XY para la primera petición, se tomó desde la posición 5 (#PE,TXX), y una longitud de 2 caracteres para identificar el dato solicitado por el ingeniero. Pero pueden existir espacios en blanco que pueden ser interpretados como ceros, para eso se realizó la comparación de espacios en blanco, para no generar errores en los reportes. Al obtener los valores de X, Y, Z, H el sistema los reconoce como cadena de caracteres, es por esto que debió convertirse a valores numéricos con la ayuda del bloque String to number, esto con el fin de poder ubicar la posición en el arreglo de vectores. La estructura Case, extrae el valor del arreglo de vectores según la posición leída como lo muestra la figura 3-53.

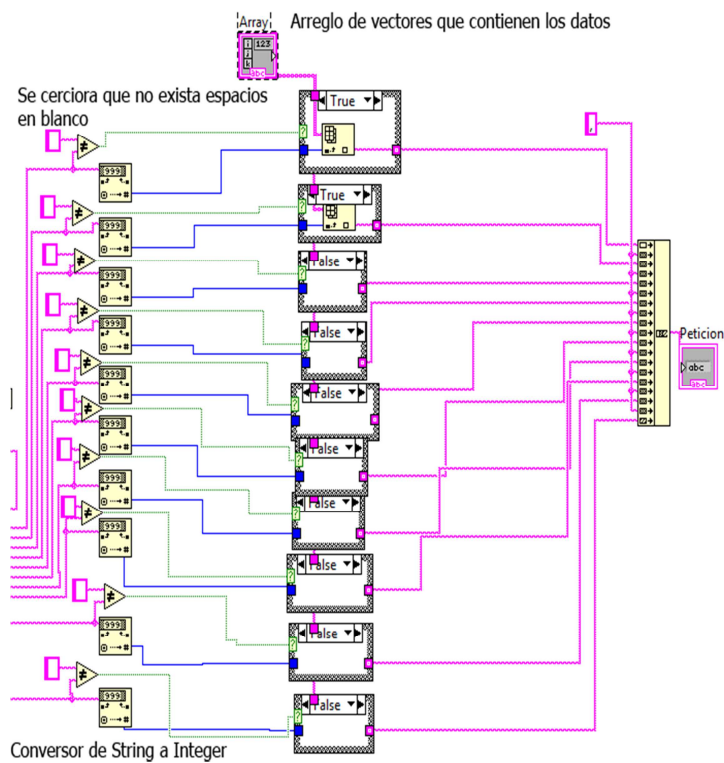


Figura 3-53: Código de extracción de información en el array que contiene los datos parte 2.

Al case de organización de datos, se le añadió el SubVi de respuesta a las peticiones con la condición de encendido como remitente el número de celular del ingeniero. Se utilizó unos indicadores por dato para visualizar las 19 variables en la interfaz y la identificación de cada dato como lo muestra la figura 3-54.



Figura 3-54: Interfaz de visualización de los datos.

El código utilizado para la visualización de los datos y respuesta a peticiones es el mostrado en la figura 3-55.

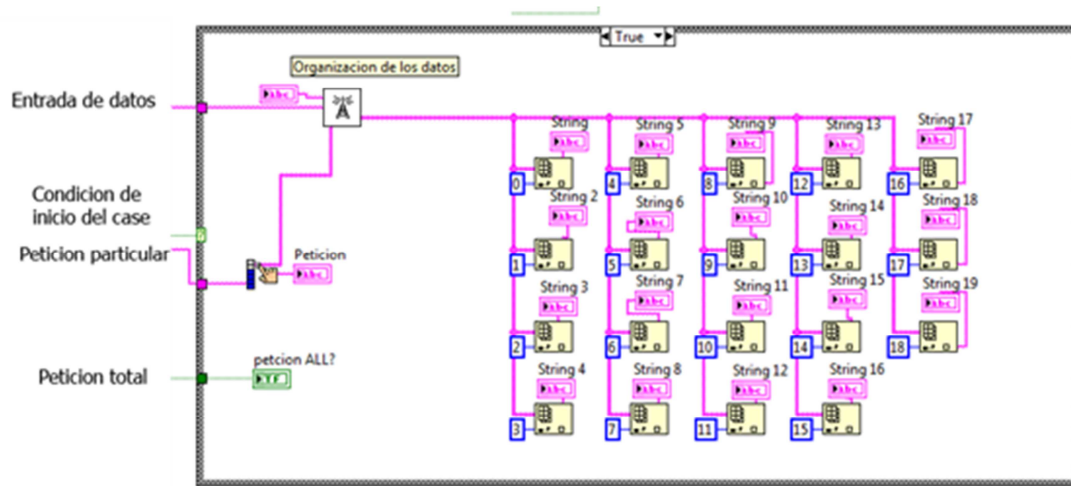


Figura 3-55: Código para la visualización de los datos y respuestas a las solicitudes.

Las condiciones de encendido de la estructura case de la figura 3-55 son 4:

- Presionando el botón de inicio de la interfaz usuario mostrado en la figura 3-43.
- La hora de envío del informe diario de los datos es igual a la hora actual. (HE)
- Si existe alguna petición en particular. (PE)
- Si existe una petición total de los datos. (ALL)

En cuanto al envío de las respuestas de las peticiones, se debió implementar la misma línea de comandos utilizadas para enviar el mensaje de configuración inicial, pero la diferencia radica en el destino del mensaje que es constante (número del ingeniero) y la información enviada. Como solo existen dos tipos de peticiones, se utilizó un selector que tienen como entrada las variables locales de las respuestas a las peticiones como lo muestra la figura 3-56.

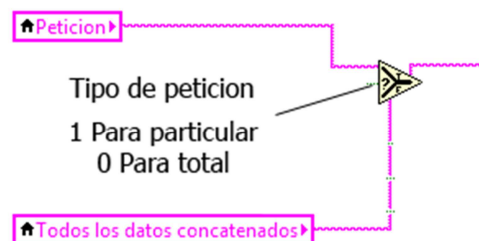


Figura 3-56: Código selector para el tipo de respuesta.

Pero en realidad no solo son dos entradas como selección del tipo de petición, existe una tercera que es implícita en el programa, y es la notificación diaria de los datos. Para esto se implementó la siguiente tabla de verdad. Véase tabla 3-4.

HORA	PETICIÓN TOTAL	PETICIÓN PARTICULAR	SALIDA
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0

Tabla 3-4: Tabla de verdad para la selección del tipo de respuesta a las solicitudes.

Como se nota en la tabla la única condición de salida verdadera es la segunda, para cuando exista alguna petición particular. Realizando suma de productos y simplificando la expresión booleana, resulto la expresión $\overline{H}PTPP$, cuya estructura booleana es la mostrada en la figura 3-57.

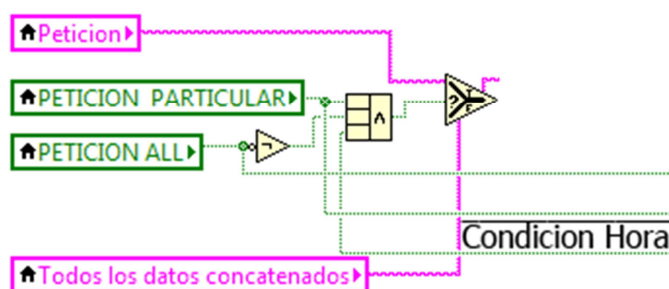


Figura 3-57: Código de selección de entrada en el selector para el tipo respuesta.

Como el reporte de los datos se hará en la última línea de comando, solo fue necesario modificarla con respecto a la cadena de valores que vienen en respuesta a la petición. Para esto se utilizó el bloque Replace Array Subset, para remplazar la cadena de valores por la cadena existente en la posición 3, el bloque Index Array para extraer la cadena de caracteres de la posición 2 y luego pasar a remplazar las 10 “T” por el número del

ingeniero, esto con ayuda del bloque Replace Substring. Y finalmente utilizar el bloque Replace Array Subset para remplazar la cadena resultante del proceso por la cadena de la posición 2 del array. El código resultando es el mostrado en la figura 3-58.

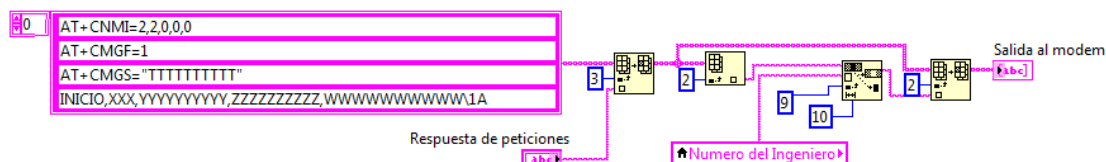


Figura 3-58: Código para asignación del número del ingeniero en la línea de comandos enviados al modem.

Para la condición de hora de envío fue solo utilizar el bloque Get Date / Time String para obtener la hora actual y compararla con la digitada por el usuario en la interfaz. Luego añadir la condición a la estructura lógica para la selección del tipo de solicitud como lo muestra la figura 3-59.

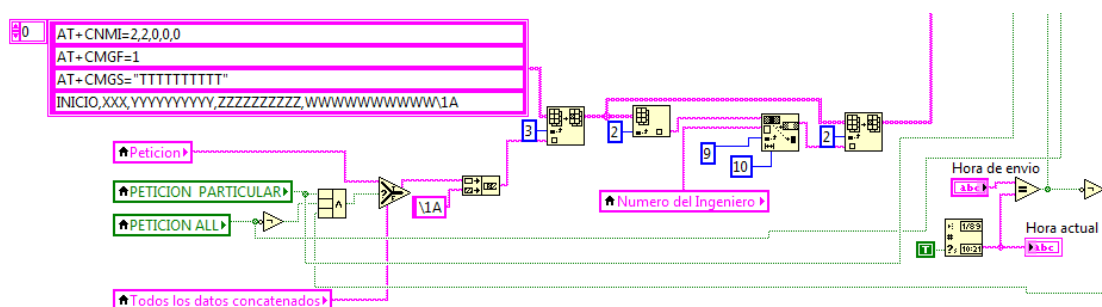


Figura 3-59: Código para las respuestas a las solicitudes y hora de envío diaria de los datos.

Los valores de las condiciones booleanas de petición particular y petición total se utilizaron para activar el case de escritura en el puerto de comunicación, además se utilizó un selector con la condición booleana del botón inicio para seleccionar la entrada de respuestas de las peticiones en la condición falsa, para que luego de haberse dado la configuración inicial, quede en espera de solicitudes por parte del ingeniero. Por último se le agregó una función que permitiera guardar las variables monitoreadas en un documento de texto, con la fecha y hora del valor mostrado como lo muestra la figura 3-60.

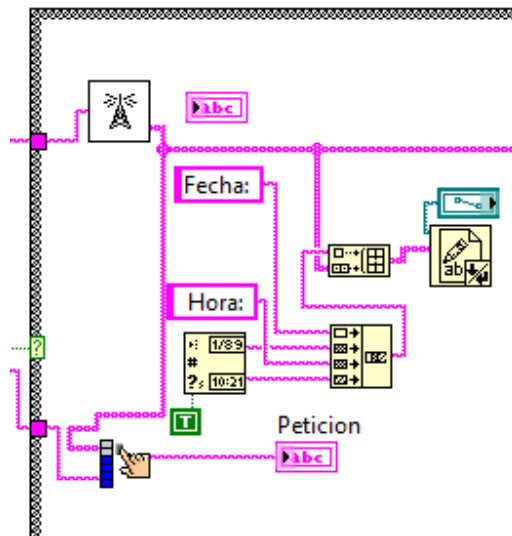


Figura 3-60: Código para guardar en un documento de texto la información de las variables.

En la interfaz de usuario se añadió un botón con la ayuda de un control en la entrada path del bloque Write to text File, esto para indicar la ubicación del documento de texto que debió ser creado con anterioridad. Véase la figura 3-61.

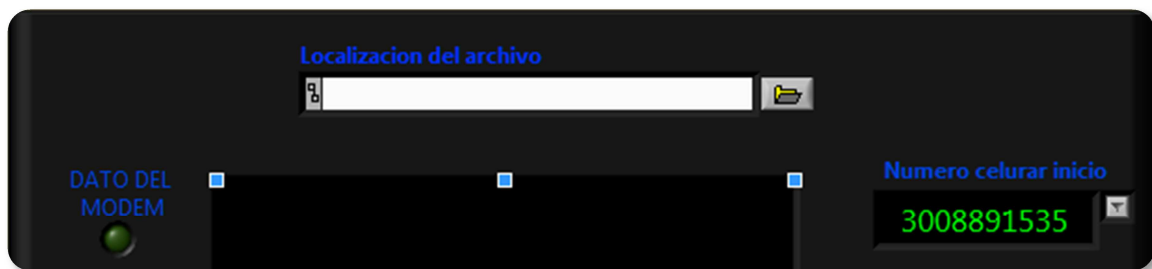


Figura 3-61: Interfaz para localización del documento de texto.

El programa completo está compuesto por dos estructuras. La primera en donde se configuran los usuarios del sistema, los números de teléfono móviles de los mismos. Se queda en espera de alguna petición particular o total, o si es la hora del envío del informe diario. Configuración, lectura y escritura del puerto de comunicación. Véase la figura 3-62.

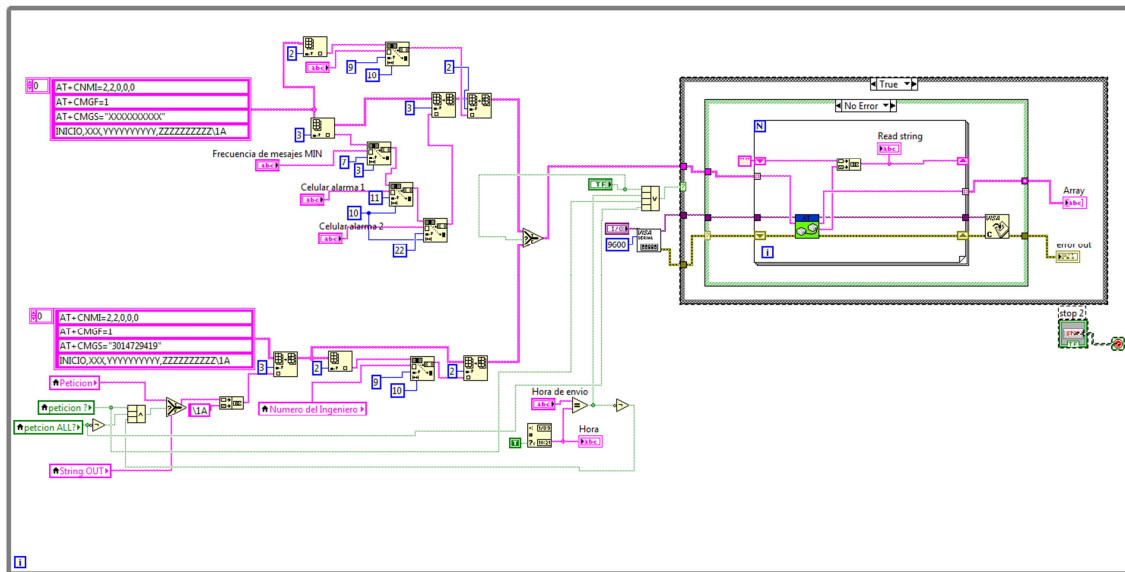


Figura 3-62: Código de configuración inicial, escritura – lectura en el puerto para peticiones y hora de envío de datos.

La segunda se encarga de leer en el puerto los mensajes recibidos e ir comparando si es un mensaje de petición particular, petición total o una entrada de datos por parte de la aplicación SISDAQ. Guarda los datos de entrada del mensaje. Responde y crea la cadena de caracteres de los valores solicitados por el ingeniero. Véase la figura 3-63.

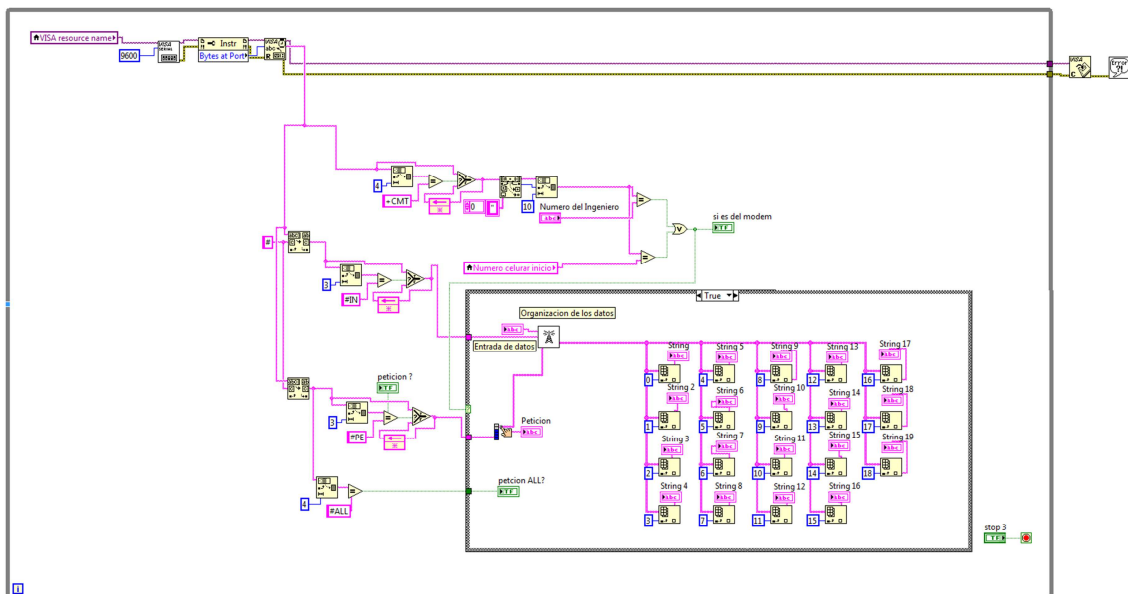


Figura 3-63: Código de identificación del remitente del mensaje, tipo de mensaje, organización de datos, peticiones y visualización de datos.

