



Proyecto de fin de carrera

*Proyecto Adquisidor Remoto
Facultad de Ingeniería Universidad de la República
Montevideo, Uruguay
Julio 2008*

*Federico Blanco – Pablo Massaferro – Sebastián Torterolo
Tutores: Ing. Juan Pablo Oliver – Ing. Nicolás Antoniello*

1 Contenido

| | | |
|-------|--|----|
| 1 | Contenido | 2 |
| 2 | Introducción | 4 |
| 2.1 | Resumen | 4 |
| 2.2 | Créditos y agradecimientos..... | 4 |
| 2.3 | Organización de la documentación | 4 |
| 3 | Motivación | 6 |
| 4 | Objetivos..... | 7 |
| 5 | Características y funcionalidades..... | 8 |
| 5.1 | Características | 8 |
| 5.2 | Funcionalidades..... | 8 |
| 5.2.1 | Adquisición..... | 8 |
| 5.2.2 | Acceso a los datos | 9 |
| 5.2.3 | Configuración | 9 |
| 5.2.4 | Consola..... | 9 |
| 5.2.5 | Alarmas..... | 9 |
| 6 | Hardware | 10 |
| 6.1 | Descripción general | 10 |
| 6.2 | Placa de Adquisición: adq-vx | 11 |
| 6.2.1 | Características generales..... | 11 |
| 6.2.2 | Descripción general..... | 12 |
| 6.2.3 | Diseño..... | 14 |
| 6.2.4 | Consumo y autonomía | 46 |
| 6.2.5 | Costos..... | 48 |
| 6.2.6 | Conclusiones..... | 49 |
| 6.3 | Placa de Protecciones | 49 |
| 6.3.1 | Características generales..... | 49 |
| 6.3.2 | Descripción general..... | 50 |
| 6.2.3 | Diseño..... | 51 |
| 6.2.4 | Costos..... | 54 |
| 6.2.5 | Conclusiones..... | 54 |
| 6.4 | Gabinete | 55 |
| 6.4.1 | Características generales..... | 55 |
| 6.4.2 | Descripción general..... | 56 |
| 6.4.3 | Diseño..... | 57 |
| 6.4.4 | Costos..... | 59 |
| 6.4.5 | Conclusiones..... | 59 |
| 7 | Software | 60 |
| 7.1 | Descripción General..... | 60 |
| 7.2 | Arquitectura de Software..... | 60 |
| 7.3 | Sistema de Archivos | 62 |
| 7.4 | Comunicación entre procesos..... | 64 |
| 7.5 | Lenguajes de programación..... | 64 |
| 7.6 | Gestión de la aplicación: Programa Main.sh | 66 |
| 7.6.1 | Descripción | 66 |
| 7.6.2 | Sincronizar relojes: SincronizarHoras.sh | 68 |
| 7.6.3 | Actualización de configuración: ActualizarConf.sh..... | 68 |
| 7.6.4 | Watchdog de software: SoftWatchdog.sh..... | 68 |
| 7.6.5 | Finalización de procesos: Boogeyman.sh | 69 |
| 7.6.6 | Modo bajo consumo: Standby.sh | 69 |
| 7.7 | Comunicación | 70 |
| 7.7.1 | Descripción | 70 |
| 7.7.2 | Gestor de comunicación - CtlModem.sh | 71 |

| | |
|---|-----|
| 7.7.3 Receptor de SMS – SMSInt.sh..... | 73 |
| 7.7.4 Control de actividad del receptor de SMS – IfSMS.sh | 76 |
| 7.7.5 Conexión a Internet – Con2Int.sh | 77 |
| 7.7.6 Consola remota..... | 78 |
| 7.7.7 Otros programas utilizados..... | 80 |
| 7.8 Adquisición – Programa CtlAdq.sh..... | 81 |
| 7.8.1 Descripción | 81 |
| 7.8.2 Medida de frecuencia | 85 |
| 7.8.3 Medición de entradas analógicas | 86 |
| 7.8.4 Sensor serie..... | 86 |
| 7.9 Configuración..... | 87 |
| 7.9.1 Parámetros de configuración..... | 87 |
| 7.9.2 Configuración local..... | 89 |
| 7.9.3 Configuración remota | 90 |
| 7.10 Programas utilitarios | 93 |
| 7.11 Sistema operativo | 94 |
| 7.12 Compilación cruzada..... | 96 |
| 7.13 Modificaciones del U-Boot..... | 96 |
| 7.14 Imagen precompilada..... | 97 |
| 7.15 Conclusiones | 98 |
| 8 Incertidumbre en la medida..... | 99 |
| 8.1 Incertidumbre en la medida de frecuencia..... | 99 |
| 8.2 Incertidumbre en la medida de señales analógicas | 101 |
| 9 Pruebas de campo..... | 103 |
| 9.1 Introducción | 103 |
| 9.2 Sensores..... | 103 |
| 9.3 Calibración | 104 |
| 9.4 Conclusiones | 109 |
| 10 Costos de fabricación | 110 |
| 10.1 Materiales | 110 |
| 10.1.1 Materiales importados | 110 |
| 10.1.2 Materiales de plaza | 111 |
| 10.2 Soldado y montaje | 112 |
| 10.3 Costo total..... | 113 |
| 10.4 Cómo disminuir costos..... | 113 |
| 11. Costos de explotación..... | 115 |
| 12. Bibliografía y glosario..... | 117 |
| 12.1 Bibliografía..... | 117 |
| 12.2 Glosario | 118 |
| 13 Apéndices | 121 |
| 13.1 Apéndice A: Esquemáticos | 121 |
| 13.1.1 Esquemáticos del circuito impreso del Adquisidor | 122 |
| 13.1.2 Esquemáticos del circuito impreso de la Placa de Protecciones..... | 130 |
| 13.2 Apéndice B: Mapeo del conector de 60 pines | 135 |
| 13.3 Apéndice C: Lista de materiales..... | 137 |
| 13.4 Apéndice D: Contenido de CD | 139 |
| 13.5 Apéndice E: Manual de usuario..... | 139 |
| 13.6 Apéndice F: Manual de fabricación y desarrollo | 139 |

2 Introducción

2.1 Resumen

Este proyecto consistió en el diseño y la implementación de un adquisidor de datos remoto y multipropósito. Surgió a partir de la necesidad del IMFIA (Instituto de Mecánica de los Fluidos e Ingeniería Ambiental, Facultad de Ingeniería) de tener un adquisidor de acceso remoto para la medición de velocidad y dirección del viento.

El adquisidor fue diseñado para ser instalado en lugares de difícil acceso. Debía cumplir con las siguientes características: bajo consumo, acceso a los datos en forma remota, gestión y configuración remota, envío de alarmas, protecciones contra descargas atmosféricas y protecciones contra filtraciones de agua y polvo.

El adquisidor utiliza la red de telefonía celular para el envío de datos y alarmas, y la gestión remota a través de un módem GSM/GPRS. Posee una computadora embebida con sistema operativo Linux y varias entradas y salidas en diferentes estándares. Esto brinda una flexibilidad que permite usar el dispositivo para diversas aplicaciones.

2.2 Créditos y agradecimientos

Nuestras familias, Juan Pablo Oliver, Nicolás Antonello, Sofía García, Gabriela Antmann, Dave Hylands, Fernando Surraco, Leonardo Etcheverry, Ciro Mondueri, Sebastián Fernandez, José Cataldo, Valeria Durañona, José Miguel García, Pablo Cancela, Pablo Pais.

2.3 Organización de la documentación

La documentación está organizada en los siguientes capítulos y apéndices:

- **Introducción**
Resumen del proyecto, agradecimientos y organización de la documentación.
- **Motivación**
Justificación del proyecto.
- **Objetivos**
Objetivos generales del proyecto.
- **Características y funcionalidades**
Características y funcionalidades del adquisidor.

- **Hardware**

Diseño del hardware.

- **Software**

Diseño del software.

- **Incertidumbre de la medida**

Análisis de la incertidumbre de las medidas del adquisidor.

- **Pruebas de campo**

Resultados de los ensayos realizados.

- **Costos de fabricación**

Análisis del costo de fabricación del adquisidor.

- **Costos de explotación**

Análisis de los costos de operación para distintas estrategias de adquisición.

- **Referencias, bibliografía y glosario**

- **Apéndices**

Esquemáticos, Mapeo de GPIO, Lista de Materiales, Manual de Usuario, Manual de Fabricación.

3 Motivación

Este proyecto surge de la necesidad del instituto IMFIA (Instituto de Mecánica de los Fluidos e Ingeniería Ambiental) de tener adquisidores de datos que puedan ser gestionados remotamente. Actualmente el instituto cuenta con adquisidores (data logger) que son utilizados para recabar información de velocidad y dirección de viento en distintos puntos del país. Estos adquisidores presentan los siguientes problemas:

1. **Costos de operación:** Los datos adquiridos deben ser extraídos localmente. Esto implica costos de traslado y mano de obra.
2. **Confiabledad:** El equipamiento puede ser instalado en lugares lejanos y de difícil acceso. En caso de fallas inesperadas, se corre el riesgo de perder parte de la información debido a que no es posible conocer el estado del adquisidor en forma remota.
3. **Asistencia técnica:** No se cuenta con asistencia técnica local.
4. **Flexibilidad:** La mayoría de los equipos son rígidos frente a cambios en la estrategia y periodo de muestreo. Además son capaces de manejar una variedad muy acotada de sensores.

4 Objetivos

El proyecto consiste en el diseño e implementación de una adquisidor de datos que cumpla con las siguientes características:

- Bajo costo de fabricación.
- Bajo consumo: Autonomía mayor o igual a siete días con una batería de gel de 12V.
- Almacenamiento de los datos: Capacidad de almacenamiento de al menos un mes de datos en una memoria no volátil extraíble.
- Extracción local de datos.
- Configuración y acceso local.
- Envío de datos a través de Internet.
- Configuración y acceso a través de Internet.
- Entradas digitales y analógicas en formatos estándar.
- Medición de frecuencia.
- Capacidad de trabajar con las estrategias de adquisición estándar para dispositivos meteorológicos.
- Protección contra descargas atmosféricas.
- Protección contra filtraciones de agua y polvo IP54.
- El adquisidor debería ser lo más robusto posible a posibles fallas inesperadas.
- Dispositivo multipropósito: Flexible a la incorporación de nuevas funcionalidades de hardware y software.

5 Características y funcionalidades

5.1 Características

- Tensión de alimentación entre 6V y 18V
- Consumo en stand by 32mA @ 12V. Consumo en modo de adquisición continua 110mA @ 12V
- 5 entradas analógicas 12 bits, configurables entre 0-5V y 0-2.5V
- 8 entradas digitales 0-5V
- 4 salidas digitales 0-3.3V
- Bus i2c
- 1 salida relay AC/DC 60V 300mA
- 5V para alimentación de sensores
- 2 puertos RS-232
- 1 puerto USB host
- Protecciones contra sobretensiones, cortocircuitos y sobrecorrientes

5.2 Funcionalidades

5.2.1 Adquisición

El adquisidor desarrollado puede adquirir datos de señales analógicas con una resolución de 12 bits. Así también, puede adquirir frecuencias de hasta 1 KHz de señales digitales. Además posee un puerto serie RS-232 en donde se puede conectar un transductor que envíe datos para ser almacenados.

Cuenta con dos estrategias para tomar los datos. La primera estrategia almacena un dato por hora. Para obtener dicho dato se realiza un promedio de las muestras tomadas durante un cierto tiempo durante la hora. El resto de la hora el adquisidor permanece en estado de bajo consumo. En la segunda estrategia se toman datos de forma continua y se guardan sin realizar promedios.

Además, el adquisidor tiene la posibilidad de convertir los datos antes de ser almacenados mediante una recta de calibración configurable.

5.2.2 Acceso a los datos

Para poder extraer fácilmente los datos, se almacenan en una memoria FLASH extraíble. De esta forma, cuando se necesita descargar los datos del adquisidor se desconecta el dispositivo FLASH y se conecta a donde se requiera descargar.

Además cuenta con la capacidad de descargar los datos a través de Internet. El adquisidor envía los datos diariamente a un sitio FTP. De esta forma, accediendo a través de Internet al sitio FTP, el usuario accede a los datos adquiridos hasta el día anterior.

Por último, mediante una solicitud SMS se puede obtener la medida actual de los sensores en menos de 3 minutos.

5.2.3 Configuración

El adquisidor se puede configurar tanto local como remotamente. La configuración local se realiza a través de un programa. Este programa guía al usuario y no permite que se ingresen valores erróneos. La configuración remota se realiza desde un sitio Web con una interfaz amigable para el usuario. Luego se manda al adquisidor un mensaje de texto para pedirle que actualice su configuración con la información del sitio Web. Por último, se puede pedir al adquisidor mediante un mensaje de texto que mande la configuración que se encuentra en uso al sitio FTP.

5.2.4 Consola

Se puede acceder al adquisidor localmente conectando un cable al puerto serie y realizando una consola mediante el protocolo RS232.

Además puede ser accedido a través de Internet mediante el protocolo SSH. De esta forma se tiene total control del adquisidor en forma remota.

5.2.5 Alarmas

El adquisidor permite configurar valores límites inferiores y superiores para cada sensor. En el caso que los datos adquiridos excedan estos límites, se lanza una alarma, la cual se configura para enviarse a través de e-mail, SMS o FTP.

Además es posible consultar el valor actual de la fuente de alimentación mediante un mensaje de texto. El adquisidor devuelve con un mensaje de texto el valor de la fuente en aproximadamente 3 minutos.

6 Hardware

6.1 Descripción general

El diseño de hardware se dividió en cuatro partes: la placa de adquisición, la placa de protecciones eléctricas, la alimentación y el gabinete (Figura 1 - División del Hardware).

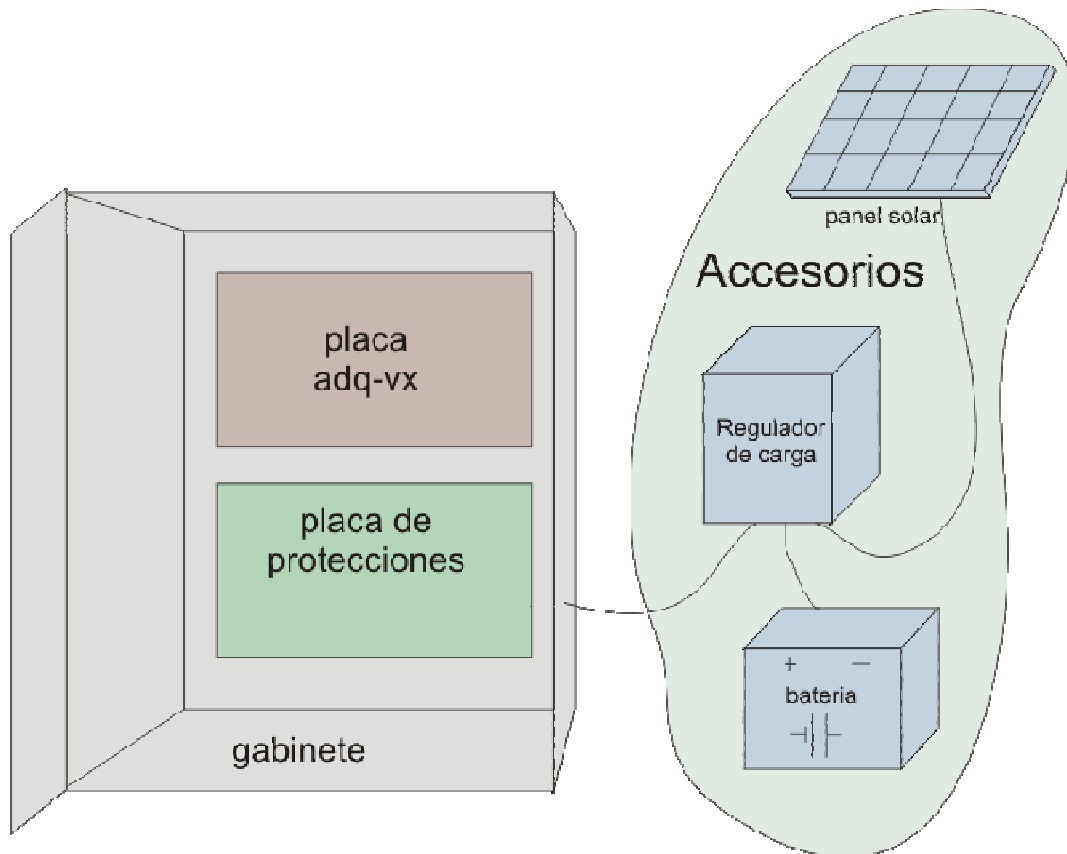


Figura 1 - División del Hardware

La placa de adquisición está compuesta por una computadora embebida, un módem GSM/GPRS y un conjunto de periféricos que permiten la interacción con el usuario y los sensores.

La placa de protecciones está constituida por elementos que dan un camino de baja impedancia a perturbaciones eléctricas. Además cuenta con fusibles que protegen el circuito frente a sobrecargas y cortocircuitos. Para su diseño se utilizó como referencia el proyecto de fin de carrera “Adquisidor de Velocidad y Dirección de Viento” de Fernando Caraballo, Luis Ferrari y Juan Pablo Oliver.

La alimentación está compuesta por una batería, un regulador de carga y un panel solar.

El gabinete está compuesto por una caja estanco y un conjunto de accesorios de fijación. Contiene al adquisidor, la placa de protecciones, la batería y el regulador de carga.

La división entre placa de adquisición y de protecciones fue realizada por motivos económicos. En primer lugar, en caso de no utilizar todas las entradas y salidas del adquisidor, se puede fabricar una placa de protecciones con menos componentes y superficie de impreso, disminuyendo los costos de producción considerablemente. En segundo lugar, si algún elemento de la placa de protecciones se dañara, es posible recambiar la placa de protecciones y no dejar fuera de servicio el adquisidor. De esta forma disminuye el costo de mantenimiento y explotación y mejora la seguridad operacional del equipamiento. Por último, la tecnología requerida para la fabricación del impreso del adquisidor es superior y más costosa que la necesaria para la placa de protecciones. Con esta división cada placa está fabricada con la tecnología mínima necesaria.

6.2 Placa de Adquisición: adq-vx

6.2.1 Características generales



Figura 2 - Foto de la placa adq-vx

- Tamaño 10cm x 16cm
- 4 puntos de fijación
- Tensión de alimentación entre 6V y 18V
- 5 entradas analógicas 12 bits configurables entre 0-5V y 0-2.5V
- 8 entradas digitales 0-5V
- 4 salidas digitales 0-3.3V
- Bus i2c
- 1 salida relay AC/DC 60V 300mA
- 5V para alimentación de sensores digitales
- 5V para alimentación de sensores analógicos
- 2 puertos RS-232
- 1 puerto USB host
- RTC con batería de backup
- 3 botones de propósito general
- Botón de reset
- Botón de apagado
- 3 LED
- Módem GSM/GPRS
- Computadora embebida Gumstix Verdex – 400MHz, 64MB de RAM, 16MB de FLASH.

6.2.2 Descripción general

El bajo costo y consumo eléctrico fueron dos aspectos especialmente cuidados en la elección de la arquitectura y los componentes.

A los efectos de presentar el hardware se divide el adquisidor en los módulos mostrados en la Figura 3.

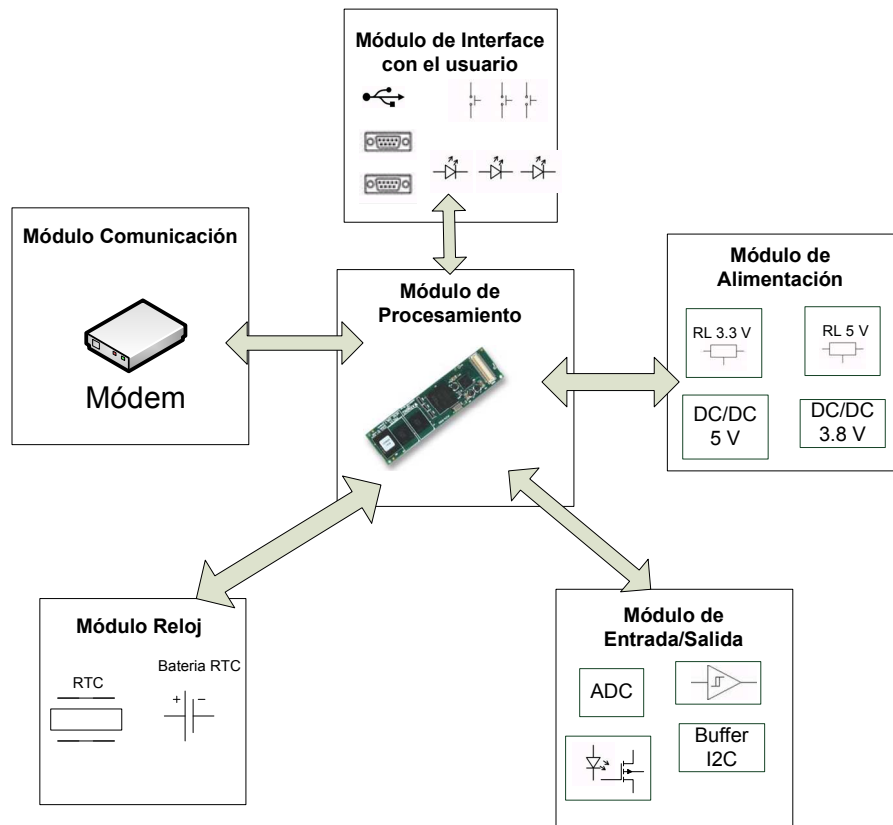


Figura 3 - División de hardware en módulos

El módulo de procesamiento es el encargado de controlar la aplicación y de procesar la información. Se implementó con una computadora embebida.

El módulo de comunicación es el que permite el envío de la información adquirida y la gestión remota del adquisidor. Dadas las características de los lugares habituales en donde el cliente instala los adquisidores, se decidió implementar este módulo utilizando la red de telefonía celular con un módem GSM/GPRS.

El módulo de entradas y salidas contiene los distintos elementos necesarios para implementar una interfaz entre el procesador y el exterior.

El módulo reloj es el conjunto batería-RTC que le permite al sistema conservar la hora durante períodos sin alimentación.

El módulo de interfaces de usuario está compuesto por los botones, LEDs, puertos RS-232 y el puerto USB.

El módulo de alimentación adapta la tensión de entrada y alimenta a los circuitos del adquisidor.

Por último, el circuito impreso es el que interconecta los distintos módulos.

A continuación se analiza con detalle el diseño de cada módulo.

6.2.3 Diseño

6.2.3.1 Procesamiento

6.2.3.1.1 Descripción

El módulo de procesamiento es el corazón del adquisidor. Es el encargado de controlar la aplicación. Debido a la flexibilidad requerida se decidió utilizar una computadora embebida. Se buscaron computadoras de bajo costo y consumo, con varias entradas y salidas de propósito general, con puerto USB host, puertos serie y bus i2c o SPI. También se buscaron computadoras que tuvieran una comunidad importante de usuarios para poder intercambiar información.

6.2.3.1.2 Computadora Embebida: Gumstix Verdex

Para el módulo de procesamiento se eligió la computadora embebida Verdex XM4 de la empresa Gumstix. A continuación se listan sus principales características:

- Tensión de alimentación entre 3.6V y 5V
- Bajo costo
- Dimensiones 80mm x 20mm x 6.3mm
- Peso 8 gramos
- Procesador PXA270 de 400MHz marca Marvell
- 64MB de memoria SDRAM
- 16MB de memoria FLASH
- Conector Hirose de 60 pines
- Conector Molex de 120 pines
- Conector de cable plano de 24 pines
- 120 entradas y salidas de propósito general
- RTC con alarma
- Puerto USB host
- 3 puertos serie

- Bus i2c
- Consumo a máxima carga del procesador menor a 250mA
- Consumo en estado inactivo menor a 50mA
- Sistema Operativo Linux
- Disponibilidad de varias placas de expansión que permiten construir plataformas de desarrollo

En la Figura 4 se muestra una foto de la computadora.



Figura 4 - Computadora embebida Gumstix Verdex

Para la etapa de desarrollo se utilizó la placa de expansión *console-vx* de Gumstix. Esta placa tiene los tres puertos serie con lógica TTL, el bus i2c y varias entradas y salidas.

Para conectar la computadora al adquisidor se utilizó el conector de 60 pines. La Figura 5 muestra cómo se utilizaron las señales disponibles en el conector.

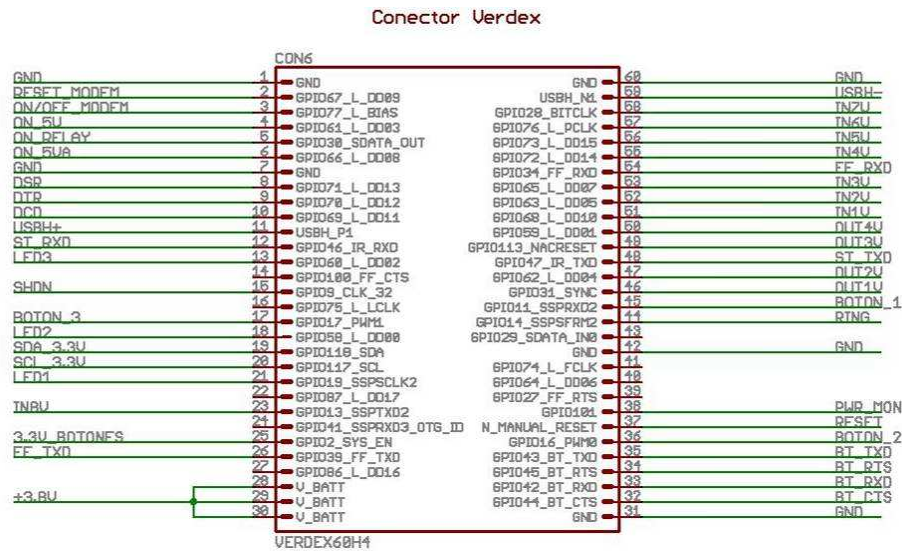


Figura 5 - Distribución de las señales en el conector de 60 pines de la computadora embebida

6.2.3.2 Comunicación

6.2.3.2.1 Descripción

El módulo de comunicación es el encargado del envío periódico de datos y el que permite la gestión remota. Dadas las características de los lugares donde va a operar el adquisidor se decidió utilizar un módem GSM/GPRS. Se buscaron módems de bajo consumo, compatibles con todos los operadores de telefonía celular en Uruguay.

6.2.3.2.2 Módem GSM/GPRS: Telit GM862

El módulo de comunicación está compuesto por el módem *GM862* de la empresa Telit. A continuación se listan sus principales características:

- Tensión de alimentación entre 3.4V y 4.2V
- Cuatro bandas 850/900/1800/1900MHz
- Potencia de salida 2W para 850/900MHz y 1W para 1800/1900MHz
- Control mediante comandos AT
- Puerto serie CMOS 2.8V
- Dimensiones 43.9 x 43.9 x 6.9 mm
- Peso 19 gramos

En la Figura 6 se muestra una foto del módem.



Figura 6 - Módem Telit GM862 GSM/GPRS

En la Figura 7 se muestra el diagrama de conexión del módem.

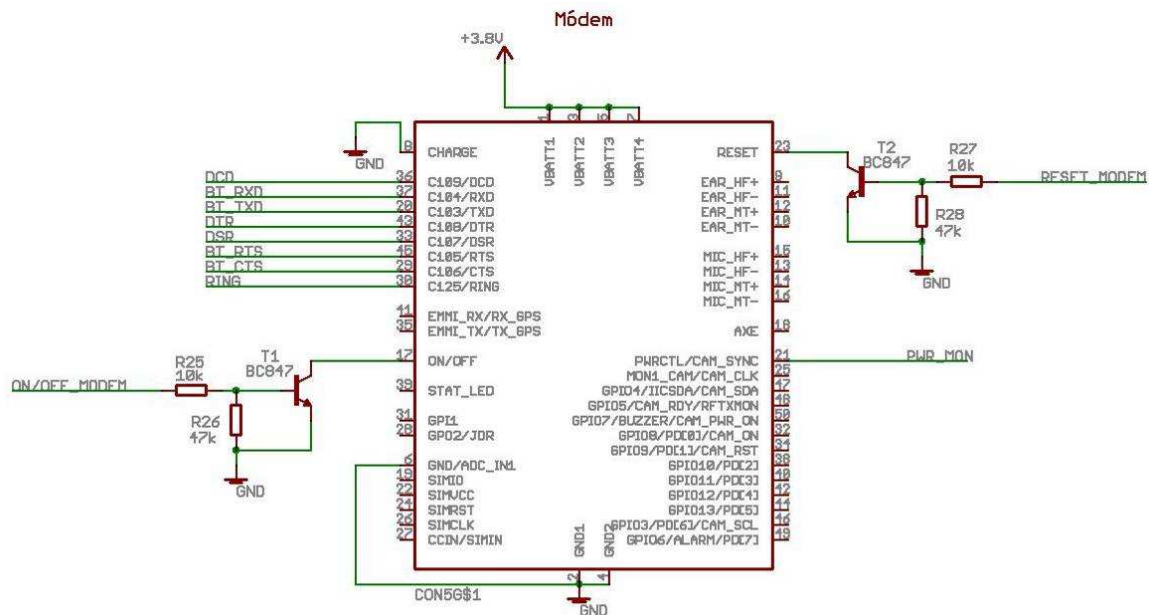


Figura 7 – Distribución de las señales del conector del módem

Para el encendido y apagado y el reseteo del módem se utilizaron transistores BJT según lo recomendado por el fabricante. Las líneas BT_RXD, BT_TXD, BT_RTS y BT_CTS se usan para la transmisión de datos entre el módem y el procesador. La línea RING se usa para avisar la llegada de un SMS. Esta línea saca al procesador del modo de bajo consumo. La línea DCD se usa para saber si el módem encontró una portadora. La línea DTR se usa para que el procesador le indique al módem que salga del modo de bajo consumo. La línea DSR se usa para indicarle al procesador que el módem está listo para recibir datos del puerto serie. La línea PWR_MON se usa para saber si el módem está encendido o apagado.

Se eligió una antena PCB de cuatro bandas con conector MMCXS coaxial de 50ohms.

6.2.3.3 Entradas y salidas

6.2.3.3.1 Descripción

Para determinar el tipo y número de entradas y salidas se evaluaron los distintos usos del adquisidor. Se definió que el número mínimo de entradas debía ser cuatro del tipo digital y una del tipo analógico. El cliente no solicitó que el adquisidor tuviera salidas. Sin embargo, por tratarse de un dispositivo multipropósito, se consideró que el adquisidor debía tener salidas digitales. Existen sensores que se alimentan en 12V. Se agregó una salida relay para alimentar estos sensores directamente desde la batería. Con el fin de no limitar el número y tipo de dispositivos a conectar, se hizo disponible el bus i2c desde el exterior. Las entradas y salidas se resumen en la siguiente lista:

- 5 entradas analógicas
- 8 entradas digitales
- 4 salidas digitales
- 1 salida relay
- Bus i2c

Estas señales, junto con la alimentación de los sensores, están disponibles en un conector IDE de 40 pines, del tipo utilizado en discos duros IDE. Las placas de adquisición y de protecciones se interconectan con un cable plano de 40 conductores AWG 28.

6.2.3.3.2 Entradas analógicas

El elemento más importante de este módulo es el convertor analógico digital. Existen varios tipos de convertidores que, entre otras cosas, se distinguen por el método de conversión, el número de bits, el número de entradas, la tensión de trabajo, la velocidad de muestreo y el formato de los datos digitales.

La frecuencia de muestro de diseño del adquisidor es del orden de los Hertz. Por este motivo se decidió utilizar un convertor de aproximaciones sucesivas. De forma de tener una resolución adecuada, la búsqueda se restringió a convertidores de más de 10 bits. También se dio preferencia a los convertidores de más de una entrada. Por último, se buscó un convertor con salida digital serie i2c con el objetivo de minimizar el uso de entradas del procesador y de simplificar el software. Dadas estas restricciones se

eligió el conversor analógico digital *ADS7828* de *Texas Instruments*¹. A continuación se listan las principales características del conversor:

- Conversor de aproximaciones sucesivas de 12 bits
- Canales: 8
- Frecuencia de muestro: 50kHz
- Tensión de trabajo: 2.7V – 5V
- Referencia interna de 2.5V
- Salida digital i2c

En la Figura 8 se muestra el circuito analógico.