CAPÍTULO IV

4 RESULTADOS

Como producto de la investigación para solución del problema planteado se fijaron los objetivos y metas del proyecto, a través del desarrollo del proyecto obtuvieron resultados que son el producto final de la investigación, en el presente capítulo se describen cada uno de los resultados obtenidos que son:

- Prototipo de prueba para la adquisición y envío de datos.
- Interfaz de adquisición y envío de datos SISDAQ.
- Interfaz de visualización y gestionamiento de datos IVIGED.
- Simulación de planta embotelladora de alcohol isopropílico.
- Interfaz de visualización de planta embotellador de alcohol isopropílico.

4.1 PROTOTIPO DE PRUEBAS PARA CONFIGURACIÓN DE MODEM GSM

Para poder realizar las debidas configuraciones en el Modem GSM se realizó el prototipo de pruebas para adquisición y envío de datos a través de mensajes de texto, a continuación se describen las especificaciones del dispositivo.

4.1.1 ESPECIFICACIONES DEL PROTOTIPO DE PRUEBAS PARA CONFIGURACIÓN DE MODEM GSM

El prototipo de prueba para la configuración del modem GSM está compuesto por dos placas de circuitos superpuestas, consta de las siguientes especificaciones:

- Dimensiones, 6 cm de alto, 8 cm ancho, 10 cm de largo.
- Interfaz serial RS-232 con conector DB9 macho para comunicación con un terminal de datos.
- Conector de 19 pines hembra para conexión de modem GSM.
- 1 Socket para SIMCARD.
- Bornera de alimentación para una entrada de voltaje de entre 9 y 25 voltios DC.
- 1 Pulsador encendido.

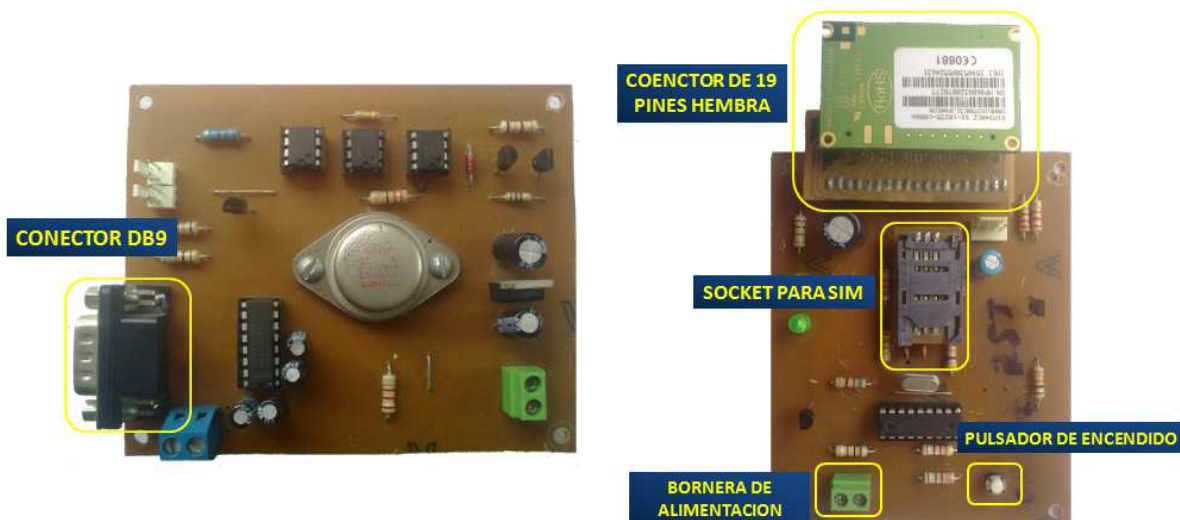


Figura 4-1: Prototipo de pruebas para configuración de modem GSM

4.1.2 INTERFAZ DE ADQUISICIÓN Y ENVÍO DE DATOS SISDAQ

La aplicación SISDAQ es un módulo que administra la comunicación entre dos interfaces seriales, la interfaz serial PC-microcontrolador y modem GSM-microcontrolador. Ver figura 4-2. Se encarga de las siguientes funciones:

- Obtener información del terminal de datos.
- Obtener información de configuración inicial de la aplicación IVIGED.
- Enviar datos de las variables suministradas por un terminal de datos a la aplicación IVIGED por medio de SMS.
- Administrar la comunicación entre terminal de datos y microcontrolador.
- Administrar la comunicación entre modem GSM y microcontrolador.

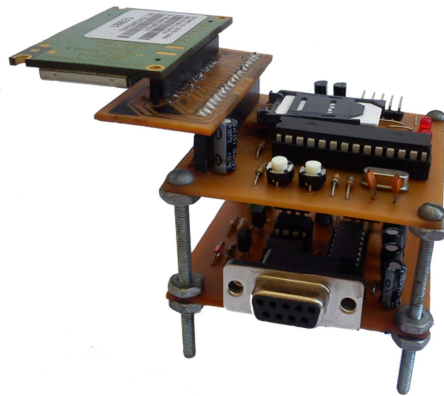


Figura 4-2: Modulo SISDAQ

4.1.3 ESPECIFICACIONES DEL MODULO SISDAQ

El módulo SISDAQ está compuesto por dos placas de circuitos superpuestas, ver figura 4-33, consta de las siguientes especificaciones:

- Dimensiones, 6 cm de alto, 5,8 cm ancho, 5,8 cm largo.
- Interfaz serial RS-232 con conector DB9 hembra para serial con un terminal de datos.
- Conector de 19 pines hembra para conexión de modem GSM.
- 1 Socket para SIMCARD.
- Bornera de alimentación para una entrada de voltaje de entre 9 y 25 voltios DC.
- 1 Pulsador encendido.
- 1 Pulsador para reset.

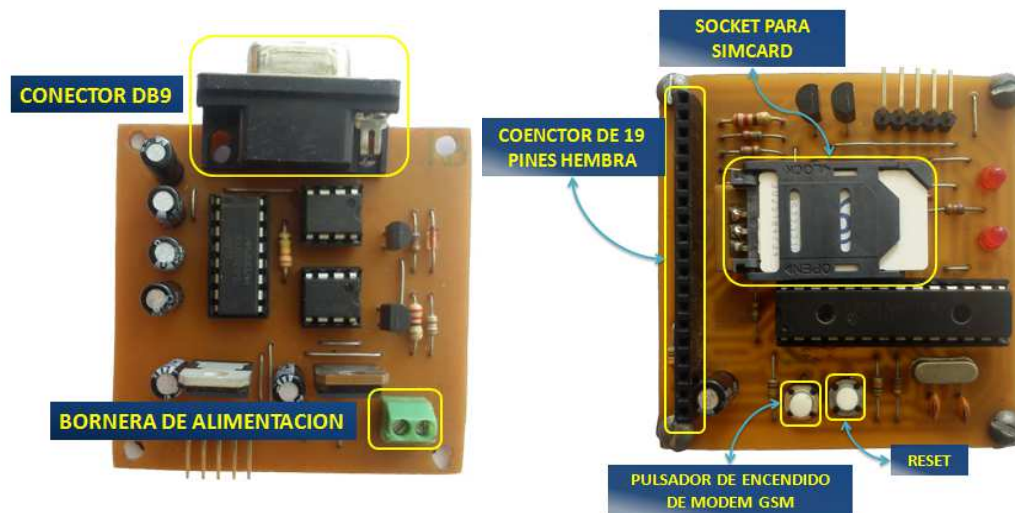


Figura 4-3: Especificaciones del módulo SISDAQ

4.2 VI DE SIMULACIÓN DE PLANTA EMBOTELLADORA DE ALCOHOL ISOPROPILICO.

Para la evaluación del proyecto se desarrolló una aplicación en LABVIEW que consiste en simulación de un terminal de datos que controla una planta embotelladora de alcohol isopropilico, el proceso consiste en la mezcla de dos líquidos contenidos en tanques de almacenamiento, cuando la mezcla esta lista pasa a un reactor donde se obtienen el alcohol isopropilico por condensación y los residuos de la mezcla.

Todas las variables implicadas en el proceso pueden ser cambiadas manualmente como el nivel, la temperatura y presión en los tanques, mezcladora y reactor, también la velocidad con la que se llenan las botellas y la cantidad acumulada.

Para la ejecución de la simulación es necesario indicar el puerto serial donde está conectado el módulo SISDAQ, colocar los datos de las variables y correr la aplicación. Ver figura 4-4.

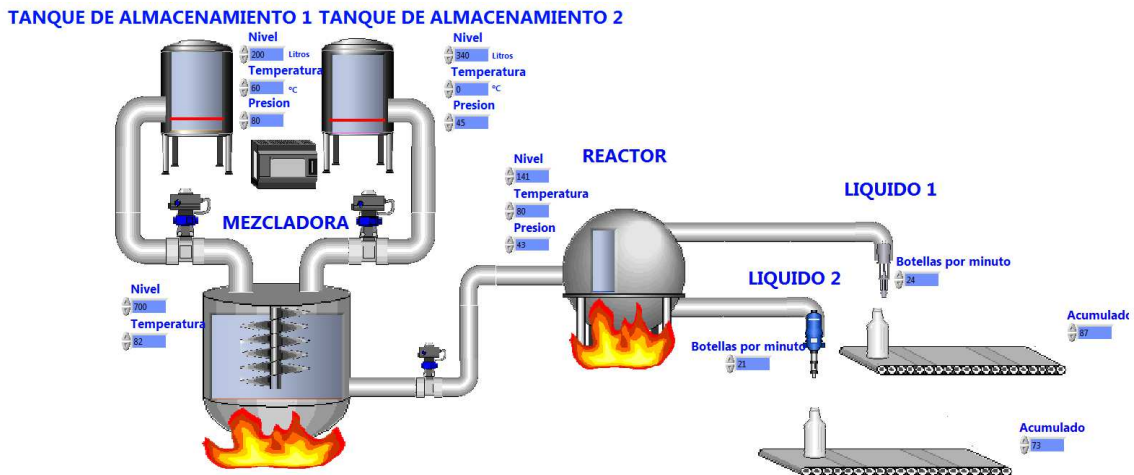


Figura 4-4: Vi de simulación de planta embotelladora de alcohol isopropílico.

4.3 INTERFAZ DE CONFIGURACIÓN INICIAL DEL SISTEMA, VISUALIZACIÓN Y GESTIONAMIENTO DE DATOS A PETICIONES IVIGED

La interfaz de configuración inicial del sistema, visualización y gestionamiento de datos a peticiones, es la aplicación servidora que configura a SISDAQ, visualiza los datos que son monitoreados por el SISDAQ y responde a solicitudes o peticiones de datos.

La interfaz está compuesta por dos bloques, uno para la configuración de los parámetros y otro para visualización de datos como lo muestra la figura 4-5.

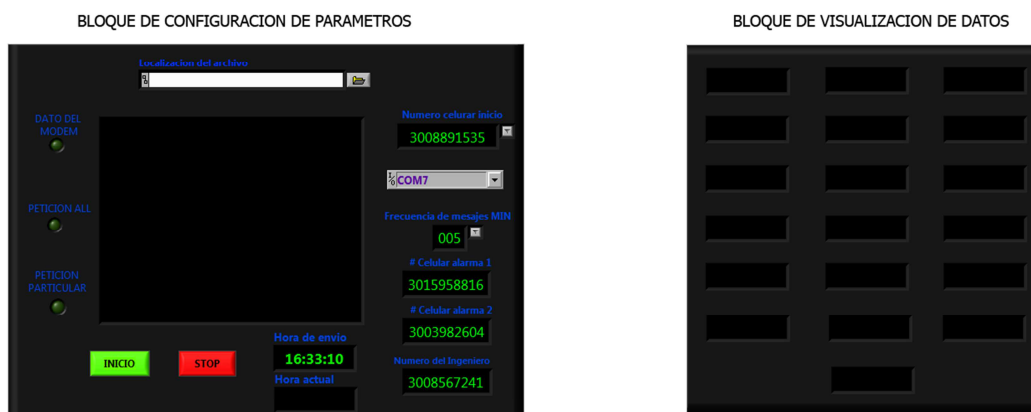


Figura 4-5: Interfaz de usuario para la configuración inicial del sistema, visualización y gestionamiento de los datos.

4.3.1 BLOQUE DE CONFIGURACIÓN DE PARAMETROS

La interfaz que se muestra en la figura 4-6 es la interfaz configuración de parámetros y está compuesta por:

- 3 leds indicadores para definir el tipo de petición o si es un mensaje de entrada de dato.
- 2 Botones para control del inicio del programa y del fin del programa.
- Una celda para indicar la hora actual.
- Una celda para digitar la hora de envío de las notificaciones diarias de las variables monitoreadas.
- Una celda para digitar el número de teléfono celular del ingeniero.
- Dos celdas para digitar los números de teléfonos celulares a los cuales les llegarán avisos de alarmas del proceso industrial.
- Un menú desplegable de las frecuencias de monitoreo de las variables, que van desde 005 minutos hasta 060 minutos.
- Un menú desplegable de los puertos de comunicaciones que están hábiles y sin uso.
- Un menú modificable de los números de teléfonos celulares asociados al modem que está conectado a la aplicación SISDAQ.
- Un botón de búsqueda del documento de texto que llevara el registro guardado de las variables del proceso en formato texto.
- Una pantalla que muestra las respuestas del modem por el puerto de comunicación.



Figura 4-6: Interfaz de usuario para la configuración inicial.

4.3.2 BLOQUE DE VISUALIZACIÓN DE DATOS

El bloque de visualización es el mostrado en la figura 4-7 y está compuesta por 19 campos que muestran la identificación de cada variable, más su valor. Cada campo mostrará 7 caracteres que tienen la forma TXX-YYY, en donde XX es la posición del dato y YYY es el valor del dato. En la figura 4-7 se muestran los campos con la estructura diseñada para mostrar los valores de las variables, los caracteres XX, ZZ, UU y VV son los que indican la posición del dato, van desde 00 hasta 18 como se ilustra en la figura. Los caracteres YYY, RRR, CCC y HHH, son los que indican el valor de la variable y van desde 000 hasta 999, ver figura 4-7.

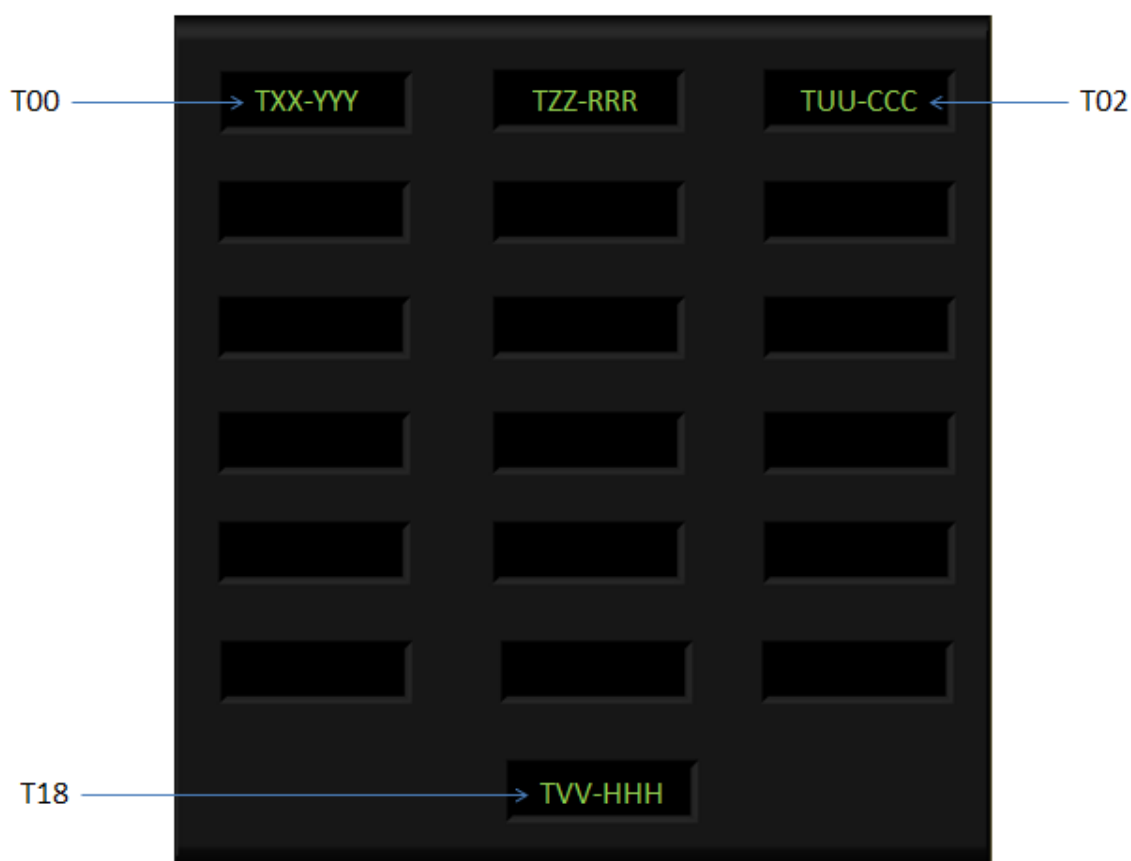


Figura 4-7: Interfaz de usuario para la visualización de los datos.

CAPÍTULO V

5 EVALUACIÓN DEL SISTEMA

Para medir el grado en que se desarrollaron los objetivos propuestos, en el presente capítulo se realiza la evaluación del funcionamiento del sistema total y en cada una de sus partes en diferentes tiempos de ejecución del proyecto, con el objeto de detectar problemas y dificultades para así desarrollar soluciones.

5.1 EVALUACIÓN DEL PROTOTIPO PARA CONFIGURACIÓN DE MODEM GSM

En el siguiente apartado se presenta el desarrollo de pruebas experimentales realizadas con el modem GSM SIM300CZ de SIMCOM, con el objeto de determinar los comandos AT necesarios para la configuración del modem GSM y el envío de SMS, y la estabilidad del comportamiento de los circuitos.

5.1.1 COMUNICACIÓN CON HYPERTERMINAL

Para el envío de comandos AT se utilizó el Hyperterminal¹ del sistema operativo de Windows y se probaron los siguientes comandos AT:

- Comandos de equipos de respuesta automática:
 - AT
 - AT&F
 - AT&W
 - ATE
 - AT+IPR
- Comandos del servicio de red:
 - AT+COPS
- Comandos del servicio de mensajes cortos:
 - AT+CMGF
 - AT+CMGS
 - AT+CNMI
 - AT+CMGD

Se realizó la siguiente configuración en Hyperterminal, velocidad de 9600bps, 8 bits de datos, sin paridad, un bit de parada y sin control de flujo y se encontraron siguientes inconvenientes:

- Retardo en la repuesta de modem.
- El modem no se registra en la red.
- Eco de los caracteres.
- No es posible el envío de mensajes de texto, leer mensajes recibidos y borrar la pila de mensajes almacenados.
- No es posible recibir notificaciones de envío.
- Se pierden los parámetros cargados al modem.

A continuación se listan lo problemas planteados y sus soluciones.

¹Hyperterminal: es una aplicación del sistema operativo Windows para conectar una computadora a un sistema remoto.

RETARDO EN LA REPUESTA DEL MODEM

Al momento de enviar primer comando al modem en Hyperterminal en la mayoría de la pruebas el modem no respondía, respondía correctamente en el tercer o cuarto intento ver figura 5-1.



Figura 5-1: Comando AT

Esto se debe a que el modem GSM se encuentra por defecto en modo autobauding, en este modo el modem detecta la velocidad de los datos enviados por el terminal de datos y se ajusta a ella, pero para ello es necesario que se hagan varios intentos de comunicación.

Dependiendo del tipo de aplicación esto puede ser ventajoso, pero para la aplicación deseada en el proyecto es necesario evitar este periodo de sincronización.

Por medio del comando AT+IPR es posible ver las distintas velocidades y fijar una tasa de baudios, para esta aplicación se escoge una velocidad de 2400bps, ver figura 5-2.

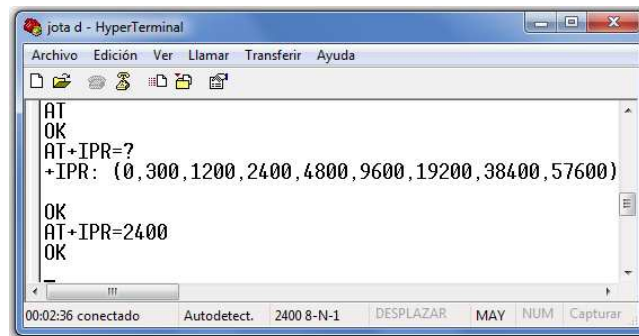
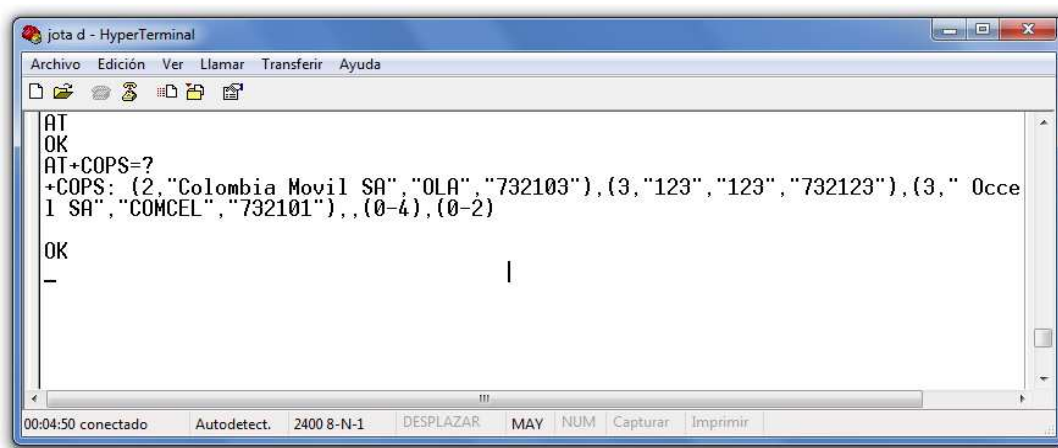


Figura 5-2: Comando IPR

EL MODEM NO SE REGISTRA EN LA RED

Para poder realizar una llamada o el envío de un mensaje de texto es necesario que el modem este registrado en la red del operador de telefonía móvil, para ello primero se hace la selección y registro de la red con el comando AT+COPS y luego con el comando AT+CREG se habilita la notificación del código de resultado no solicitado que indica el registro de esta.

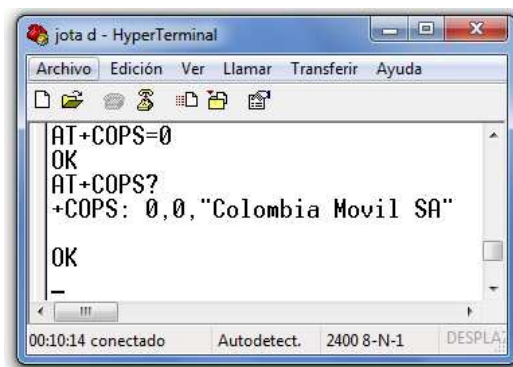
Con este comando se pueden ver las redes disponibles a la que el modem GSM se puede registrar. Ver figura 5-3.



```
jota d - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
AT
OK
AT+COPS=?
+COPS: (2,"Colombia Movil SA","01A","732103"),(3,"123","123","732123"),(3," 0cce
1 SA","COMCEL","732101"),,(0-4),(0-2)
OK
-
```

Figura 5-3: Operadores disponibles-comando COPS

Le asignamos el siguiente comando AT+COPS=0 para que se registre de forma automática, en este caso a la red de TIGO, ver figura 5-4.



```
jota d - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
AT+COPS=0
OK
AT+COPS=?
+COPS: 0,0,"Colombia Movil SA"
OK
-
```

Figura 5-4: Registro en la red-Comando COPS

ECO DE LOS CARACTERES

Para el desarrollo del proyecto se considera un inconveniente el eco de los caracteres por parte del modem ya que es una información innecesaria que microcontrolador debe procesar, por ello se desactivo el eco de caracteres, como se muestra en la figura 5-5.



Figura 5-5: Comando ate

NO ES POSIBLE EL ENVÍO DE MENSAJES DE TEXTO, NOTIFICACIÓN DE MENSAJES RECIBIDOS Y BORRAR LA PILA DE MENSAJES ALMACENADOS

Este problema se muestra en la figura 5-6.

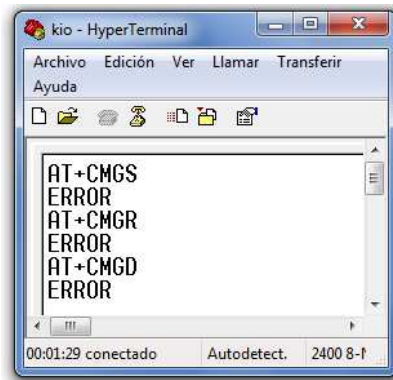


Figura 5-6: Error en comandos para SMS

Esto se debe a que todavía no se ha configurado el formato de texto del modem, este soporta dos formatos el PDU que es basado en números y el de texto que utiliza caracteres de texto, para habilita el formato texto AT+CMGF=1. Ver figura 5-7.



Figura 5-7: Comando CMGF

En la siguiente imagen se muestra el envío de un mensaje.

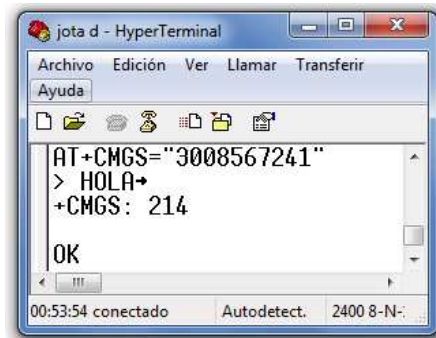


Figura 5-8: Comando CMGS

El código de resultado no solicitado +CMGS aparece entre 1 y 2 segundos después de enviado el mensaje como confirmación de que se realizó exitosamente el envío del mensaje.

NO ES POSIBLE RECIBIR NOTIFICACIONES DE ENVÍO

Al llegar un SMS el modem por defecto no notifica la llegada de este, es necesario habilitar el código de resultado necesario para ello y esto se realiza por medio de comando AT+CNMI, ver figura 5-9.

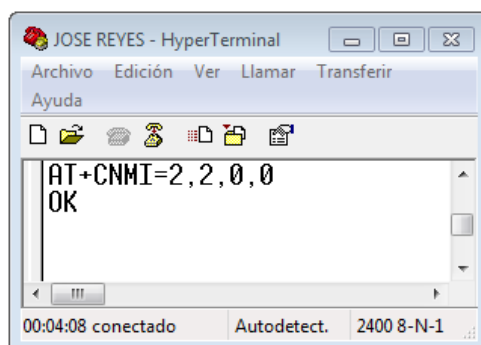


Figura 5-9: Comando CNMI

Con el comando AT+CNMI=2,2,0,0 se configura el modem GSM para que reciba mensajes de texto y los envíe directamente al terminal de datos y no los almacene en la SIMCARD así no se llena la memoria de la SIM, que tiene espacio para 15 SMS.

En la siguiente figura se muestra la notificación de un mensaje recibido en el TE, se observa el código de resultado no solicitado +CMT, el número que envió el mensaje la fecha, la hora y el contenido del mensaje.

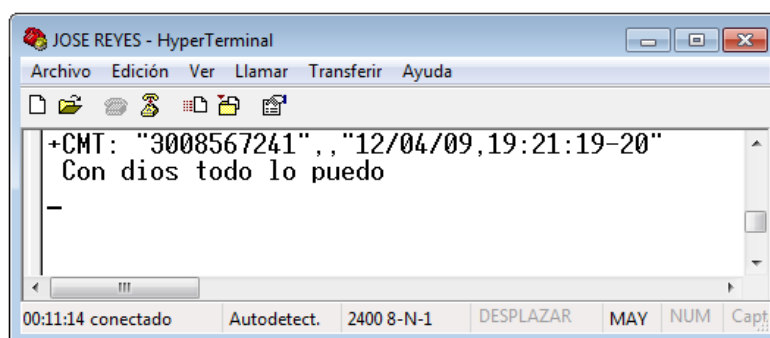


Figura 5-10: Mensaje recibido

SE PIERDEN LOS PARÁMETROS CARGADOS AL MODEM

Se observó que después de apagar el modem todas las configuraciones realizadas anteriormente se perdían por lo que en necesario guardar un perfil de usuario por medio del comando AT&W. Ver figura 5-11.

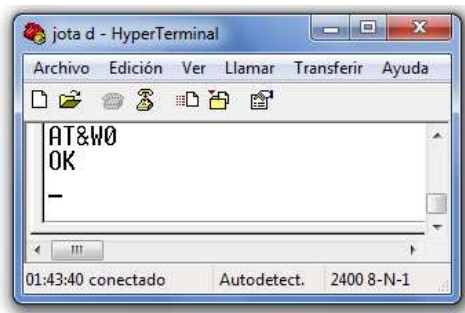


Figura 5-11: Comando AT&W

Al reiniciar el modem todas las configuraciones quedaron guardadas

5.1.2 PRUEBAS ELECTRICAS

Con el objeto de optimizar el diseño de la aplicación final se realizaron pruebas eléctricas relacionadas con la eficiencia de los reguladores de voltaje y la corriente máxima consumida en la carga.

ANÁLISIS DE LOS REGULADORES DE VOLTAJE

En la tabla 5-1 se registraron los valores de voltaje de salida de los reguladores de voltaje ante variaciones del voltaje de entrada.

Voltaje de entrada	Voltaje de salida del regulador de 5 voltios DC	Voltaje de salida del regulador de 4 voltios DC
5 V DC	4,4 V DC	3,2 V DC
7,5 V DC	4,9 V DC	3,8 V DC
10 V DC	5 V DC	3,9 V DC
15 V DC	5,01 V DC	4 V DC
20 V DC	5,12 V DC	4,03 V DC
25 V DC	5,2 V DC	4,14 V DC
30 V DC	5,4 V DC	4,2 V DC
35 V DC	5,9 V DC	4,5V DC

Tabla 5-1: Respuesta de reguladores ante variaciones de voltaje en la entrada

Se puede concluir que los reguladores presentan valores de salida estables para un rango de valores de 8 a 30 V DC, para el desarrollo de la aplicación se escogió una fuente DC de 15 V.

ANÁLISIS DE MÁXIMA CORRIENTE CONSUMIDA POR LA CARGA

En la tabla 5-2 se registraron los valores de corriente medidos cuando se enviaban mensajes de texto, es en este momento cuando el modem registra mayor consumo de corriente, las mediciones se realizaron en serie con el elemento de control de fuente como se muestra en la figura 5-12.

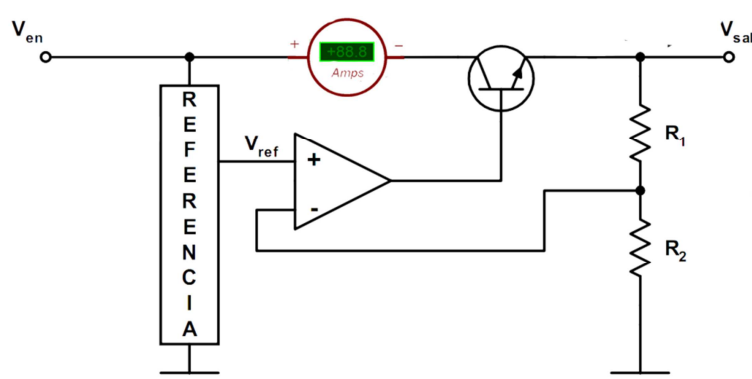


Figura 5-12: Medición de corriente del modem GSM

# Envío de Mensaje	Medición de corriente en el colector
1	10,7 mA DC
2	10,1 mA DC
3	11,2 mA DC
4	10 mA DC
5	10,5 mA DC
6	11,7 mA DC
7	9,8 mA DC
8	10,5 mA DC

Tabla 5-2: Medición de corriente consumo del modem GSM por envío de mensaje

El valor promedio de corriente en 8 mensajes enviados es $\frac{10,7+10,1+11,2+10+10,5+11,7+9,8+10,5}{8} = 10,5 \text{ mA}$, podemos deducir que el uso del transistor 2N3055 en este caso está sobredimensionado, para la aplicación final se utilizara un transistor TIP41C que garantiza una corriente 4 amperios en el colector.

5.2 EVALUACIÓN DE LA APLICACION SISDAQ

En el siguiente apartado se evalúa el funcionamiento de la aplicación SISDAQ, donde se evalúa la principal función de este módulo que es, capturar datos del TE, empaquetarlos en un mensaje de texto y enviarlos a un número telefónico celular, en la siguiente figura se muestra un diagrama de bloques de la conexión realizada para la prueba.

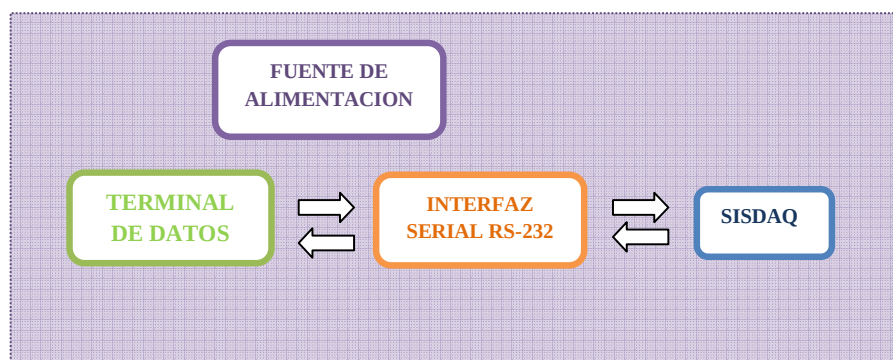


Figura 5-13: Diagrama de bloques de conexión de elementos para evaluación de módulo SISDAQ

Para efectuar las pruebas se desarrollaron 2 algoritmos, uno para simular un terminal de datos a través de LABVIEW y el otro algoritmo es el que rige el funcionamiento del módulo SISDAQ.