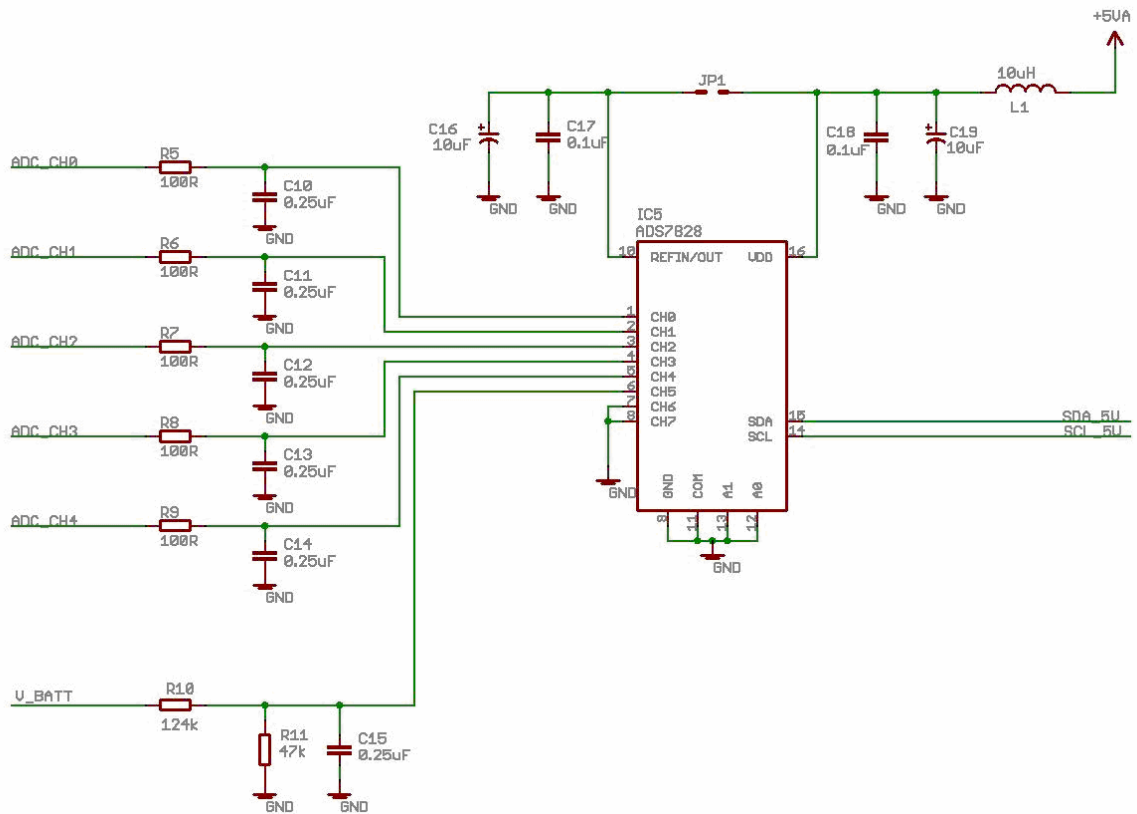


Figura 8 - Diseño del circuito analógico

Tan importantes como el conversor son los circuitos auxiliares. Estos son los que garantizan que el mismo funcione de acuerdo a lo especificado por el fabricante. En

¹ <http://focus.ti.com/lit/ds/symlink/ads7828.pdf>

este caso se tuvo especial cuidado en el diseño del circuito de adaptación de las señales analógicas y el de referencia de tensión. Para el diseño de estos circuitos se utilizó como material de referencia los siguientes artículos:

- Data loggin goes wireless²
- Signal acquisition and conditioning³

Para alimentar los circuitos analógicos se escogió un regulador lineal para la referencia del conversor. Esto es así porque los reguladores lineales tienen un excelente rechazo al ruido de entrada y un muy bajo ripple a la salida. Para obtener una medida de 12 bits de resolución, el ripple de la fuente debía ser menor a 0.6mVpp. El ripple de salida del regulador lineal elegido es menor a 0.1mVpp. Se puede observar en la Figura 8 que próximo al conversor se agregó en la alimentación un filtro pasa bajos. El objetivo de este filtro es eliminar ruidos de alta frecuencia provenientes del regulador lineal o ruidos captados por la línea entre el regulador y el conversor. Se agregó un jumper que permite usar la referencia interna del conversor.

Generalmente los conversores necesitan un circuito de acondicionamiento de la señal de entrada. Se debe asegurar que la señal varíe dentro del rango de conversor. Además es necesario aplicar un filtro para eliminar las frecuencias fuera del rango de interés. Por último, es necesario desacoplar la impedancia de salida del sensor de la impedancia de entrada del conversor para evitar que afecte la medida.

El conversor posee un circuito de muestreo y retención encargado de retener el valor de tensión de entrada durante todo el proceso de conversión. De forma simplificada este circuito consiste en una llave y un capacitor.

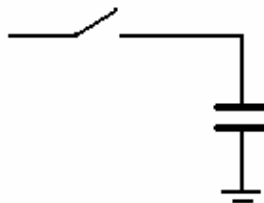


Figura 9 - Esquema del circuito de muestreo y retención

En la primera parte del ciclo de conversión se cierra la llave y se carga el capacitor con la tensión de entrada. Cuando la impedancia de salida del sensor es demasiado

² http://www.eetindia.co.in/ARTICLES/2007APR/PDF/EEIOL_2007APR10_NETD_EMS_TA.pdf

grande, el tiempo de carga del capacitor puede no ser suficiente. En convertidores con varios canales este problema se ve agravado debido a que se puede muestrear de forma consecutiva un canal a tierra y otro canal a la máxima tensión manejable por el dispositivo.

La hoja de datos del convertidor no especifica el tiempo de carga del capacitor. Sin este valor no se puede estimar la impedancia máxima de salida admitida para los sensores. De hojas de datos de convertidores de similares características, el tiempo de carga del capacitor de muestreo es aproximadamente 250ns.

El valor máximo de resistencia para que el capacitor se cargue a menos de 1/2LSB es:

$$R_{\max} = \frac{t}{C \ln(2^{13})}$$

Donde t es el tiempo de carga y C es el valor del capacitor de muestreo. El capacitor del convertidor es de 25pF. Entonces si no se desea utilizar un circuito de desacople la resistencia de salida máxima de los sensores debe ser menor a 1.1kΩ. Dado que los sensores a utilizar pueden exceder este valor de impedancia de salida, fue necesario proveer un circuito de desacople.

Durante la etapa de diseño se realizaron pruebas que confirmaron la necesidad de este circuito de adaptación. Se construyó una placa de pruebas con un convertidor *ADS7828*.

El tipo más común de sensores analógicos a utilizar son potenciómetros de 10kΩ. El equivalente Thévenin es una resistencia variable entre 0Ω y 2.5kΩ. Con una pila y una resistencia de 2.5kΩ se simuló una transición de 0V a 3V. Se compararon las medidas del adquisidor con las de un multímetro. La Figura 10 muestra el circuito de prueba.

³ <http://focus.ti.com/lit/ml/slap082/slap082.pdf>

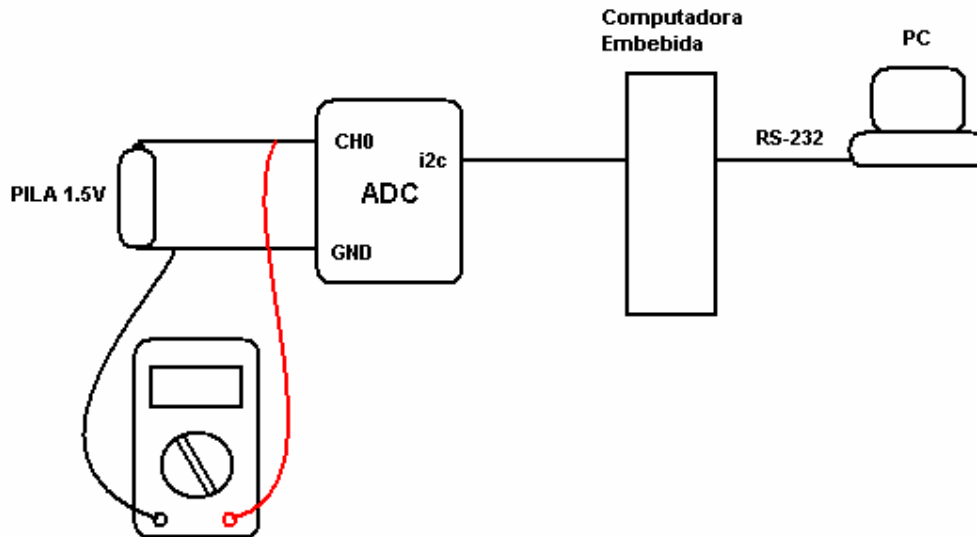


Figura 10 - Circuito de prueba del ADC

El procedimiento fue el siguiente:

- Se puso el canal 0 a tierra
- Se adquirieron varias medidas del canal 0
- Se conectó al batería al canal 0 a través de la resistencia
- Se adquirieron varias medidas del canal 0

Se repitió el procedimiento para la batería conectada directamente a la entrada sin la resistencia. Los resultados se resumen en la Tabla 1.

| Condición | Lectura del Conversor | Lectura del Multímetro |
|-------------------------|-----------------------|------------------------|
| Canal a tierra | 0mV | 0mV |
| Batería con resistencia | 1148mV | 1185mV |
| Batería sin resistencia | 1179mV | 1185mV |

Tabla 1 - Resultado de las pruebas del circuito de muestro y retención

Se puede ver que la resistencia hace que la constante de tiempo del circuito sea demasiado grande para la ventana de muestreo del convertidor. Las pruebas muestran que si no se utiliza algún tipo de adaptación de la señal de entrada, el circuito de muestreo y retención no es capaz de funcionar adecuadamente.

La alternativa más común es usar amplificadores operacionales para lograr esta adaptación. Sin embargo agregar amplificadores operacionales implica complicar el diseño y aumentar los costos. Además el rango de salida de los amplificadores

operacionales es distinto al rango de alimentación. Esto último introduce un error en la medida. Aprovechando las características del tipo de señales a manejar, se decidió utilizar capacitores para lograr esta adaptación.

Para calcular el valor mínimo del capacitor de desacople se plantea el circuito de la Figura 11, R_{mux} y C_{SH} son la resistencia de llave y el capacitor del circuito de muestreo y retención respectivamente y C_1 es el capacitor de desacople. Se estudia el circuito en el instante en que se cierra la llave.

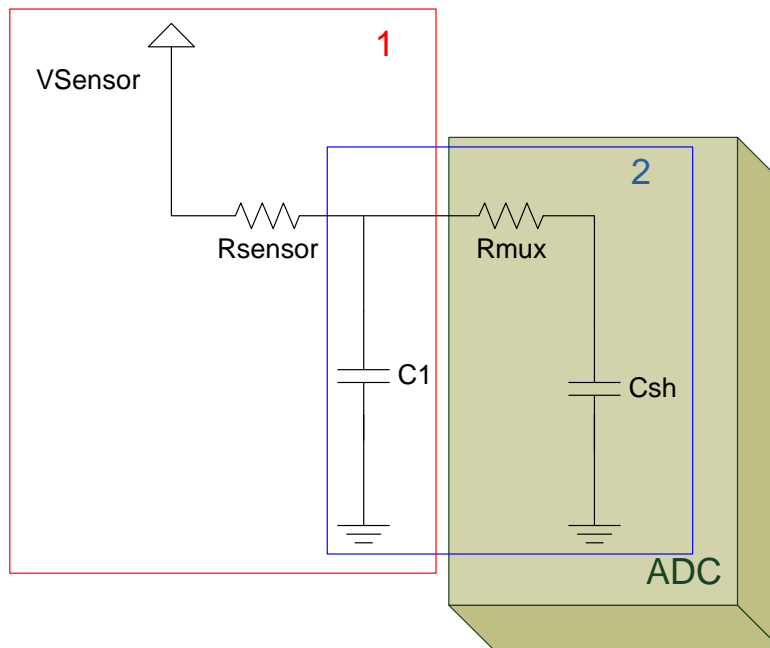


Figura 11 - Interacción entre sensor y circuito de muestreo y retención

Asumiendo que el capacitor de desacople es mucho más grande que el capacitor de muestreo, el sub-circuito 1 tiene una constante de tiempo mucho mayor que la del sub-circuito 2. Por este motivo sólo se analiza el sub-circuito 2. Resolviendo el circuito se llega a que la tensión en el capacitor C_{SH} cumple con la ecuación 1.

$$\frac{\partial v_{SH}(t)}{\partial t} = -v_{SH}(t) \frac{C_1 + C_{SH}}{C_1 C_{SH} R_{mux}} + \frac{V_{10} C_1 + V_{SH0} C_{SH}}{C_1 C_{SH} R_{mux}}$$

Ecuación 1

Entonces la constante de tiempo del sistema es $\frac{C_1 C_{SH} R_{mux}}{C_1 + C_{SH}}$.

Si C_1 es mucho mayor que C_{SH} , entonces la constante se puede aproximar por $R_{mux} C_{SH}$. Esta es la constante de tiempo inherente al conversor y por lo tanto es lo

suficientemente chica como para cargar el capacitor de muestreo en el tiempo de apertura. Entonces si el capacitor de desacople es lo suficientemente grande se logra desacoplar la entrada del conversor del sensor.

El valor mínimo para el capacitor de desacople se determina de manera que éste tenga una variación en su tensión de menos de 1/2LSB cuando el capacitor de muestreo está descargado y el capacitor de desacople está cargado a la tensión máxima. La ecuación 2 muestra la relación entre la variación de la tensión del capacitor de desacople y la tensión inicial para las condiciones descritas.

$$\Delta V_1 = V_{10} \frac{C_{SH}}{C_1 + C_{SH}}$$

Ecuación 2

Siendo la máxima variación de tensión admisible 0.6mV, el mínimo valor del capacitor de desacople es 0.21uF.

Esta solución introduce un nuevo problema. Como los sensores del cliente tienen una impedancia de salida variable, al agregar este capacitor se tiene un filtro pasa bajos con frecuencia de corte variable. Por ejemplo, para el potenciómetro de 10kΩ, la resistencia de salida varía entre 0Ω y 2.5kΩ. Esto implica una frecuencia de corte variable teórica desde infinito hasta 300Hz. Para limitar la frecuencia de corte se decidió agregar una resistencia en serie de 100Ω. Esto acota la frecuencia máxima de corte a 7.5kHz. Este rango de frecuencias es aceptable para la aplicación del cliente. Como contrapartida a la sencillez del diseño se tiene una frecuencia de corte variable.

Algo muy importante a destacar de la solución es que la tensión de referencia del conversor es la misma utilizada para alimentar los sensores. En el caso de los sensores implementados con potenciómetros esto implica que el error introducido por concepto de conversiones de magnitudes es cero. No es necesario suponer un valor de tensión de referencia, el conversor simplemente mide un porcentaje del valor máximo posible del sensor.

Para validar los razonamientos expuestos se repitieron las pruebas con un capacitor de desacople de 10uF. Los resultados se resumen en la Tabla 2.

| Condición | Lectura del Conversor | Lectura del Multímetro |
|-------------------------|-----------------------|------------------------|
| Canal a tierra | 0mV | 0mV |
| Batería con resistencia | 1179mV | 1185mV |
| Batería sin resistencia | 1179mV | 1185mV |

Tabla 2 - Resultado de las pruebas del circuito de muestreo y retención con capacitor de desacople

En contraste con las primeras pruebas, no hay diferencias entre las medidas con y sin resistencia. Eso muestra que el capacitor desacopla los circuitos.

Se eligieron capacitores de 0.27uF para el circuito de desacople. Junto con la resistencia serie de 100Ω determinan un filtro pasa bajos con una frecuencia máxima de corte de 5.9kHz. La frecuencia de corte mínima la determina el sensor a utilizar. Para el caso de un potenciómetro de 10kΩ, la frecuencia de corte varía entre 235Hz para el punto medio del sensor y 5.9kHz para los extremos.

Como lo indica la Figura 8 para la medida de tensión de batería se utilizó una de las entradas del conversor con un divisor resistivo y un capacitor de desacople. Se utilizaron valores grandes de resistencia para minimizar el consumo. Esto determina una frecuencia de corte muy baja para el filtro, lo cual es adecuado para el tipo de medida.

6.2.3.3 Entradas y salidas digitales

Para no conectar las salidas del procesador directamente al exterior se decidió agregar algún tipo de interfaz. Esta interfaz debía proveer protección eléctrica y filtrar posibles oscilaciones del sensor por fuera de los márgenes de ruido del procesador. La búsqueda se restringió a circuitos Schmitt-Trigger implementados con una tecnología CMOS de 3.3V tolerante con niveles TTL de 5V. Además el consumo debía ser bajo.

Se eligió el Schmitt-Trigger *SN74AHC14* de Texas Instruments⁴. A continuación se listan sus principales características:

- 6 canales
- Schmitt-Trigger inversor

⁴ <http://focus.ti.com/lit/ds/symlink/sn74ahc14.pdf>

- Tensión de alimentación entre 2V y 5.5V
- Tensión de entrada entre 0V y 5.5V
- Tensión de salida entre 0V y Vcc
- Consumo menor a 20uA
- Consumo por entrada menor a 1uA

En la Figura 12 se muestra el circuito de entradas y salidas.

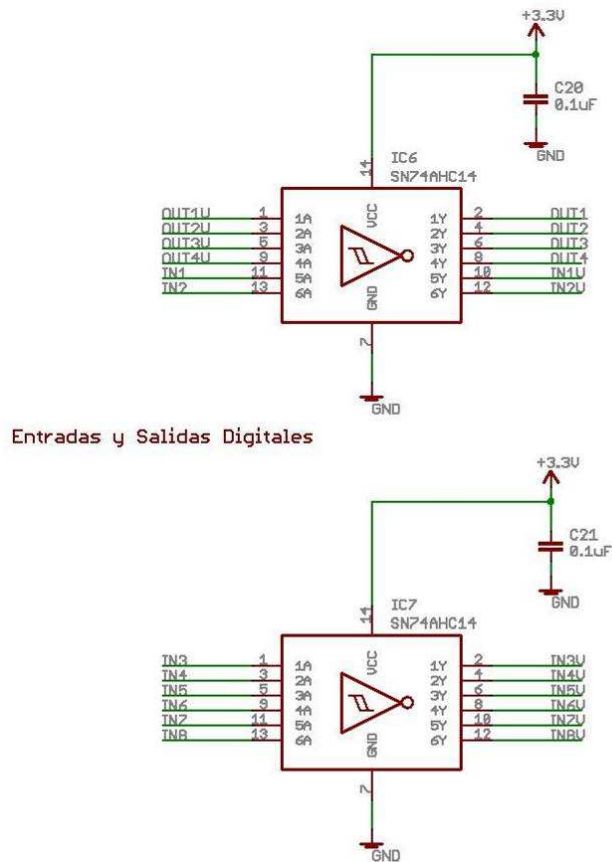


Figura 12 - Diseño de las entradas y salidas digitales

Se alimentaron estos integrados en 3.3V de forma de tener en la entrada del procesador una tensión de hasta 3.3V.

Las entradas del adquisidor pueden manejar señales de hasta 5V y las salidas son de 3.3V. Los Schmitt-Trigger invierten la señal.

6.2.3.3.4 Salida Relay

La motivación de disponer de una salida relay surgió de la necesidad de alimentar sensores en 12V. El consumo de estos sensores es importante, por lo que se debía proveer algún mecanismo para encenderlo y apagarlo. Estos sensores han de conectarse directamente a la batería a través de la salida relay. Por su bajo consumo se buscó un relay de estado sólido.

En la Figura 13 se muestra el diseño de la salida relay.

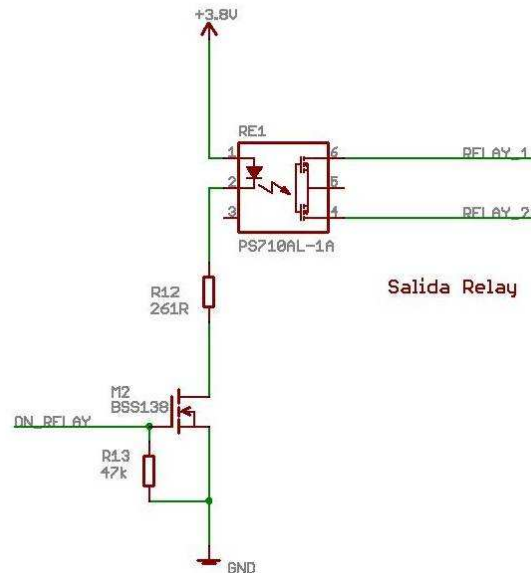


Figura 13 - Diseño de la salida relay

Se eligió el relay PS710AL de California Eastern Laboratorios⁵. A continuación se listan sus principales características:

- Resistencia 0.1Ω
- Corriente máxima de 1.8A
- Tensión máxima 60V
- AC/DC

Se conectó el dispositivo de forma que pueda manejar corriente continua y alterna. En esta configuración el dispositivo soporta hasta 1.8A. Sin embargo, dado que la corriente del relay pasa a través de dos conductores AWG 28, ésta se debe limitar a 300mA.

⁵ <http://www.cel.com/pdf/datasheets/ps710a1a.pdf>

La corriente de excitación nominal del relay es 10mA. Esta corriente es mayor que los 5mA que pueden manejar las salidas del procesador. Por lo tanto se tuvo que alimentar el LED de excitación del relay desde la fuente de 3.8V a través de un MOSFET. La resistencia del circuito de excitación se dimensionó para obtener la corriente de excitación nominal. Como para el resto de las llaves del adquisidor, se agregó una resistencia de pull-down para evitar encendidos involuntarios en el arranque del sistema.

6.2.3.3.5 Bus i2c

Para no limitar el número y tipo de dispositivos a conectar, se hizo disponible el bus i2c desde el exterior. Fue necesario entonces agregar un repetidor i2c de forma que las señales del procesador no estuvieran directamente accesibles desde el exterior. Además este repetidor permite interconectar la salida i2c del convertor analógico digital con la salida i2c del procesador. El convertor tiene niveles lógicos en 5V y el procesador en 3.3V.

En la Figura 14 se muestra el repetidor i2c.

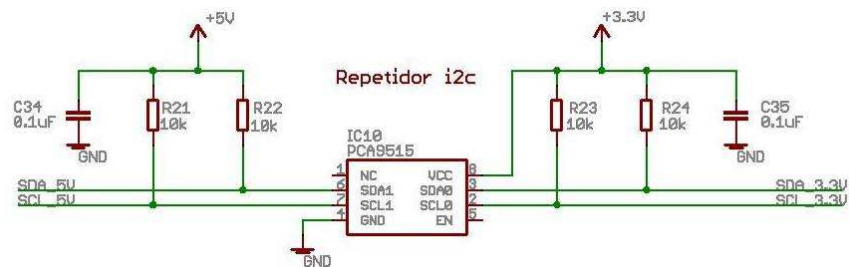


Figura 14 - Diseño del repetidor i2c

Se utilizó el repetidor i2c *PCA9515A* de Texas Instruments⁶. A continuación se listan sus principales características:

- Tensión de alimentación entre 2.3V y 3.6V
- Tolerante a tensiones de hasta 5.5V
- Consumo menor a 4mA
- Protección ESD de 200V

Para el bus i2c se utilizaron resistencias de pull-up de 10kΩ.

⁶ <http://focus.ti.com/lit/ds/symlink/pca9515a.pdf>

6.2.3.3.6 Conector IDE

Las placas de adquisición y de protecciones se conectan a través de un cable plano de 40 conductores AWG 28 del tipo utilizado en computadoras. Se distribuyeron las señales en el cable plano de forma de disminuir las interferencias. Se colocó una línea de tierra entre cada señal analógica. La Figura 15 muestra esta distribución.

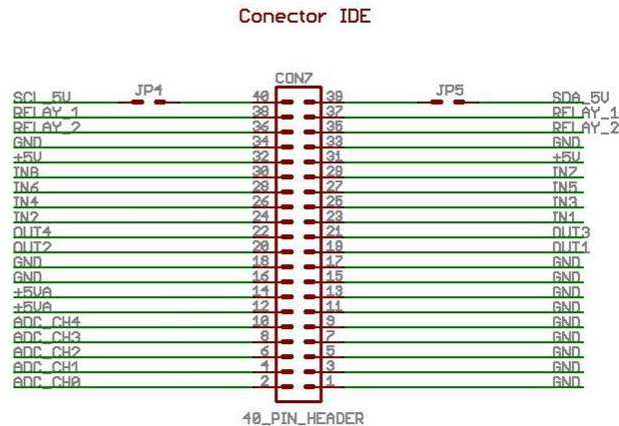


Figura 15 - Distribución de señales en el conector IDE

6.2.3.4 Reloj

6.2.3.4.1 Descripción

Tan importante como las medidas de los sensores es la medida del tiempo. Se debió proveer un mecanismo para asegurarse que la hora del adquisidor fuera correcta, incluso frente a períodos sin alimentación. Se analizaron las siguientes alternativas:

- Actualizar la hora desde Internet
- Usar el RTC del procesador
- Usar la red celular para sincronizar el reloj
- Usar un RTC externo

Usar el RTC del procesador fue descartado porque, debido a problemas de diseño de la computadora embebida, no es posible mantener la hora en períodos sin alimentación, a pesar de usar una batería de respaldo. Se optó por usar un RTC externo por ser la solución más confiable. Fue necesario entonces agregar un reloj externo con una batería auxiliar.

6.2.3.4.1 RTC externo

En la Figura 16 se muestra el RTC (Real Time Clock) utilizado.

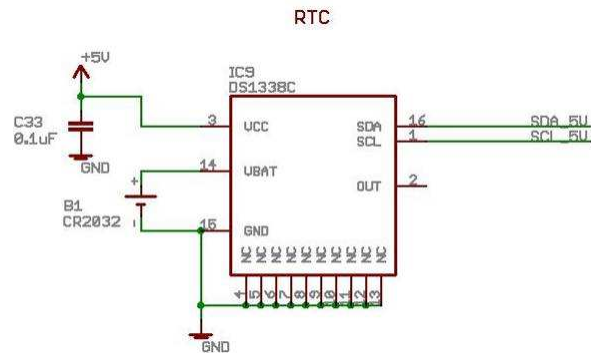


Figura 16 - Diseño del RTC

Se restringió la búsqueda a relojes con salida i2c y oscilador incorporado. Se dio preferencia a aquellos relojes que cuentan con controladores para Linux. Se utilizó el RTC *DS1338C-33* de Maxim⁷.

A continuación se listan sus principales características:

- Alimentación entre 3.3V y 5.5V
- Consumo en estado activo menor a 325uA
- Consumo en estado inactivo menor a 100nA
- Salida digital i2c
- Oscilador incorporado
- Back-up con batería de litio de 3V

Para mantener la hora en períodos sin alimentación se agregó un receptáculo para baterías CR2032. Estas baterías son las usadas en computadoras para el respaldo del RTC. Con una batería de 200mAh se logra una autonomía que excede largamente la vida útil del adquisidor.

6.2.3.5 Interfaces con el usuario

6.2.3.5.1 Descripción

Este módulo contiene dos puertos serie RS-232, un puerto USB host, tres botones genéricos, un botón de apagado, un botón de reset y tres LEDs genéricos.

⁷ <http://datasheets.maxim-ic.com/en/ds/DS1338-DS1338Z.pdf>

Uno de los puertos serie permite al usuario establecer una consola con una computadora. Por este puerto se realiza el mantenimiento del sistema. El segundo puerto serie no tiene una función específica y puede ser utilizado, por ejemplo, para conectar sensores con salida RS-232.

El puerto USB host se utiliza para conectar un dispositivo externo de almacenamiento de datos.

Los botones y LEDs genéricos permiten al usuario obtener información del sistema sin tener que conectar una computadora. Su objetivo principal es facilitar la tarea del instalador. Además se agregó un botón para apagar de forma segura el equipo y un botón de reset.

Para el diseño del puerto USB y los puertos RS-232 se utilizaron como referencia los esquemáticos de las placas de expansión de la empresa Gumstix⁸.

6.2.3.5.2 Puerto USB

En la Figura 17 se muestra el diseño del puerto USB.

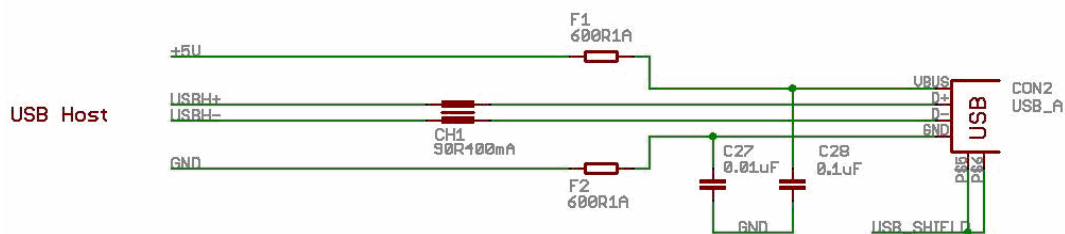


Figura 17 - Diseño del puerto USB

Se utilizaron ferritas y capacitores para filtrar ruidos en las líneas de alimentación y tierra. Como el puerto USB es diferencial, se agregó un filtro de rechazo al modo común entre las líneas de datos. La necesidad de estos componentes surgió del estudio de los esquemáticos de las placas de expansión de la empresa Gumstix.

El conector USB es hembra tipo A.

6.2.3.5.3 Puertos Serie RS-232

En la Figura 18 se muestra el diseño de los puertos serie.

⁸ <http://pubs.gumstix.org/boards/CONSOLE/VX/PCB10003-R1851/PCB10003.sch>

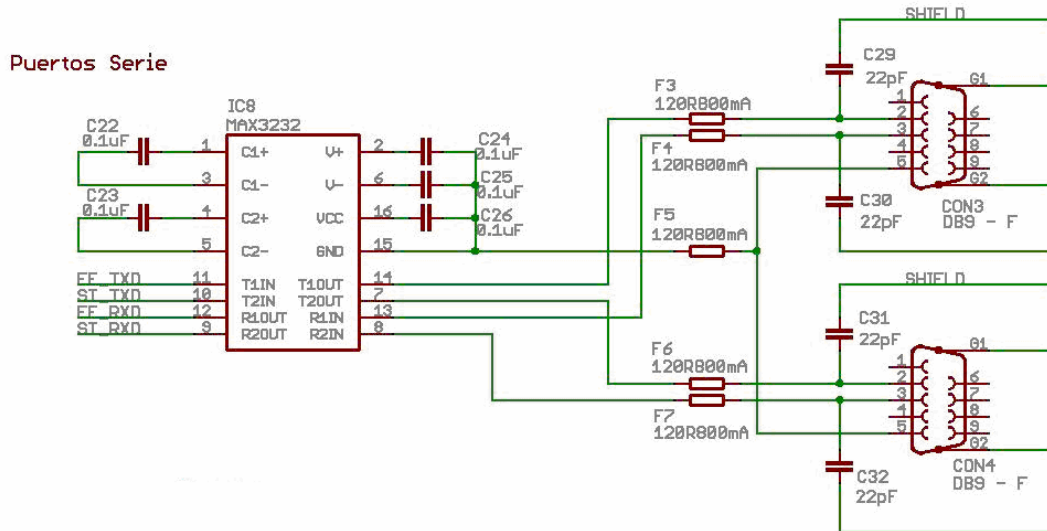


Figura 18 - Diseño de los puertos serie

Para poder conectar una computadora al adquisidor fue necesario agregar un adaptador de niveles. El procesador del adquisidor maneja una lógica CMOS de 3.3V y las computadoras manejan una lógica TTL de $\pm 15V$. Para esta conversión se eligió el integrado *MAX3232* de Maxim⁹. A continuación se listan sus principales características:

- Tensión de alimentación 3.3V
- 2 receptores y 2 transmisores
- Lógica de salida de receptores 0V – 3.3V
- Lógica de entrada de receptores $\pm 25V$
- Lógica de entrada de transmisores 0V – 3.3V
- Lógica de salida de transmisores $\pm 5.4V$
- Consumo menor a 1mA

Se utilizaron ferritas y capacitores para filtrar ruidos.

Se eligieron conectores DB9 hembra.

6.2.3.5.4 Botones y LEDs

⁹ <http://datasheets.maxim-ic.com/en/ds/MAX3222-MAX3241.pdf>

Los botones y LEDs permiten al usuario interactuar con el sistema sin la necesidad de una computadora. La Figura 19 muestra su diseño.

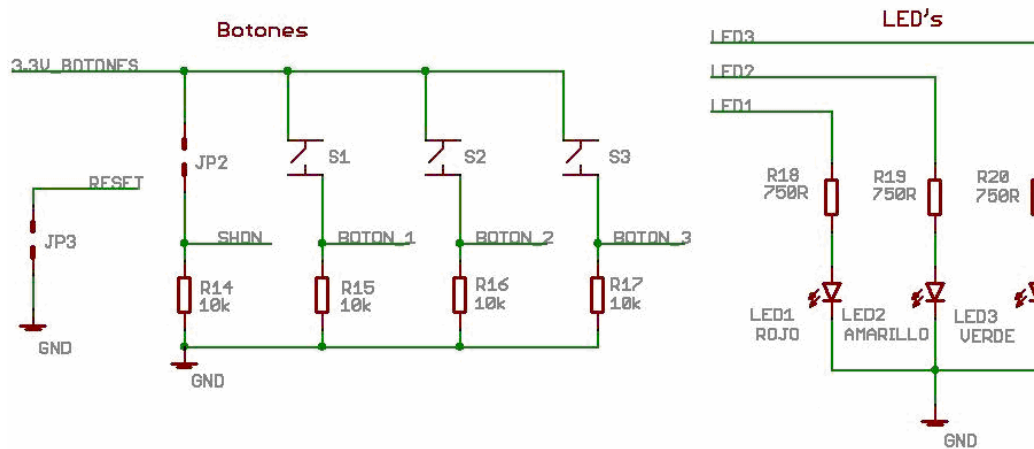


Figura 19 - Diseño de los botones y LED's

Se utilizaron LEDs de bajo consumo para poder alimentarlos directamente desde las salidas del procesador. Las resistencias fueron dimensionadas para que la corriente de excitación sea 2mA. Las salidas del procesador pueden entregar continuamente hasta 5mA.

Una de las funciones de los botones es despertar al sistema en caso de que esté en bajo consumo. En bajo consumo la fuente de 3.3V está apagada, por lo tanto no se pudo utilizar esta fuente para alimentar los botones. La solución más eficiente fue utilizar la salida SYS_EN del procesador. Esta salida está siempre a 3.3V mientras el sistema está encendido. Sólo se consume corriente en el momento de presionar un botón. Se agregaron resistencias de pull-down para evitar que las entradas queden con un valor de tensión indefinido. El botón de reset está conectado al pin de reset del procesador. Es activo por nivel bajo y tiene conectado internamente una resistencia de pull-up a una de las fuentes del procesador.

6.2.3.6 Alimentación

6.2.3.6.1 Descripción

El adquisidor se diseñó para ser alimentado con una batería de gel 12V. El primer paso del diseño del módulo fue analizar cuáles eran las tensiones necesarias para alimentar los distintos componentes. Se debía proveer una tensión de valor y calidad adecuada a los sensores, a la computadora embebida, al módem y a los circuitos integrados. Se definió que la tensión a suministrar a los sensores analógicos y

digitales debía ser 5V. Una vez elegidos el módem y la computadora embebida quedaron definidas todas las tensiones.

- 5V – Sensores analógicos y digitales y puerto USB
- 3.8V – Computadora embebida y módem
- 3.3V – Circuitos integrados

Existe un tipo especial de sensor llamado “ultrasónico” que se alimenta con una tensión entre 12V y 24V. Este sensor ha de conectarse directamente a la batería a través de una salida relay como se detalla más adelante.

Para elegir los componentes del módulo se tuvo en cuenta la eficiencia de los circuitos de adaptación y la calidad de tensión requerida. Dada la diferencia entre la tensión de entrada y las tensiones de salida, se dio preferencia al uso de fuentes conmutadas. El módulo está compuesto por dos fuentes conmutadas y dos reguladores lineales. Las fuentes conmutadas adaptan la tensión de entrada a 3.8V y 5V. La tensión 3.3V se obtiene de los 5V a través de un regulador lineal. Esto se debe a que el consumo en 3.3V es despreciable y la eficiencia de una fuente conmutada para corrientes pequeñas es muy baja. El gran problema de las fuentes conmutadas es el ruido que introducen en el circuito y el ripple de la tensión de salida. Es por eso que para la alimentación del conversor analógico-digital y los sensores analógicos se utilizó un regulador lineal directamente conectado a la tensión de entrada.

El adquisidor se alimenta con un conector Molex MATE-N-LOK de cuatro posiciones. Este tipo de conector es el usado en computadoras para alimentar los discos duros.

6.2.3.6.2 Fuentes conmutadas

Los 3.8V se adaptan a través de una fuente conmutada conectada a la tensión de entrada. Para dimensionar el circuito se analizó el consumo de los componentes alimentados por esta tensión.

Computadora embebida – Gumstix Verdex

- Consumo en Standby - 50mA
- Consumo en Idle – 60mA
- Consumo máximo – 220mA

Módem GSM/GPRS – Telit GM862

- Consumo en Standby – 5mA
- Consumo en Idle – 24mA
- Consumo máximo transmitiendo datos – 371mA promedio con picos de frecuencia 216Hz de hasta 2A.

Dada la característica de consumo del módem fue necesario conseguir una fuente que se adaptara rápidamente frente a cambios de corriente y que soportara 2.22A de corriente pico.

Los 5V para los circuitos digitales también se obtienen a través de una fuente conmutada conectada a la tensión de entrada. La carga de esta fuente es principalmente el puerto USB y los sensores digitales. Se estima un consumo máximo de 800mA.

Como forma de simplificar el diseño se decidió usar el mismo modelo de integrado para ambas tensiones.

Se utilizaron dos fuentes conmutadas *PTH08080* de Texas Instruments¹⁰. A continuación se listan las características más importantes de estas fuentes:

- Inductor y llave incorporados. Sólo es necesario dimensionar los capacitores de entrada y salida para disminuir el ripple
- Corriente constante de salida 2.25A
- Tensión de entrada entre 4.5 y 18V
- Tensión de salida ajustable por medio de una resistencia de precisión
- Control On/Off
- Protección contra sobrecarga
- Protección contra sobretemperatura

La Figura 20 muestra el diseño de las fuentes conmutadas.

¹⁰ <http://focus.ti.com/lit/ds/symlink/pth08080w.pdf>

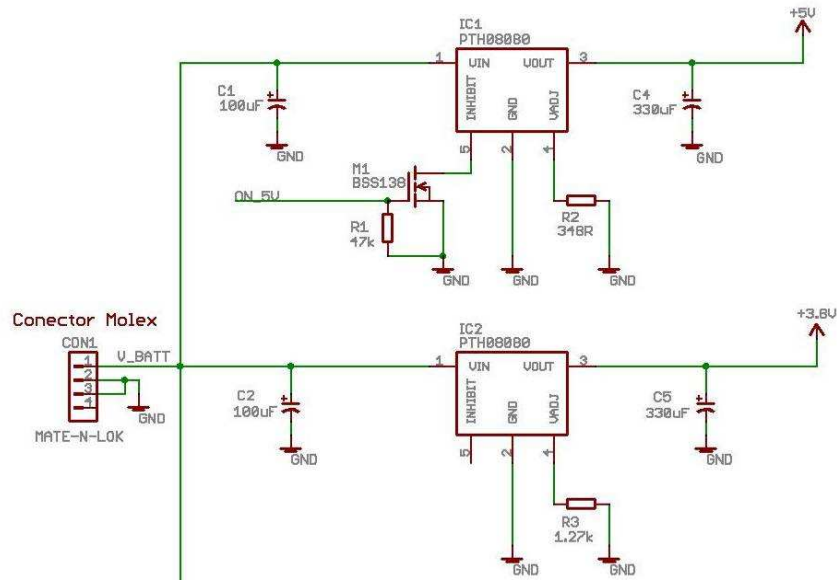


Figura 20 - Diseño de las fuentes conmutadas

Las resistencias de ajuste se dimensionaron de acuerdo a la hoja de datos. Se utilizaron resistencias de precisión al 1% con una estabilidad con respecto a la temperatura de $\pm 100\text{ppm}/^\circ\text{C}$. El valor de las resistencias es $1.27\text{k}\Omega$ y 348Ω para las fuentes de 3.8V y 5V respectivamente.

Para el dimensionado de los capacitores de entrada y salida se utilizó la hoja de datos de la fuente y el reporte *"Input and Output Capacitor Selection"* de Texas Instruments¹¹.

Los capacitores de entrada pueden ser tan grandes como se quiera. Cuanto mayor es la capacidad, menor es el ripple a la entrada de la fuente. Los capacitores con bajo ESR son preferibles debido a que frente a cambios repentinos de la carga producen menor caída de tensión. A partir de las recomendaciones del fabricante, para los valores esperados de variación de carga se llegó a que los capacitores de entrada debían cumplir con los siguientes límites.

- Capacidad mayor o igual a 100uF
- ESR menor o igual a 150mΩ
- Corriente de ripple mayor o igual a 650mA

¹¹ <http://focus.tij.co.jp/jp/lit/an/slta055/slta055.pdf>

- Tensión máxima mayor a 18V

El capacitor de entrada es el mismo para las fuentes de 3.8V y 5V. Para reducir el ESR se decidió utilizar dos capacitores iguales en paralelo. Se eligió un capacitor electrolítico de 100uF 35V con un ESR de 150mΩ y una corriente de ripple de 670mA.

En el caso del capacitor de salida también se cumple que a mayor tamaño menor ripple. Pero a diferencia del caso anterior, el capacitor de salida no puede ser indefinidamente grande. Esto se debe a que para grandes valores de capacidad, la corriente inicial de carga puede quemar la fuente. Esto también impone restricciones al valor mínimo del ESR. La resistencia equivalente del capacitor ayuda a limitar el valor inicial de corriente. Aplicando las recomendaciones del fabricante, el capacitor de salida debía cumplir los siguientes requerimientos:

- Capacidad menor que 330uF
- ESR entre 14mΩ y 33mΩ
- Tensión máxima mayor a 10V

Considerando estas restricciones se eligió para ambas fuentes un capacitor electrolítico de 330uF 16V con un ESR de 14mΩ y una corriente de ripple de 5A.

La fuente de 5V se debe apagar cuando no se estén adquiriendo datos. Por este motivo se agregó el MOSFET recomendado por el fabricante con una resistencia de pull-down de 47kΩ.

6.2.3.6.3 Fuente de 5V para los circuitos analógicos

Los sensores analógicos y el conversor analógico-digital debían alimentarse con una fuente limpia, separada del mundo digital y con el menor ripple posible. Analizando la incertidumbre en la medida del conversor se concluyó que no era plausible obtener una resolución adecuada en la medida si se usaba para el circuito analógico la tensión de salida de la fuente conmutada de 5V. El ripple de la fuente conmutada de 5V según el fabricante es 30mVpp. Esto implica que alimentando el circuito analógico con la fuente de 5V no se puede tener una medida con más de 7 bits de resolución, lo cual es inaceptable. Se analizaron dos alternativas, filtrar la alimentación o usar un regulador lineal. Los reguladores lineales tienen un excelente rechazo al ruido de entrada y un muy bajo ripple a la salida. La gran desventaja es que son ineficientes. Por otro lado, es difícil filtrar altas frecuencias sin complicar el diseño. La solución surgió a partir de un análisis de la eficiencia de una y otra alternativa. La fuente conmutada, para la

carga estimada tiene una eficiencia de 90%. La eficiencia del regulador lineal para 12V de entrada y 5V de salida es 42%. El conmutador es más del doble de eficiente que el regulador lineal. Sin embargo este consumo extra representa para un caso típico de uso, menos del 1% del consumo total. Por lo tanto se sacrifica un poco de eficiencia a cambio de una mayor resolución en la medida sin complicar innecesariamente el diseño.

El regulador debía suministrar como mínimo la corriente necesaria para 5 sensores de 1k Ω , que equivale a 50mA. El consumo del convertor analógico digital es despreciable frente a este valor. Para obtener una medida de 12 bits de resolución, el ripple de la fuente debía ser menor a 0.6mVpp. Además, como la diferencia de tensión de entrada y salida es grande, el regulador tenía que ser capaz de disipar adecuadamente el calor generado para la corriente de diseño.

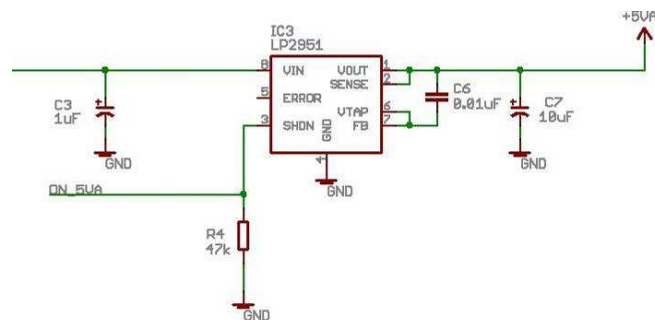


Figura 21 - Diseño del regulador lineal de 5V

A partir de estas restricciones se eligió el regulador lineal *LP2951-50* de Texas Instruments¹². A continuación se listan las principales características:

- Tensión de salida ajustable o fija en 5V
- Tensión de entrada máxima 30V
- Corriente de salida 100mA
- Precisión de 2% en todo el rango de temperaturas
- Regulación de línea menor al 0.4% para todo el rango de tensión de entrada
- Regulación de carga menor al 0.3% para todo el rango de corriente
- Ripple de salida menor a 100uApp para la configuración recomendada
- Control On/Off

¹² <http://focus.ti.com/lit/ds/symlink/lp2950-50.pdf>

- Temperatura máxima de juntura 125°C
- Impedancia térmica 97°C/W

En la hoja de datos se dan recomendaciones para el dimensionado de los capacitores de entrada y salida. El valor de ESR del capacitor de salida debe estar entre 30mΩ y 5Ω. Esto es muy importante porque determina la estabilidad del regulador. Para la entrada se utilizó un capacitor de tantalio de 1uF 35V. Para la salida se utilizó un capacitor de tantalio de 10uF 10V, con un ESR de 2.1Ω. El fabricante recomienda colocar un capacitor de cerámico de 0.01uF entre la salida y la entrada “feedback”. Con esto se logra un mejor desempeño a altas frecuencias reduciendo el ripple a la salida.

De acuerdo con los datos térmicos indicados por el fabricante, la fuente puede suministrar los 100mA de diseño incluso con una temperatura ambiente de 50°C.

Para el apagado y prendido del regulador se utiliza la entrada SHDN con una resistencia de pull-down de 47kΩ. Mientras SHDN tiene una tensión aplicada menor a 2V el regulador permanece encendido.

6.2.3.6.4 Fuente de 3.3V

La computadora embebida maneja una lógica CMOS de 3.3V. Por este motivo se eligieron todos los circuitos integrados para trabajar con esta tensión. El consumo en 3.3V es menor a 6mA.

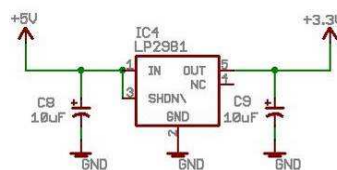


Figura 22 - Diseño del regulador lineal de 3.3V

Se eligió el regulador lineal LP2981 de Texas Instruments¹³. A continuación se listan las principales características:

- Tensión de salida fija 3.3V
- Corriente de salida 100mA

¹³ <http://focus.ti.com/docs/prod/folders/print/lp2981-28.html>

- Bajo Dropout – 0.2V para 100mA

En la hoja de datos se recomienda utilizar capacitores de tantalio. También se establece un valor mínimo de 1uF y 3.3uF para los capacitores de entrada y salida respectivamente. Además se indican valores máximos y mínimos para el ESR del capacitor de salida para asegurar la estabilidad del regulador. Para la entrada y salida se escogió un capacitor de tantalio de 10uF 10V con un ESR de 2.1mΩ.

6.2.3.7 Circuito impreso

6.2.3.7.1 Descripción

Para el diseño del impreso se utilizó el programa EAGLE de la empresa Cadsoft. Se tuvieron en cuenta los siguientes puntos:

- Corriente de operación de cada dispositivo
- Corriente admisible de pistas y vías
- Ruido introducido por las fuentes conmutadas
- Ruido introducido por el módem
- Ruido introducido por los distintos componentes digitales
- Aislamiento de los circuitos analógicos

La disposición de los componentes se hizo de forma de minimizar las interferencias de señales. En particular, se tuvo especial cuidado en separar los circuitos analógicos de los digitales. Siguiendo las recomendaciones del fabricante del módem y dada la complejidad del diseño se decidió utilizar un impreso de cuatro capas. Las dos capas internas son de alimentación y tierra y las dos externas son de señales.

El circuito impreso se fabricó en Sunstone Circuits. A continuación se listan sus principales características:

- 4 capas
- Máscara antisoldante en ambos lados
- Máscara silkscreen del lado de los componentes
- Agujeros metalizados
- Espesor del cobre de las capas: 1oz
- Capas externas con baño de plata: 1oz

- Ancho mínimo de pista: 6mil
- Vía mínima: 8mil

6.2.3.7.2 Diseño

Para determinar el tamaño mínimo de pistas y vías se utilizaron como referencia los siguientes sitios de Internet:

- <http://home.comcast.net/~pcb.george/trace.html>
- http://www.pcb3d.com/uploads/via_Current.xls

A continuación se listan los tamaños de pistas y vías utilizados:

- Alimentación de módem y computadora: planos y vías de 20mil
- Salida relay: pistas de 65mil
- Alimentación USB: pistas de 30mil y vías de 20mil
- Alimentación principal: planos y vías de 20mil
- Alimentación 5V: planos y vías de 20mil
- Alimentación 3.3V: planos, pistas de 6mil y vías de 10mil
- Señales: pistas de 6mil y vías de 8mil

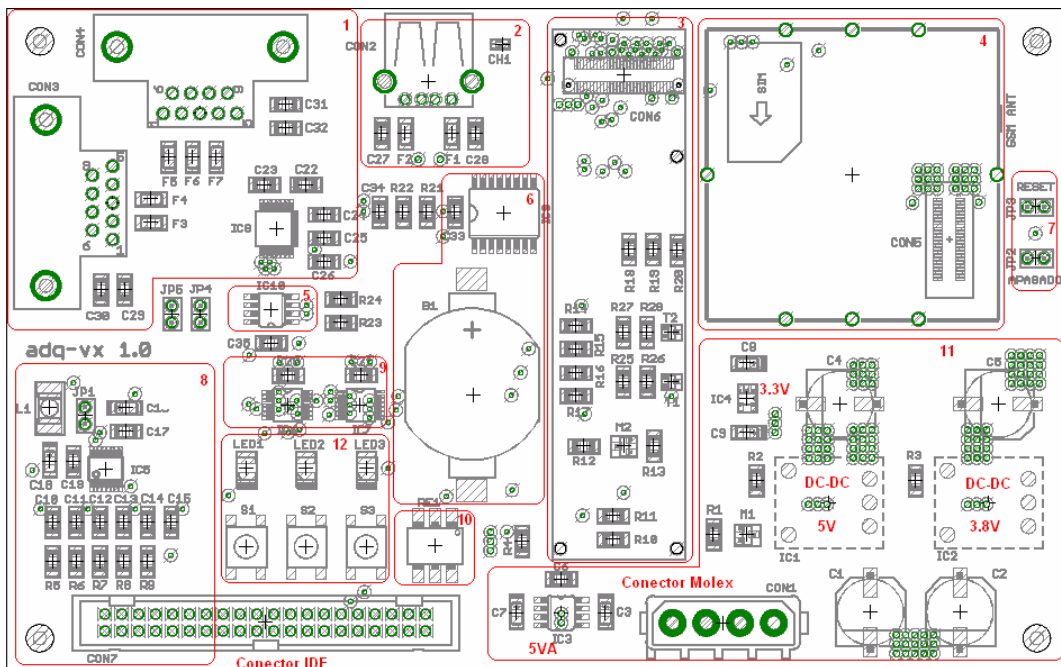


Figura 23 - Ubicación de los componentes de la placa adq-vx

La Figura 23 muestra una imagen del diseño del impreso. En la figura se indica la disposición de los componentes. La figura está dividida en 12 zonas:

1. Puertos serie
2. Puerto USB
3. Computadora embebida
4. Módem
5. Buffer i2c
6. RTC y batería
7. Botones de apagado y reset
8. Circuito analógico
9. Adaptación de señales digitales – Schmitt Trigger
10. Relay de estado sólido
11. Módulo de alimentación
12. Botones y LED's

Para minimizar la interferencia introducida por los circuitos digitales, se intentó alejarlos lo más posible de los circuitos analógicos. A la derecha se ubicaron las fuentes conmutadas, el módem, la computadora y el RTC. Estos son los componentes que introducen más ruido. En la esquina inferior izquierda se ubicó el conversor analógico digital.

La Figura 24 muestra un detalle del circuito analógico. Se agregó una guarda de tierra a su alrededor para aislarlo del resto de los componentes. El conversor se conecta con el bus i2c a través de dos vías.

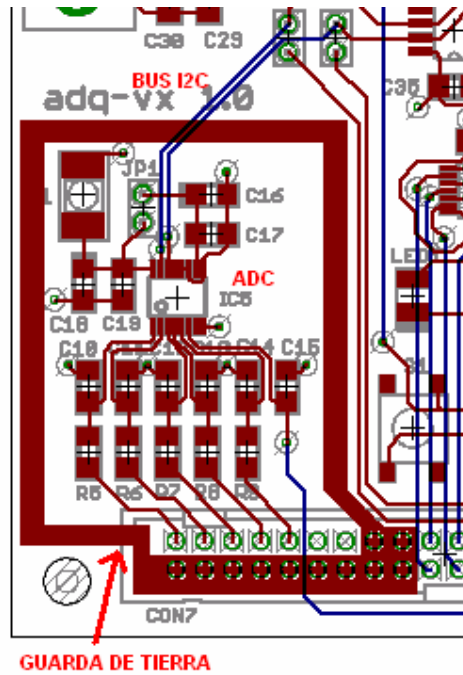


Figura 24 - Detalle del circuito analógico

La Figura 25 muestra un detalle de los puertos serie y USB. Se puede apreciar cómo se combinaron los planos de tierra y las ferritas para lograr el filtrado de la señal. Esta técnica es la que utilizan las placas de expansión de la empresa Gumstix¹⁴.

¹⁴ <http://pubs.gumstix.org/boards/CONSOLE/VX/PCB10003-R1851/PCB10003.brd>

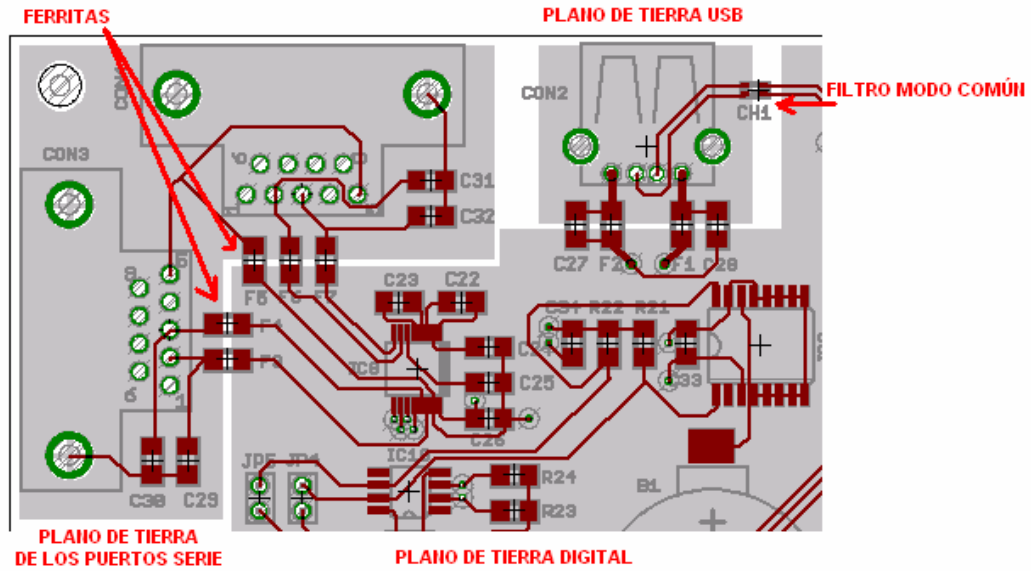


Figura 25 - Detalle del los puertos serie y USB

La Figura 26 muestra un detalle del conector de lo computadora embebida.

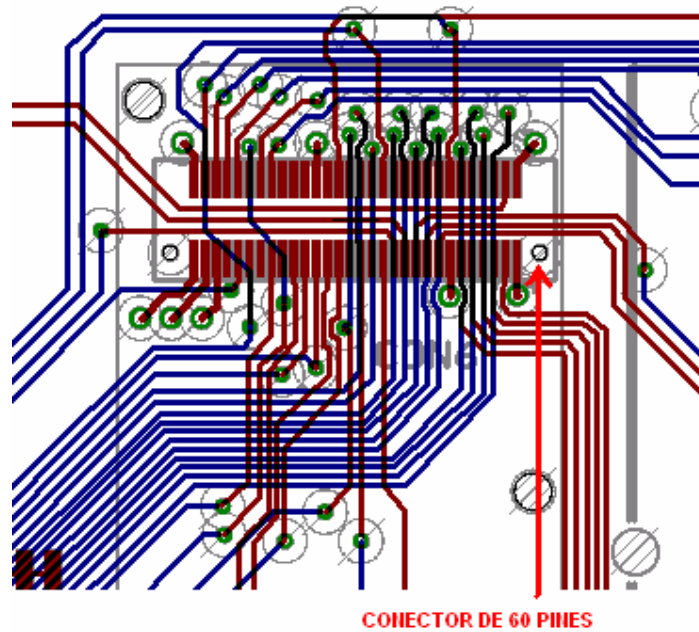


Figura 26 - Detalle del conector de 60 pines de la computadora embebida

La Figura 27 muestra un detalle del conector del módem. Para conectar los planos de tierra y alimentación se utilizaron varias vías de 20mil. Esto mejora la conductividad entre planos.

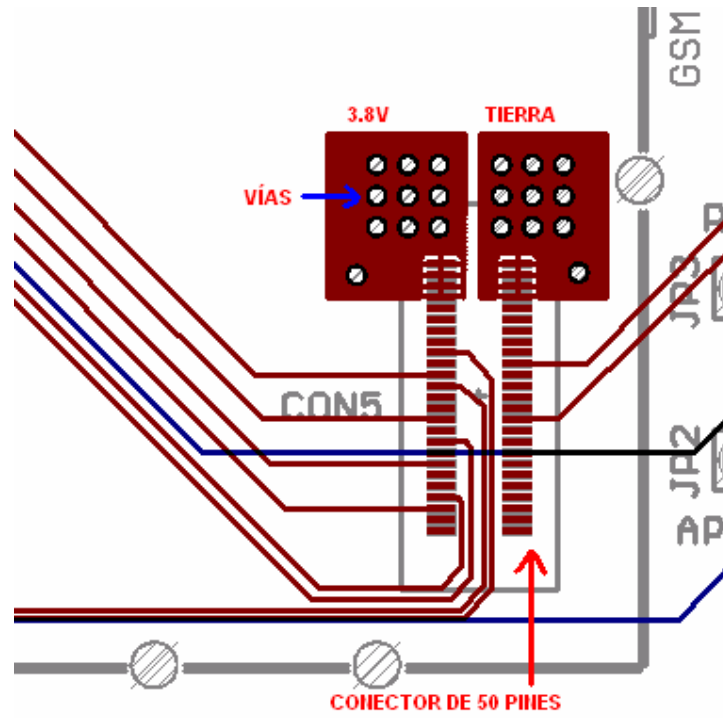


Figura 27 - Detalle del conector de 50 pines del módem

En la Figura 28 y la Figura 29 se puede ver cómo se dividieron los planos de alimentación y tierra.

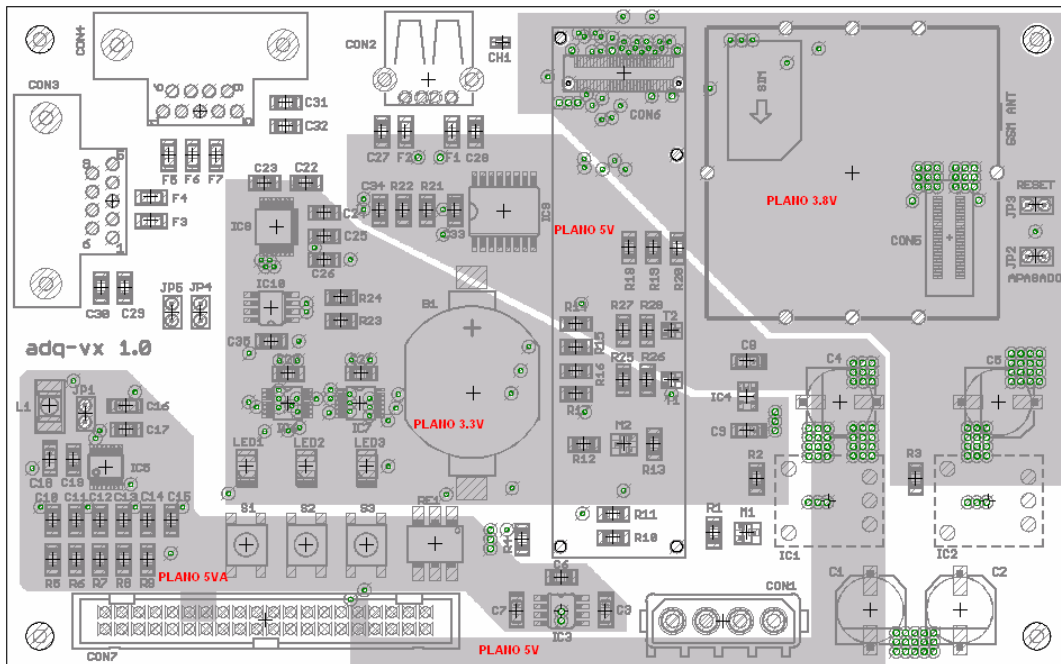


Figura 28 - Plano de alimentación de la placa adq-vx

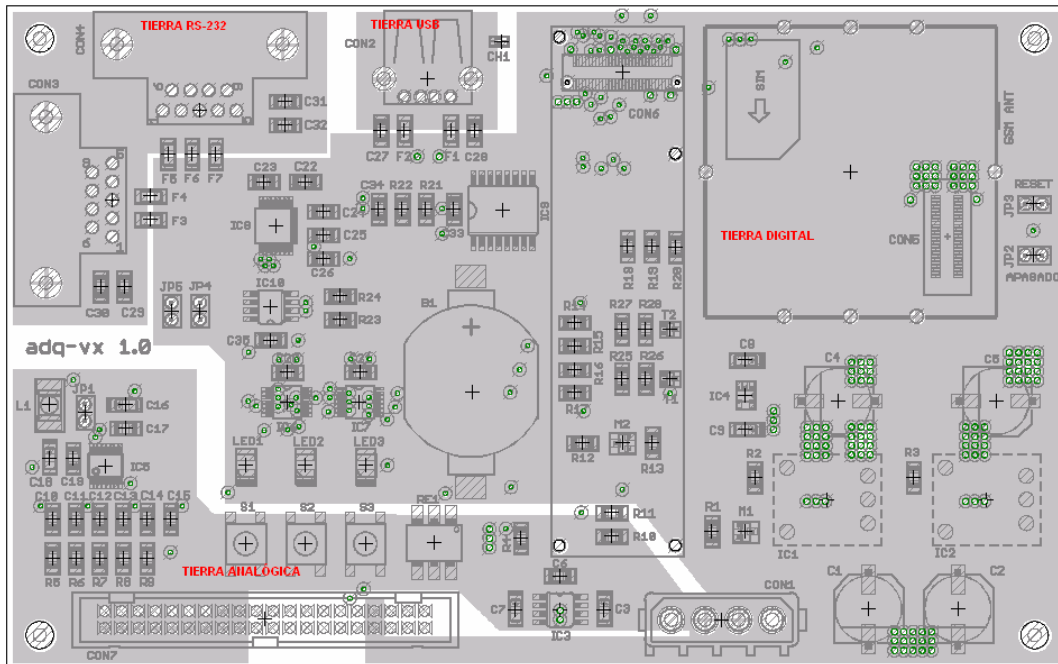


Figura 29 - Plano de tierra de la placa adq-vx

6.2.4 Consumo y autonomía

El consumo medio del adquisidor depende de la estrategia de muestreo utilizada y de la cantidad de datos enviados. La Tabla 3 muestra los consumos instantáneos de adquisidor alimentado en 12V.

| Estado | Consumo (mA) |
|----------------|--------------|
| Standby | 32 |
| Adquisición | 110 |
| Envío de Datos | 210 |

Tabla 3 - Consumo instantáneo del adquisidor

Con estos datos es posible estimar el consumo medio del adquisidor para distintas estrategias de muestreo. Para determinar el tiempo de envío de datos en función de la cantidad de datos a enviar, se puede asumir una velocidad de transferencia de 24kbps. En los capítulos 6.2.4.1 y 6.2.4.2 se resume el cálculo de consumo medio para dos estrategias de muestreo típicas.