5.2.1 ALGORITMO PARA SIMULACIÓN DEL TERMINAL DE DATOS

En primera instancia se configura el puerto serial que se va utilizar para la comunicación, luego es necesario saber si hay datos en el buffer de recepción para recibir la petición por parte de SISDAQ, la cadena de caracteres que representa esta petición es “DATO” en mayúscula, una vez se detecta que hay cuatro caracteres en el buffer se lee el puerto serial y se verifica la validez de la cadena que llegó, si es válida se envía la cadena “#IN,T00-345,T01-23,T02-456,T03-122,T04-090,T05-133,T06-001,T07-078,T08-021,T09-312,T10-255,T12-344,T13-012,T14-556,T15-035,T16-007 HH:MM:SS”. Esta cadena está compuesta por un conjunto de nombres y valores de variables y por la hora actual a la que se capturan los datos donde HH es la hora, MM son minutos y SS los segundos

respectivamente, a continuación se muestra un diagrama de flujo del algoritmo implementado, ver figura 5-14.

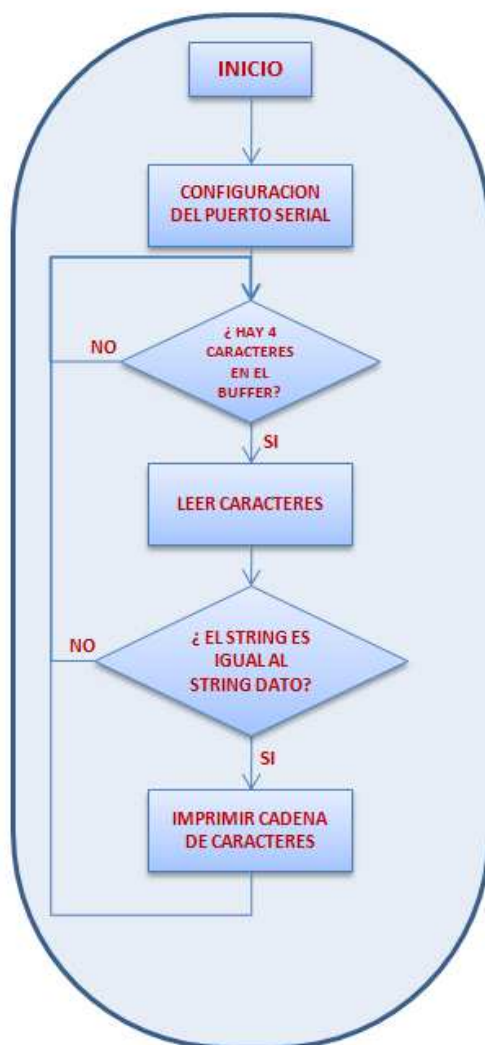


Figura 5-14: Diagrama de flujo de algoritmo simulación de TE

5.2.2 ALGORITMO PARA PRUEBA DE EVALUACIÓN DE LA APLICACIÓN SISDAQ

Al igual que en el anterior algoritmo, la primera actividad que se debe realizar es la configuración del puerto serial, luego se hace una temporización para no llenar el buffer del terminal de datos y se envía la petición de datos al TE, este debe responder con la cadena de caracteres descrita anteriormente y se almacena en la memoria EEPROM del microcontrolador, en esta etapa solo resta verificar la validez de los datos, para ello se pregunta por la cabecera #IN, si el dato es válido se envía un mensaje de texto donde el contenido es la información arrojada por él TE y si no es válido se espera por la llegada de un nuevo dato y se realizan los pasos anteriores, en la figura 5-15 se muestra el algoritmo descrito.

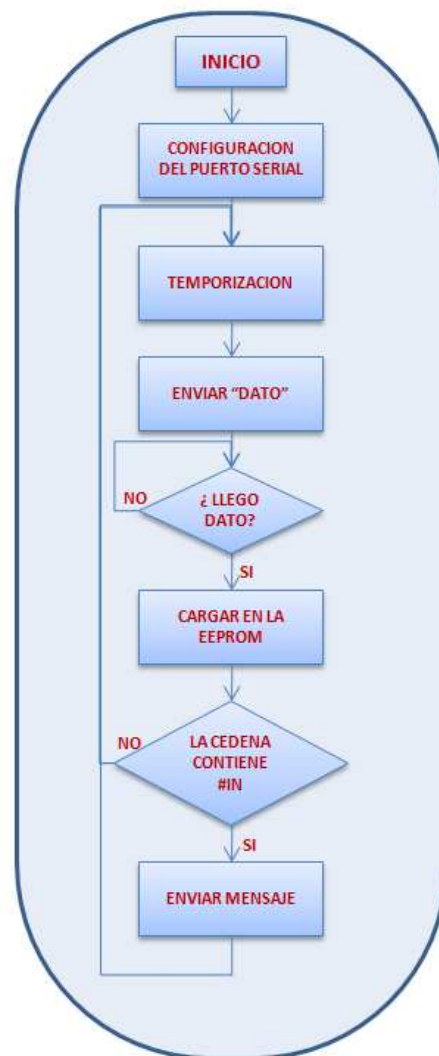


Figura 5-15: Diagrama de flujo de algoritmo para aplicación 1 en prueba experimental

5.2.3 DESARROLLO DE PRUEBAS DE LA APLICACIÓN SISDAQ

Para desarrollo de la evaluación de sistema se utilizó una SIMCARD del operador de red de telefonía móvil TIGO y un teléfono móvil suscrito a la misma red, el procedimiento a seguir es la conexión de SISDAQ en el conector de DB9 del puerto serial, encender el la aplicación y correr la aplicación de simulación del TE en LABVIEW.

Se programó el microcontrolador para ejercer una temporización de 10 segundos entre cada petición, en la siguiente figura se observa la interfaz de visualización en LABVIEW y la respuesta del TE cuando SISDAQ realiza una petición. Ver figura 5-16.

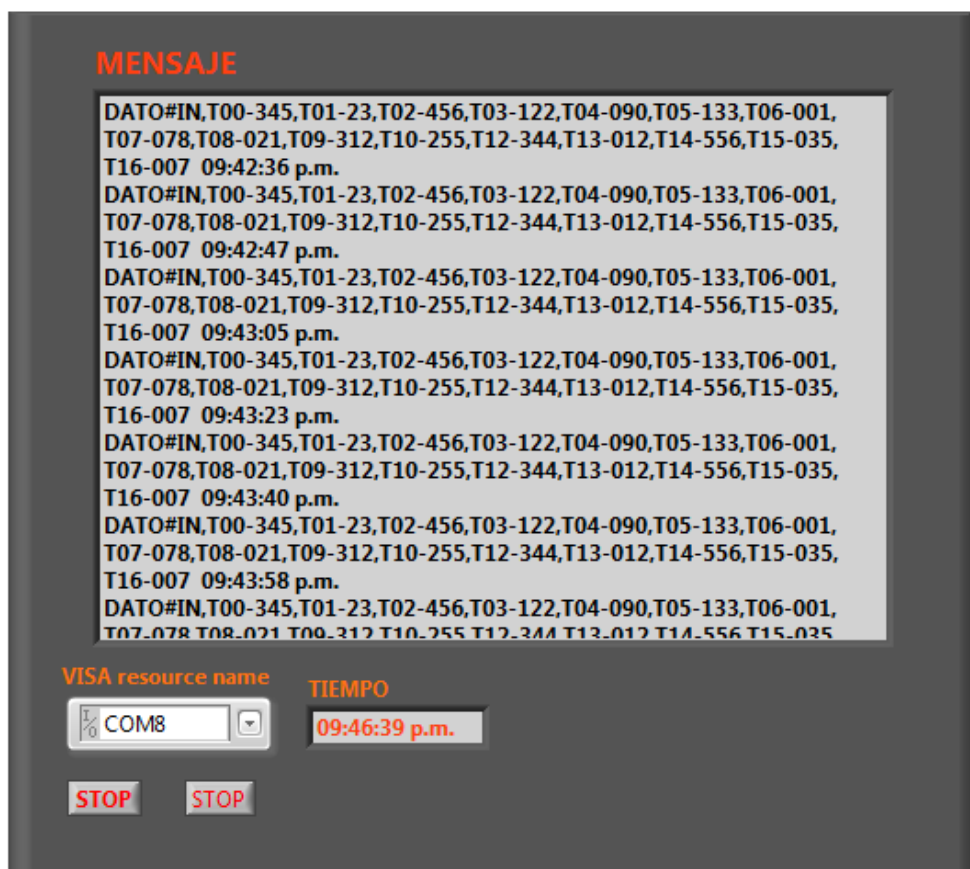


Figura 5-16: Interfaz de visualización de simulador de terminal de datos

Se enviaron 14 mensajes de texto consecutivamente según el algoritmo de la aplicación SISDAQ, se utilizó un teléfono Nokia 5130 como receptor de mensajes, los cuales se visualizan en el Nokia PC Suite que es el software de gestión para estos equipos móviles, ver figura 5-17.

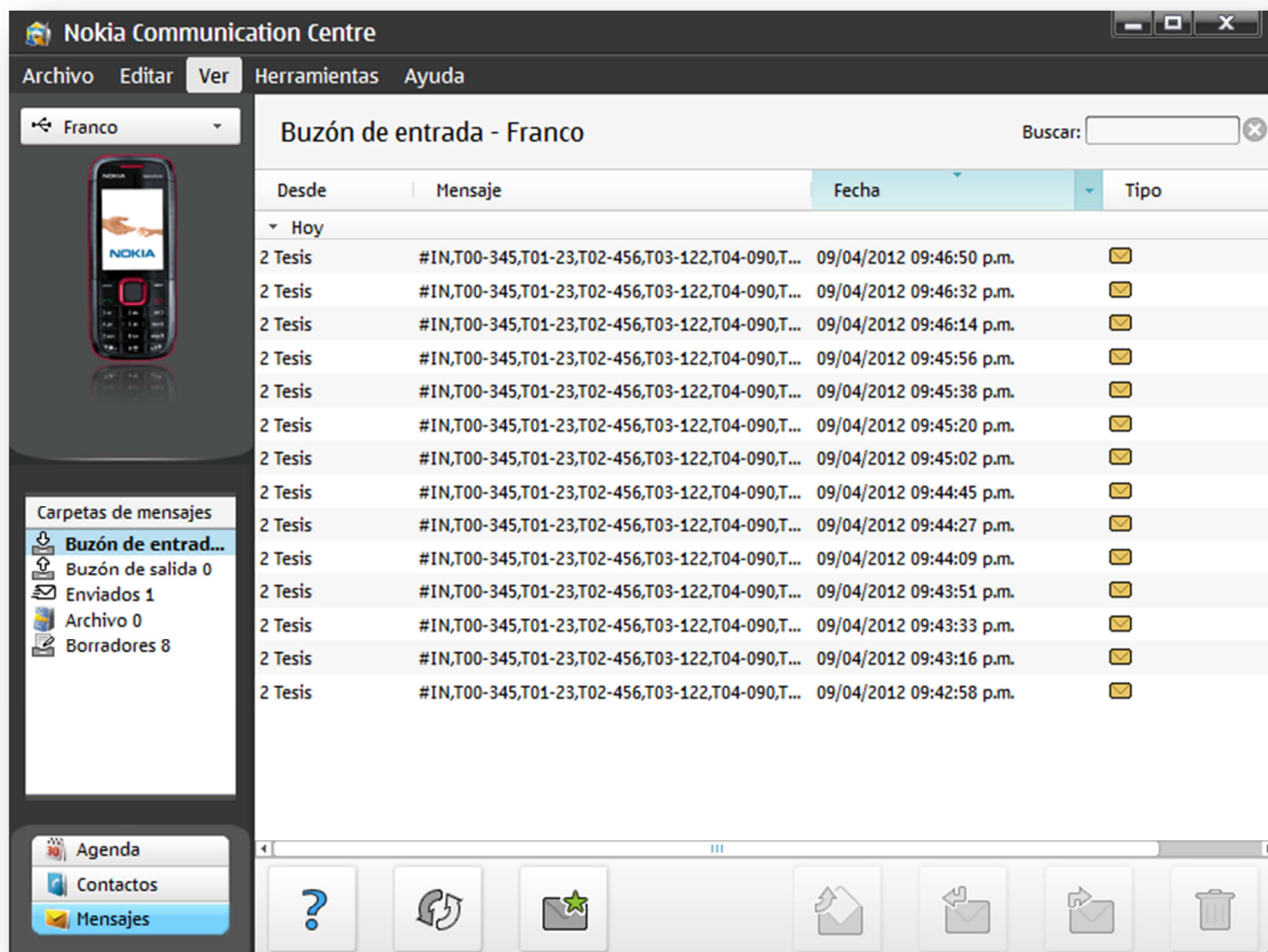


Figura 5-17: Recepción de mensajes en Nokia PC Suite

En la tabla 5-3 se compara el tiempo que toma capturar un mensaje y enviarlo a un destino y la calidad de la información es decir si hubo pérdida de datos, se expresa este valor en porcentaje, donde el 100% indica que todos los datos llegaron.

# DE MENSAJE	TIEMPO DE SALIDA	TIEMPO DE LLEGADA	TIEMPO TRANSCURRIDO	CALIDAD DE LA INFORMACIÓN
1	09:42:36	09:42:58	22 s	100%
2	09:42:47	09:43:16	29 s	100%
3	09:43:05	09:43:33	28 s	100%
4	09:43:23	09:43:51	28 s	100%
5	09:43:40	09:44:09	29 s	100%
6	09:43:58	09:44:27	30 s	100%
7	09:44:16	09:44:45	29 s	100%
8	09:44:34	09:45:02	29 s	100%
9	09:44:51	09:45:20	29 s	100%
10	09:45:09	09:45:38	29 s	100%
11	09:45:27	09:45:56	29 s	100%
12	09:45:41	09:46:14	33 s	100%
13	09:46:03	09:46:32	29 s	100%
14	09:46:21	09:46:50	21 s	100%
PROMEDIO DE DATOS			28,14 s	100%

Tabla 5-3: Tabla de comparación de tiempo de envío y calidad de la información

El tiempo promedio que toma capturar un dato empaquetarlo en un mensaje y enviarlo a un destino es de 28,14 s con una alta confiabilidad en la calidad de la información ya que, aunque sea un método lento de transmisión de datos es poco probable que se pierdan datos en la comunicación pero se obtiene la gran ventaja de poder acceder a la información en cualquier lugar y momento.

5.3 EVALUACIÓN DEL FUNCIONAMIENTO DE LA APLICACIÓN IVIGED

Para la evaluación de la aplicación IVIGED, se toma bloque a bloque y se evalúa el funcionamiento del mismo. Es necesario entender el funcionamiento de cada bloque para poder tener una idea en general de cómo funciona el proyecto y cuáles fueron los inconvenientes que se presentaron a medida que se desarrollaba, la definición de cada bloque se encuentra en la sección 2.3.8.

5.3.1 EVALUACIÓN DEL ENVÍO DEL MENSAJE DE TEXTO CON LA CONFIGURACIÓN INICIAL

Se utilizó la interfaz de usuario para realizar la evaluación del envío correo del mensaje de configuración. Los valores son los mostrados en la figura 5-18.



Figura 5-18: Envío de un SMS con la configuración inicial al número de celular de inicio.

El mensaje fue recibido con éxito como lo muestra la figura 5-19.

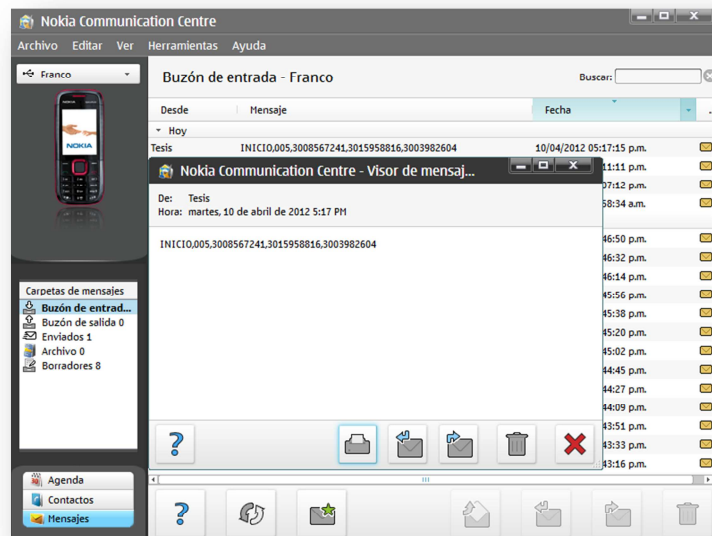


Figura 5-19: Mensaje recibido con la plantilla de inicio.

En la figura 5-19, se alcanza a notar la plantilla de inicio del sistema, llega con éxito a su destino con los números modificados en la interfaz de la figura 5-18. Con esto se garantiza que el sistema comience a monitorear las variables del proceso y a enviarlas por medio de SMS a la aplicación servidora. Es por esto que se debió buscar la manera de identificar de alguna manera los mensajes que indiquen la actualización del algún dato.

5.3.2 EVALUACIÓN DE LA RECEPCION DE MENSAJES Y VISUALIZACION DE LOS MISMOS.

Se envió un mensaje de texto con la plantilla de entrada de datos desde un número particular asociado como numero de celular de inicio. Se utilizó el visor de mensajes de la aplicación Nokia PC suite como se muestra en la figura 5-20.