6.2.4.1 Consumo y autonomía: medidas de los diez últimos minutos de una hora

Una estrategia muy utilizada en este tipo de adquisidores es promediar las medidas de diez minutos consecutivos una vez por hora. Esta estrategia permite que la mayor

parte del tiempo el adquisidor trabaje en modo de bajo consumo. Debido a que sólo se guarda un valor por hora es razonable suponer 100KB como cantidad máxima de datos a transmitir por día. Bajo estas hipótesis, el tiempo de envío de datos es menor a un minuto por día. La Tabla 4 muestra el cálculo del consumo medio de esta estrategia.

| Estado | Fracción de Tiempo | Consumo Instantáneo | Consumo Medio |
|----------------------------|--------------------|---------------------|---------------|
| Standby | 5/6 | 32mA | 26.66mA |
| Adquisición | 1/6 | 110mA | 18.33mA |
| Envío de Datos | 1/1440 | 210mA | 0.15mA |
| Consumo Medio Total | | | 45.1mA |

Tabla 4 - Consumo medio de la estrategia intermitente

El consumo medio para esta estrategia es 45.1mA. La Tabla 5 muestra la autonomía del adquisidor bajo esta estrategia para distintos valores de capacidad de batería.

| Batería | Autonomía |
|---------|-----------|
| 9Ah | 8 días |
| 12Ah | 11 días |
| 17Ah | 16 días |
| 24Ah | 22 días |
| 45Ah | 41 días |

Tabla 5 - Autonomía del adquisidor en modo intermitente

El panel solar se debe elegir de forma que en las horas pico de luz suministre la energía consumida en el día. El adquisidor consume 0.54W. Asumiendo 6 horas de carga por día, el panel solar debe ser de una potencia mayor o igual a 2.2W.

6.2.4.1 Consumo y autonomía: modo continuo y sensor ultrasónico

El cliente tiene un tipo de sensor llamado ultrasónico que puede adquirir hasta diez datos por segundo. Su consumo es de 110mA en 12V. Usualmente se usa para adquirir datos de forma continua. El sensor genera 2MB de datos por día. Esto equivale a diez minutos diarios de transferencia de datos. La Tabla 6 muestra el cálculo del consumo medio de esta estrategia.

| Estado | Fracción de Tiempo | Consumo Instantáneo | Consumo Medio |
|----------------|--------------------|----------------------------------|----------------|
| Adquisición | Todo el tiempo | 110mA adquisidor 110mA sensor | 220mA |
| Envío de Datos | 10/1440 | 210mA | 1.5mA |
| | | Consumo Medio Total | 221.5mA |

Tabla 6 - Consumo medio del adquisidor en modo continuo con un sensor ultrasónico

El consumo medio para esta estrategia es 221.5mA. La Tabla 7 muestra la autonomía del adquisidor bajo esta estrategia para distintos valores de capacidad de batería.

| Batería | Autonomía |
|---------|-----------|
| 9Ah | 1.7días |
| 12Ah | 2.2 días |
| 17Ah | 3.2 días |
| 24Ah | 4.5 días |
| 45Ah | 8.5 días |

Tabla 7 - Autonomía del adquisidor en modo continuo con un sensor ultrasónico

El panel solar se debe elegir de forma que en las horas pico de luz suministre la energía consumida en el día. El conjunto adquisidor-sensor consume 2.7W. Asumiendo 6 horas de carga por día, el panel solar debe ser de una potencia mayor o igual a 10.6W.

6.2.5 Costos

La Tabla 8 resume los costos de fabricación de uno, diez y cien placas adq-vx. Por una lista detallada de componentes ver el Manual de Fabricación y Desarrollo.

| Componente | Una Unidad (USD) | Diez Unidades (USD) | Cien Unidades (USD) |
|--------------|------------------|---------------------|---------------------|
| Computadora | 135 | 135 | 135 |
| Módem | 116.5 | 99.8 | 87.9 |
| Impreso | 152 | 51.2 | 2.1 |
| Componentes | 103 | 79 | 58.6 |
| Soldado | 129 | 90 | 70 |
| Total | 635.5 | 455 | 353.6 |

Tabla 8 - Costo de fabricación de la placa adq-vx

6.2.6 Conclusiones

Se logró un diseño de hardware que cumple con todos los requerimientos del cliente. La flexibilidad que brinda la computadora embebida permite usar el adquisidor para varias aplicaciones.

El diseño llevó más tiempo que el planificado inicialmente. Esto se debió a la búsqueda de un producto lo más genérico posible, analizándose varios casos de uso. Además cada detalle fue verificado y corregido varias veces, ya que luego de su implementación los errores en el hardware son difíciles y costosos de corregir. Es importante destacar que luego del soldado de los componentes no hubo que realizar ninguna corrección. El hardware pasó todas las pruebas con éxito.

6.3 Placa de Protecciones

6.3.1 Características generales

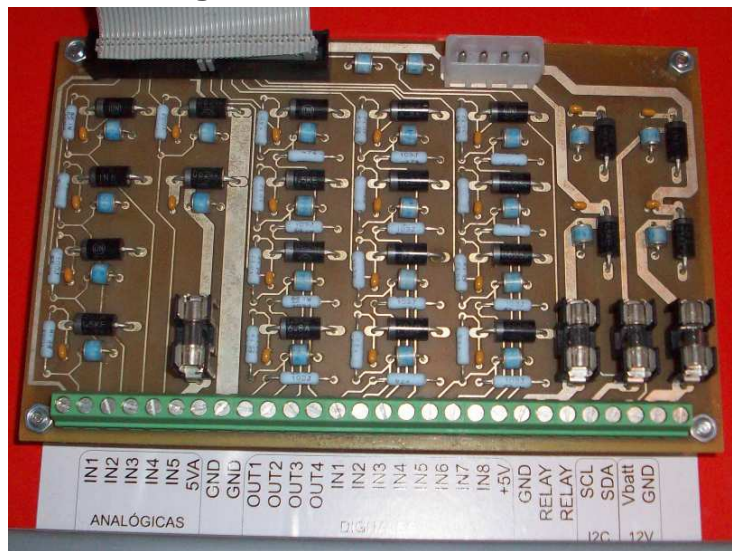


Figura 30 - Foto de la placa de protecciones

- Tamaño 10cm x 16cm.
- 4 puntos de fijación.
- 5 entradas analógicas: Limita tensión a 6.8V.
- 8 entradas digitales: Limita tensión a 6.8V.
- 4 salidas digitales: Limita tensión a 6.8V.
- Bus i2c: Limita tensión a 90V.
- 1 salida relay: Limita tensión a 43V. Fusible de 300mA.

- 5V para sensores analógicos: Limita tensión a 6.8V. Fusible 300mA.
- 5V para sensores digitales: Limita tensión a 6.8V. Fusible 300mA.
- Alimentación: Limita tensión a 20V. Fusible 3A.

6.3.2 Descripción general

El adquisidor está diseñado para funcionar a la intemperie, en ambientes donde es probable la ocurrencia de descargas atmosféricas. Por este motivo se debió proveer un mecanismo de protección frente a las posibles sobretensiones generadas por descargas atmosféricas próximas al punto de instalación del adquisidor. La placa de protecciones está constituida por elementos que dan un camino de baja impedancia a perturbaciones eléctricas. Además cuenta con fusibles que protegen el circuito frente a sobrecargas y cortocircuitos. Para su diseño se utilizó como referencia el proyecto de fin de carrera “*Adquisidor de Velocidad y Dirección de Viento*” de Fernando Caraballo, Luis Ferrari y Juan Pablo Oliver.

A continuación se describen los componentes utilizados:

- **Resistencias:** Las resistencias a tierra proveen un camino a la perturbación. Las resistencias en serie permiten desacoplar las protecciones.
- **Descargadores Gaseosos:** Estos componentes son aisladores hasta que se supera una tensión umbral. Luego de superada esta tensión, se ioniza el gas y se produce una descarga entre los bornes. El componente se comporta como una resistencia negativa. Tienen una gran capacidad de conducción de corriente y una muy baja capacidad parásita. Sin embargo son muy lentos y las tensiones umbrales disponibles son superiores a 40V.
- **Diodos Zener:** Los diodos zener limitan la tensión en sus bornes. Son baratos y actúan rápido. No son capaces de conducir corrientes elevadas y su capacitancia puede afectar el funcionamiento de circuitos digitales.
- **Capacitores:** Proveen un camino de muy baja impedancia frente transitorios de alta frecuencia. Son baratos y de rápida actuación.

6.2.3 Diseño

6.2.3.1 Líneas de potencia

La Figura 31 muestra el diseño de las protecciones de las líneas de potencia. Se evitó el uso de resistencias para no disipar energía. Los fusibles protegen frente a sobrecargas y cortocircuitos.

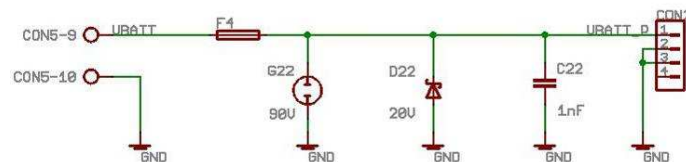


Figura 31 - Diseño de las protecciones de las líneas de potencia

6.2.3.2 Líneas de datos digitales

La Figura 32 muestra el diseño de las protecciones de las líneas de datos digitales. La resistencia serie permite desacoplar el descargador gaseoso del diodo y junto con el capacitor forman un filtro pasa bajos. La resistencia en paralelo da un camino a la perturbación. Además sirve como resistencia de pull-down.

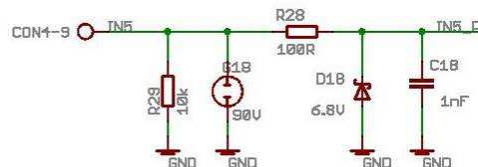


Figura 32 - Diseño de las protecciones de las líneas digitales

6.2.3.3 Líneas de datos analógicos

La Figura 33 muestra el diseño de las protecciones de las líneas de datos analógicos. Su diseño es similar a las líneas digitales. No tienen la resistencia en paralelo porque ésta puede afectar la medida del convertor.

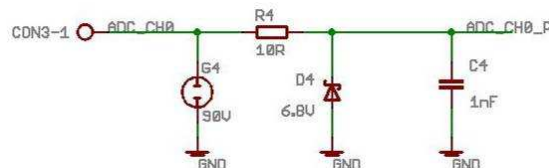


Figura 33 - Diseño de las protecciones de las líneas analógicas

6.2.3.4 Bus i2c

La Figura 34 muestra el diseño de las protecciones del bus i2c. Dado que el valor de capacitancia del bus i2c no puede superar los 400pF, no se pudo utilizar diodos zener ni capacitores. El bus se protegió con descargadores gaseosos que tienen una capacitancia menor a 1pF. El buffer i2c usado en la placa adq-vx tiene una protección ESD de 200V.

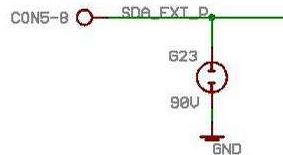


Figura 34 - Diseño de las protecciones de las líneas del bus i2c

6.2.3.5 Circuito impreso

El circuito impreso se fabricó en Sunstone Circuits. A continuación se listan sus principales características:

- 2 capas
- Agujeros metalizados
- Espesor de las capas 1oz
- Capas con baño de plata
- Ancho mínimo de pista: 6mil
- Vía mínima: 8mil

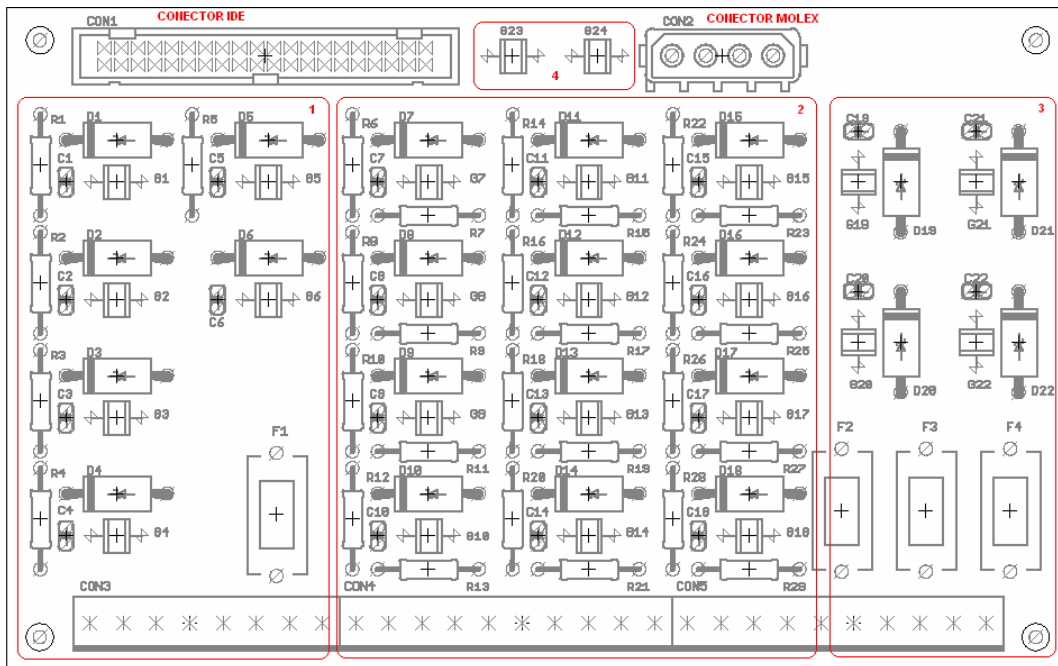


Figura 35 - Ubicación de los componentes de la placa de protecciones

La Figura 35 muestra una imagen del diseño del impreso. En la figura se indica la disposición de los componentes. Está dividida en 4 zonas:

1. Protecciones de líneas analógicas y 5VA
2. Protecciones de líneas digitales
3. Protecciones de líneas de potencia
4. Protecciones del bus i2c

La capa de inferior es el plano de tierra. Las pistas se hicieron lo más anchas posible para disminuir su resistencia. Del lado de los componentes se separó el circuito digital del analógico mediante una guarda de tierra. La Figura 36 muestra la división del plano de tierra.

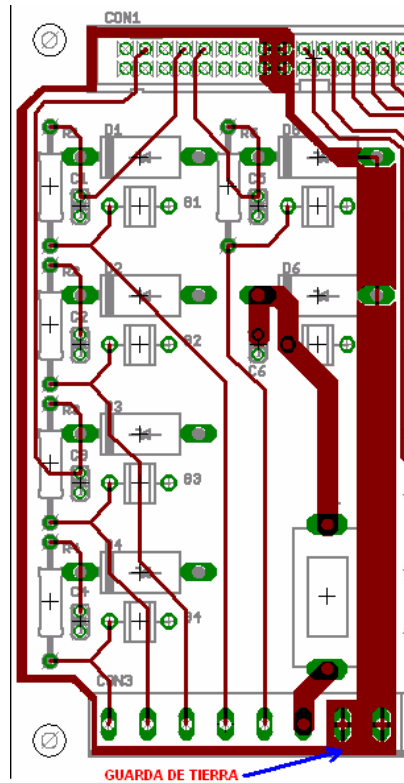


Figura 36 - Detalle del circuito analógico de la placa adq-vx

6.2.4 Costos

La Tabla 9 resume los costos de fabricación de uno, diez y cien placas de protecciones. Es importante destacar que en caso de no utilizar todas las entradas y salidas es posible fabricar una placa de protecciones con menos componentes y superficie de impreso, disminuyendo los costos de fabricación considerablemente. Por una lista detallada de componentes ver el Manual de Fabricación y Desarrollo.

| Componente | Una unidad (USD) | Diez unidades (USD) | Cien unidades (USD) |
|--------------|------------------|---------------------|---------------------|
| Impreso | 55 | 20.7 | 1.2 |
| Componentes | 88.6 | 61.5 | 41.8 |
| Soldado | 87 | 60 | 50 |
| Total | 230.6 | 142.2 | 93 |

Tabla 9 - Costo de fabricación de la placa de protecciones

6.2.5 Conclusiones

Se diseñó una placa que protege cada una de las entradas y salidas del adquisidor frente a sobretensiones. En la mayoría de las aplicaciones no son necesarias todas las

entradas y salidas. Para estos casos se recomienda implementar las protecciones mínimas necesarias, con lo cual se reducen los costos de fabricación.

6.4 Gabinete

6.4.1 Características generales

Para la instalación y protección del adquisidor se seleccionó el gabinete AGP 4232 TO 1P de la línea Argenpol de la empresa Conextube. Este gabinete es de fabricación argentina bajo las normas de calidad IRAM y sus características técnicas son las siguientes:

- Material: Policarbonato gris Ral 7035
- Tapa: Policarbonato opaca
- Grado de protección: IP 549 - IEC 60670
- Autoextinguibilidad: Cumple 850°C - IEC 60695
- Resistencia a los rayos UV: Sí
- Homologaciones: IRAM IEC 60670
- Dimensiones: 420x320x200 (mm)

El código 1P del modelo significa que el gabinete cierra en un sólo punto. Es esto lo que determina que su grado de protección sea IP54. El modelo 3P con cierre en tres puntos tiene un grado de protección IP65.



Figura 37 - Gabinete

6.4.2 Descripción general

El montaje del adquisidor en el interior del gabinete se realizó sobre una placa metálica de dimensiones estándar. Para un gabinete de 42cm de alto por 32cm de ancho la dimensión de la placa es de 36cm de alto por 24.5cm de ancho con un espesor de 2mm.

A continuación se pueden apreciar la placa adq-vx y la placa de protecciones montadas sobre la bandeja metálica en el interior del gabinete.



Figura 38 - Gabinete armado

El montaje de la bandeja dentro de la caja se realizó con tornillos de 6mm, con cabeza hexagonal, doble tuerca y arandela, utilizando un accesorio del gabinete que además de permitir la fijación de éste en el exterior mantiene la estanqueidad. La Figura 39 muestra un gráfico de este montaje.



Figura 39 - Montaje del gabinete

El gabinete cuenta además con 3 adhesivos. La Figura 40 muestra una imagen del adhesivo colocado debajo de la bornera. En él se identifican cada uno de los bornes.



Figura 40 - Adhesivo

Otros dos adhesivos con el logo del adquisidor se encuentran ubicados a ambos lados de la tapa del gabinete.

6.4.3 Diseño

Dado que no se imponen restricciones volumétricas, se optó por un gabinete amplio que permitiese la manipulación del adquisidor sin dificultades en la conexión de sensores, el manejo de los puertos serie y el conexionado del pendrive en el puerto USB.

El siguiente plano muestra las dimensiones de la bandeja metálica con los agujeros necesarios para el montaje (las dimensiones están en milímetros). En líneas punteadas se representan las placas y el ducto plástico para los cables.

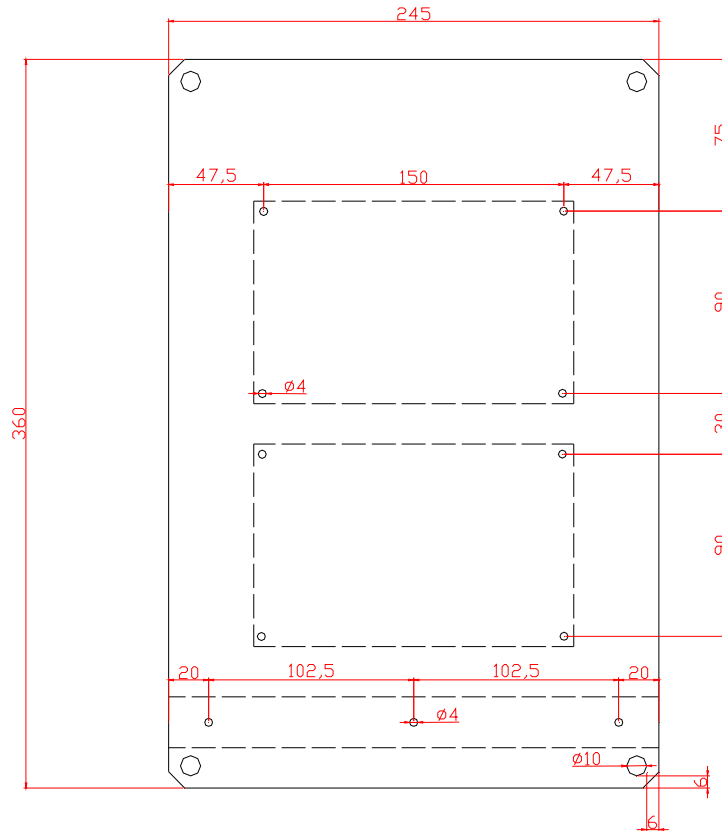


Figura 41 - Diseño de la bandeja

La fijación de las placas en la bandeja se realizó con 4 tornillos 5/32" y 3 tuercas. La primera de las tuercas se utilizó para fijar el tornillo a la chapa, la segunda para determinar la altura de las placas respecto a la bandeja y la tercera para afirmar las placas.



Figura 42 - Tornillos de fijación

El ducto de los cables también se fijó con tornillos 5/32" que, a diferencia de los anteriores, se colocaron en agujeros roscados. La función del ducto es que todos los

sensores que sean conectados al adquisidor pasen por él, manteniendo el orden y la seguridad dentro del gabinete.

6.4.4 Costos

| Elemento | Costo unitario | Costo por 10 | Costo por 100 | Proveedor |
|-----------------------------|-----------------|-------------------|-----------------|------------|
| Gabinete | 84 U\$S + IVA | 75.6 U\$S + IVA | 67.2 U\$S + IVA | Fivisa |
| Bandeja | 10 U\$S + IVA | 10 U\$S + IVA | 10 U\$S + IVA | Tefix S.A. |
| Tornillería | 2 U\$S | 2 U\$S | 2 U\$S | Ferretería |
| Adhesivos | 1.2 U\$S | 1.2 U\$S | 1.2 U\$S | 4Tintas |
| Mano de obra | 10 U\$S | 5 U\$S | 3.5 U\$S + IVA | Tefix S.A. |
| Total (IVA incluido) | 128 U\$S | 112.6 U\$S | 101 U\$S | |

Tabla 10 - Costos de gabinete

6.4.5 Conclusiones

Con la elección de este gabinete se consiguió un nivel de protección suficiente para la aplicación.

El gabinete es fácilmente reproducible dada la simplicidad de su armado y que sus elementos pueden ser conseguidos en plaza.

7 Software

7.1 Descripción General

Previo a dar inicio a la programación, se realizó una selección del sistema operativo a utilizar, del entorno de desarrollo, de los lenguajes de programación, así como de la forma en la que se comunicarían entre sí los diferentes programas.

El software se dividió en tres módulos: el módulo de adquisición, el módulo de comunicación y el módulo de gestión de la aplicación. El módulo de adquisición se encarga de adquirir y almacenar los datos de los sensores. El módulo de comunicación se encarga del envío y recepción de información. El módulo de gestión de la aplicación ejecuta los programas de adquisición, comunicación, tareas de rutina y maneja el modo de bajo consumo.

Se desarrolló además una interfaz amigable para configurar el adquisidor de forma local y un sitio Web para la configuración remota.

Además de los programas utilizados por los módulos anteriormente mencionados se crearon programas que permiten al usuario tener acceso al Hardware de forma sencilla y segura.

Finalmente, para facilitar la reproducción del sistema, se creó una imagen que incluye el sistema operativo, los programas utilizados y las aplicaciones programadas.

7.2 Arquitectura de Software

Se analizaron dos tipos de arquitecturas de software, una arquitectura Round-Robin en la cual todas las tareas se ejecutan de manera secuencial, y una arquitectura donde las tareas se ejecutan en paralelo y son administradas por el sistema operativo.

La ventaja de la primera opción es que todos los recursos del sistema son destinados a la tarea en ejecución. Por ejemplo, las rutinas de adquisición que necesitan precisión en el tiempo, es conveniente que dispongan de toda la capacidad de procesamiento. Sin embargo, se ocupa gran parte del tiempo en estas rutinas, imposibilitando la ejecución de las tareas de comunicación. Además se subutilizan los recursos de hardware. La segunda alternativa es más modular, delegando al sistema operativo algunas tareas como ser el control de los tiempos de ejecución. Además permite dividir la programación, reduciendo los tiempos de desarrollo.

Considerando la aplicación y sus requerimientos, se concluyó que la segunda opción era la más indicada.

Se dividió la aplicación en tres módulos: el módulo de adquisición de datos, el módulo de comunicación y el módulo de gestión de la aplicación. Este último es el encargado de la ejecución de las tareas de adquisición y comunicación y del control del modo de bajo consumo. Cada módulo fue concebido como una unidad funcional independiente de las demás. Estos módulos se ejecutan en paralelo. La Figura 43 muestra un esquema de esta división.

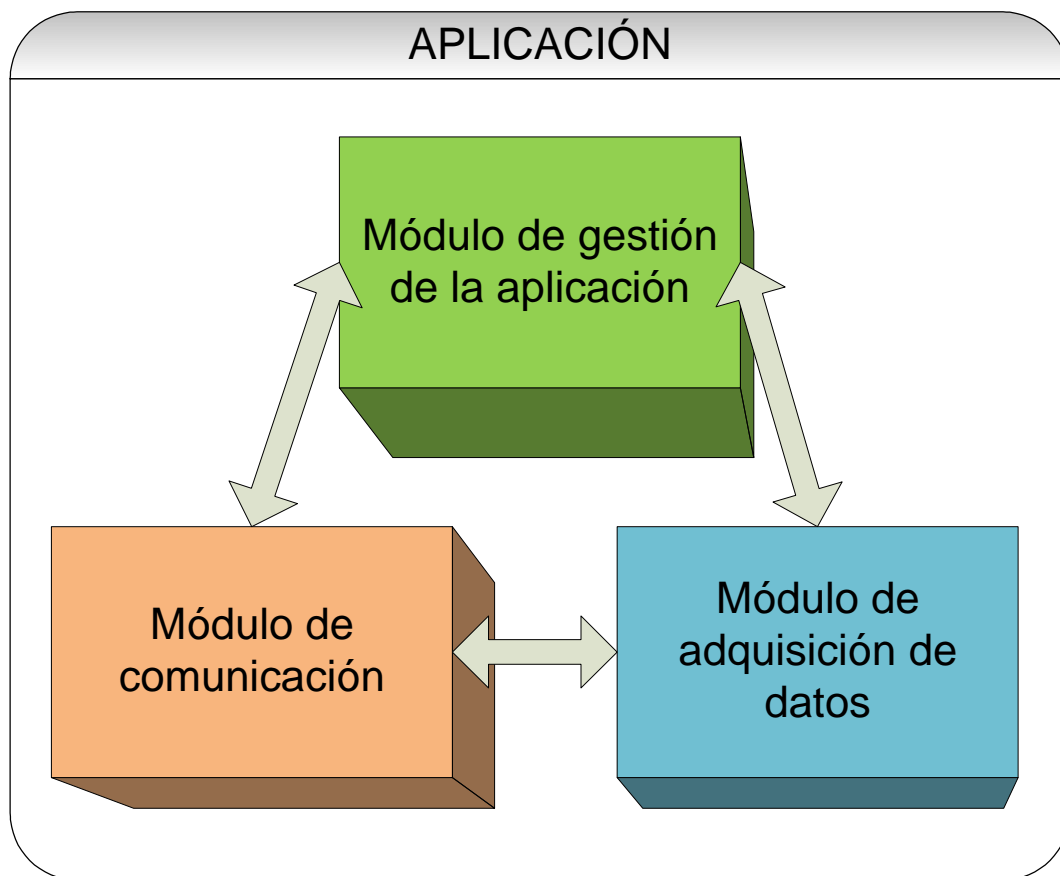


Figura 43 - División del programa en módulos

La aplicación se diseñó para funcionar de forma autónoma, sin supervisión, en lugares de difícil acceso. Debido a esto fue necesario lograr una solución robusta, que sobreponga al sistema frente a fallas inesperadas. Para esto se utilizó el watchdog del procesador, que reinicia el sistema en caso de que se bloquee. Además, los programas tienen un tiempo límite de vida luego del cual son finalizados forzosamente por el módulo de gestión de la aplicación.

7.3 Sistema de Archivos

El sistema de archivos es la forma en que se estructura y gestiona la información en memoria. Como todo sistema operativo, Linux tiene su propio sistema de archivos. Si bien pueden encontrarse variaciones mínimas según la distribución de Linux utilizada, la estructura principal se mantiene. El sistema operativo instala una estructura de carpetas para organizar y gestionar sus aplicaciones y recursos. Se creó una estructura de archivos para organizar la información de la aplicación que fuera coherente con el sistema de archivos de Linux.

Se distinguen tres memorias que almacenan diferentes tipos de información: la memoria FLASH integrada en la placa base donde se aloja el sistema operativo y la aplicación, la memoria RAM donde se almacena información temporal del sistema operativo e información temporal de la aplicación y la memoria FLASH extraíble (pendrive) donde se guardan los datos adquiridos. La carpeta /tmp/ en Linux permite escribir directamente en la memoria RAM por lo que todos los datos temporales de la aplicación se alojan dentro de dicha carpeta. La memoria FLASH extraíble se accede a través de una carpeta. La Figura 44 muestra un esquema de la organización de carpetas de la aplicación.

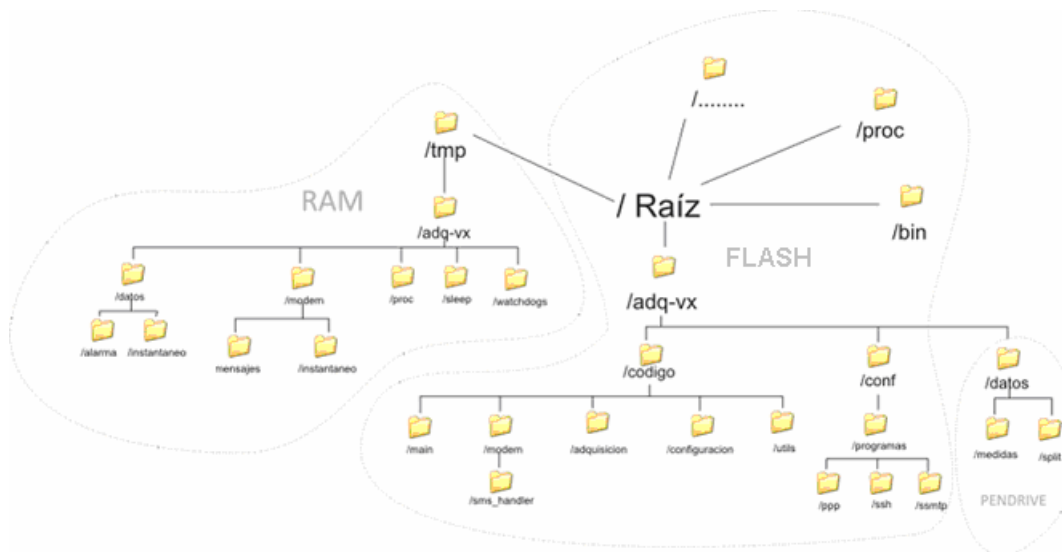


Figura 44 - Sistema de archivos

A continuación se detalla en forma resumida la estructura de directorios y su contenido:

/adq-vx: Códigos e información de la aplicación

/codigo: Programas de la aplicación

/adquisición: Programas del módulo de adquisición

/modem: Programas del módulo de comunicación

/sms_handler: Programas asociados al manejo de SMS

/main: Programas asociados al gestor de la aplicación

/utils: Utilitarios de las aplicaciones que son de para uso externo a la aplicación

/configuracion: Programas que permiten la configuración de la aplicación

/conf: Archivos de configuración de la aplicación y de otras aplicaciones que afectan el funcionamiento de la aplicación

/programas: Archivos de configuración asociados a programas externos a la aplicación

/ppp: Archivos de configuración de la comunicación mediante el protocolo PPP

/ssh: Archivos de configuración del cliente SSH

/ssmtp: Archivos de configuración del cliente de correo SMTP

/datos: Datos adquiridos

/medidas: Datos de medidas adquiridas

/split: Datos de medidas adquiridas a ser enviados por FTP

/tmp: Datos temporales en la memoria RAM

/adq-vx: Datos temporales de la aplicación

/módem: Datos temporales utilizados por módulo de comunicación

/mensajes: Mensajes SMS en formato de archivo

/instantaneo: Datos a ser enviados en respuesta a un pedido de datos instantáneo

/proc: Información de procesos padres que están corriendo

/sleep: Información de procesos que evitan que el adquisidor entre en modo standby (bajo consumo)

/datos: Información de medidas temporales

/instantaneo: Medidas instantáneas adquiridas en respuesta a un SMS pidiendo datos instantáneos

/alarma: Medidas que se salen de los parámetros configurados y que disparan alarmas

/watchdogs: Archivos utilizados por el watchdog de software

7.4 Comunicación entre procesos

Se buscó una forma de comunicación entre procesos sencilla que aprovechara la ventaja que ofrece el Sistema Operativo. La dificultad más importante en el intercambio de información es evitar los problemas de concurrencia. Los problemas de concurrencia se presentan al tratar de leer y escribir a la vez el mismo lugar de memoria. Por simplicidad se seleccionó la comunicación mediante archivos. Cada programa que quiere comunicarse con otro escribe en una carpeta especial, definida previamente, un archivo cuyo nombre contiene la información deseada. De esta forma no es necesario abrir el archivo para obtener la información evitando el problema de concurrencia. Esta forma de comunicación entre procesos es fácil de implementar con scripts de bash.

7.5 Lenguajes de programación

La mayoría de las aplicaciones fueron programadas usando scripts de bash. Este es un intérprete de comandos (shell) de Unix utilizado en la mayoría de los sistemas Linux. Permite realizar guiones de comandos (scripts) desde los cuales se pueden ejecutar otros programas, realizar operaciones, funciones lógicas. La ventaja más importante de este lenguaje es la gran cantidad de herramientas que permite utilizar; se puede emplear cualquier programa dentro de un script como si fuera una función.

Además, la ejecución del programa no requiere la realización de un proceso de compilación. Esto permite programar en cualquier editor de textos y probar su funcionamiento en forma inmediata, simplemente configurando los atributos del archivo para hacerlo ejecutable.

Para la comunicación con el módem se utilizó C-Kermit, un lenguaje de scripting orientado a la comunicación entre dispositivos con comandos especializados para este fin. Cuenta además con varias herramientas de programación como funciones lógicas y manejo de archivos.

Dado que algunos programas requerían acceso directo al hardware, se programaron en C. Este es un lenguaje de bajo y medio nivel utilizado tanto para programar grandes sistemas operativos como para aplicaciones.

Para el diseño del sitio Web se utilizaron los lenguajes HTML y PHP. El lenguaje HTML fue utilizado para el diseño de la estructura del sitio y su formato y el lenguaje PHP para desarrollar scripts.

7.6 Gestión de la aplicación: Programa Main.sh

7.6.1 Descripción

El programa Main.sh es el encargado de ejecutar los programas que controlan la adquisición y envío de datos. Ejecuta tareas por minuto, hora y día y se encarga de controlar el modo de bajo consumo del adquisidor. A continuación se listan las tareas realizadas por Main.sh:

- Controla la ejecución periódica de CtlAdq.sh
- Controla la ejecución de CtlModem.sh
- Acaricia el watchdog de hardware
- Actualiza la configuración del sistema
- Controla los watchdogs de software
- Sincroniza la hora del sistema y del RTC del procesador de acuerdo a la hora del RTC externo
- Guarda de forma periódica el log del sistema en el pendrive
- Borra los archivos de datos y log que superen la cantidad de días especificados en el archivo de configuración main.conf.
- Termina los procesos que hayan superado el tiempo de vida declarado en /tmp/adq-vx/proc.
- Controla el modo de bajo consumo
- Verifica cuál es el evento que saca al sistema del modo de bajo consumo

La Figura 45 muestra el diagrama de flujo del programa.

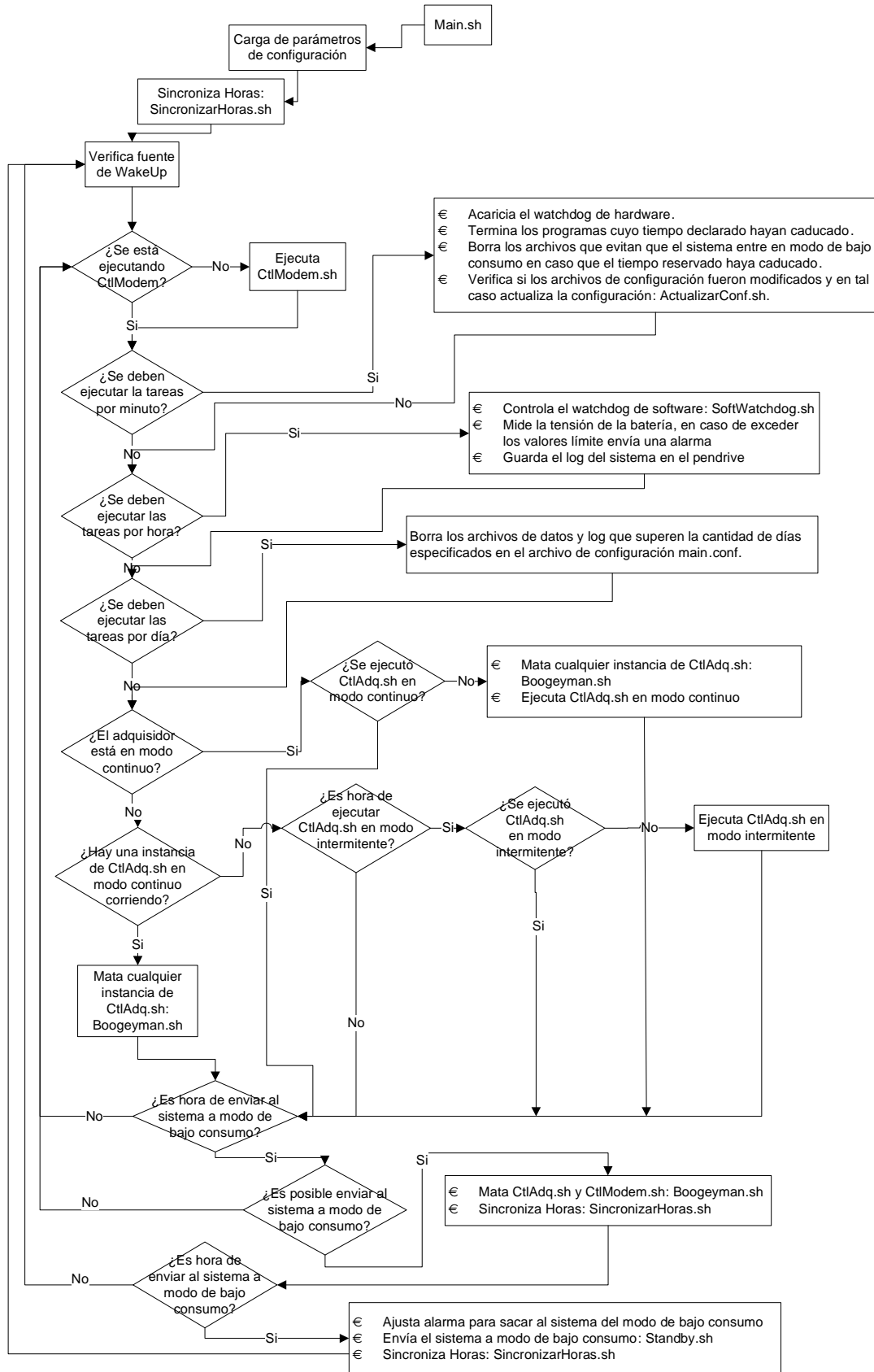


Figura 45 - Diagrama de flujo de Main.sh

Main.sh está compuesto por varios scripts. A continuación se detallan los más relevantes.

7.6.2 Sincronizar relojes: SincronizarHoras.sh

La aplicación tiene tres relojes: el reloj del sistema operativo, el RTC del procesador y el RTC externo. Los tres relojes son usados, por lo que fue necesario garantizar su sincronización. Se comprobó que al entrar y salir del modo de bajo consumo, el reloj del sistema operativo y el del procesador se retrasaban. Por esto se utilizó el RTC externo como referencia.

7.6.3 Actualización de configuración: ActualizarConf.sh

La configuración del sistema se realiza mediante archivos que contienen los parámetros usados por los programas. Para realizar una actualización de la configuración se deben modificar estos archivos. Se tuvo que implementar un mecanismo de actualización de configuración que evitara problemas de concurrencia. La concurrencia se da cuando un programa intenta modificar un archivo y otro intenta leer su contenido. Este evento puede causar resultados inesperados. ActualizarConf.sh implementa este mecanismo.

Existen tres archivos que contienen los parámetros de configuración: `adq.conf`, `modem.conf` y `main.conf`. Estos son los archivos accesibles para los usuarios. Sin embargo no son los utilizados por los programas para obtener los parámetros de configuración. ActualizarConf.sh genera tres archivos ocultos: `.adq.conf`, `.modem.conf` y `.main.conf`. Estos son los archivos de configuración usados por la aplicación.

ActualizarConf.sh verifica si hubo cambios en cada archivo de configuración. Por ejemplo, verifica si `adq.conf` y `.adq.conf` son diferentes. En caso de detectar un cambio de configuración, finaliza los procesos que usan el archivo modificado. Luego crea el archivo oculto a partir del archivo modificado. Por ejemplo, crea `.adq.conf` a partir de `adq.conf`. En caso de ser necesario vuelve a ejecutar los procesos finalizados.

7.6.4 Watchdog de software: SoftWatchdog.sh

El adquisidor fue diseñado para reponerse frente a fallas inesperadas. Esto es fundamental considerando que puede ser instalado por periodos prolongados en lugares alejados y sin supervisión. Main.sh acaricia una vez por minuto el watchdog del procesador. Si por algún motivo el watchdog permanece sin ser acariciado por más de veinte minutos, se reinicia el sistema operativo y la aplicación.

Además del watchdog del procesador se implementó un watchdog por software. Su necesidad surgió del análisis de las posibles fallas. Por ejemplo, en caso que CtlModem.sh falle, el adquisidor podría quedar incomunicado.

SoftWatchdog.sh verifica la fecha de modificación de todos los archivos ubicados en /tmp/adq-vx/watchdogs. En caso que la fecha de modificación de alguno de estos archivos no haya sido actualizada por más de dos días, se reinicia el sistema operativo y la aplicación. De esta forma, todos los procesos que necesiten un watchdog, sólo deben guardar un archivo en la dirección especificada y actualizar periódicamente su fecha de modificación.

7.6.5 Finalización de procesos: Boogeyman.sh

Una de las funciones de Main.sh es inicializar y finalizar los programas de adquisición y comunicación. Boogeyman.sh es un script que mata a los procesos que se pasen como argumentos de entrada y recursivamente a todas las generaciones posteriores. De esta forma se garantiza que, una vez finalizado el proceso padre, sus hijos no queden ejecutándose.

Boogeyman.sh manda la señal SIGSTOP al proceso para que no pueda seguir ejecutándose. De esta forma se garantiza que no ejecute más procesos. Luego busca cuáles de los procesos en ejecución son sus hijos. El sistema operativo guarda información de cada proceso en ejecución. En particular, cuál es su proceso padre. Esta información está accesible en la carpeta /proc. Por último, se ejecuta Boogeyman.sh para cada uno de los procesos hijos.

7.6.6 Modo bajo consumo: Standby.sh

En la estrategia de nuestro “intermitente” se envía al sistema a modo de bajo consumo durante los períodos en que no se adquieren datos. La distribución de Linux instalada en la computadora embebida tiene problemas con el manejo de los modos de bajo consumo. En particular, existen situaciones en las cuales no es posible sacar al sistema del modo “standby”. En el foro de la empresa Gumstix varios usuarios relataban este problema, aunque nadie identificaba la causa ni sugería una posible solución. Realizando varias pruebas se descubrió que cuando un proceso escribe en “stdout” (salida estándar) justo antes de enviar el sistema a “standby”, éste no es capaz de salir del estado. Para evitar este problema, Standby.sh realiza las siguientes acciones:

- Busca las consolas abiertas y las termina

- Mueve el dispositivo /dev/ttyS0 para que no se despliegue el login
- Mata al proceso que despliega el login
- Borra los buffer stdout (salida estándar) y stderr (error estándar)
- Envía el sistema a “standby”

Cuando un evento saca al sistema del modo “standby”, Standby.sh reestablece el dispositivo /dev/ttyS0 para que se despliegue el login del sistema.

7.7 Comunicación

7.7.1 Descripción

El módulo de comunicación comprende todos los programas que utilizan el módem. Se encarga de realizar las siguientes tareas:

- En el caso de solicitar datos instantáneos se encarga de responder de acuerdo a lo configurado, ya sea por e-mail, por FTP o por SMS.
- En caso de solicitar una consola remota crea una conexión SSH entre el adquisidor y el usuario. Además envía por e-mail la clave pública para autenticar la sesión.
- Se encarga de enviar los datos adquiridos durante el día a un servidor FTP.
- Envía datos de alarma por FTP, SMS o e-mail según lo que esté configurado.
- Descarga una nueva configuración desde el sitio Web cuando se lo solicita por SMS.
- Envía la configuración actual cuando se lo solicita por SMS.

A continuación se detallan los programas principales que componen dicho módulo se describe cómo se gestionan las tareas. En el capítulo 7.7.2 se describe al gestor principal del módulo de comunicación CtlModem, en el capítulo 7.7.3 se describe al programa SMSInt encargado de recibir los mensajes SMS y ejecutar las acciones correspondientes y en el capítulo 7.7.4 se describe al controlador de actividad de SMSInt, IfSMS. Estos son los programas fundamentales en la gestión de las tareas de comunicación.

Además de esto se describen las soluciones a dos de los principales problemas afrontados. En el capítulo 7.7.5 se describe la forma en que se conecta el adquisidor a Internet y en el capítulo 7.7.6 se describe cómo se logra establecer el túnel SSH.

Por último, se describen brevemente en el capítulo 7.7.7 los programas que se utilizaron para poder mandar e-mail, enviar y recibir archivos con un servidor FTP y establecer una conexión PPP.

7.7.2 Gestor de comunicación - CtlModem.sh

Durante el desarrollo del software del adquisidor surgió la necesidad de gestionar las tareas de comunicación en un único programa que se encargase de administrar el acceso al módem. La razón de esto radica en los problemas que pueden surgir al tratar de utilizar en simultáneo el módem para diferentes propósitos.

El gestor de comunicación tiene conocimiento de la tarea que está usando el módem y cuánto tiempo debe esperar para que finalice. De esta forma, si el tiempo máximo en que la tarea puede estar en ejecución es alcanzado, el gestor la finaliza forzosamente previniendo que fallas inesperadas desestabilicen el sistema. A dicho gestor se lo llamó CtlModem (Controlador de módem).

El módulo de comunicación funciona independientemente de los demás módulos. Así como los otros módulos, registra sus programas en los directorios /tmp/adq-vx/proc y /tmp/adq-vx/sleep para que el gestor principal Main sepa si puede o no entrar a modo de bajo consumo.

SMSInt es un programa independiente de CtlModem. El módem y el procesador están configurados para que cada SMS que llegue al módem genere una interrupción al procesador y se ejecute una instancia de SMSInt. Dicho programa se ejecuta de forma asíncrona pudiendo generar problemas si se ejecuta conjuntamente con CtlModem y ambos intentan utilizar simultáneamente el módem.

La solución a este problema está en la forma en que se implementó CtlModem. El diagrama de flujo de la Figura 46 muestra dicha solución.

Diagrama de flujo – CtlModem.sh

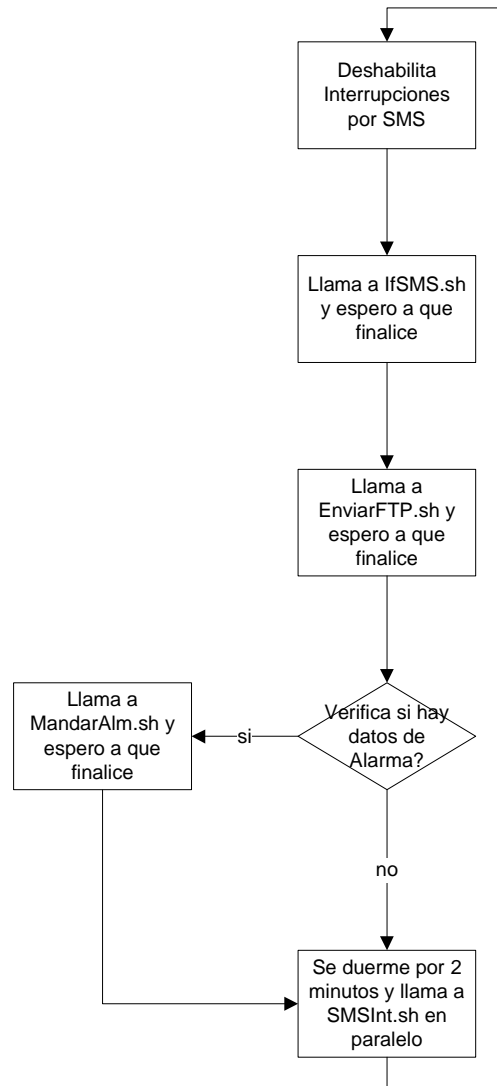


Figura 46 - Diagrama de flujo de CtlModem.sh

Para resolver el problema de concurrencia con el programa SMSInt, la primera tarea de CtlModem antes de comenzar con subprogramas que utilizan el módem es deshabilitar las interrupciones por SMS y verificar la existencia de alguna instancia de SMSInt mediante el programa IfSMS. Al deshabilitar las interrupciones CtlModem se asegura de que será el único en tratar de acceder al módem. IfSMS verifica en qué estado está el módem y espera a que finalice la instancia de SMSInt que se encuentra en ejecución. Si IfSMS considera que se excedió el tiempo límite para atender el mensaje solicitado finaliza forzosamente la instancia de SMSInt. De esta forma se garantiza que el módulo de comunicación continúe con las demás tareas. Luego de

finalizar IfSMS CtlModem se asegura que no hay ningún proceso utilizando el módem y entonces ejecuta las tareas de rutina. En el capítulo 7.7.3 se detalla el funcionamiento de IfSMS.

La primera tarea de rutina que ejecuta CtlModem es verificar si existen datos adquiridos a enviar y si es la hora correspondiente mediante el script EnviarFTP. Si EnviarFTP encuentra que es momento de mandar los datos llama a los scripts Selecciona y SuboDatos, programas que se encargan de comprimir la información y luego enviarla. La segunda tarea de rutina que ejecuta CtlModem es verificar la existencia de alarmas para enviar. En este caso ejecuta el script MandarAlm que se encarga de mandar los datos de alarma por el medio especificado el archivo modem.conf.

Durante el tiempo que CtlModem ejecuta las tareas de rutina, no es capaz de atender los mensajes de texto entrantes. Por esta razón, como última tarea antes de dormirse, CtlModem lanza una instancia de SMSInt en paralelo y se duerme por dos minutos. De esta forma CtlModem se asegura que los mensajes que pudieron haber llegado mientras las interrupciones por SMS estaban deshabilitadas son atendidos inmediatamente luego de que finaliza la ejecución de las tareas de rutina. Al finalizar, SMSInt vuelve a habilitar las interrupciones por SMS. Esto permite que cualquier mensaje que llegue sea atendido inmediatamente.

7.7.3 Receptor de SMS – SMSInt.sh

SMSInt se ejecuta cuando se recibe una interrupción del módem indicando que hay un nuevo mensaje SMS. SMSInt se encarga de atender todos los pedidos por SMS. Se comunica con el módem para obtener la información del mensaje, atiende el pedido y luego borra el mensaje.

Al comenzar el desarrollo de las aplicaciones se notó una gran dificultad al intentar obtener del módem los datos específicos de un SMS. Era prácticamente igual de complejo preguntar solamente por el teléfono del remitente que obtener toda la información (teléfono, mensaje y lugar en donde se almacenaba el mensaje). Así también, cuando se intentaba borrar un mensaje y no se tenía éxito se generaban problemas pues, al no tener forma de marcarlo como ya atendido con éxito, se volvía a atender dicho mensaje. Se decidió entonces crear una copia en la memoria RAM de la información contenida en el módem sobre los mensajes de texto. En la Figura 47 se puede ver el diagrama de flujo de SMSInt

Diagrama de flujo de SMSInt.sh

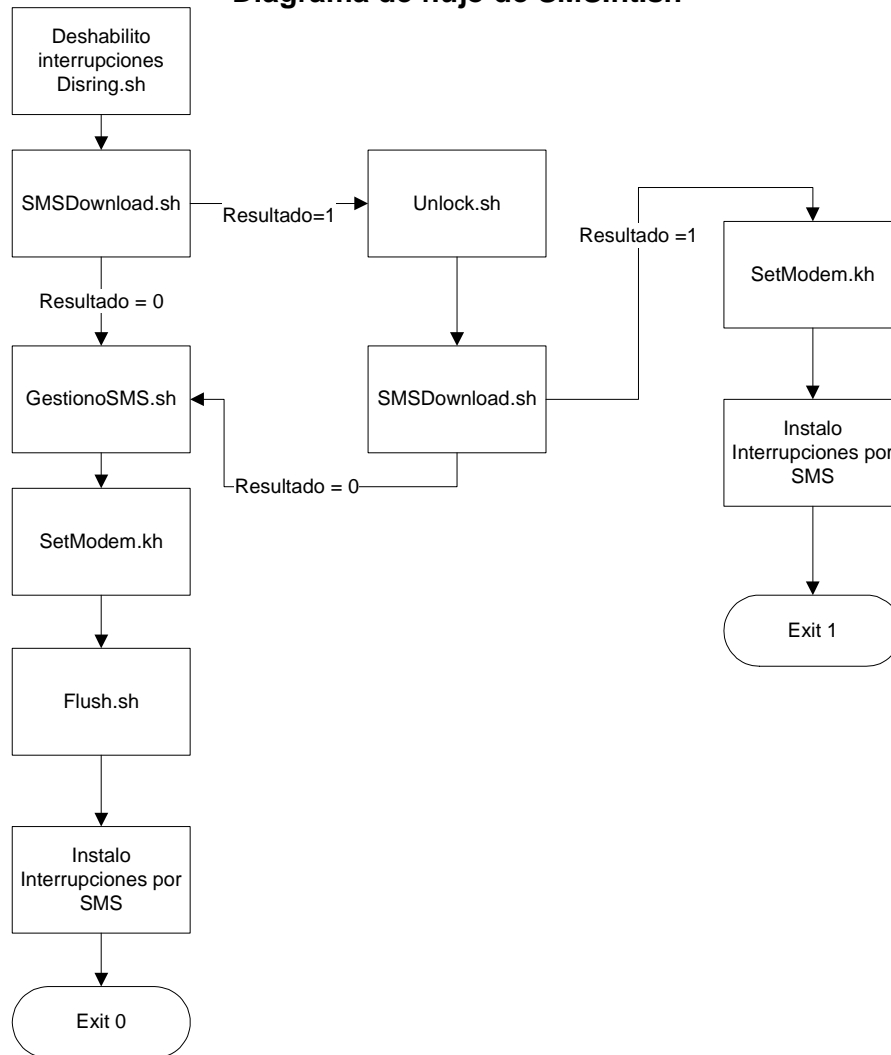


Figura 47 - Diagrama de flujo SMSInt.sh

Lo primero que ejecuta SMSInt es el script DisRing (Disconnect Ring), el cual deshabilita las interrupciones. Esto asegura que no se ejecute otra instancia de SMSInt mientras haya una en progreso.

Luego el programa SMSDownload transforma cada mensaje de texto en un archivo en el directorio /tmp/adq-vx/modem/mensajes. El nombre del archivo es un número, correspondiente al lugar donde se aloja en la tarjeta SIM del módem el mensaje, seguido de un punto y una letra. Dicha letra puede ser una "u" (Unattended) indicando que dicho mensaje no está procesado, o una "d" (Delete) indicando que dicho mensaje

ya se atendió y debe borrarse. El propósito de guardar la ubicación del mensaje en la SIM es para facilitar borrarlo. El contenido del archivo es el siguiente:

- MENSAJE=[mensaje del SMS]
- PHONE=[teléfono del remitente]

Esta forma de almacenar la información del SMS facilita el manejo de datos. Si se quiere saber el mensaje o el teléfono se carga con el comando “source” de bash el archivo y se tiene guardado dentro del programa el mensaje y el teléfono en las variables MENSAJE y PHONE, respectivamente.

Cuando no se tiene éxito al descargar los mensajes se reintenta, habiendo ejecutado previamente un programa llamado Unlock. Dicho programa procura dejar el módem desbloqueado. Busca por conexiones que estén ocupando el puerto de comunicación del módem y las finaliza. Asimismo, busca conexiones GPRS que estén activas o se hayan cerrado mal y las finaliza correctamente. Como último paso vuelve a configurar el módem.

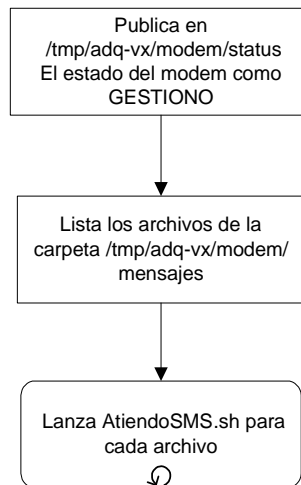


Figura 48 - Diagrama de flujo GestionoSMS.sh

Teniendo en la carpeta /tmp/adq-vx/modem/mensajes la información actualizada, SMSInt ejecuta GestionoSMS el cual lista los mensajes y los atiende uno por uno. Por cada mensaje, GestionoSMS llama a AtiendoSMS pasándole como parámetro el mensaje a atender. AtiendoSMS ejecuta una acción en función al contenido del mensaje. En la Figura 48 se puede ver el diagrama de flujo de GestionoSMS.

Luego que se termina de atender el mensaje y se considera que el mismo se puede borrar se cambia la extensión de .u por .d y se trata de borrar del módem. En caso que

no se tenga éxito al comunicarse con el módem, no se borra el mensaje de la tarjeta SIM pero igualmente el archivo queda con extensión .d. Cuando se traten de bajar nuevos mensajes, si se encuentra que ya existe el mismo archivo con la extensión .d no se descarga, previniendo la atención reiterada de un mensaje. Cuando se tiene éxito en borrar el mensaje en el módem se borra el archivo de la carpeta /tmp/adq-vx/modem/mensajes.

Luego de atendidos todos los mensajes se ejecuta SetModem, el cual reconfigura el módem, previniendo que éste quede mal configurado.

Al finalizar, se ejecuta el programa Flush, el cual chequea si existen mensajes para borrar, y si es así se conecta con el módem para borrarlos.

Con esta solución se hace mínima la cantidad de accesos al módem y mejora enormemente la facilidad con la que se maneja la información.

7.7.4 Control de actividad del receptor de SMS – IfSMS.sh

Al comenzar el bucle de CtlModem, el primer subprograma que se ejecuta es IfSMS. Es de vital importancia asegurarse que el acceso al módem va a estar disponible durante el chequeo de rutina por el eventual uso del mismo. Es por esto que existe IfSMS. Este programa se asegura de finalizar cuando no haya ningún subproceso utilizando el módem. Pueden distinguirse dos casos diferentes al entrar al bucle: el primer caso es que IfSMS encuentre una instancia de SMSInt en progreso, lo cual significa que se está atendiendo un pedido y por lo tanto utilizando el módem, y el segundo caso es que no se encuentre una instancia de SMSInt en progreso, lo que significa que se puede continuar con el bucle sin problemas pues no se está utilizando el módem. El primer caso es donde IfSMS adquiere sentido.

IfSMS se asegura de que, de haber una instancia de SMSInt utilizando el módem, se va a esperar un tiempo límite donde se asume que la instancia de SMSInt está funcionando mal o está colgada y, en tal caso, finaliza la instancia de SMSInt. Este tiempo límite varía según el mensaje que se esté atendiendo, por esto es que hay un archivo en /tmp/adq-vx/modem llamado status donde cada programa que está atendiendo un pedido escribe en dicho archivo lo que está haciendo. De esta forma IfSMS diferencia según la instancia de SMSInt que se está ejecutando. Por ejemplo, si se está atendiendo un pedido de SSH IfSMS toma como tiempo máximo 2 horas y también chequea la actividad de la conexión, si detecta inactividad por más de 6

minutos considera que la conexión está “colgada”. Si es un pedido de envío de la clave SSH IfSMS toma como tiempo máximo 5 minutos.

En caso de que se agote el tiempo y que se considere necesario finalizar la instancia de SMSInt por presumir que se encuentra de alguna forma “colgada” se realiza un procedimiento de finalización. Esto consiste en finalizar todo tipo de comunicación con el módem, reconfigurarlo y resetear el servicio de SSH. Es necesario resetear el servicio SSH ya que al finalizar una conexión forzosamente se generan conflictos que de esta forma se solucionan.

7.7.5 Conexión a Internet – Con2Int.sh

La conexión a Internet del adquisidor se realizó a través de la red GPRS. La misma debe tener el máximo de confiabilidad posible ya que de ella depende el acceso remoto. Para establecer una conexión, el adquisidor pide al módem que se conecte a la red GPRS. Luego crea una sesión PPP con el módem. En la Figura 49 se muestra un esquema de la conexión.

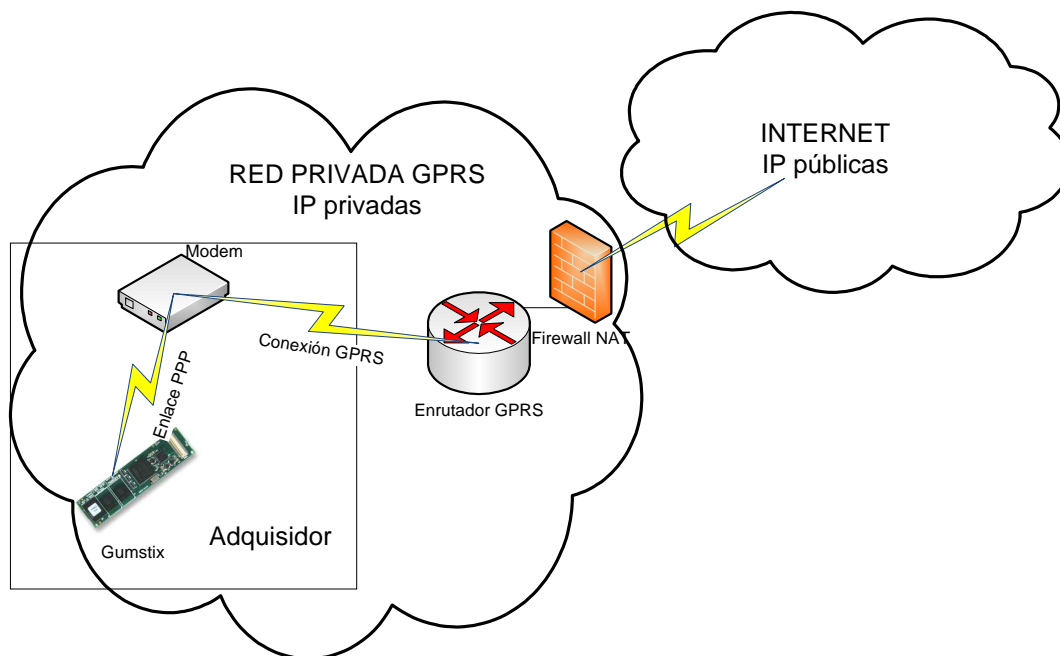


Figura 49 - Esquema de la conexión del adquisidor a Internet

En ocasiones el módem realiza el procedimiento de conexión con la red GPRS de forma exitosa. Sin embargo, a menudo cuando se trata de utilizar la conexión se encuentra que no hay respuesta del servidor. Analizando el problema se encontraron dos posibles fuentes de error, la red y el módem. Para descartar problemas con la red

GPRS se probó la tarjeta SIM en otros dispositivos como celulares y palms. En dichos dispositivos se encontró que la conexión con la red GPRS tenía conectividad el 100% de las veces. De esta manera se descartó que el problema fuera de la red. Para descartar problemas de comunicación entre el adquisidor y el módem se realizaron pruebas de conexión directamente desde el módem (este posee un stack TCP/IP embebido lo que permitió dichas pruebas). En el módem no se encontraron problemas de configuración ni de otro tipo. Se encontró que en las mismas condiciones de configuración en un intento lograba conectividad y en otro no. Se buscó obtener una respuesta del fabricante pero no se logró encontrar una explicación. Frente a este problema, el adquisidor debe probar la conectividad antes de tratar de utilizar la conexión para descartar un error de la aplicación que utiliza la conexión. Frente a la exigencia de la aplicación en cuanto a la confiabilidad de la conexión se determinó que los programas que necesitaran conexión a Internet deberían utilizar un mecanismo de verificación de la conectividad que reintentara en caso de falla.