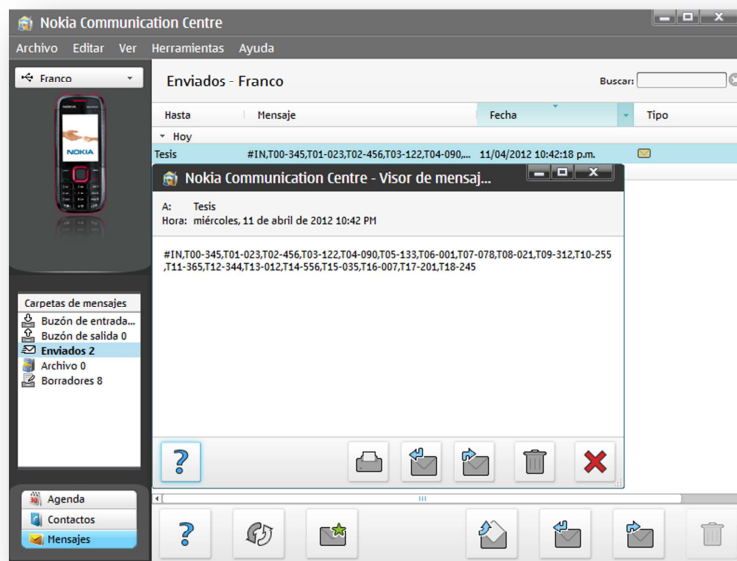


Figura 5-20: Envío de SMS a la aplicación IVIGED

El número de destino del mensaje enviado es el número de teléfono del modem asociado a la aplicación servidora con el contenido de la plantilla característica de entrada de datos. La aplicación servidora recibe el mensaje y lo visualiza como lo muestra la figura 5-21.



Figura 5-21: Interfaz de visualización de datos.

5.3.3 EVALUACIÓN DE LAS RESPUESTAS A LAS SOLICITUDES

Para la evaluación de las respuestas a las solicitudes particulares se envió un mensaje de texto con la plantilla de petición de datos desde un número de teléfono asociado como número del ingeniero con los datos de interés T01, T00, T13, T17 y T08 al número de teléfono del modem asociado a la aplicación IVIGED. Ver figura 5-22.

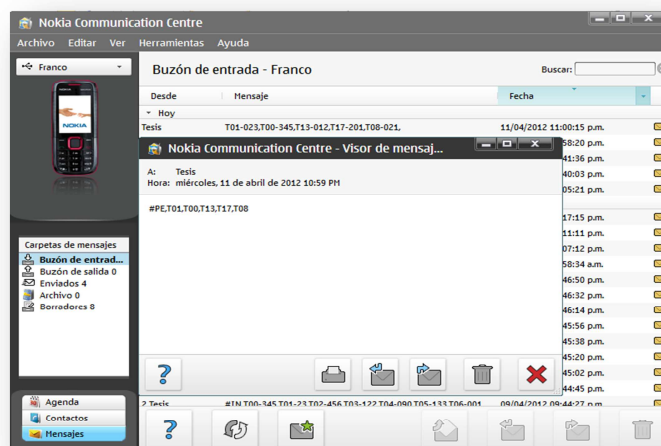


Figura 5-22: Envío de mensaje de texto con plantilla de petición de datos.

El mensaje recibido se muestra en la figura 5-23.

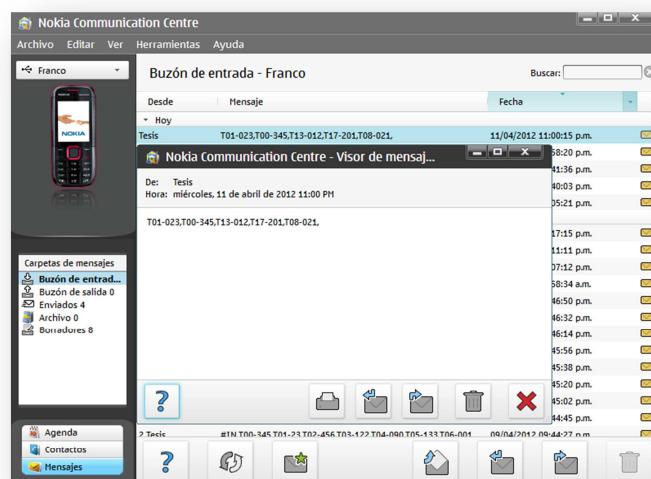


Figura 5-23: Mensaje recibido con la información de los datos solicitados.

Para la evaluación de la respuesta a petición total, se envió un mensaje de texto con la plantilla de petición de datos total desde un número de teléfono asociado como número del ingeniero, al número de teléfono del modem asociado a la aplicación IVIGED. Ver figura 5-24.

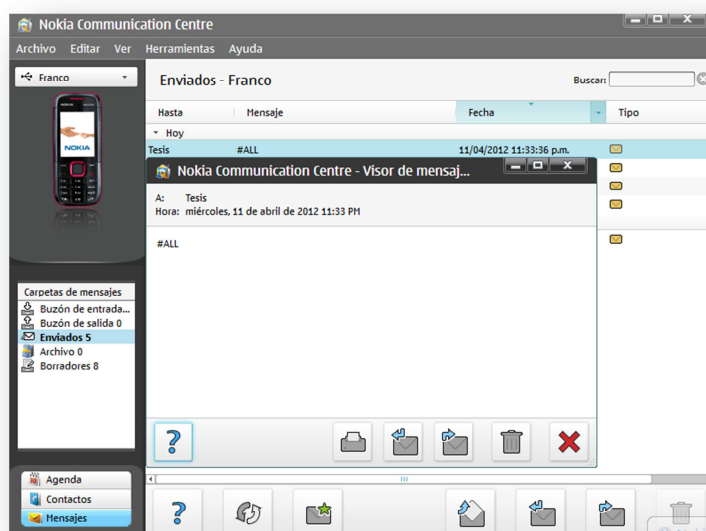


Figura 5-24: Mensaje de petición total enviado.

El mensaje recibido se muestra en la figura 5-25.

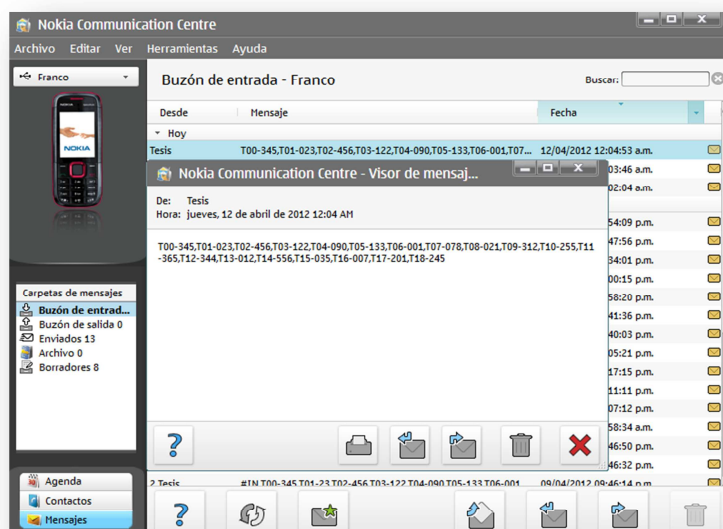


Figura 5-25: Mensaje recibido como respuesta a la solicitud total de datos.

5.3.4 EVALUACIÓN DE INFORME DIARIO DE LOS TODOS LOS DATOS A UNA HORA ESTIPULADA

Hasta este punto se han realizado las evaluaciones de los dos tipos de peticiones, falta realizar la evaluación de la notificación diaria de todos los mensajes hacia el ingeniero. Se estableció una hora cercana a la hora actual para corroborar el correcto funcionamiento de la función. La hora establecida se muestra en la figura 5-26.



Figura 5-26: Hora de envío diaria de todos los datos.

La respuesta es la mostrada en la figura 5-27. Se puede corroborar la hora de llegada del mensaje en la misma figura.

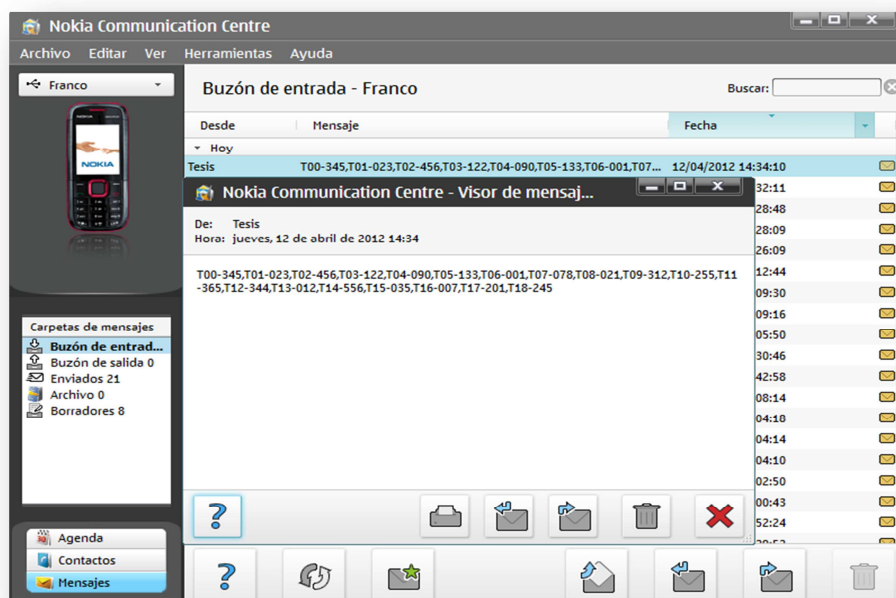


Figura 5-27: Mensaje recibido de la aplicación IVIGED con la información de todos los datos.

5.4 COMPARACION CON OTRAS TECNOLOGÍAS DE MONITOREO

Se realizó una comparación con algunos sistemas de monitoreo que se implementan tradicionalmente en el país con el fin de encontrar diferencias, ventajas y desventajas. Se encontraron tres tipos de tecnologías que se utilizan para el monitoreo de procesos industriales actualmente:

- Sistemas SCADA
- Monitoreo de procesos industriales por medio de internet
- Exemys

La descripción se detalla en la tabla

TECNOLOGÍA	DESCRIPCIÓN	COSTO DE IMPLEMENTACIÓN	VENTAJAS / DESVENTAJAS
SCADA	• Sistema que utiliza computadores que permiten supervisar y controlar variables del proceso localmente. • Requiere dispositivos anexos para informe de datos por SMS. • Parte de la obtención de datos desde el PLC o desde otros controladores. • Se requiere licencia por estación donde está corriendo.	Alto costo	Es un sistema robusto para el monitoreo y control de procesos industriales, presenta notificaciones de alarmas localmente, pero no notifica remotamente acerca de alarmas. No dispone de respuestas de información a dispositivos móviles en cuanto a las variables que se están monitoreando.
MONITOREO DE PROCESOS INDUSTRIALES POR INTERNET	• La obtención de los datos parte desde el PLC o desde otros controladores, utilizando una computadora como servidor y la cantidad de computadores por cliente. • Utiliza la nube de internet para enviar	Bajo costo	Presenta bajo costo y puede monitorear variables del mismo modo que cualquier otra tecnología. Notifica acerca de alarmas pero siempre y cuando exista conexión entre el cliente y el servidor. No presenta alguna

	la información de las variables que se están monitoreando.		interface GSM para notificación a dispositivos móviles. No dispone de respuestas de información a dispositivos móviles en cuanto a las variables que se están monitoreando. Depende la calidad del servicio de internet para la visualización de las variables
EXEMYS	• Utiliza un dispositivo llamado DABin en la cual almacenan una o varias páginas WEB a las que puede acceder desde cualquier navegador estándar. • Se requieren computadoras para la visualización de las variables en el software de monitoreo. • Utiliza e-mails para informar acerca de alarmas.	Alto costo	Trabaja con un dispositivo solido que brinda rigidez y estabilidad en el monitoreo de las variables. Es necesario de la disposición computadoras o dispositivos móviles de alta tecnología que cuenten con un navegador y el plan de navegación requerido. Notifica por medio de e-mails hacer de alarmas, lo que no es oportuno para el aviso inmediato de la alarma.

Tabla 5-4: Comparación con otros sistemas de monitoreo

CAPÍTULO VI

6 CONCLUSIONES Y RECOMENDACIONES

En este capítulo se da a conocer las conclusiones que se obtuvieron en el desarrollo y evaluación del proyecto. Las recomendaciones que se deben tener en cuenta para desarrollos futuros en sistemas de monitoreo remoto que empleen la redes de telefonía móvil celular, recomendaciones y conclusiones acerca de la implementación del lenguaje de programación grafico LabVIEW para sistemas monitoreados remotamente.

- El SMS a través de la red del operador telefonía móvil le permite al SISTEMA DE MONITOREO REMOTO DE UN PROCESO INDUSTRIAL EMPLEANDO LAS REDES DE TELEFONIA MOVIL CELULAR EN COLOMBIA la transmisión de datos de variables alfanuméricas. El software LabVIEW facilita la visualización y gestión de los datos en un entorno industrial.
- La aplicación SISDAQ tiene la capacidad de pedir información de variables a un terminal de datos y enviar ésta a través de un mensaje texto a un equipo móvil, con una velocidad de envío de mensaje máxima de 1 mensaje cada 29 segundos, puede muestrear 19 variables sin pérdida de información siempre y cuando éstas tengan un periodo de cambio mayor a 29 segundos.
- La aplicación IVIGED tiene la capacidad de recibir información, visualizar y gestionar información de datos de un proceso industrial, responder a solicitudes realizadas por el personal administrativo encargado del proceso y realizar notificaciones diarias con la información de todos los datos.
- El proyecto SISTEMA DE MONITOREO REMOTO DE UN PROCESO INDUSTRIAL EMPLEANDO LAS REDES DE TELEFONIA MOVIL CELULAR EN COLOMBIA permite el monitoreo automático de variables alfanuméricas y el acceso de las mismas a través del SMS en cualquier lugar y momento siempre que exista cobertura de la red de telefonía móvil celular del operador contratado.
- Para la implementación del SISTEMA DE MONITOREO REMOTO DE UN PROCESO INDUSTRIAL EMPLEANDO LAS REDES DE TELEFONIA MOVIL CELULAR EN COLOMBIA se deben cumplir los siguientes requerimientos:
 - Contar con 2 modem GSM con interfaz RS232 y 2 SIMCARD suscritas a un operador de red.
 - El terminal de datos debe reconocer el significado de las cadenas de caracteres "DATO", "XOFF", "XONN".
 - Digitar la información de usuario: número de ingeniero, números de alarma, periodo de muestreo, número de la SIMCARD de la aplicación SISDAQ y el puerto serial del modem GSM asociado a la aplicación IVIGED.
- Se espera que el que sistema de monitoreo remoto produzca un gran impacto en las tecnologías de información para monitoreo de procesos industriales, puesto que aunque muchas de las tecnologías actuales en Colombia cuentan con la característica de visualizar variables local y remotamente, no explotan la capacidad

de movilidad que ofrece la red GSM y sistema propuesto en este libro, a un bajo costo.

- Para el desarrollo futuro del proyecto de investigación se recomienda, explotar todas las ventajas de las nuevas generaciones de las redes de telefonía móvil celular, 3.5G y 4G por medio de aplicaciones móviles, se podría desarrollar una aplicación móvil que muestre en el teléfono de forma gráfica todas las variables de un proceso, permita el control y monitoreo de variables de forma remota.

BIBLIOGRAFÍA

- Acedo, J. (2007). *Instrumentacion y control basico de procesos*. España: Diaz de Santos.
- Alvarado S., A., Vinnett P., L., & Grote H., W. (2004). *Análisis de Tasa Efectiva de Servicio y Retardo en SMS*. Valdivia, Chile.
- Angulo Usategui, J. M., Angulo Martínez, I., & Etxebarria Ruiz, A. (2007). *Microcontroladores "PIC": diseño práctico de aplicaciones. PIC12F508 y PIC16F84A, lenguajes Ensamblador, C y PBASIC*. McGraw-Hill.
- Arduino. (21 de Noviembre de 2011). RS 232. Obtenido de <http://arduino.cc/es/Tutorial/ArduinoSoftwareRS232>
- Contraloria general de la republica. (2010). *ESTADO ACTUAL DEL SERVICIO DE TELEFONÍA MÓVIL EN COLOMBIA*. Bogota.
- Contraloria general de la republica. (s.f.). *Estado actual del servicio de telefonía movil en Colombia (85113-059-05)*. Bogota .
- Conzales Castellanos, A. C., Jauregui Suarez, R. A., & Melchor Mejia, O. A. (2002). *COMUNICACIONES MOVILES GSM*. Guatemala: Universidad Francisco Marroquin.
- ECE. (02 de Marzo de 2012). *Amplificador Operacional*. Obtenido de http://www.ece.rice.edu/~jdw/243_lab/exp5.1.html
- Escuela Politecnica Superior de Alcoy. (02 de Febrero de 2012). *Redes GSM*. Obtenido de http://server-die.alc.upv.es/asignaturas/lased/2002-03/Redes_GSM/principal.htm
- EVA. (10 de Diciembre de 2011). *Estandar RS232*. Obtenido de <http://www.eva.uji.es/mcxcf/DocSerie.html>
- GENIA. (10 de Diciembre de 2011). *Entornos Integrados de Automatizacion*. Obtenido de <http://isa.uniovi.es/genia/spanish/info/index.htm>
- Gomez, J. (2010). *Telefonia Movil Celular - Radiocomunicaciones II*. Santa Marta.
- Gonzalez Gomez, J. (2002). EL SERVICIO SMS: UN ENFOQUE PRACTICO. En J. Gonzalez Gomez, *Trabajo de doctorado de la asignatura "Nuevas tecnologías para las comunicaciones"* (págs. 9 - 45).
- Institute European Telecommunications Standards - ETSI TS 100 901 V7.3.0 . (1999-11). *Technical realization of the Short Message Service (SMS)*.
- Jeff. (16 de Octubre de 2008). *Kioeskea*. Recuperado el 13 de Marzo de 2012, de Estándar GSM (Sistema global de comunicaciones móviles): <http://es.kioeskea.net/contents/telephonie-mobile/gsm.php3>
- Lammertbies. (21 de Noviembre de 2011). *RS 232 Null modem*. Obtenido de http://www.lammertbies.nl/comm/info/RS-232_null_modem.html
- Lázaro Laporta, J., & Miralles Aguiñiga, M. (2005). *Fundamentos de telemática*. Valencia: Univ. Politéc.
- Manuel, A. (2001). *Instrumentación virtual: adquisición, procesado y análisis de señales*. Edicions UPC.
- Mata, I. N. (2001). *Fuentes de alimentacion reguladas lineales*. Recuperado el 15 de Enero de 2012, de http://www.ing.uc.edu.ve/~ajmillan/Docencia/Material/RegLin_02.pdf
- Maxim Integrated Products. (2010). *+5V-Powered, Multichannel RS-232 Drivers/Receivers*. Sunnyvale, CA: Maxim Integrated Products, Inc.

- Maximic. (17 de Enero de 2012). *MAX232*. Obtenido de <http://www.maximic.com/datasheet/index.mvp/id/1798>
- National Instruments. (6 de Junio de 2006). *COMUNICACION SERIAL: CONCEPTOS GENERALES*. Recuperado el 16 de Enero de 2012, de Identificación del Documento: 1M9E1L6Q: <http://digital.ni.com/public.nsf/allkb/039001258CEF8FB686256E0F005888D1>
- National Instruments. (15 de Marzo de 2012). *¿Cómo Puedo Usar NI LabVIEW?* Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/labview/applications/esa/>
- National Instruments. (15 de Marzo de 2012). *¿Que es LabVIEW?* Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/labview/whatis/esa/>
- National Instruments. (15 de Marzo de 2012). *National Instruments VISA*. Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/visa/>
- National Instruments. (15 de Marzo de 2012). *NI LabVIEW para Adquirir Datos y Procesar Señales*. Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/labview/applications/daq/esa/>
- National Instruments. (15 de Marzo de 2012). *NI LabVIEW para Automatizar Sistemas de Pruebas y Validación*. Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/labview/applications/test/esa/>
- National Instruments. (15 de Marzo de 2012). *NI LabVIEW para Enseñanza Académica*. Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/labview/applications/academic/esa/>
- National Instruments. (15 de Marzo de 2012). *NI LabVIEW para Monitorear y Controlar Sistemas Embebidos*. Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/labview/applications/embedded/esa/>
- National Instruments. (s.f.). *NI LabVIEW para Control de Instrumentos*. Recuperado el 15 de Marzo de 2012, de <http://www.ni.com/labview/applications/instrument-control/esa/>
- Octoplus. (18 de Febrero de 2012). *Arquitectura de microcontroladores*. Obtenido de <http://www.octoplus.com.co/foros/index.php?topic=10.0>
- Penin, R. (2008). *Sistema Scada*. Barcelona: Marcombo.
- Rivas, L. (6 de Octubre de 2010). *Nubitek*. Obtenido de <http://www.nubitek.com>
- SIMCOM. (2005). *SIM340 AT Commands Set - SOFTWARE SPECIFICATION*. Doc. Id: SIM340_ATC_V1.00 .
- SIMCOM. (2007). *SIM340 Hardware Design* . Doc. Id: SIM340_HD_V2.01 .
- Tanenbaum, A. S. (2003). *Redes de Computadoras*. Pearson Educación.
- Tarazona Anteliz, J. O., Casadiego, C., & Alexander, S. (2008). *Servicio de recepcion y envio de mensajes SMS para consultar la informacion academica de la Universidad Francisco de paula santander*. Cucuta.
- Tecmoviles. (26 de Agosto de 2011). *Estacion Movil*. Obtenido de <http://tecmoviles.com>
- Wikipedia. (15 de Noviembre de 2011). *GSM*. Obtenido de <http://en.wikipedia.org/wiki/GSM>

ANEXOS

ANEXO A1

CÓDIGO DE PROGRAMACIÓN PARA
EL FIRWARE DE LA APLICACIÓN
SISDAQ

```

1: int vm,pos_epm;
2: int flag=0,cont=0,j=0,adress=0,y=0,cdat=0;
3: short band=0;
4: int i=0,Li=0;//la variable se usa para retardo de fncion RECIBIR y tambien en car
   rgar_EEpron
5: unsigned int m=0,Lm=0;// Li...es limite de veces que se va desbordar el TMRO es de
   ecir el limite de i
6:
7: char k;
8: char n_msj;
9: char dato[20];
10: char dato_te[15];
11: char CR=0x0D;
12: char CMLL=34;
13: char SUB=26;
14: ///////////////////////////////////
15: void interrupt()
16: {
17:     if(PIR1.F0==1 & Lm!=0)
18:     {
19:         PIR1.TMR1IF = 0;    // Reiniciar el bit TMR1IF
20:         TMR1H = 0x00;      // Ajustar el valor inicial del temporizador TMR1
21:         TMR1L = 0x00;
22:         m++;
23:         if(m>=Lm)
24:         {
25:             band=1;
26:             m=0;
27:         }
28:     }
29:     if(INTCON.F2==1)
30:     {
31:         i++;// esta interrupcion cuenta 400 veces el debordamiento del TMR0 para logra
   ar un temporizacion de 2seg
32:         TMR0=0;
33:         INTCON.F2=0;// Reseteamos la bandera de desbordamiento pero que queda habilira
   do el vector de interrupcion golabal
34:
35:         if (i>=Li)
36:         {
37:             //Soft_UART_Break();
38:             flag=1;
39:             INTCON=0b11000000;
40:             i=0;
41:         }
42:     }
43: }
44:
45: ///////////////////////////////////
46: void SOFT_ENVIAR (char *point)//Funcion para ENVIAR datos del SOFT UART
47: {
48:     k=4;
49:     i=0;
50:     while(i<k)
51:     {
52:         Soft_UART_Write(point[i]);
53:         i++;
54:     }
55: }
56: ////
57: ///////////////////////////////////
58: void WARNING (void)

```

```
59: {
60: while(1)
61: {
62:     PORTB.F1=1;
63:     Delay_ms(2000);
64:     PORTB.F1=0;
65:     Delay_ms(2000);
66: }
67: }
68: //////////////////////////////////////
69: void ENVIAR (char *point)//Funcion para ENVIAR datos del UART
70: {
71:     UART1_Write_Text(point);
72:     UART1_Write(CR);
73: }
74: //////////////////////////////////////
75: //Funcion para RECIBIR un dato y compararlo para que sea la respuesta deseada po
    medio del uart
76: void RECIBIR (char *punter)
77: {
78:     char *result;
79:     cont=0;
80:     flag=0;
81:     i=0;
82:     RCSTA.F7=1;
83:     RCSTA.F4=1;
84:     memset(dato,0,15);
85:     OPTION_REG =0x07;
86:     INTCON = 0b11100000;
87:     TMR0=0;
88:     while(1)
89:     {
90:         if (UART1_Data_Ready()== 1)
91:         {
92:             dato[cont]=UART1_Read();
93:             cont++;
94:             TMR0=0;//cada vez que se recibe un dato reiniciamos TMR0
95:             i=0;//y se reinicia el contador
96:         }
97:         if (flag==1)
98:         {
99:             break;
100:        }
101:    }
102:
103: result=strstr(dato,punter);
104: if (result==0)
105: {
106:     flag=0;
107:     j++;
108:     if(j>=8)
109:     {
110:         WARNING();
111:     }
112: }
113: if (result!=0)
114: {
115:     j=0;
116:     flag=1;
117: }
118: }
119:
```

```
120: ////////////////////////////////////////////
121: //Funcion para ENVIAR SMS
122: void MENSAJE (void)
123: {
124:     char *RTA,*vector;
125:     atras:
126:     flag=0;
127:     RTA=">";
128:     while(flag==0)
129:     {
130:         UART1_Write_Text("AT+CMGS=");
131:         UART1_Write(CMLL);
132:         k=0;
133:         while(k<10)
134:         {
135:             UART1_Write(EEPROM_Read(pos_epm+k));
136:             k++;
137:         }
138:         UART1_Write(CMLL);
139:         UART1_Write(CR);
140:         RECIBIR(RTA);
141:     }
142:     flag=0;
143:     k=0;
144:     while(k<cdat)
145:     {
146:         UART1_Write(EEPROM_Read(k));
147:         k++;
148:     }
149:     UART1_Write(SUB);
150:     while(PORTC.F7==1){}
151:     RTA="+CMGS";
152:     RECIBIR(RTA);
153:     if(flag==0)
154:     {
155:         vector="AT";
156:         RTA="OK";
157:         while(flag==0)
158:         {
159:             ENVIAR(vector);
160:             RECIBIR(RTA);
161:         }
162:         goto atras
163:     }
164: }
165:
166: ////////////////////////////////////////////Esta funcion Verifica si lo que llego al puerto se t
ata de un SMS
167: void VERIFICAR_SMS(void)//// y carga a n_msj el numero donde esta guardado el mens
saje
168: {
169:     flag=0;
170:     if(EEPROM_Read(2)=='+'|EEPROM_Read(3)=='C'|EEPROM_Read(4)=='M'|EEPROM_Read(5)=='
='T'|EEPROM_Read(6)==':')
171:     {
172:         flag=1;
173:     }
174: }
175: ////////////////////////////////////////////
176:
177: void CARGAR_EEPROM (void)
178: {
```

```
179:     char dat,error;
180:     cont=0;
181:     OPTION_REG = 0x07;
182:     INTCON = 0b11100000;
183:     TMR0 = 0;
184:     flag=0;
185:     i=0;
186:
187:     if(y==0)
188:     {
189:         while(flag==0)//Primero guardamos el mensaje COMPLETO
190:         {
191:             if (UART1_Data_Ready() == 1)
192:             {
193:                 EEPROM_Write(cont,UART1_Read());
194:                 TMR0=0;//cada vez que se recibe un dato reiniciamos TMR0
195:                 i=0;//y se reinicia el contador
196:                 cont++;
197:             }
198:         }
199:         if(cont!=0)
200:             Delay_ms(30);
201:     }
202:     if(y==1)
203:     {
204:         while(flag==0)//Primero guardamos el mensaje COMPLETO
205:         {
206:             if(PORTB.F7==0)
207:             {
208:                 do
209:                 {
210:                     dat=Soft_UART_Read(&error);
211:                     TMR0=0;
212:                     i=0;
213:                 }while (error);
214:                 //Soft_UART_Break();
215:                 EEPROM_Write(cont,dat);
216:                 cont++;
217:             }
218:         }
219:         if(cont!=0)
220:             Delay_ms(30);
221:     }
222: }
223:
224: void CARGAR_VECTOR (void)
225: {
226:     cont=0;
227:     OPTION_REG=0x07;
228:     INTCON=0b11100000;
229:     TMR0=0;
230:
231:     flag=0;
232:     i=0;
233:     j=0;
234:     while(flag==0)
235:     {
236:         if (UART1_Data_Ready() == 1)
237:         {
238:             dato[cont]=UART1_Read();
239:             cont++;
240:             TMR0=207;//cada vez que se recibe un dato reiniciamos TMR0
```

```
241:     i=0;//y se reinicia el contador
242: }
243: }
244: }
245:
246: void GUARDAR_NUM(void)
247: {
248:     cont=0;
249:     while(cont<10)
250:     {
251:         dato[cont]=EEPROM_Read(cont+k);
252:         cont++;
253:     }
254:     Delay_ms(40);
255:     cont=0;
256:     while(cont<10)
257:     {
258:         EEPROM_Write(adress+cont,dato[cont]);
259:         cont++;
260:     }
261:     Delay_ms(40);
262:     k=0;
263:     adress=0;
264: }
265:
266: void CHAR_TO_INT()
267: {
268:     char  z='0';
269:     short w=0;
270:     while(z<':')
271:     {
272:         if(z==EEPROM_Read(201))
273:         {
274:             vm=w*10;
275:             break;
276:         }
277:         z++;
278:         w++;
279:     }
280:     z='0';
281:     w=0;
282:     while(z<':')
283:     {
284:         if(z==EEPROM_Read(202))
285:         {
286:             vm=vm+w;
287:             break;
288:         }
289:         z++;
290:         w++;
291:     }
292: }
293:
294: // Programa Principal
295: void main (void)
296: {
297:     char *vector;
298:     char *RTA,error;
299:     // Inicializacion de puertos y uart
300:     TRISB.F0=1;TRISB.F1=0;
301:     PORTB.F1=0;
302:     UART1_Init(2400);
```

```
303:
304: error = Soft_UART_Init(&PORTB, 7, 6, 2400, 0); //
305:   if (error > 0)
306:   {
307:       while(1)
308:       {
309:           PORTB.F1=1;
310:           Delay_ms(100);
311:           PORTB.F1=0;
312:           Delay_ms(100);
313:       }
314:   }
315: //.....
316: //VERIFICAR EL ENCENDIDO DEL MODEM
317: while(PORTB.F0==0)
318: {
319:     PORTB.F1=1;
320:     Delay_ms(500);
321:     SOFT_ENVIAR("!ON;");
322:     PORTB.F1=0;
323:     Delay_ms(500);
324: }
325: i=0;
326: while(i<=75)
327: {
328:     PORTB.F1=1;
329:     Delay_ms(100);
330:     PORTB.F1=0;
331:     Delay_ms(100);
332:     i++;
333: }
334: //.....
335: //INICIALIZACION DE MODEM GSM
336: vector="AT";
337: RTA="OK";
338: flag=0;
339: Li=45;
340: i=0;
341: while(flag==0)
342: {
343:     ENVIAR(vector);
344:     RECIBIR(RTA);
345: }
346: vector="AT+CMGF=1"; // CONFIGURAR EL MODEM EN MODO TEXTO
347: RTA="OK";
348: flag=0;
349:
350: while(flag==0)
351: {
352:     ENVIAR(vector);
353:     RECIBIR(RTA);
354: }
355: //.....
356: /// INICIO DEL PROGRAMA ///
357: INICIO:
358: cont=0;
359: while(cont<3)
360: {
361:     PORTB.F1=0;
362:     Delay_ms(1000);
363:     PORTB.F1=1;
364:     Delay_ms(300);
```

```
365:   cont++;
366: }
367:   PORTB.F1=0;
368:
369: while(1)///Estado de espera de MSJ de inicio..y numros de alarma
370: {
371:   if(PORTC.F7==0)
372:   {
373:     y=0; ///para cargar los datos que vienen del modem
374:     CARGAR_EEPROM();///despues de usar esta funcion el numero de datos que se carg
375:     aron esta en cont
376:     VERIFICAR_SMS();///si la bandera se activa....es un sms
377:     if(flag==0)
378:     goto INICIO;///NO ERA UN SMS
379:     /////////////Como es el primer MENSAJE se compara inicio y se almacenan los n
380:     umeros
381:     cont=0;
382:     while(cont<6)
383:     {
384:       vector[cont]=EEPROM_Read(cont+46);
385:       cont++;
386:     }
387:     if(vector[0]!='I'|vector[1]!='N'|vector[2]!='I'|vector[3]!='C'|vector[4]!='I'
388: |vector[5]!='O')
389:     goto INICIO;
390:     vector[0]=EEPROM_Read(53);
391:     vector[1]=EEPROM_Read(54);
392:     vector[2]=EEPROM_Read(55);
393:     Delay_ms(40);
394:     EEPROM_Write(200,vector[0]);
395:     EEPROM_Write(201,vector[1]);
396:     EEPROM_Write(202,vector[2]);
397:     Delay_ms(40);
398:     CHAR_TO_INT();
399:     cont=0;
400:     if(EEPROM_Read(9)!='3')
401:     goto INICIO;
402:     k=9;
403:     adress=205;
404:     GUARDAR_NUM();
405:     if(EEPROM_Read(57)!='3')///el MENSAJE debe llegar al menos con dos numeroos el
406:     numero que lo envio y un numero de alarma
407:     goto INICIO;
408:     k=57;
409:     adress=215;
410:     GUARDAR_NUM();
411:     if(EEPROM_Read(68)!='3')
412:     goto SALIDA
413:     k=68;
414:     adress=225;
415:     GUARDAR_NUM();
416:     if(EEPROM_Read(79)!='3')
417:     goto SALIDA
418:     k=79;
419:     adress=235;
420:     GUARDAR_NUM();
421:     SALIDA:
422:     SOFT_ENVIAR("M-OK");
423:     break;
424:   }
```

```

423:
424:
425: Lm=100*(vm/5);           //400 ES APREXIMANDAMENTE 40 SEGUNDOS
426: PIR1.F0=0;
427: if(Lm!=0)
428: {
429:     INTCON = 0b11100000; // Interrupción habilitada (bits GIE y PEIE)
430:     PIR1.TMR1IF = 0;      // Reiniciar el bit TMR1IF
431:     TMR1H = 0x00;         // Ajustar el valor inicial del temporizador TMR1
432:     TMR1L = 0x00;
433:     PIE1.TMR1IE = 1;      // Habilitar la interrupción al producirse un desbordamiento
to
434:     T1CON = 0b00110001;   // Configurar el temporizador TMR1
435: }
436: band=0;
437:
438:
439: ESTADO1:
440: while(1)
441: {
442:     T1CON.F0=1;
443:
444:     if(PORTB.F7==0) //PREGUNTAMOS SI HAY DATOS DEL TE
445:     {
446:         T1CON.F0=0;
447:         y=1; //PARA QUE SE CARGUEN LOS DATOS DEL TE A LA EEPROM...
448:         CARGAR_EEPROM ();
449:         cdat=cont;
450:         if(EEPROM_Read(0)!='A'|EEPROM_Read(1)!='L'|EEPROM_Read(2)!='A'|EEPROM_Read(3)
!='R'|EEPROM_Read(4)!='M'|EEPROM_Read(5)!='A')
451:             goto ESTADO1;
452:         pos_epm=215; //indica el numero en la eeprom
453:         MENSAJE();
454:         if(EEPROM_Read(225)=='3')
455:         {
456:             pos_epm=225;
457:             MENSAJE();
458:         }
459:         if(EEPROM_Read(235)=='3')
460:         {
461:             pos_epm=235;
462:             MENSAJE();
463:         }
464:         SOFT_ENVIAR("A-OK");
465:     }
466:
467:
468:     if(band==1)
469:     {
470:
471:         T1CON.F0=0;
472:         AGAIN:
473:         SOFT_ENVIAR("XOFF");
474:         Delay_ms(10);
475:         SOFT_ENVIAR("DATO");
476:         i=0; // activamos el temporizador
477:         flag=0;
478:         OPTION_REG =0x07;
479:         INTCON = 0b11100000;
480:         TMR0=0; //.....
481:         while(flag==0 & PORTB.F7==1)
482:         {

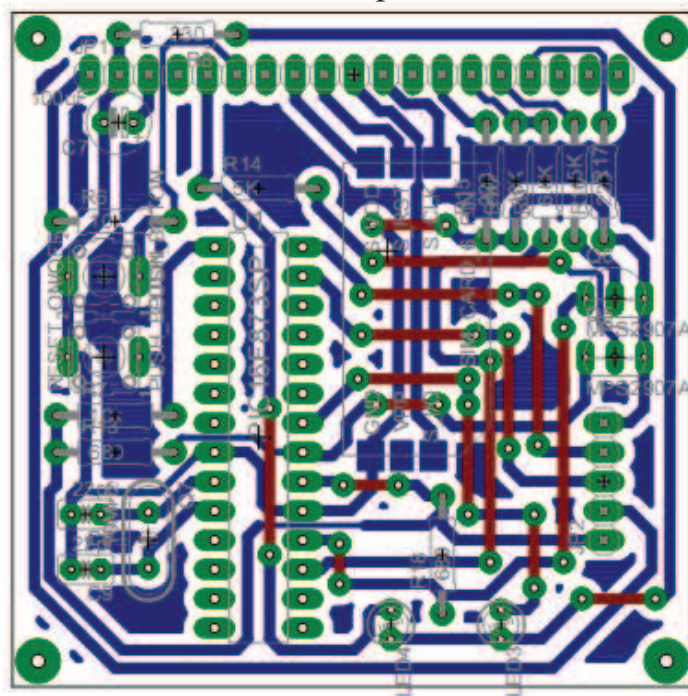
```

```
483:
484:     }
485:     if (flag==1 & PORTB.F7==1)
486:     goto AGAIN;
487:     INTCON=0b11000000;
488:     y=1; // PARA QUE SE CARGUEN LOS DATOS DEL TE A LA EEPROM...
489:     CARGAR_EEPROM ();
490:
491:     cdat=cont;
492:     if (cont==0)
493:     goto AGAIN;
494:     if (EEPROM_Read(0) != '#')
495:     goto EXIT; // SI NO LLEGO NADA QUE SALGA
496:     pos_epm=205;
497:     MENSAJE();
498:
499:     EXIT:
500:     band=0;
501:     SOFT_ENVIAR("XONN");
502: }
503: }
504: }
```