El programa Con2Int (Conexión a Internet) luego de conectar al módem con GPRS y levantar una sesión PPP manda una serie de paquetes de PING y si no recibe respuesta toma a la conexión como fallida. En caso de falla prende y apaga al módem, se asegura de que haya quedado realmente desbloqueado y configurado y sigue reintentando hasta obtener una respuesta al paquete enviado mediante PING. De llegar al quinto intento, el módem es reseteado por Con2Int y se reintenta tres veces más. Generalmente no se llega al paso donde se debe resetear el módem, pero si se llegara a este punto, es muy probable obtener conectividad.

7.7.6 Consola remota

Como requerimiento de la aplicación se solicitó poder acceder al equipo desde un PC a través de Internet para poder gestionar, corregir fallas y tomar total control del adquisidor. Se utilizó el protocolo SSH (Secure Shell). Este protocolo permite tomar control total a través de un intérprete de comandos y a diferencia de protocolos como TELNET, los datos viajan encriptados.

El principal problema para cumplir con este requerimiento es que el adquisidor no puede funcionar como servidor a través de la red GPRS. Esto es porque la red GPRS es privada la cual se encuentra a través de un Firewall NAT que imposibilita a un usuario desde Internet comenzar una comunicación con el Adquisidor. El problema

entonces se traduce en que el primer paso en cualquier comunicación desde Internet con el adquisidor lo debe realizar siempre el adquisidor.

Para esto, se ideó una solución donde el usuario logra contactar al adquisidor para notificar que necesita establecer una comunicación SSH, el adquisidor entonces se conecta con el usuario en Internet y luego el usuario sí puede conectarse con el adquisidor. Para notificar al adquisidor que se quiere establecer un enlace SSH, el usuario manda un mensaje SMS al adquisidor. En la Figura 50 se muestra esto de forma gráfica.

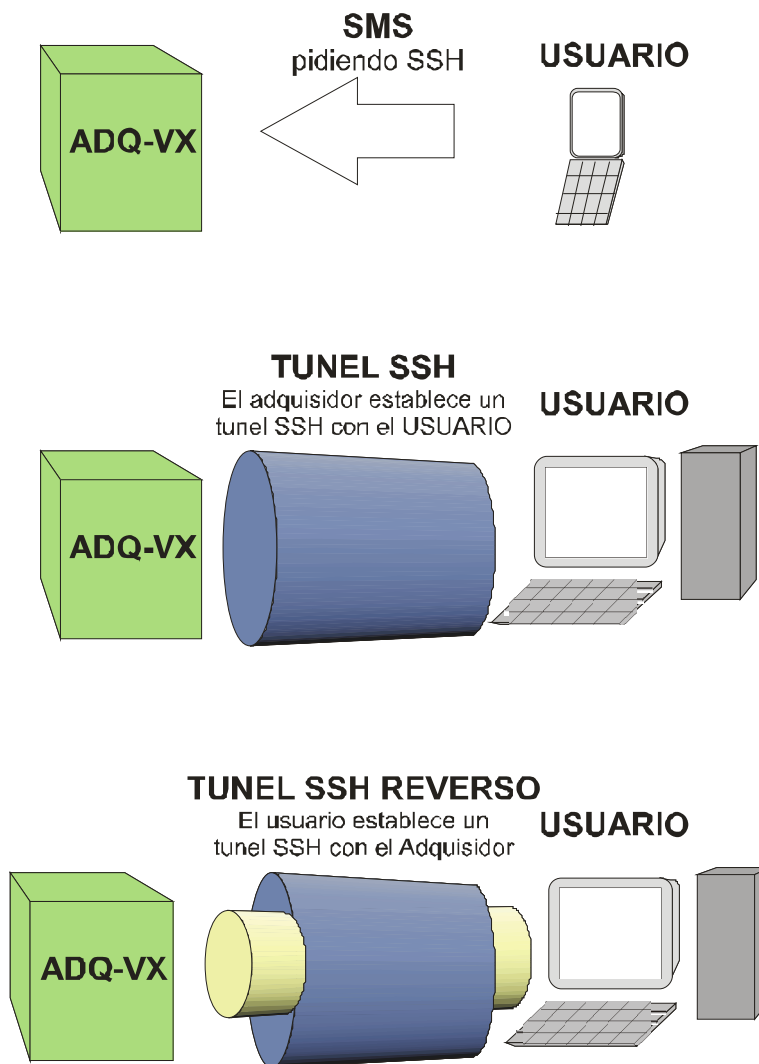


Figura 50 - Proceso para conexión mediante SSH

El usuario entonces manda un mensaje SMS al adquisidor con la palabra clave SSH, un nombre de usuario y su dirección IP pública, de manera que el adquisidor pueda localizarlo en Internet.

Además de este problema había que idear una solución para que luego de tener la dirección IP del usuario, el mismo pudiera conectarse con el adquisidor. Esto se hizo a través un túnel reverso SSH. En el túnel reverso SSH se necesitan dos equipos con servidor y cliente SSH. El equipo A que se encuentra atrás del Firewall NAT establece un túnel SSH con el equipo B que posee una IP pública. Seguidamente, el equipo A redirecciona un puerto P en el equipo B para que todo lo que se le solicite en el equipo B al puerto P se redireccione automáticamente hacia el equipo A. Luego el equipo B establece un túnel SSH hacia sí mismo pero a través del puerto P, lo que da como resultado que se redireccione el túnel hacia el equipo A obteniendo un túnel SSH desde B hacia A. En nuestro caso particular, el equipo A es el Adquisidor y el equipo B es el usuario.

Un último problema que se debió afrontar para la automatización mediante scripts del proceso de conexión con el usuario fue el manejo de password. Además de ser inseguro mandar por un mensaje SMS el usuario y password, es muy difícil para un script de BASH esperar el pedido del password y luego seguir con el proceso automático. Para poder hacer esto se necesitan programas como Expect donde el script espera una palabra en particular para continuar con la ejecución, pero no fue posible instalar dichos programas dentro del adquisidor. Se resolvió utilizar claves compartidas. Las claves compartidas funcionan como se detalla a continuación:

- El adquisidor genera las claves, una privada para uso propio y una pública.
- La clave pública es entregada al usuario que quiere acceder al adquisidor vía e-mail.
- El adquisidor establece la conexión con el usuario.

7.7.7 Otros programas utilizados

Dentro del módulo de comunicación se utilizaron programas no desarrollados por el grupo, que permitieron realizar tareas complejas de una forma simple. A continuación se describen brevemente:

- CURL: Es un potente cliente para mandar o recibir archivos mediante los siguientes protocolos HTTP, HTTPS, FTP o TELNET. La forma en que interactúa con el usuario es a través de líneas de comandos, lo que permite utilizarlo dentro de los scripts. Este programa fue utilizado para automatizar el intercambio de archivos con el servidor FTP, ya sea para subir los datos diarios o subir y bajar la configuración.

- SSMTTP: Es un cliente para mandar correo electrónico mediante el protocolo SMTP (Simple Mail Transfer Protocol). Es muy sencillo de usar y se ejecuta mediante líneas de comando.
- PPP: Es una herramienta para establecer enlaces entre dos equipos utilizado en conexiones ADSL, conexiones con módems, conexiones de discado y otros tipos de conexiones punto a punto. Los drivers de PPP trabajan conjuntamente con el demonio pppd para establecer y mantener un enlace entre los equipos y para negociar las direcciones IP de cada lado de la conexión. Esta herramienta se utilizó conjuntamente con el programa Chat para crear un enlace con el módem y conectarse a la red GPRS.
- SSH: Es el cliente de consola remota.
- SSHD: Es el servidor de SSH. El mismo espera conexiones SSH entrantes para atenderlas, autenticarlas y dar acceso a una consola.

7.8 Adquisición – Programa CtlAdq.sh

7.8.1 Descripción

La adquisición de datos se enfocó hacia una arquitectura que tuviese la flexibilidad para agregar nuevos sensores con muy pocas modificaciones y que fuese independiente de las rutinas que se encuentren en ejecución simultáneamente. Basta con agregar una nueva rutina específica si es que las existentes no contemplan los requerimientos, la cual será gestionada por CtlAdq.

El script CtlAdq se encarga de ejecutar las rutinas de medida identificando el tipo de sensor del archivo de configuración *adq.conf*. Este archivo contiene todos los parámetros necesarios para el funcionamiento y la identificación de los sensores (ver 7.9.1).

CtlAdq.sh \$modo

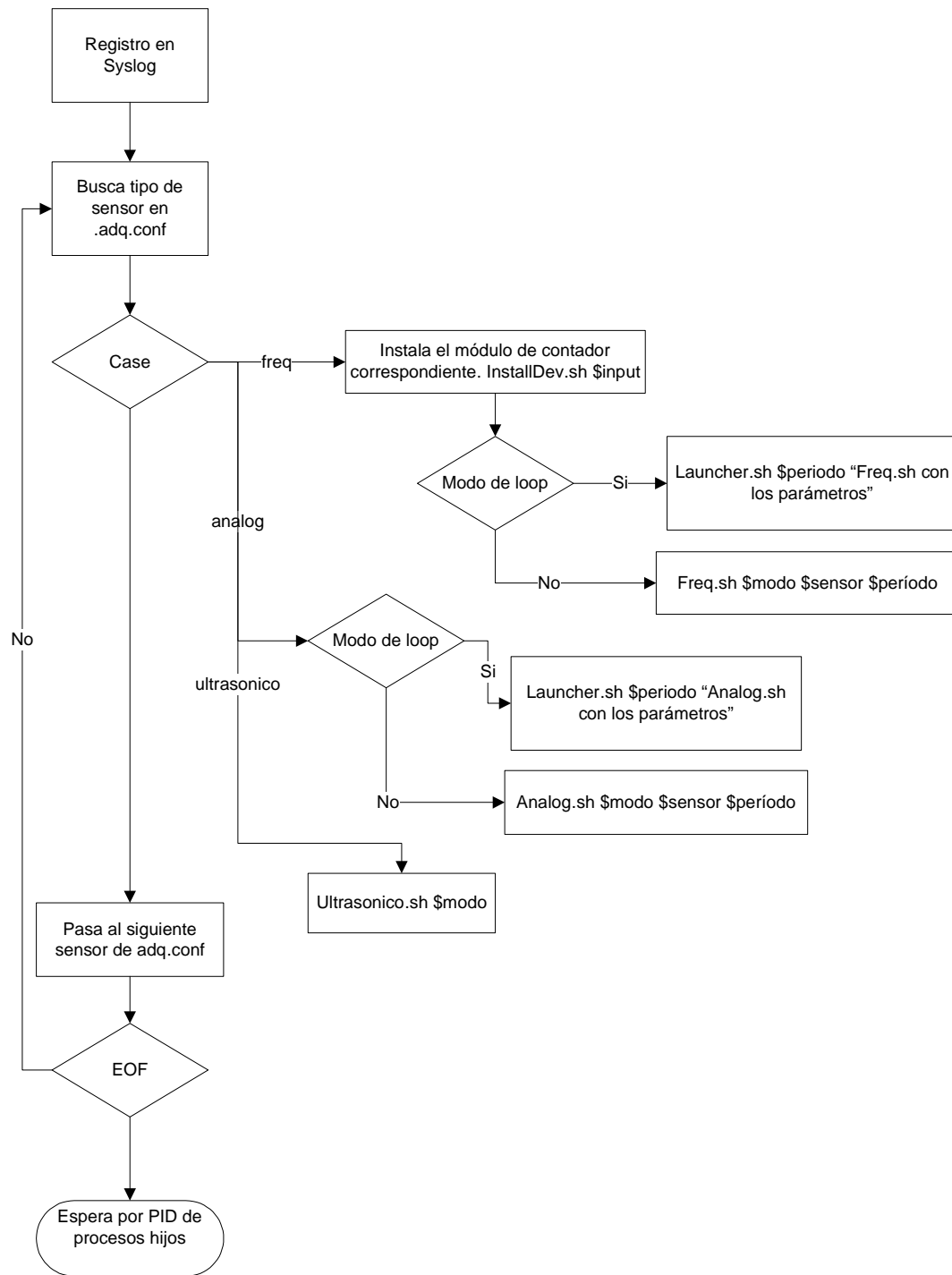


Figura 51- Diagrama de flujo de CtlAdq.sh

Este programa debe registrar su tiempo estimado de vida en un directorio específico ya que su ejecución condiciona el cambio del sistema a un modo de bajo consumo (capítulo 7.6.4).

La configuración de cada sensor tiene un parámetro llamado tipo que identifica el programa de adquisición a utilizar.

CtlAdq recibe como único parámetro el modo de adquisición. Este parámetro además de ser utilizado para estimar el tiempo de vida del proceso, es pasado como parámetro de entrada a los procesos hijos que son los encargados de la adquisición.

Los modos de adquisición son: intermitente, continuo, instantáneo y verboso.

En modo intermitente las rutinas de adquisición guardan un único valor promedio en un período de adquisición definido en la configuración (ver capítulo 7.9.1). Un caso típico de uso de este modo es adquirir un dato de velocidad y dirección del viento por hora, utilizando los últimos 10 minutos de cada hora. La rutina encargada de definir cuándo se realiza la adquisición es Main ya descrita en el capítulo 7.6.

En modo continuo las rutinas de adquisición son ejecutadas una única vez y permanecen adquiriendo y guardando datos periódicamente según los parámetros de configuración. Este es uno de los casos en que CtlAdq no termina de ejecutarse y el fin de los procesos es determinado por Main.

Para los dos modos de adquisición ya descritos, los datos son guardados en archivos independientes por sensor y por día. Estos archivos son nombrados de la siguiente manera: NombreDelSensor_DD-MM-AAAA (ejemplo: Veleta_30-06-2008).

El modo instantáneo de adquisición es utilizado para las solicitudes por SMS. Este es un modo en el cual los procesos de adquisición toman un único valor en un periodo de tiempo corto que es configurable y lo graban en un directorio específico. Este archivo es usado por las rutinas de comunicación para enviar la respuesta a la solicitud de datos instantáneos.

El modo verboso fue creado para realizar pruebas. En este modo los procesos de adquisición no guardan los datos sino que los despliegan en la consola.

Durante el proceso de prueba de la primera versión del software de adquisición, se detectó que en los modos continuo y verboso el tiempo de ejecución de las sentencias de los scripts tenía un efecto de defasaje en los timestamp de las medidas. Para cada medida el programa ejecutaba varias sentencias cuyos tiempos de ejecución son

variables. Esto generaba un defasaje aleatorio en el timestamp que sumado en cada iteración introducía un defasaje significativo.

Para solucionar este problema se decidió que las rutinas de adquisición debían devolver un único valor y que otro proceso se encargara en forma paralela de esperar el momento exacto para ejecutarlas. Este programa es *Launcher*, que en función a los períodos configurados para los modos continuo y verboso y la hora del sistema, ejecuta una instancia del programa de adquisición. Se puede pensar esta solución como un scheduler.

Una vez que *CtlAdq* ejecuta todas las rutinas de adquisición espera a que éstas terminen para finalizar. De esta forma se asegura que no queden procesos “huérfanos”.

Los script de adquisición (Freq y Analog) además chequean que los valores leídos no estén dentro del rango de alarma, si es que las alarmas fueron configuradas para el sensor.

En caso de detectar una condición de alarma, el programa registra este dato en un directorio específico que es chequeado periódicamente por las rutinas de comunicación para luego ser enviada.

Podemos resumir la estructura de los programa de medida de la siguiente forma:

1. Se cargan los parámetros de configuración del sensor
2. Se realiza la lectura de la señal correspondiente
3. Se escala el resultado de la lectura
4. Se verifica la ocurrencia de una condición de alarma
5. Se imprimen los datos en el archivo correspondiente o en consola según el modo de adquisición

El formato de datos adquiridos puede ser exportado a Excel facilitando el tratamiento de la información para el usuario. A continuación se presenta un ejemplo de dicho formato:

Anemómetro medida: 2.5 m/s TimeStamp: 30-06-2008 21:00:00

7.8.2 Medida de frecuencia

Los anemómetros estándar devuelven una onda cuadrada cuya frecuencia es proporcional a la velocidad del viento. Para medir la frecuencia se optó por desarrollar un módulo de kernel para contar interrupciones en un GPIO del procesador. De esta forma se puede utilizar la señal digital como generadora de interrupciones y contarlas durante una ventana de tiempo conocida.

Para programar este módulo fue utilizado el lenguaje C. Se pueden identificar cuatro partes fundamentales de este módulo: la inicialización, el irq-handler, la lectura de datos y la remoción.

La inicialización se realiza cuando el módulo es instalado. En primer lugar se obtiene un número mayor, el cual identifica la clase del módulo y es asignado por el kernel. Luego se registra el módulo y se crea la clase, lo que genera un archivo en /dev. Por último se inicia el monitoreo de las interrupciones con el irq-handler.

El irq-handler implementa un contador circular que con cada interrupción en la GPIO específica se incrementa en uno.

Una lectura del contador devuelve tres valores, el primero es el número actual del contador, el segundo es la hora actual en segundos y el tercero son los microsegundos de la hora actual. Los valores de tiempo están referidos al momento de la consulta y no al tiempo exacto de la última interrupción. Este punto será tratado con más detalle en el capítulo 8: Incertidumbres en la medida.

Por último, al remover el módulo del sistema se detiene el monitoreo de interrupciones, se destruye la clase creada y se desregistra el driver.

Utilizando esta herramienta, la medida de frecuencia consiste en realizar dos lecturas del contador espaciadas un tiempo conocido. Dividiendo la cantidad de interrupciones registradas entre el tiempo se obtiene un valor promedio de la frecuencia de la onda en ese periodo.

Para integrar esta solución al adquisidor se implementaron ocho módulos contador, uno para cada entrada digital. Los mismos son instalados por CtlAdq con el script InstallDev que verifica que el módulo no esté instalado para realizar su instalación.

Una vez instalado el módulo, es el programa Freq el encargado de realizar las consultas al contador y calcular la frecuencia.

7.8.3 Medición de entradas analógicas

Las entradas analógicas son convertidas a señales digitales mediante un conversor analógico digital de 12 bits que se comunica con el procesador con el protocolo I2C. Por lo tanto, para obtener el valor de una entrada analógica es necesario contar con un driver que maneje este estándar.

Para interactuar con el driver se utilizó un programa que implementa una interfaz en shell para la comunicación con cualquier dispositivo i2c estándar. Este software devuelve el valor de la entrada en hexadecimales respecto a la tensión de referencia del conversor.

El programa de medida Analog identifica el canal a leer del conversor y escala la respuesta entre un máximo y un mínimo definidos en la configuración. Al igual que en la medición de frecuencia, este programa devuelve un valor promedio del periodo de lectura. Para esto se creó un parámetro de configuración que indica el período de muestreo. Si se configura la entrada analógica para obtener un dato en una ventana de X segundos con un período de muestreo de T segundos, el valor a guardar será el promedio de las X/T mediciones.

7.8.4 Sensor serie

Además de los sensores que traducen las magnitudes medidas en algún tipo de señal eléctrica, existen sensores que cuentan con un módulo de procesamiento que les permite trabajar con las señales de los transductores, generar información y comunicarla mediante algún protocolo de comunicación serie.

Dentro de esta categoría se encuentra el sensor ultrasónico que es utilizado para medir velocidad y dirección del viento. Este sensor realiza la medición de velocidad del viento en las tres dimensiones devolviendo las componentes del vector velocidad. Una de las modalidades de comunicación de estos sensores es mediante el protocolo RS-232. Una vez configurado el sensor devuelve todas las medidas en formato texto por el puerto serie.

Para realizar la comunicación se configura el puerto a la velocidad establecida en la configuración del sensor. Con la comunicación establecida el sensor ultrasónico redirecciona la salida del puerto serie al archivo correspondiente del directorio /adq-vx/datos/medidas/ que se encuentra en el pendrive.

La adquisición desde el puerto serie también utiliza los modos de adquisición continuo e intermitente, definiendo los rangos de tiempo en que la comunicación debe ser redireccionada al archivo de datos.

7.9 Configuración

Los parámetros de configuración del adquisidor se dividieron en tres archivos: *adq.conf*, *modem.conf* y *main.conf*. Todos los scripts que utilizan algún parámetro configurable se dirigen a estos archivos para obtener la información.

adq.conf – Contiene toda la información necesaria para configurar un sensor, así como los parámetros generales requeridos por las rutinas de adquisición.

modem.conf – Contiene toda la información necesaria para configurar las rutinas de comunicación respecto al envío de e-mails, sms y transferencias FTP.

main.conf – Contiene parámetros que utiliza el script Main para la gestión global del equipo.

Para poder modificar los parámetros del equipo se desarrollaron dos interfases de usuario que permiten alterar los tres archivos. La primera es un script que despliega menús de opciones en la consola del adquisidor (configuración local) y la segunda es un sitio Web que, a través del uso de formularios, permite modificar los archivos, que luego serán descargados por el adquisidor (configuración remota).

7.9.1 Parámetros de configuración

El archivo de configuración *adq.conf* contiene 4 parámetros generales referidos a los periodos de adquisición, uno por modo.

T_INTERMITENTE: Es el tiempo total de adquisición en minutos por hora con el adquisidor en modo intermitente.

T_CONTINUO: Es el tiempo en minutos entre dos datos consecutivos en modo continuo.

T_VERBOSO: Es el tiempo en segundos entre dos datos consecutivos en modo verboso.

T_INSTANTANEO: Es el tiempo de adquisición en segundos para una solicitud de datos instantáneos.

Todas las rutinas de adquisición, independientemente del tipo de sensor, consultan estos datos para cargar sus parámetros.

Además de estos cuatro parámetros, *adq.conf* contiene la configuración individual de cada sensor. Se diferencian en este punto tres conjuntos de parámetros, uno por cada tipo de sensor.

Para los sensores cuya salida es en frecuencia, los parámetros son los siguientes:

tipo: Indica el tipo del sensor, en este caso es "freq".

nombre: Es el nombre que se le da al sensor.

unidad: Unidad de medida del valor obtenido.

fc1: Es la pendiente de la curva de calibración (unidad/Hz).

fc2: Es el termino independiente de la curva de calibración.

max: Es el valor máximo que puede tomar la medida antes de generar alarma.

min: Es el valor mínimo que puede tomar la medida antes de generar alarma.

Para los sensores cuya salida es una señal analógica, los parámetros son los siguientes:

tipo: Indica el tipo del sensor, en este caso es "freq".

nombre: Es el nombre que se le da al sensor.

unidad: Unidad de medida del valor obtenido.

fe: Es el valor entero correspondiente al fondo de escala del conversor AD.

me: Es el valor entero correspondiente al cero del conversor AD.

T: Es el periodo de muestreo. Este parámetro define la cantidad de muestras a promediar.

max: Es el valor máximo que puede tomar la medida antes de generar alarma.

min: Es el valor mínimo que puede tomar la medida antes de generar alarma.

Para los sensores cuya comunicación con el adquisidor se hace mediante un puerto serie los parámetros son los siguientes:

tipo: Indica el tipo del sensor, en este caso es "ultrasónico".

nombre: Es el nombre que se le da al sensor.

speed: Es la velocidad de la comunicación serie en bps.

Cada sensor está identificado por un número en el archivo que es utilizado de forma interna por los programas.

El archivo de configuración *modem.conf* contiene los siguientes parámetros:

INSTANTANEO: Indica el modo de respuesta a una solicitud de datos instantáneos, toma los valores SMS o EMAIL.

FTPSERVER: Es la dirección del servidor FTP utilizado por el sistema.

FTPUSER: Es el nombre de usuario para ingresar al servidor FTP.

FTPPASS: Es la contraseña utilizada para ingresar al servidor FTP.

ALARM: Indica el modo de envío de alarmas, toma los valores SMS, EMAIL o FTP.

ALARMEMAIL: Es la dirección de e-mail para el envío de alarmas.

ALARMSMS: Es el número de celular para el envío de alarmas.

El archivo de configuración *main.conf* es utilizado con tres fines diferentes: para comunicación entre procesos, para fijar los límites de alarma de la alimentación, y para determinar la cantidad de días de respaldo de los datos adquiridos.

MODO: Indica el modo de funcionamiento actual del adquisidor. Este parámetro puede tomar los valores continuo o intermitente.

vbatt.max: Es el valor máximo de alimentación del sistema a partir del cual es generada una alarma.

vbatt.min: Es el valor mínimo de alimentación del sistema a partir del cual es generada una alarma.

tiempo.archivos: Es el tiempo de vida de un archivo de datos en el sistema.

7.9.2 Configuración local

Dado que una modificación errónea de un archivo de configuración puede causar problemas en las aplicaciones, se decidió desarrollar una interfaz de usuario amigable para realizar la configuración de adquisidor. El programa verifica que cada parámetro ingresado. En caso de que este valor no sea correcto el usuario debe ingresar nuevamente el valor.

Todos los datos son guardados en un archivo temporal en */tmp/adq-vx* para realizar las modificaciones una vez que el usuario está seguro de los cambios que va a realizar.

El menú principal es desplegado por el script *Conf*. Éste contiene nueve opciones de configuración como se ve a continuación:

```
=====Elija una de las siguientes opciones=====
1 -Agregar nuevo sensor
2 -Modificar parámetros generales
3 -Modificar configuración actual de sensores
4 -Borrar configuración de sensores
5 -Visualizar configuración actual de sensores
6 -Modificar parámetros de comunicación
7 -Cambiar modo de operación [ continuo ]
8 -Modificar valores de alarma de Batería
9 -Modificar N° de días de almacenamiento [ 100 ]
0 -Salir
=====
```

Este script permite correr uno de los nueve scripts encargados de cada una de las configuraciones en función a la entrada del usuario.

7.9.3 Configuración remota

Dentro del sitio Web del proyecto¹⁵ se incluyó una página de configuración que permite modificar las versiones de los tres archivos de configuración en el servidor. Para que estos archivos sean descargados y utilizados por el adquisidor, el usuario debe enviarle un SMS con una palabra clave.

Para poder acceder a la página es necesario autenticarse ya que la información que contiene no puede ser alterada por personas que no estén relacionadas con la administración del adquisidor y sus datos. Para este propósito se realizó un script en PHP que se ejecuta con el botón entrar y compara los datos del usuario con un valor fijado dentro del código, y que si la condición verdadera, redirecciona hacia la página de configuración. Esta es la única manera de acceder a la página de configuración ya que ésta al iniciarse corre un código PHP que verifica la autenticación de usuario.

¹⁵ www.argos.edu.uy/adgvx



Figura 52 - Home Page Página Web

Una vez realizada la autenticación de usuario, todos los parámetros de configuración quedan accesibles para su modificación como se ve en la Figura 53.

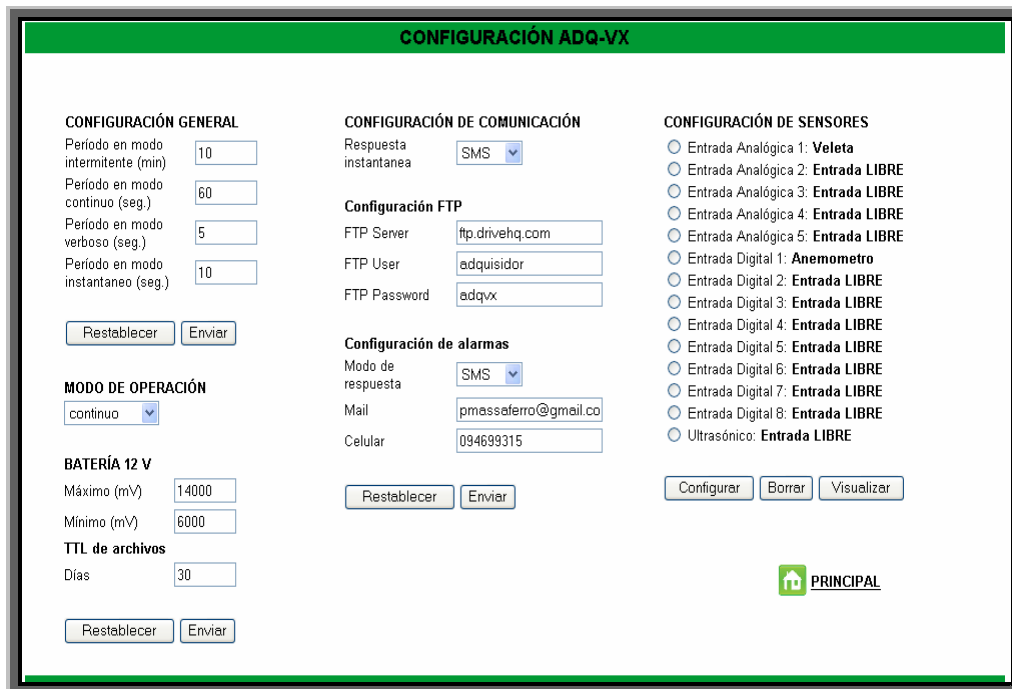


Figura 53 - Página Web de configuración

Esta página programada en html contiene cuatro formularios, dos vinculados a *adq.conf*, uno a *modem.conf* y uno a *main.conf*.

Cada formulario ejecuta un script luego de ser completado. Estos script trabajan directamente sobre las versiones de los archivos de configuración que se encuentran en el servidor. Una vez ejecutada la acción del formulario se despliega una página que permite visualizar el archivo de configuración modificado.

Los valores que se encuentran en los campos del formulario son recogidos de los archivos de configuración.

Para facilitar la lectura de los códigos se desarrolló una librería de funciones php (libreria1.php), la cual contiene funciones como *cargar_dato()* que es utilizada un gran número de veces para buscar en un archivo el valor de un parámetro en función a una serie de argumentos que recibe.

El primer formulario, que se encuentra sobre el lado izquierdo de la página de configuración, es el de la **configuración general**. La acción del botón enviar modifica los cuatro parámetros generales dentro del archivo *adq.conf* (cuyo nombre en el servidor es *adq_conf.txt*).

El formulario de **configuración de sensores** tiene tres posibles acciones: modificar, borrar y visualizar. Para ejecutar cualquiera de ellas es necesario marcar alguna de las opciones de la tabla, ya sea una entrada digital, una analógica o el puerto serie que contiene el sensor ultrasónico.

Si el archivo de configuración ya contiene algún sensor, su nombre aparecerá en la página a la derecha de la correspondiente entrada. Todas las entradas que no tengan asignado ningún sensor quedarán marcadas como "Entrada LIBRE" y en caso de haber un error en algún sensor también aparecerá un mensaje en este punto.

Tanto un sensor existente como una entrada libre pueden ser modificadas con el botón "Configurar".

En el centro de la página de configuración se encuentra el formulario de **configuración de la comunicación** que contiene los siete parámetros de configuración del archivo *modem.conf* (cuyo nombre en el servidor es *modem_conf.txt*).

Al pulsar el botón “Enviar” es remplazado el archivo de configuración con los parámetros que se encuentren en los campos del formulario. Una vez que se actualiza la página se muestran los valores actuales del archivo de configuración.

Por último, sobre el margen inferior izquierdo de la página de configuración se encuentra el formulario que maneja los cuatro parámetros de *main.conf* (cuyo nombre en el servidor es *main_conf.txt*).

Como se mencionó anteriormente, una vez que el usuario haya realizado los cambios a través de la interfaz Web deberá enviar un SMS con la palabra CONFDOWN al adquisidor. Una vez recibido el SMS el adquisidor descarga los archivos de configuración del servidor y los guarda en el directorio */adq-vx/conf* sustituyendo los anteriores.

7.10 Programas utilitarios

Se crearon varios programas utilitarios que facilitan la interacción del usuario con el hardware. Fueron programados en bash, lo que permite leer y modificar el código fácilmente. La cabecera de cada archivo describe la utilidad del programa. Están ubicados en */adq-vx/codigo/utils*. Esta ubicación está incluida en la variable PATH del usuario root, por lo que los programas pueden ser ejecutados desde cualquier ubicación. A continuación se da una breve descripción de cada uno:

- **ActualizarHora.sh:** Actualiza la hora del sistema y de los dos RTC's.
- **Boton.sh:** Devuelve el estado del botón especificado.
- **DCDState.sh:** Devuelve el estado de la salida DCD del módem. Esta salida es 1 ó 0 si el módem tiene o no portadora.
- **DSRState.sh:** Devuelve el estado de la salida DSR del módem. Esta salida indica si el módem está preparado para recibir datos del puerto serie.
- **DTR.sh:** Establece el estado de la entrada DTR del módem. Se utiliza para sacar al módem del modo de bajo consumo.
- **DTRState.sh:** Devuelve el estado de la entrada DTR del módem.
- **Fuente5V.sh:** Enciende y apaga la fuente conmutada de 5V.
- **Fuente5VState.sh:** Devuelve el estado de la fuente conmutada de 5V.
- **Fuente5VA.sh:** Enciende y apaga el regulador lineal de 5V.

- **Fuente5VAState.sh:** Devuelve el estado del regulador lineal de 5V
- **InA.sh:** Devuelve el valor de la entrada analógica especificada. Devuelve un valor entre 0 y 1.
- **InD.sh:** Devuelve el valor de la entrada digital especificada.
- **Led.sh:** Enciende y apaga el led especificado.
- **LedState.sh:** Devuelve el estado del led especificado.
- **Out.sh:** Enciende y apaga la salida especificada.
- **OutState.sh:** Devuelve el estado de la salida especificada.
- **Parpadear.sh:** Parpadea el led especificado. Acepta como parámetros de entrada el led, la cantidad de veces a parpadear, el tiempo de encendido y el tiempo de apagado.
- **PowerModem.sh:** Enciende y apaga el módem.
- **PowerMon.sh:** Indica si el módem está encendido o apagado.
- **Relay.sh:** Enciende y apaga la salida relay.
- **RelayState.sh:** Devuelve el estado de la salida relay.
- **ResetModem.sh:** Resetea el módem.
- **RingState.sh:** Devuelve el estado de la salida RING del módem. Esta salida indica que hay una llamada o la llegada de un SMS.
- **VBatt.sh:** Devuelve el valor de tensión de entrada en volts.
- **Verboso.sh:** Ejecuta los programas de medida para cada sensor especificado en el archivo adq.conf. Devuelve los valores en pantalla.
- **MatoVerboso.sh:** Finaliza la ejecución del programa CtlAdq.sh en modo verboso. Se recomienda utilizarlo para finalizar el programa Verboso.sh en caso que este haya sido ejecutado en background.

7.11 Sistema operativo

El sistema operativo es la pieza del software que gestiona los recursos de hardware y proporciona accesibilidad al usuario para poder crear y ejecutar programas, así como acceder a los recursos de hardware y al sistema de archivos.

Se seleccionó como sistema operativo una distribución de Linux 2.6.21 específicamente desarrollada por el fabricante de la placa base Gumstix que se adapta a la arquitectura del sistema y sus prestaciones. Linux tiene como ventajas el hecho de que es un sistema operativo libre, por lo que no es necesario el pago de licencias para su uso, el ser de código abierto, lo que implica que éste puede ser modificado y ajustado a las necesidades del usuario, y el contar con una variedad de herramientas que facilitaron la tarea de programar.

Se analizó la posibilidad de cambiar el sistema operativo por uno de tiempo real como RTLinux. Estos sistemas tienen tareas llamadas “de tiempo real” que, junto con las interrupciones, tienen máxima prioridad; todas las demás tareas, incluyendo las del propio Linux, se ejecutan en paralelo con menor prioridad. Dado que la aplicación en cuestión es un adquisidor de señales, las tareas de adquisición deberían tener prioridad máxima a la hora de adquirir. Se consideró, entonces, como escenario posible, uno en el que las tareas de adquisición fueran “tareas de tiempo real” y las demás, como las de comunicación, fueran de baja prioridad. Considerando los requerimientos del cliente como la velocidad de adquisición y las estrategias de muestreo se decidió que no era necesario cambiar de sistema operativo para obtener resultados óptimos.

7.12 Compilación cruzada

Para crear la imagen de kernel y la del sistema operativo en los programas escritos en lenguaje C se realizó una compilación cruzada. Esto significa que la compilación de un código se hace para una arquitectura objetivo (en este caso, ARM) desde una plataforma de desarrollo con una arquitectura diferente (para el presente proyecto, x86).

Las restricciones de memoria y procesamiento dificultan la realización del proceso de compilación en la computadora embebida. Resultaría imposible, por ejemplo, recompilar el kernel del sistema operativo dentro de la computadora embebida. Es por esto que se programó desde una computadora personal para luego transportar los programas, imagen de kernel y sistema de archivos. Esto requirió la instalación de un ambiente de desarrollo apropiado para realizar la compilación dentro de la computadora utilizada.

Se usaron dos ambientes de desarrollo, Buildroot y OpenEmbedded (OE). Los primeros desarrollos se realizaron en Buildroot ya que OE no estaba tan difundido cuando se comenzó con la programación de la aplicación. Dada las prestaciones de OE, se decidió adaptar las aplicaciones ya realizadas a este nuevo ambiente de desarrollo.

Tanto la imagen como la compilación de los programas y el sistema final de archivos se desarrollaron en su totalidad en el entorno OE, lo que facilitó la instalación de programas. Este fue el motivo fundamental del cambio de entorno de desarrollo

7.13 Modificaciones del U-Boot

La computadora embebida tiene un procesador con 120 entradas y salidas programables (GPIO). Cada GPIO puede ser programada como entrada, salida o para cumplir una función determinada. El estado de cada GPIO está unido unívocamente al valor de una dirección específica de la memoria RAM. Para modificar el estado de una GPIO se debe modificar el valor de la dirección correspondiente de la RAM.

El uboot es el arrancador del sistema y se encarga de inicializar las GPIO y ejecutar el sistema operativo. El arrancador instalado de fábrica en la computadora embebida está compilado para trabajar con las placas de expansión de la empresa Gumstix. Esto significa que inicializa las GPIO para trabajar con estas placas de expansión. Fue

necesario entonces recompilar el arrancador para que inicializara las GPIO de acuerdo al hardware del adquisidor.

Si al adquisidor se le conecta una computadora embebida sin haber previamente modificado el arrancador, se generan cortocircuitos que pueden dañar el hardware.

Se tuvo especial cuidado al momento de modificar, compilar e instalar el nuevo arrancador porque un error en alguno de estos pasos puede dejar inutilizada la computadora.

Por información acerca de cómo instalar y modificar el arrancador consultar el *Manual de Fabricación y Desarrollo*.

7.14 Imagen precompilada

Para facilitar la producción en serie, se creó una imagen que incluye el sistema operativo, los programas utilizados y las aplicaciones programadas. Esta imagen es un conjunto de archivos binarios que se pueden cargar en la memoria flash de la computadora embebida. El objetivo es tener una forma rápida y segura de instalar la aplicación. Está formada por tres archivos:

- uboot.bin: El arrancador del sistema.
- ulmage-2.6.21-r1-gumstix-custom-verdex.bin: El kernel de Linux.
- adq-vx-image-gumstix-custom-verdex.jffs2: El sistema de archivos. Incluye las aplicaciones específicas del adquisidor.

El archivo uboot.bin fue creado en el entorno de desarrollo buildroot. Por más información acerca de este entorno de desarrollo se puede consultar el siguiente sitio de Internet:

<http://docwiki.gumstix.org/Buildroot>

Los archivos ulmage y rootfs fueron creados en el entorno de desarrollo Open Embedded. Por más información acerca de este entorno de desarrollo se puede consultar el siguiente sitio de Internet:

<http://www.gumstix.net/>

Por información acerca de cómo modificar e instalar estas imágenes consultar el *Manual de Fabricación y Desarrollo*.

7.15 Conclusiones

Luego del desarrollo del software y de realizadas las pruebas, se concluye que se alcanzaron los objetivos planteados. Se consiguió un software robusto con todas las funcionalidades requeridas. Se logró una arquitectura flexible que posibilita la incorporación de nuevas funcionalidades realizando pequeñas modificaciones en el software.

Aún así, se podría mejorar la performance de los scripts de adquisición. Esto reflejaría una mejora en la carga total del sistema, especialmente cuando se adquiere en modo continuo con tiempos de adquisición bajos.

8 Incertidumbre en la medida

Este capítulo presenta un breve análisis sobre la incertidumbre introducida por el adquisidor en las medidas de frecuencia y en las lecturas analógicas.

8.1 Incertidumbre en la medida de frecuencia

Para evaluar la incertidumbre de la medida de frecuencia se considera el caso de una onda cuadrada con frecuencia constante.

El método de medida de frecuencia consiste en contar los flancos de subida y bajada de la señal dentro de una ventana de tiempo fijo. Dado que el tiempo de la ventana es independiente de la señal, se introduce un error aleatorio.

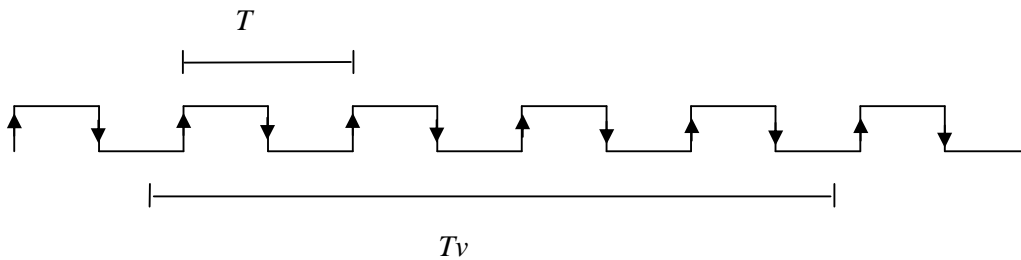


Figura 54 - Incertidumbre en la medida de frecuencia

T_v es el tiempo de la ventana de muestreo, T es el período de la señal a medir y k es la cantidad de flancos de subida y bajada sumados durante el tiempo T_v . Se miden kT flancos durante T_v si se cumple la siguiente condición:

$$(k-1)\frac{T}{2} < T_v < (k+1)\frac{T}{2} \quad (1).$$

Se define la variable aleatoria δ de la siguiente manera:

$$\delta\frac{T}{2} = T_v - k\frac{T}{2}$$

El hecho de que el instante en que comienza el muestreo es independiente de la señal a medir hace de δ una variable aleatoria con distribución uniforme.

Además por la ecuación (1) se tiene que $T_v > k \frac{T}{2} - \frac{T}{2}$ por lo que $T_v - k \frac{T}{2} > -\frac{T}{2}$

entonces $\delta > -1$.

También se cumple que:

$$T_v < k \frac{T}{2} + \frac{T}{2}$$

Entonces:

$$T_v - k \frac{T}{2} < \frac{T}{2} \text{ y } \delta < 1.$$

Se concluye que $\delta \sim U[-1,+1]$

$$\text{Luego } \delta \frac{T}{2} = T_v - k \frac{T}{2} \therefore T_v = \frac{T}{2}(k + \delta) \therefore T = \frac{2T_v}{(k + \delta)}$$

$$f = \frac{1}{T} = \frac{k + \delta}{2T_v} = \frac{k}{2T_v} + \frac{\delta}{2T_v} \quad \boxed{f = \frac{k}{2T_v} \pm \frac{1}{2T_v}}$$

Por lo que la incertidumbre en la medida es inversamente proporcional al tiempo de la ventana de muestreo. A modo de ejemplo la incertidumbre de una medida de frecuencia para una ventana de un minuto es aproximadamente $\pm 0.0083Hz$.

El error relativo de la frecuencia queda dada por la siguiente ecuación:

$$error_relativo = \frac{1}{2T_v f}$$

En la Figura 55 se presenta el error relativo de la medida de frecuencia para $T_v = 1\text{min}$. Se puede ver como para frecuencias inferiores a 3Hz el error relativo supera el 1%.

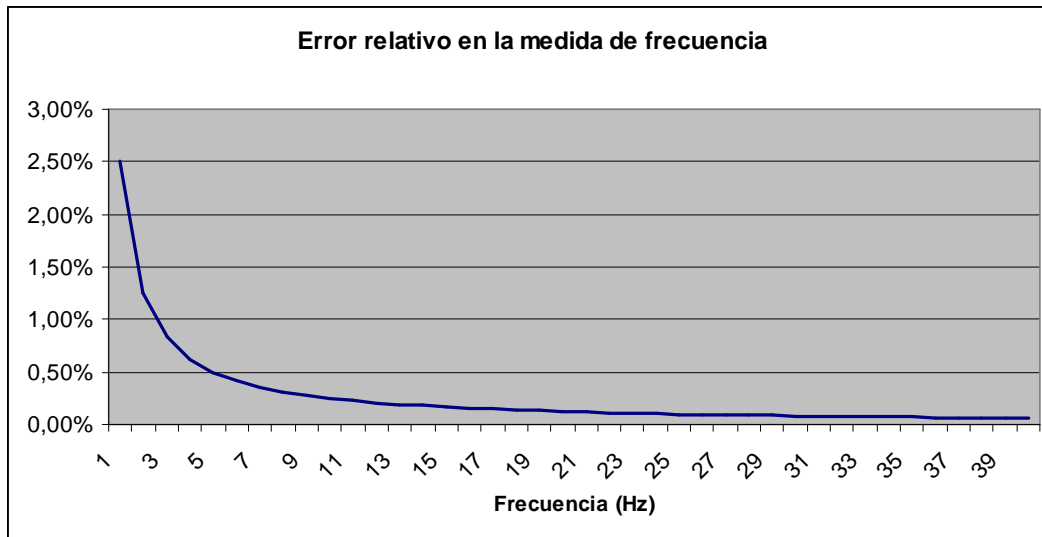


Figura 55- Error relativo en la medida de frecuencia

8.2 Incertidumbre en la medida de señales analógicas.

Se presentan en este capítulo los valores de la incertidumbre de las entradas analógicas según el fabricante del conversor analógico-digital. Como se demuestra en el capítulo 6.2.3.2.2 la incertidumbre introducida en el sistema por el filtro RC implementado para la entrada analógica, es menor que medio LSB.

| PARAMETER | ADS7828 | | |
|------------------------------|---------|------|------|
| | Min. | Typ. | Max. |
| Integral Linearity error | | ±1.0 | ±2.0 |
| Differential Linearity error | | ±1.0 | |
| Offset error | | ±1.0 | ±3.0 |
| Offset error match | | ±0.2 | ±1.0 |
| Gain error | | ±1.0 | ±4.0 |
| Gain error match | | ±0.2 | ±1.0 |

Tabla 11 - Especificación de error ADS7828

Para probar las entradas analógicas se realizaron una serie de medidas con el adquisidor y el generador de señales HP 3245A del Instituto de Ingeniería Eléctrica. Los resultados obtenidos se muestran a continuación.

| Tensión de referencia HP 3245A | Medida del adquisidor (cuentas) | Medida esperada (cuentas) | Diferencia |
|--------------------------------|---------------------------------|---------------------------|------------|
| 0 | 0 | 0 | 0 |
| 0,2 | 163 | 164 | 1 |
| 0,4 | 327 | 328 | 1 |
| 0,6 | 490 | 491 | 1 |
| 0,8 | 654 | 655 | 1 |
| 1 | 818 | 819 | 1 |
| 1,2 | 982 | 983 | 1 |
| 1,4 | 1146 | 1147 | 1 |
| 1,6 | 1310 | 1310 | 0 |
| 1,8 | 1473 | 1474 | 1 |
| 2 | 1637 | 1638 | 1 |
| 2,2 | 1801 | 1802 | 1 |
| 2,4 | 1964 | 1966 | 2 |
| 2,6 | 2128 | 2129 | 1 |
| 2,8 | 2292 | 2293 | 1 |
| 3 | 2456 | 2457 | 1 |
| 3,2 | 2620 | 2621 | 1 |
| 3,4 | 2783 | 2785 | 2 |
| 3,6 | 2947 | 2948 | 1 |
| 3,8 | 3111 | 3112 | 1 |
| 4 | 3275 | 3276 | 1 |
| 4,2 | 3439 | 3440 | 1 |
| 4,4 | 3602 | 3604 | 2 |
| 4,6 | 3766 | 3767 | 1 |
| 4,8 | 3930 | 3931 | 1 |
| 5 | 4093 | 4095 | 2 |

Tabla 12 - Resultados de la prueba de las entradas analógicas

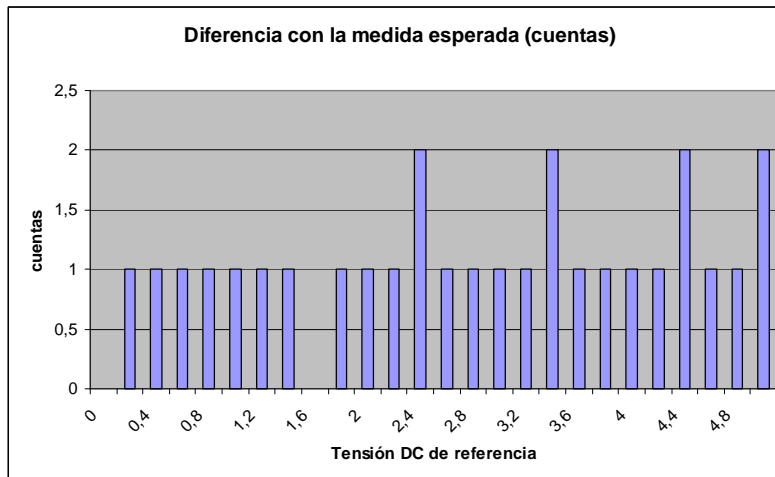


Figura 56 - Diferencia en la medida analógica

9 Pruebas de campo

Este capítulo fue incluido a efectos de mostrar una aplicación del adquisidor. No pretende ser un estudio riguroso de la calibración.

9.1 Introducción

En general, los adquisidores utilizados para la medición de condiciones climáticas son comercializados conjuntamente con los sensores. Para que el adquisidor desarrollado cumpliera eficientemente los requerimientos establecidos, se calibró para un anemómetro y una veleta con los que ya cuenta el cliente. Estos sensores corresponden a un adquisidor marca Bapt modelo EDV-1B.

9.2 Sensores

El sensor utilizado para medir la velocidad del viento fue el anemómetro marca Bapt modelo SV-1 de tres copelas troncocónicas de 50mm de diámetro y vértice semiesférico. Su radio de giro es de 97mm y el umbral de funcionamiento es menor a 0.4m/seg. Este anemómetro puede medir velocidades de hasta 60m/s con un error de $\pm 0.2\text{m/seg}$.



Figura 57 - Anemómetro de cazoletas

El anemómetro de copela es uno de los más utilizados para la medición de la velocidad horizontal del viento. Este instrumento consiste en un conjunto de tres copelas centralmente conectadas a un eje vertical para la rotación. La forma aerodinámica de las copelas convierte la fuerza generada por la presión del viento en torque, provocando el movimiento rotatorio¹⁶. Este movimiento es utilizado para abrir y cerrar un contacto en el interior del anemómetro.

¹⁶ *Nociones generales de energía eólica.* www.fing.edu.uy/imfia/rige/cur_pas/material/Cuba/Cap6.pdf

El sensor utilizado para medir la dirección del viento fue una veleta marca Bapt modelo SD-1. La posición de la veleta es determinada mediante una referencia analógica que es obtenida con una resistencia variable de $1k\Omega$ montada en el interior de la misma.

9.3 Calibración

La calibración del adquisidor se realizó en el Túnel de Viento de la Facultad de Ingeniería con el apoyo del IMFIA.

El Túnel de Viento, que es utilizado para la investigación de fenómenos relacionados al viento atmosférico mediante la modelación física, tiene una zona de trabajo de 17 metros de largo y una sección de $2.25 \times 1,80$ metros. El ventilador de 2,5 metros de diámetro, está acoplado al eje de un motor de 750rpm que mediante el accionamiento de un variador de frecuencia genera vientos de hasta 30 m/s.



Figura 58 – Layout túnel de viento



Figura 59 - Foto túnel de viento

Para calibrar el anemómetro se realizó una tabla de calibración, registrando los valores de frecuencia medidos por el adquisidor y los valores necesarios para el cálculo de la velocidad del viento (diferencia de presión en un tubo Pitot, presión atmosférica y temperatura del aire dentro del túnel).

Se tomó como referencia de funcionamiento del túnel la frecuencia de alimentación del motor tomando una serie de valores cada 5hz desde 5 hasta 35hz, 7 valores

aumentando la velocidad y 6 valores disminuyéndola, para evaluar una posible histéresis.

Los valores de frecuencia fueron medidos en ventanas de 60 segundos, mientras que el valor de diferencia de presión fue el promedio de 6 valores tomados cada 10 segundos. Los valores de temperatura y presión atmosférica corresponden a una única medida realizada en el mismo intervalo.

Para el cálculo de la velocidad del viento se utilizó la siguiente fórmula:

$$v = \sqrt{2 \cdot g \cdot H_{aire}} \quad , \quad \text{donde la altura del aire es } H_{aire} = \frac{\rho_w \cdot H_{H_2O}}{\rho_a} \quad , \quad \text{siendo } \rho_w \text{ la densidad}$$

del agua, ρ_a la densidad del aire y H_{H_2O} la diferencia de presión medida en el tubo pitot por el micromanómetro. Para el cálculo de la densidad del aire se supone $\rho \cdot P \cdot T = cte$ y se utilizan los valores medidos de presión y temperatura, y la densidad del aire en condiciones normales (0°C y 1atm).

La Tabla 13 muestra los resultados obtenidos.

| Medida Nº | Túnel variador (Hz) | ADQ-VX anemómetro (Hz) | Condiciones para cálculo de v | | | Velocidad v(m/s) |
|--------------|---------------------------|------------------------------|-------------------------------|-------|---------|---------------------|
| | | | dP (mmH2O) | T(°C) | PA(Hpa) | |
| 1 | 5 | 4.127 | 0.468 | 14.1 | 1021.5 | 2.727 |
| 2 | 10 | 9.875 | 1.99 | 13.9 | 1021.4 | 5.621 |
| 3 | 15 | 15.725 | 4.49 | 14.1 | 1021.5 | 8.446 |
| 4 | 20 | 21.698 | 8.13 | 14.1 | 1021.7 | 11.364 |
| 5 | 25 | 27.892 | 12.7 | 14.2 | 1021.7 | 14.205 |
| 6 | 30 | 34.221 | 18.6 | 14.3 | 1021.9 | 17.193 |
| 7 | 35 | 40.315 | 25.2 | 14.6 | 1021.9 | 20.022 |
| 8 | 30 | 34.154 | 18.6 | 14.6 | 1022 | 17.201 |
| 9 | 25 | 28.043 | 12.8 | 14.6 | 1021.9 | 14.270 |
| 10 | 20 | 21.881 | 8.1 | 14.6 | 1022 | 11.351 |
| 11 | 15 | 15.853 | 4.56 | 14.5 | 1022 | 8.515 |
| 12 | 10 | 9.731 | 1.95 | 14.5 | 1022 | 5.568 |
| 13 | 5 | 3.980 | 0.464 | 14.5 | 1022.1 | 2.716 |

Tabla 13 - Resultados de calibración en el túnel de viento

En Figura 60 se presenta el gráfico de la curva de calibración obtenida, que relaciona la velocidad calculada y la frecuencia medida con el anemómetro Bapt.

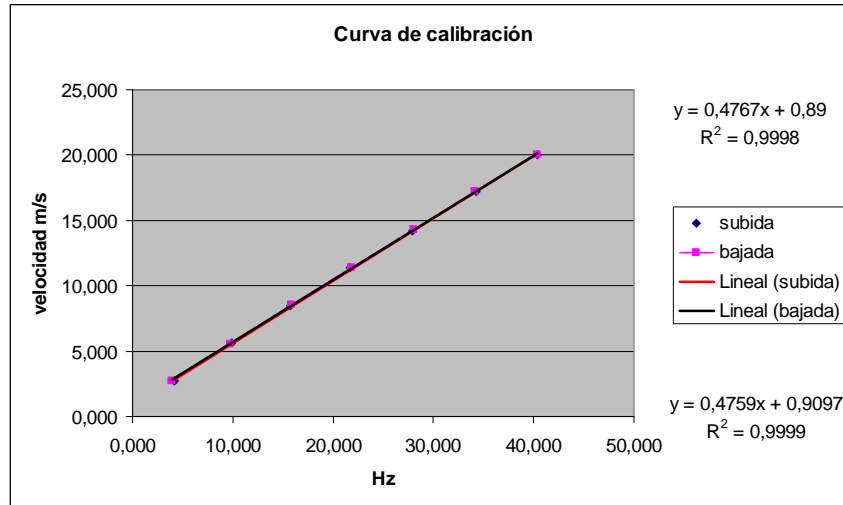


Figura 60 - Gráfico de los datos de la calibración del anemómetro Bapt

Para la calibración de la veleta se la posicionó sobre un papel con las líneas marcadas de las 16 posiciones de la rosa de los vientos y se registraron las lecturas del adquisidor para cada posición. Para no introducir errores de cálculo en la medida se registró directamente la lectura del ADC de 12 bits en cuentas (0 a 4095). Los resultados obtenidos se muestran en la Tabla 14 y en la Figura 61 - Resultados de la calibración de la veleta BaptFigura 61.

| Posición (grados) | Medida (cuentas) |
|-------------------|------------------|
| 0 | 5 |
| 22.5 | 174 |
| 45 | 422 |
| 67.5 | 712 |
| 90 | 972 |
| 112.5 | 1234 |
| 135 | 1524 |
| 157.5 | 1745 |
| 180 | 1982 |
| 202.5 | 2242 |
| 225 | 2519 |
| 247.5 | 2829 |
| 270 | 3035 |
| 292.5 | 3309 |
| 315 | 3594 |
| 337.5 | 3866 |
| 360 | 4087 |

Tabla 14 - Resultados de la calibración de la veleta Bapt

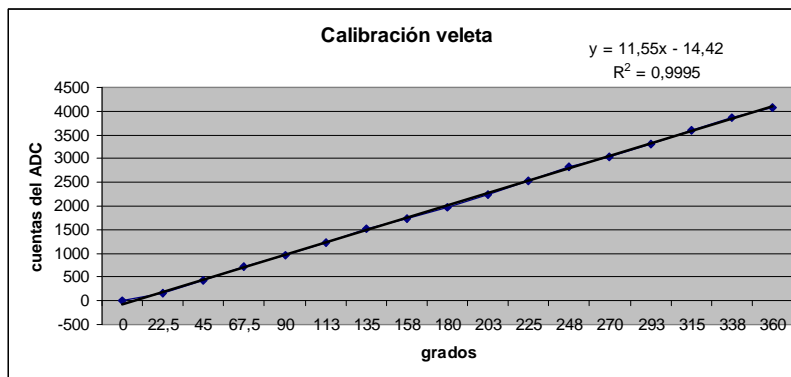


Figura 61 - Resultados de la calibración de la veleta Bapt

Se detectó que al comenzar el movimiento de la veleta, la lectura mínima fue de 5 cuentas en lugar de ser 0 y este valor se mantuvo por algunos grados. Asimismo, se detectó que al llegar a 4087 cuentas el valor permaneció constante por algunos grados en torno a 360.

Si se considera la resistencia de los cables utilizados (cables de cobre de 15.4 metros por 0.5mm de diámetro), el divisor resistivo que implementa la veleta queda representado por la siguiente ecuación: $V = V_{cc} \cdot \frac{R \cdot x + R_C}{R + 2 \cdot R_C}$, donde V es la tensión en la

entrada analógica, R la resistencia total de la veleta, Vcc la alimentación de la veleta, x

la posición porcentual de la resistencia variable y R_c la resistencia del cable. Utilizando esta fórmula y con $R=1K \Omega$ y $R_c=1,3 \Omega$ se obtiene para $x=0$ una tensión de 6,5mV que equivale a 5 cuentas y para $x=1$ una tensión de 4,9935V que equivale a 4089 cuentas. Esto explicaría los máximos y mínimos registrados en la prueba.

Este tipo de veleta presenta una banda muerta (ver Figura 62 y Figura 63) que, en este caso, se puede calcular utilizando la pendiente obtenida de 11,55 cuentas/grado y el rango de 4082 obtenido en la práctica para el recorrido del 100% de la resistencia variable. El cálculo de la banda muerta es entonces el siguiente:

$$banda_muerta = 360 - \frac{4082}{11,55} = 6.6$$

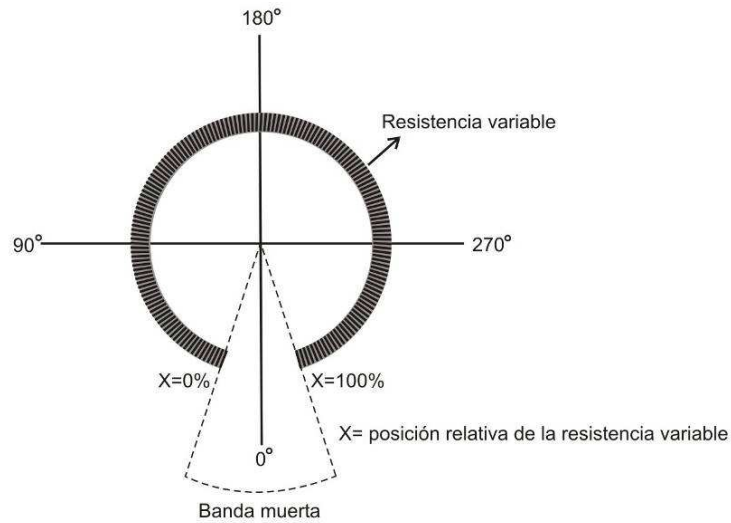


Figura 62 - Veleta Bapt

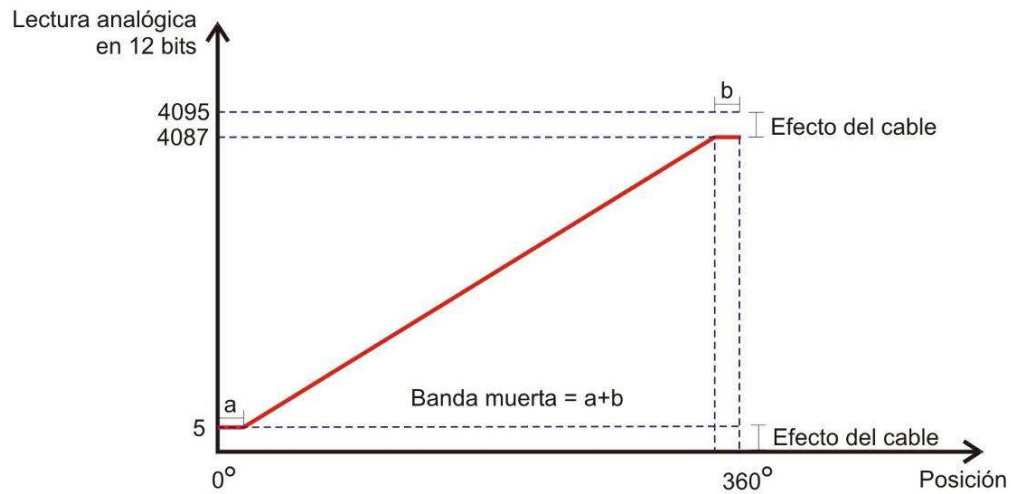


Figura 63 - Gráfico de calibración de la Veleta

9.4 Conclusiones

Se verificó la linealidad para la respuesta del anemómetro, ya que el coeficiente de linealidad resultó muy próximo a 1. Por otro lado, se identificó que para realizar la calibración del anemómetro Bapt en el adquisidor son necesarios dos parámetros: la pendiente y el cruce por cero. Estos parámetros son 0,4767 (Hz por m/s) para la pendiente y 0,89 (m/s) para el cruce por cero. Dada esta calibración, no se registran valores de velocidad de viento inferiores a 0,89 m/s.