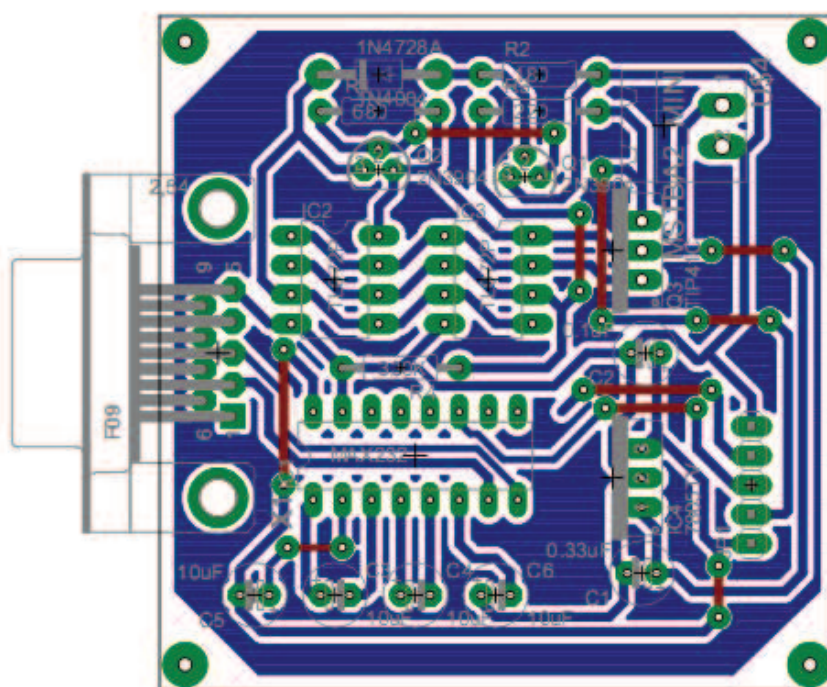

ANEXO A2

DIAGRAMA DEL PCB DE LA
APLICACIÓN SISDAQ

Placa superior



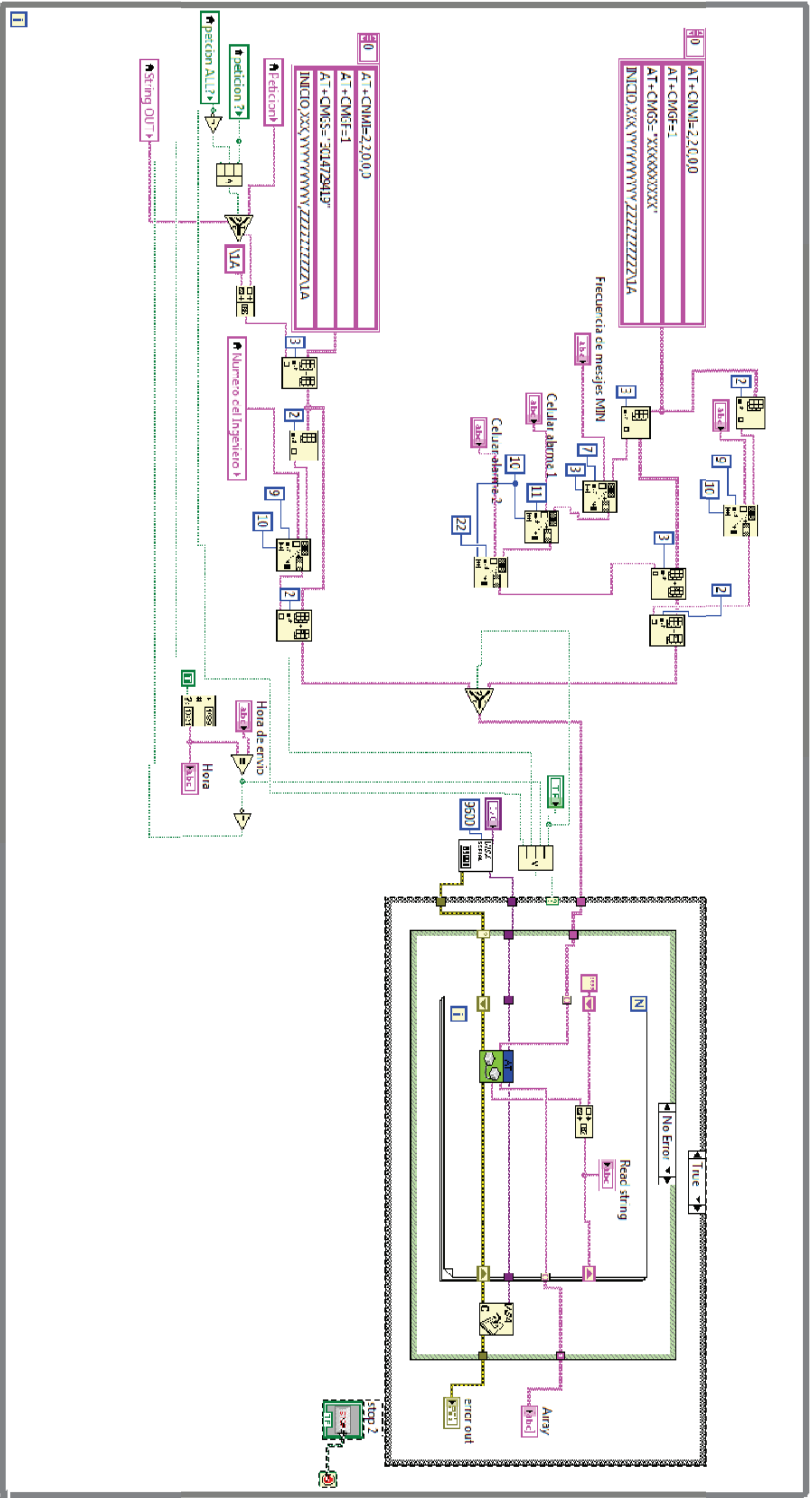
Placa inferior



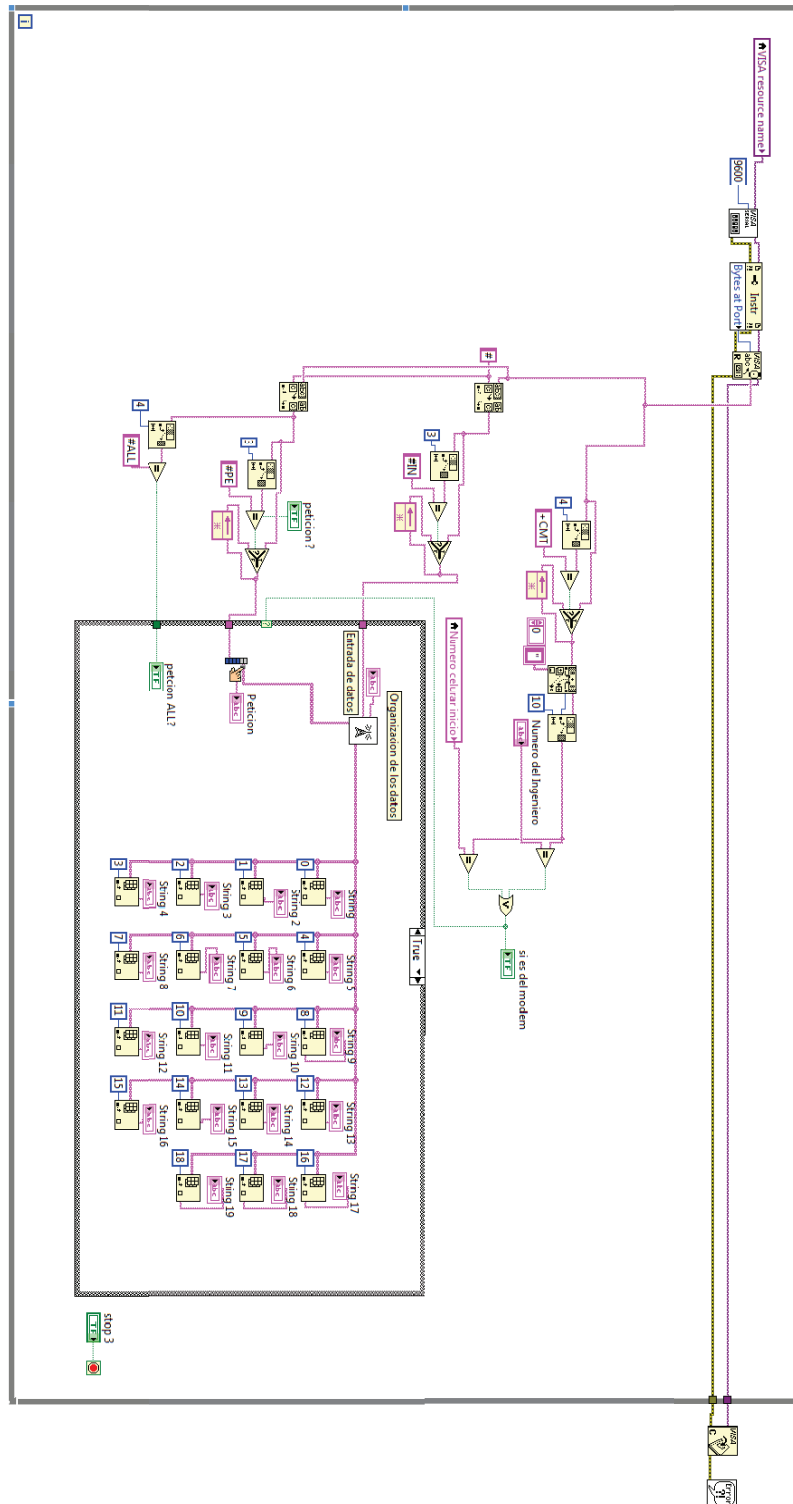
ANEXO A3

CODIGO DE PROGRAMACION EN LABVIEW PARA LA APLICACIÓN IVIGED

Código de configuración inicial



Código de visualización y almacenamiento de datos



ANEXO A4

MANUAL DE USUARIO IVIGED

En el presente informe se encontrará el manual de cómo debe operar un usuario la plataforma IVIGED. Se explica detalladamente como los datos deben ser ingresados en la interfaz de usuario. Cuáles son los pasos a seguir para que el sistema funcione adecuadamente y las precauciones que se deben tener antes de dar inicio al programa para que no se presenten inconvenientes el desarrollo del sistema.

CONFIGURACION DE INICIO DEL SISTEMA

El Sistema es dependiente de la configuración que le es suministrada en la interfaz de usuario IVIGED. Es en esta donde se le indica al sistema, cuales son los números de celulares a notificar acerca de alguna alarma y el numero de celular del Ingeniero encargado del proceso industrial. Además, se le indica al sistema el periodo de muestreo de las variables, la ubicación del archivo de texto en donde serán guardados los datos de las variables localmente, la identificación del puerto de comunicación en donde esta conectado el modem a trabajar con IVIGED, y el numero de celular de inicio el cual es el numero de celular del modem asociado a SISDAQ.

DIGITACION DE LOS NUMEROS DE CELULAR DE ALARMA E INGENIERO

El orden de ingreso de los datos no esta establecido, puede ser cualquiera. Se debe ingresar los números de celulares de alarma e ingeniero y garantizar el correcto ingreso de los datos, puesto que no se pueden cambiar los números de celulares, una vez se inicie el sistema. Se debe ser precavido en los números de celulares digitados, son 10 dígitos y no deben existir espacios al comienzo del número, ni al final.

The screenshot displays the IVIGED configuration window. At the top, there is a field for 'Localización del archivo' with the path 'C:\Users\user\Desktop\franco.txt'. Below this, on the left, are three status indicators: 'DATO DEL MODEM', 'PETICION ALL', and 'PETICION PARTICULAR', each with a green dot. In the center is a large black rectangular area. To the right of this area, there are several input fields: 'Numero celular inicio' (3008891535), a dropdown for 'COM7', 'Frecuencia de mensajes MIN' (005), '# Celular alarma 1' (3015958816), '# Celular alarma 2' (3003982604), and 'Numero del Ingeniero' (3008567241). At the bottom left are 'INICIO' and 'STOP' buttons. At the bottom center, there are fields for 'Hora de envío' (16:33:10) and 'Hora actual'. A blue arrow points from the text 'No deben existir espacios' to the 'Numero del Ingeniero' field.

Figure 1: Digitación de los números de celulares

SELECCIÓN DE LA FRECUENCIA DE MENSAJES

Para la asignación de la frecuencia de mensajes, la interfaz arroja una lista desplegable con los valores que van desde 005 minutos hasta 060 minutos como lo muestra la figura 2. Solo se debe seleccionar el valor deseado.

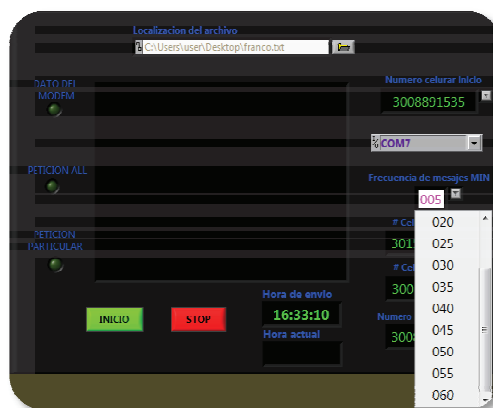


Figure 2: Selección de la frecuencia de mensajes

SELECCIÓN DEL PUERTO DE COMUNICACIÓN

Para el campo de identificación del puerto de comunicación, también existe una lista desplegable que muestra los puertos que están disponibles para trabajar. Si el puerto de comunicación no aparece en la lista se recomienda utilizar las propiedades avanzadas del sistema operativo para reconocer el puerto de comunicación, una vez identificado presionar la opción “Refresh” y seleccionar el puerto de comunicación asociado al modem.

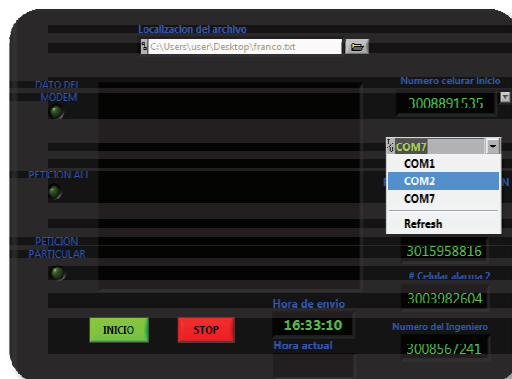


Figure 3: Selección del puerto de comunicación

SELECCIÓN DEL NUMERO DE CELULAR INICIO Y EDITAR CAMPOS

En el campo Número celular inicio, existe una lista desplegable con los números de celulares asociados al modem conectado a SISDAQ. Este campo es editable al igual que el campo frecuencia de mensajes.

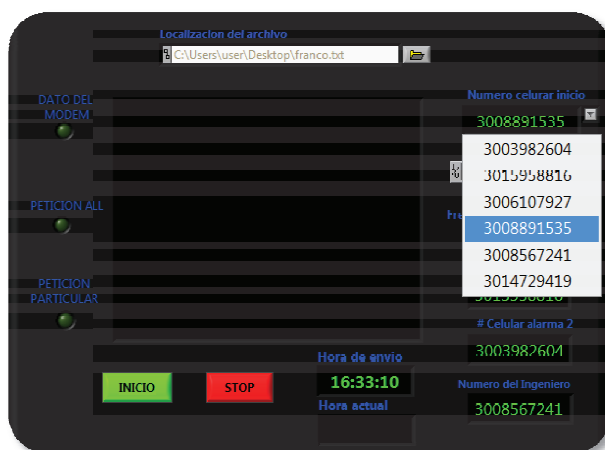


Figure 4: Selección del número celular inicio

Para editar estos campos se debe ubicar el cursor sobre el campo, hacer click derecho y seleccionar la opción EditItems. Véase figura 5.

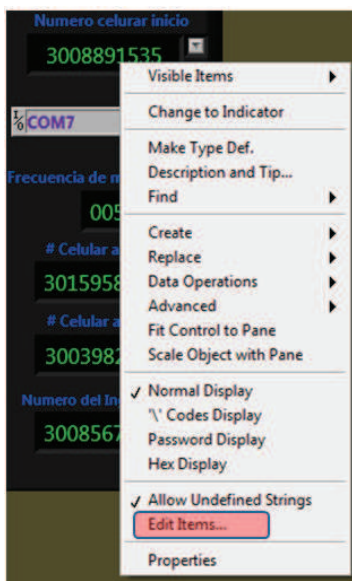


Figure 5: Asignación de valores en la lista desplegable.

Luego de esto aparecerá una ventana en donde se realizarán los cambios de valores de la lista. Véase la figura 6.

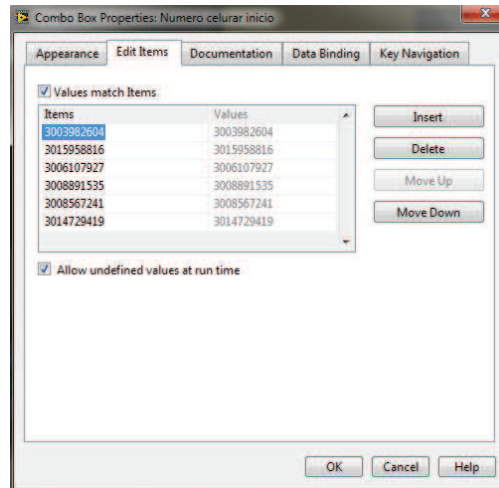


Figure 6: Valores del campo Número celular inicio

Para cambiar alguno de estos valores se debe ubicar el cursor encima del valor a cambiar en las celdas “Items” y hacer doble click, luego digitar el valor deseado. Automáticamente el valor de las celdas “Values” tomará el valor de la celda “Items”. Recuerde que los números de celulares de inicio no deben contener espacios. Al igual para el campo de frecuencia de mensajes, los valores deben contener 3 caracteres y no debe exceder los 999 minutos, ni tampoco contener el valor 000 minutos. La ventana de la figura 6 es intuitiva, se pueden añadir, borrar y mover valores.

ARCHIVO DE TEXTO PARA EL ALMACENAMIENTO DE LOS DATOS

Para el almacenamiento de los datos, se debe crear un documento de texto en la ubicación deseada por el usuario. La creación del archivo se debe realizar desde las ventanas del sistema operativo. Véase la figura 7.

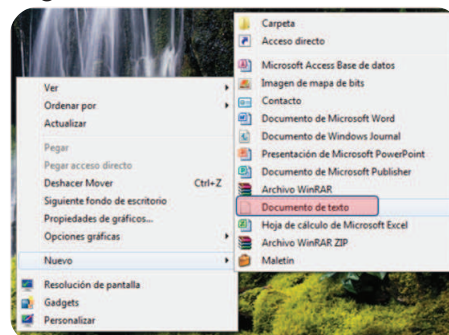


Figure 7: Creación del documento de texto.

Luego de haber creado el documento se pasa a la localización del archivo de texto por medio de la interfaz presionando sobre el botón que se muestra en la figura 8.



Figure 8: Abrir ventana de ubicación del archivo

Al hacer click sobre este botón, aparecerá una ventana en donde se debe buscar el archivo creado. Ver figura 9.

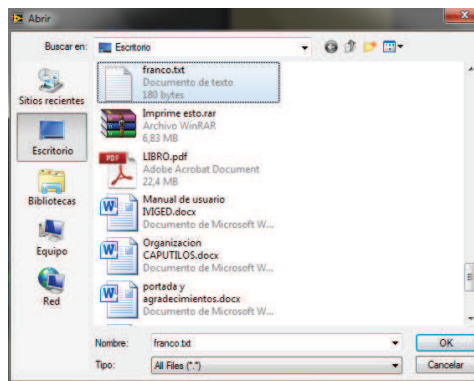


Figure 9: Ubicación del archivo de texto

Luego de seleccionar el archivo y presionar sobre el botón OK. El campo localización del archivo mostrara la ruta de ubicación del archivo como se en la figura 10.



Figure 10: Ruta de ubicación del archivo

LEDs INDICADORES DE LA INTERFAZ

Los indicadores DATO DEL MODEM, PETICION ALL y PETICION PARTICULAR se detallan en la figura 11.

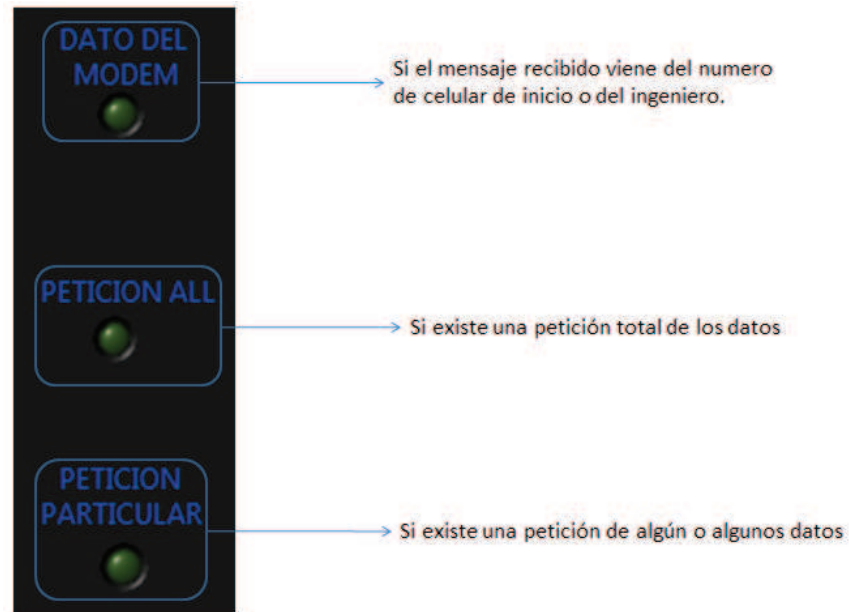


Figure 11: Descripción de los leds indicadores de la interfaz.

ASIGNACION DE LA HORA DE ENVIO DIARIO DE LOS DATOS

Para asignar el valor del campo Hora de envío en la figura 12, se debe ubicar el cursor encima del campo y hacer click, luego indicar la hora en formato militar a la cual el sistema va enviar la notificación diaria de todos los datos.



Figure 12: Asignación de la hora de envío

BOTON DE INICIO Y STOP DE LA INTERFAZ

El botón de inicio solo se debe presionar cuando todos los datos estén ingresados correctamente como lo muestra la figura 13. Con este se da inicio al programa, enviando toda la información ingresada a SISDAQ, como la configuración inicial del sistema para las notificaciones y demás. El botón de stop detiene el flujo del programa. Al detener el flujo del programa, no se responderán las solicitudes, ni tampoco se tendrá actualizada el valor de los datos en la interfaz de visualización de datos mostrada en la figura 14.

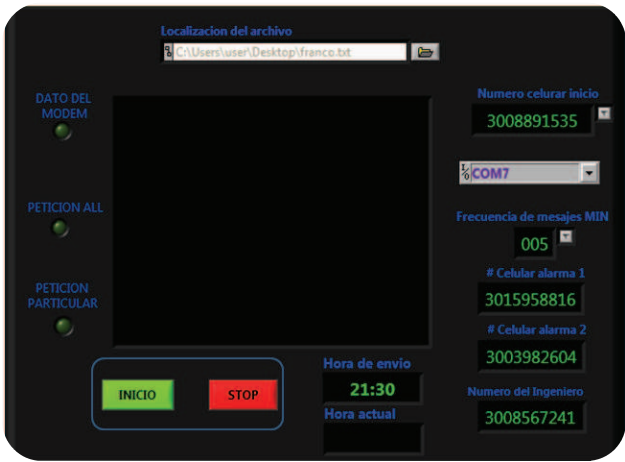


Figure 13: Botón de inicio y stop de la interfaz.

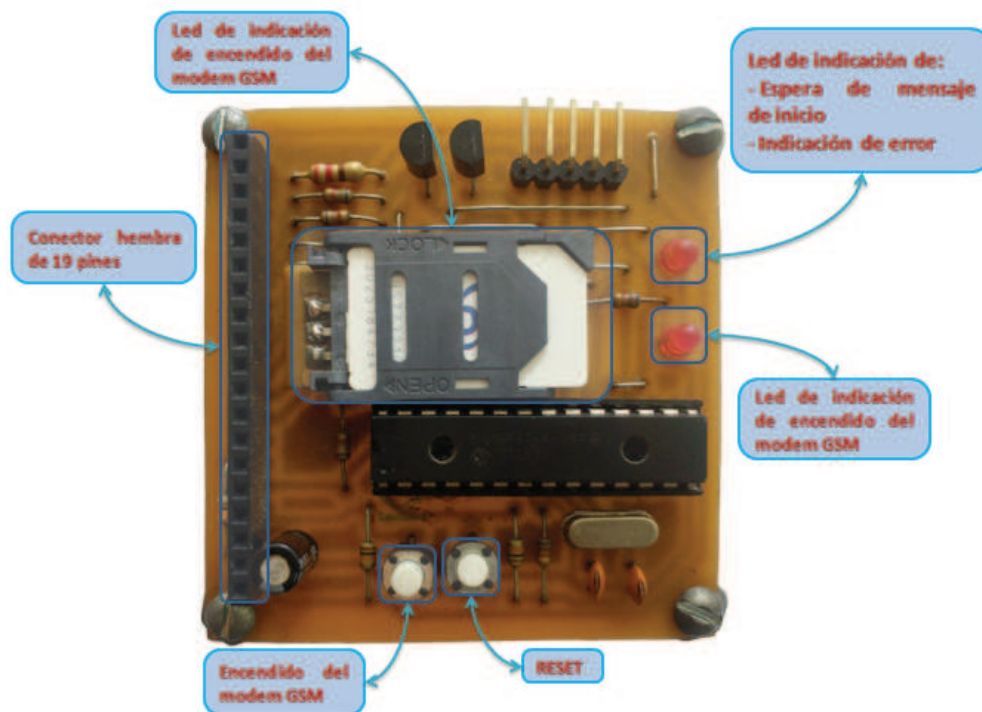


Figure 14: Interfaz de visualización de los datos.

ANEXO A5
MANUAL DE USUARIO SISDAQ

MANUAL DE USUARIO PARA LA APLICACIÓN SISDAQ

A continuación se describen los pasos para la implementación del sistema de monitoreo remoto. En la siguiente figura se muestra los dispositivos más relevantes en la aplicación SISDAQ.



NOTA: Para el correcto funcionamiento de la aplicación SISDAQ el usuario debe tener claro el significado de las cadenas de caracteres “XONN”, “XOFF” y “DATO” para la comunicación con el terminal de datos.

INSTRUCCIONES

1ero

Introducir SIMCARD en el socket para SIM ubicado en la parte superior ver figura 1. En la figura se observa la dirección en que se abre y asegura el socket para SIM. Una vez introducida la SIM debe que quedar asegurada de lo contrario

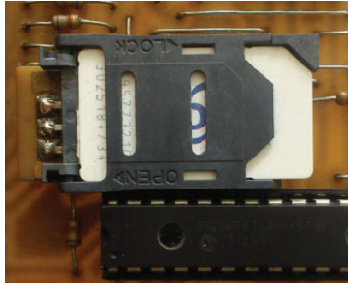


Figura 1

2do

Introducir el modem GSM con el conector de 19 pines macho en el conector hembra de 19 pines como se observa en la figura2.

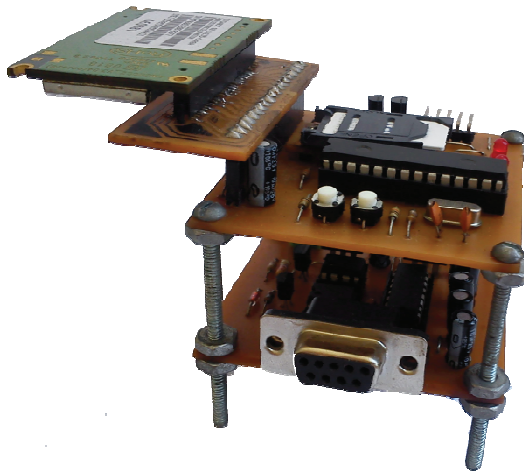


Figura 2

3ro

Conectar la antena al modem SIM300CZ como se muestra en la figura.

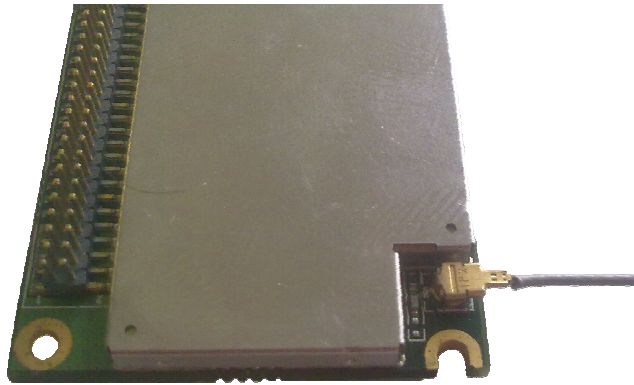


Figura 3

4to

Conecte el conector DB9 hembra del modulo SISDAQ al conector DB9 macho del Terminal de datos.

5to

Conectar la fuente de alimentación, incrustar el conector macho de la fuente al conector hembra, observe que el led sombreado en la figura empieza a parpadear, esto indica que el microcontrolador está esperando el encendido del modem.

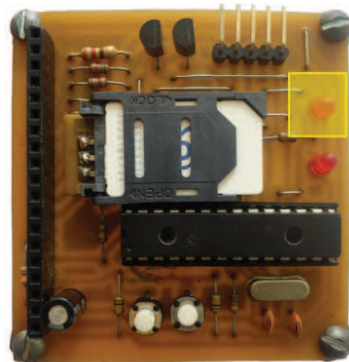


Figura 4

6to

Encender el modem GSM, para ello presione el pulsador que se muestra en la figura por más de 1,5 segundos, una vez encendido el modem verá que ambos led's parpadean con una frecuencia de 5 Hz esto se mantendrá durante 15 segundos y después de esto la aplicación está preparada para recibir el mensaje de inicio de la aplicación servidora.

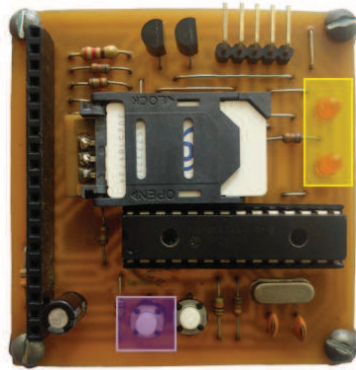


Figura 5