Se verificó también la linealidad de la respuesta de la veleta y la existencia de una banda muerta de aproximadamente 6,6 grados. Asimismo, se comprobó que deben tomarse en cuenta las características de los cables a utilizar para compensar los efectos que estos generan en la lectura, en particular si la instalación requiere cables de varios metros.

10 Costos de fabricación

El presente capítulo resume los costos de los materiales necesarios para la fabricación del adquisidor. Incluye además los costos de envío de los componentes importados, los costos de aduana y los costos de armado de las placas y el gabinete.

Se presentan los resultados para la fabricación de un adquisidor, diez adquisidores y cien adquisidores.

Asimismo, se proponen algunas alternativas para disminuir los costos según la aplicación específica.

10.1 Materiales

10.1.1 Materiales importados

Los principales proveedores del proyecto son de Estados Unidos lo que implica incluir en este análisis los costos de envíos y los gastos de aduana.

Los proveedores de los materiales para ambas placas se muestran en Tabla 15.

| Proveedor | Web | Insumo |
|-------------------|--|-------------------------|
| Sunstone Circuits | www.sunstone.com | Impresos |
| Gumstix | www.gumstix.com | Verdex XM4 y accesorios |
| Janus | www.janus-rc.com | Módem y accesorios |
| Digikey | www.digikey.com | Componentes |
| Mouser | www.mouser.com | Componentes |

Tabla 15 - Proveedores de materiales

El método más económico para la importación de estos componentes es realizar la cantidad mínima de envíos necesarios dentro de Estados Unidos (una por proveedor) a un PO Box y luego un único envío a Montevideo para minimizar los costos fijos de aduana.

La Tabla 16 resume los costos de las importaciones para ambas placas considerando insumos para un adquisidor, diez y cien. Todos los precios son presentados en dólares americanos, con impuestos incluidos.

| | Insumo | 1 unidad | | | 10 unidades | | | 100 unidades | | |
|-----------------|-------------------------|----------|-------|----------------|-------------|-------|---------------|--------------|-------|---------------|
| | | Precio | Envío | Total | Precio | Envío | Total | Precio | Envío | Total |
| | | USA | | | USA | | | USA | | |
| adq-vx | Impreso | 152.00 | 0.00 | 152.00 | 51.20 | 0.00 | 51.20 | 2.07 | 0.00 | 2.07 |
| | Verdex XM4 y accesorios | 135.00 | 7.79 | 142.79 | 135.00 | 0.78 | 135.78 | 135.00 | 0.15 | 135.15 |
| | Módem y accesorios | 116.5 | 6.49 | 122.99 | 99.83 | 0.65 | 100.48 | 87.93 | 0.15 | 88.08 |
| | Componentes periféricos | 102.98 | 14.84 | 117.82 | 79.32 | 1.48 | 80.80 | 58.60 | 0.23 | 58.83 |
| | Total materiales en USA | 506.48 | 29.12 | 535.60 | 365.35 | 2.91 | 368.26 | 283.60 | 0.53 | 284.13 |
| protecc. | Impreso | 55.00 | 0.00 | 55.00 | 20.70 | 0.00 | 20.70 | 1.16 | 0.00 | 1.16 |
| | Componentes | 88.60 | 0.00 | 88.60 | 61.53 | 0.00 | 61.53 | 41.76 | 0.00 | 41.76 |
| | Total materiales en USA | 143.60 | 0.00 | 143.60 | 82.23 | 0.00 | 82.23 | 42.92 | 0.00 | 42.92 |
| | Total materiales en USA | 650.09 | 29.12 | 679.21 | 447.58 | 2.91 | 450.49 | 326.52 | 0.53 | 327.05 |
| | Envío Miami Box | | | 6.00 | | | 6.00 | | | 6.00 |
| | Aduana impuesto | | | 359.73 | | | 239.66 | | | 174.85 |
| | Costo fijo Aduana | | | 70.00 | | | 7.00 | | | 0.70 |
| | Despachante | | | 50.00 | | | 5.00 | | | 0.50 |
| | Total Aduana y envío | | | 485.73 | | | 257.66 | | | 182.05 |
| | TOTAL | | | 1164.94 | | | 708.14 | | | 509.11 |

Tabla 16 - Resumen de costos de materiales importados

Nota 1: Los precios de envío dentro de EE.UU. corresponden a la modalidad *ground*, que tiene una demora de cinco días. En el caso del proveedor de los impresos esta modalidad de envío es gratuita.

Nota 2: El costo de envío de los componentes de la placa de protecciones está incluido dentro del costo de envío de los componentes periféricos de la placa adq-vx, puesto que son adquiridos en el mismo proveedor.

10.1.2 Materiales de plaza

El gabinete y todos los accesorios de fijación de las placas fueron comprados en plaza. Son varias las razones para comprar estos insumos a proveedores locales. Por un lado, el volumen del gabinete encarece los costos de envío. Asimismo, en la región, la fabricación de estos materiales es de buena calidad y los precios son adecuados.

Los materiales utilizados para la alimentación externa del adquisidor, batería, panel solar y regulador de carga, son productos que tienen un importante mercado desarrollado dentro de Uruguay por su utilización en diversas aplicaciones, sobre todo a nivel rural. Por esto es que se cuenta con una buena oferta en plaza.

La Tabla 17 presenta estos materiales y sus costos para los tres escenarios ya planteados. Todos los precios son presentados en dólares americanos, con impuestos incluidos.

| Elemento | Costo unitario | Costo por 10 | Costo por 100 | Proveedor |
|-----------------------|----------------|---------------|---------------|--------------------|
| Gabinete* | 102.48 | 92.23 | 81.98 | Fivisa |
| Bandeja* | 12.2 | 10.98 | 9.76 | Tefix S.A. |
| Tornillería | 2 | 2 | 2 | Ferretería |
| Adhesivos | 1.2 | 1.2 | 1.2 | 4Tintas |
| Panel Solar 10W | 185 | 166.65 | 148.4 | Terko |
| Regulador de carga 5A | 65.75 | 59.2 | 52.6 | Terko |
| Batería 12V 17AH* | 70 | 63 | 56 | La casa de la pila |
| Otros | 10 | 9 | 8 | Fivisa |
| Total | 448.63 | 404.26 | 359.94 | |

Tabla 17 - Costo de materiales comprados en plaza

*Estimación para 10 y 100

Nota: El rubro *Otros* incluye pasa-cables, cables para la alimentación en 12V, terminales, etc.

10.2 Soldado y montaje

Dada la tecnología utilizada en la placa adq-vx es necesario que el soldado de sus componentes sea realizado con técnicas especiales, que requieren cierta infraestructura. En el caso del prototipo diseñado en el proyecto se recurrió a una empresa nacional para la realización de este trabajo. Si bien los componentes de la placa de protecciones son *Through Hole* y su soldado es de menor complejidad, se tomó en cuenta también el costo de este trabajo cotizado por la misma empresa.

El último ítem del costo de fabricación es el armado. Esta tarea incluye la adaptación de la bandeja metálica del tablero (según diseño mostrado en el capítulo 6.4.3), fijación de las placas, colocación de pasa-cables, montaje de batería y regulador de carga en el interior del gabinete. Este precio fue calculado estimando el tiempo que la tarea puede insumir a una persona considerando un precio determinado por hora.

| Elemento | Costo unitario | Costo por 10 | Costo por 100 |
|-------------------------------------|----------------|--------------|---------------|
| Soldado de placa adq- vx | 129 | 90 | 70 |
| Soldado de placa de protecciones | 87 | 60 | 50 |
| Armado | 12 | 10 | 7 |
| Total | 228 | 160 | 127 |

Tabla 18 - Costo de soldado y montaje

10.3 Costo total

En la Tabla 19 continuación se presenta el costo total para la fabricación del adquisidor remoto.

| Elemento | Costo unitario | Costo por 10 | Costo por 100 |
|-----------------------|----------------|----------------|---------------|
| Materiales importador | 1164.94 | 708.14 | 509.11 |
| Materiales plaza | 448.63 | 404.26 | 359.94 |
| Ensamblado | 228 | 160 | 127 |
| Total | 1841.57 | 1272.40 | 996.05 |

Tabla 19 - Resumen de costo total

Nota: Los precios incluyen impuestos.

10.4 Cómo disminuir costos

A continuación se presentan diferentes alternativas que, según la aplicación, permiten reducir los costos del adquisidor. Todos los valores están en dólares americanos.

- En primer lugar, si los adquisidores son desarrollados por una empresa, ésta puede descontar el IVA, tanto de los materiales y servicios adquiridos en plaza como de los cargos de aduana. En la Tabla 20 se muestran las posibles reducciones de costo.

| | 1 | 10 | 100 |
|------------------------------|--------|--------|--------|
| Deducciones de importaciones | 149,43 | 99,11 | 71,95 |
| Deducciones de plaza | 122,02 | 101,75 | 87,8 |
| Reducción de costo | 271,45 | 200,86 | 159,75 |

Tabla 20 - Reducciones de costos descontando IVA

- La placa de protecciones del dispositivo puede ser adaptada a la aplicación específica reduciendo la cantidad de componentes utilizados. Si se utilizan sólo una entrada analógica y una digital, y se mantienen las protecciones de la fuente tanto interna como externa, se puede reducir el costo total. En la Tabla 21 se puede visualizar dicha información.

| | 1 | 10 | 100 |
|--------------------|-----|----|-----|
| Reducción de costo | 108 | 75 | 51 |

Tabla 21 - Reducción de costos con una placa de protecciones restringida

- En caso de no utilizar la placa de protecciones, como puede ser el caso de un adquisidor que sólo utiliza sensores con comunicación serie, se puede descontar el costo de los componentes importados y de soldado. La Tabla 22 muestra la reducción de costos lograda.

| | 1 | 10 | 100 |
|--------------------|-----|-----|-----|
| Reducción de costo | 306 | 185 | 115 |

Tabla 22 - Reducción de costos prescindiendo de la placa de protecciones

- En caso de que se decida no proteger el sistema contra cortocircuitos y sobretensiones se puede desarrollar y fabricar una placa económica en placa que simplemente contenga un conector para sensores y el conector que la comunica con la placa adq-vx. En la Tabla 23 se muestra dicha reducción de costos.

| | 1 | 10 | 100 |
|--------------------|-----|-----|-----|
| Reducción de costo | 206 | 155 | 105 |

Tabla 23 - Reducción de costos suplantando la placa de protecciones por una placa de conexión

- En caso de que el adquisidor fuera instalado en una zona donde pueda ser alimentado directamente de la red eléctrica, no serían necesarios los componentes del sistema de alimentación externa. En la Tabla 24 se visualiza dicha información.

| | 1 | 10 | 100 |
|--------------------|-----|-----|-----|
| Reducción de costo | 321 | 289 | 257 |

Tabla 24 - Reducción de costos prescindiendo de los accesorios de alimentación

11. Costos de explotación

El adquisidor de datos necesita tener acceso a la red GSM/GPRS para poder operar correctamente. El único costo de explotación del adquisidor es el arrendamiento de este servicio.

El acceso a la red GSM permite al adquisidor recibir y enviar mensajes de texto y el acceso a la red GPRS permite al adquisidor conectarse a Internet. Esto es imprescindible para operar el adquisidor remotamente.

Actualmente en Uruguay operan tres compañías que ofrecen este servicio: Movistar, Ancel y Claro.

Existen dos alternativas:

1. Realizar un contrato por un volumen de datos prefijado (costo fijo).
2. Utilizar la modalidad de conexión con tarjeta prepaga (costo por volumen de transferencia).

La alternativa a utilizar depende del volumen de datos a transferir por el adquisidor. Si este es mayor a 340 KBytes por día la alternativa más económica es realizar un contrato (Figura 64). A la fecha de realización de este informe la opción más conveniente es el plan Ceibo ofrecido por la empresa ANCEL, donde se arrienda el servicio con un costo inicial de \$1464 (impuestos incluidos) para habilitar el servicio GPRS. Luego se compra un paquete mensual de 150 MBytes con un costo de \$305 (impuestos incluidos).

Si el volumen es menor a 340 KBytes la opción más económica es utilizar tarjetas prepagas. La compañía con el precio más barato para trabajar en esta modalidad es CLARO que actualmente cobra \$0.03 el KBytes.

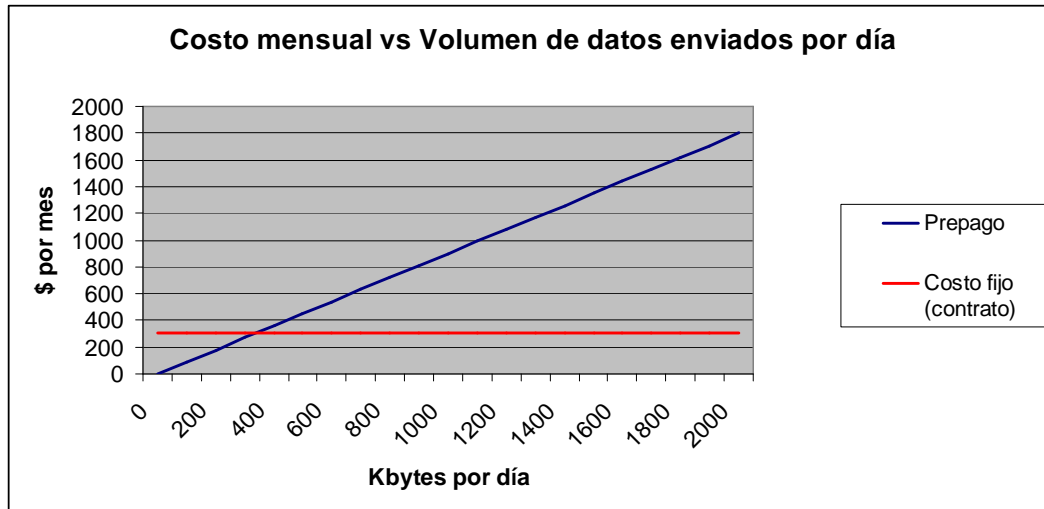


Figura 64 – Gráfico de costos del acceso al servicio GPRS

A continuación se analizan dos posibles escenarios y sus costos:

- Adquiriendo datos desde las entradas analógicas y digitales en modo continuo y con un período de un minuto, se tiene un volumen de datos de 7 KBytes diarios por sensor. Independientemente de la cantidad de sensores utilizados no se superan los 340 KBytes diarios por lo que la alternativa más conveniente es utilizar tarjeta prepaga con un costo mensual de \$6.3 por sensor.
- Adquiriendo datos desde un sensor ultrasónico con una frecuencia de muestreo de 10Hz en modo continuo, se tiene un volumen de datos de 2 MBytes diarios. Dado que este valor supera los 340 KBytes la alternativa más conveniente es utilizar el contrato anteriormente mencionado.

12. Bibliografía y glosario

12.1 Bibliografía

CORBET, Jonathan, RUBINI, Alessandro, KROAH-HARTMAN, Greg, *Linux Device Drivers*, 2nd Edition, O'Reilly, Junio 2001.

JOYANÉS AGUILAR, Luis, ZAHONERO, Ignacio, *Programación en C*, McGraw-Hill, España, Abril 2005.

KERNIGHAN, Brian, RITCHIE, Dennis, *The C programming language*, Prentice-Hall, United States of America, 1978.

COOPER, Mendel, *Advanced Bash-Scripting Guide*, Mayo 2008.
<http://tldp.org/ldp/abs/html>

Gumstix, Home Page, www.gumstix.com

Gumstix Developer Site, Home Page, www.gumstix.net

Gumstix Support Wiki, Home Page, <http://docwiki.gumstix.org>

HOELKE, Ben, *Embedded Internet – Easy GPRS*, Julio 2003.
www.roudsolutions.com/pdf/Telit/RS-Easy-GPRSv2.pdf

GM862-QUAD Hardware User Guide, Telit Communications, Revisión 5, Setiembre 2007. www.roudsolutions.com/techdocs/docs

GM862-QUAD Software User Guide, Telit Communications, Revisión 3, Octubre 2007.
www.roudsolutions.com/techdocs/docs

GM862-QUAD AT Commands Reference Guide, Telit Communications, Revisión 4, Noviembre 2007. www.roudsolutions.com/techdocs/docs

OLIVER, Juan Pablo, *Diseño y construcción de un registrador de velocidad y dirección de viento para la evaluación del potencial eólico*. iie.fing.edu.uy/jpo/papers/quito4.pdf

CATALDO, José (Resp. Científico), *La Ingeniería del viento al servicio de la construcción y del confort urbano*, IMFIA, www.fing.edu.uy/imfia/eolodinamica/tunel.html

CATALDO, José (Resp. Científico), *Nociones generales de energía eólica*, IMFIA, Capítulo 6: Instrumentos para el monitoreo de las estaciones meteorológicas. www.fing.edu.uy/imfia/rige/cur_pas/material/Cuba/Cap6.pdf

FERNÁNDEZ, Sebastián, MONDUREI, Ciro, *Proyecto de fin de carrera: IIE-PCI una plataforma de desarrollo para el bus PCI*. www.mondurei.com/iipec/01-DocIIEPCI.pdf

Esquemáticos y layout de placa de expansión Console-vx de la empresa Gumstix. pubs.gumstix.org/boards/CONSOLE/VX/PCB10003-R175/

Hojas de datos de los fabricantes

Nabble, Hardware Gumstix Forum, www.nabble.com/gumstix-f22543.html

12.2 Glosario

Data Logger: Dispositivo para la adquisición y almacenamiento de datos mediante la conexión de sensores.

GSM (Global System for Mobile Communications): Sistema estándar para comunicación utilizando teléfonos móviles que incorporan tecnología digital. Por ser digital cualquier cliente de GSM puede conectarse a través de su teléfono con su ordenador y puede hacer, enviar y recibir mensajes por e-mail, faxes, navegar por Internet, acceso seguro a la red informática de una compañía (LAN/Intranet), así como utilizar otras funciones digitales de transmisión de datos, incluyendo el Servicio de Mensajes Cortos (SMS).

GPRS (General Packet Radio Service): Servicio de datos móvil orientado a paquetes. Está disponible para los usuarios de GSM así como para los teléfonos móviles que incluyen el sistema IS-136. Permite velocidades de transferencia de 56 a 114 kbps. GPRS se puede utilizar para servicios tales como Wireless Application Protocol (WAP), servicio de mensajes cortos (SMS), servicio de mensajería multimedia (MMS), Internet y para los servicios de comunicación, como el correo electrónico y la World Wide Web (WWW). La transferencia de datos de GPRS se cobra por megabyte de capacidad, mientras que la comunicación de datos a través de conmutación de circuitos tradicionales se factura por minuto de tiempo de conexión, independiente de si el usuario utiliza la capacidad o está en un estado de inactividad.

Computadora embebida: Sistema informático de uso específico construido dentro de un dispositivo mayor. Los sistemas integrados se utilizan para usos muy diferentes de los usos generales para los que se emplea una computadora. En un sistema integrado la mayoría de los componentes se encuentran incluidos en la placa base.

I²C: Bus de comunicaciones serie. Su nombre viene de Inter-Integrated Circuit (Circuitos Inter-Integrados). La versión 1.0 data del año 1992 y la versión 2.1 del año 2000, su diseñador es Philips. La velocidad es de 100Kbits por segundo en el modo estándar, aunque también permite velocidades de 3.4 Mbit/s. Es un bus muy usado en la industria, principalmente para comunicar microcontroladores y sus periféricos en sistemas embebidos y generalizando más para comunicar circuitos integrados entre sí que normalmente residen en un mismo circuito impreso. La principal característica de I²C es que utiliza dos líneas para transmitir la información: una para los datos y por otra la señal de reloj.

SSH (Secure Shell): Nombre de un protocolo y del programa que lo implementa, que sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos. Además de la conexión a otras máquinas, SSH permite copiar datos de forma segura (tanto ficheros sueltos como simular sesiones FTP cifradas), gestionar claves RSA para no escribir claves al conectar a las máquinas y pasar los datos de cualquier otra aplicación por un canal seguro mediante un túnel SSH.

Watchdog (perro guardián): Mecanismo de seguridad que provoca un reset del sistema en caso de que éste se haya bloqueado. Consiste en un temporizador que irá continuamente decrementando un contador, inicialmente con un valor alto. Cuando este contador llega a cero, se resetea el sistema. Así que se debe diseñar el programa de manera que refresque o inicialice el perro guardián antes de que provoque el reset. Si el programa falla o se bloquea, al no poder actualizar el contador del perro guardián a su valor de inicio, éste llegará a decrementarse hasta cero y se reseteará el sistema.

Silkscreen: Impresión superficial sobre PCB con la información sobre la posición de los componentes y sus códigos.

RTC (Real Time Clock): Dispositivo utilizado para mantener la hora de los sistemas.

Buildroot: Conjunto de makefiles y patches que genera un entorno de desarrollo para la compilación cruzada y la generación de un sistema de archivos para el sistema Linux destino. Es muy útil para trabajar con pequeños sistemas embebidos

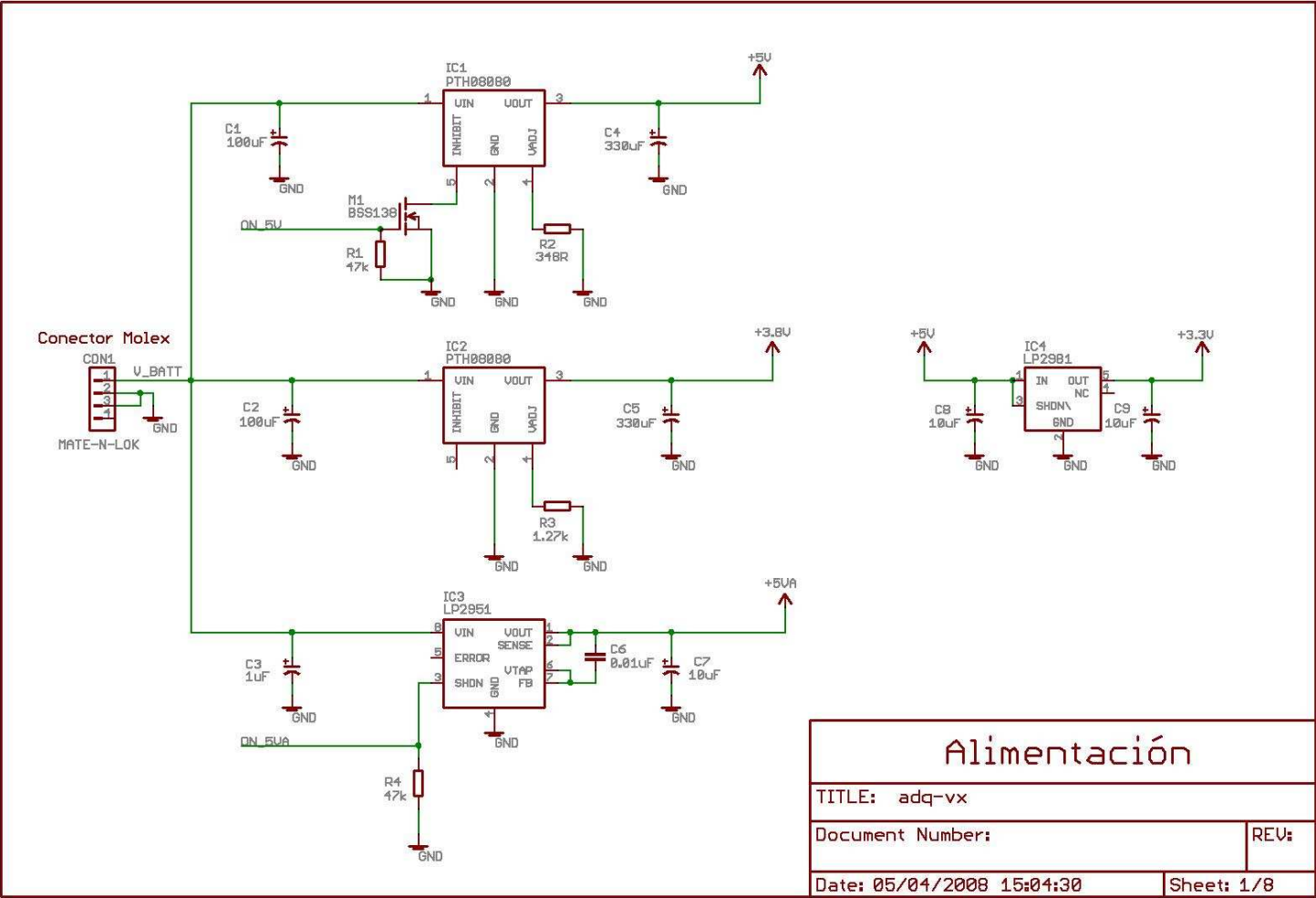
OpenEmbedded: Entorno de desarrollo para crear distribuciones de Linux para sistemas embebidos. Posee un conjunto de herramientas que permiten la compilación cruzada.

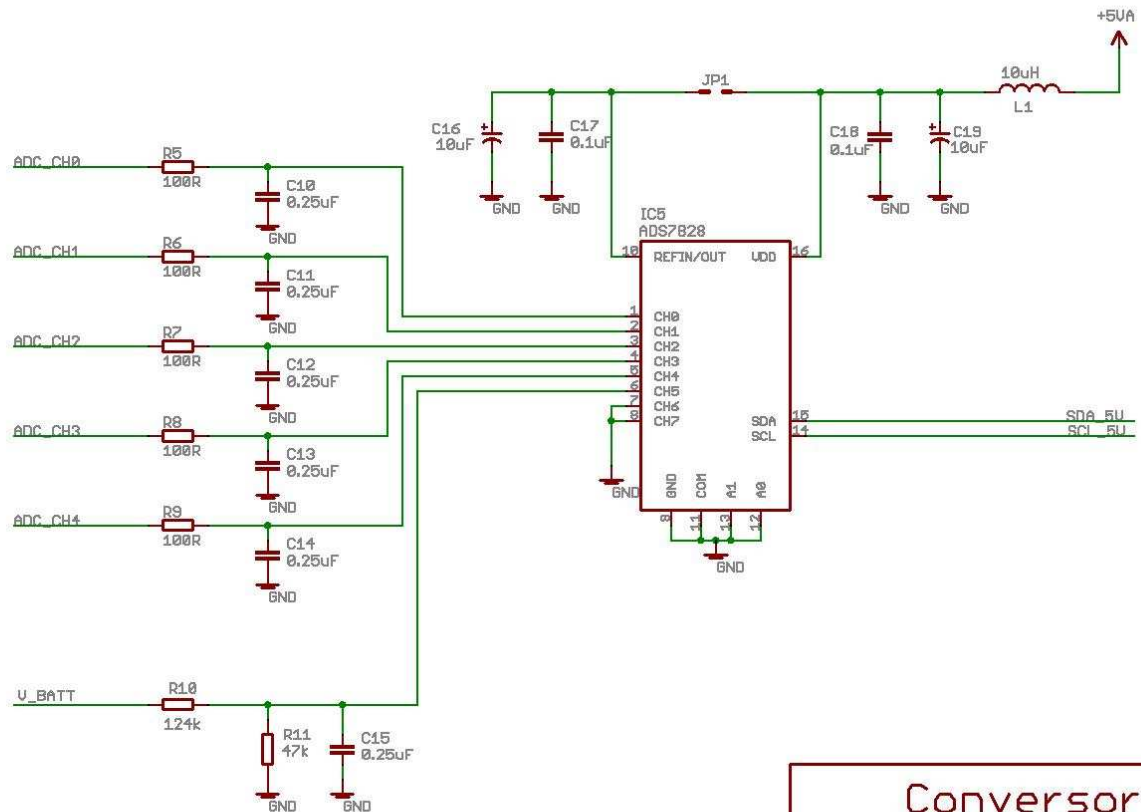
Script: Conjunto de instrucciones que permiten la automatización de tareas creando pequeñas utilidades. Es muy utilizado para la administración de sistemas UNIX. Son ejecutados por un intérprete de línea de órdenes y usualmente son archivos de texto.

13 Apéndices

13.1 Apéndice A: Esquemáticos

13.1.1 Esquemáticos del circuito impreso del Adquisidor





Conversor A/D

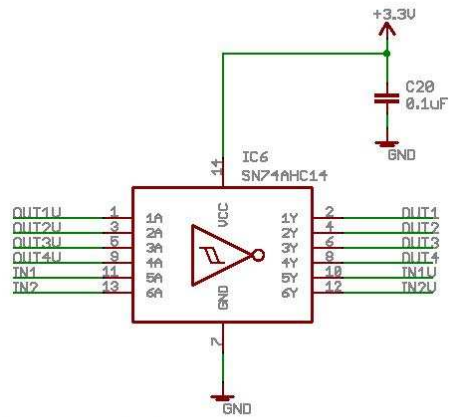
TITLE: adq-vx

Document Number:

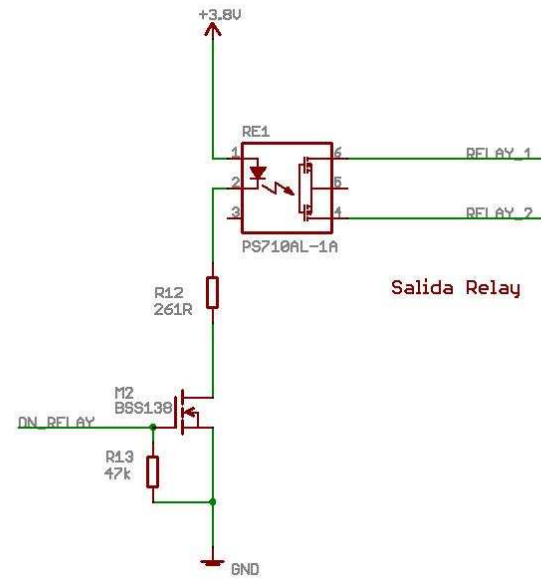
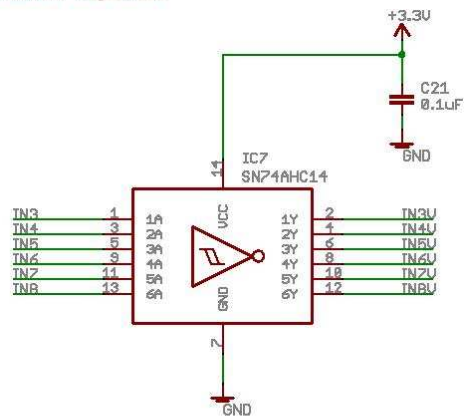
REV:

Date: 05/04/2008 15:04:30

Sheet: 2/8



Entradas y Salidas Digitales



Entradas y Salidas

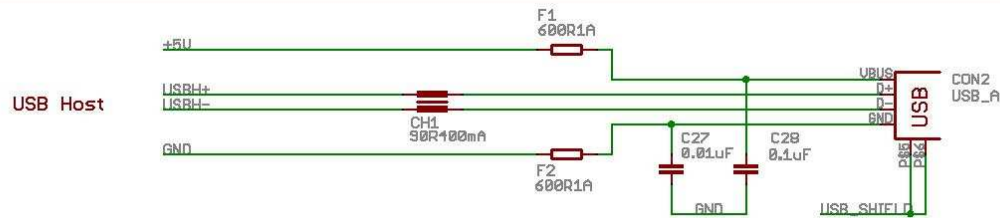
TITLE: adq-vx

Document Number:

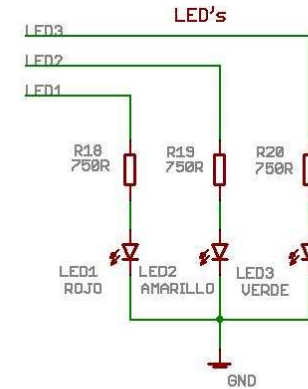
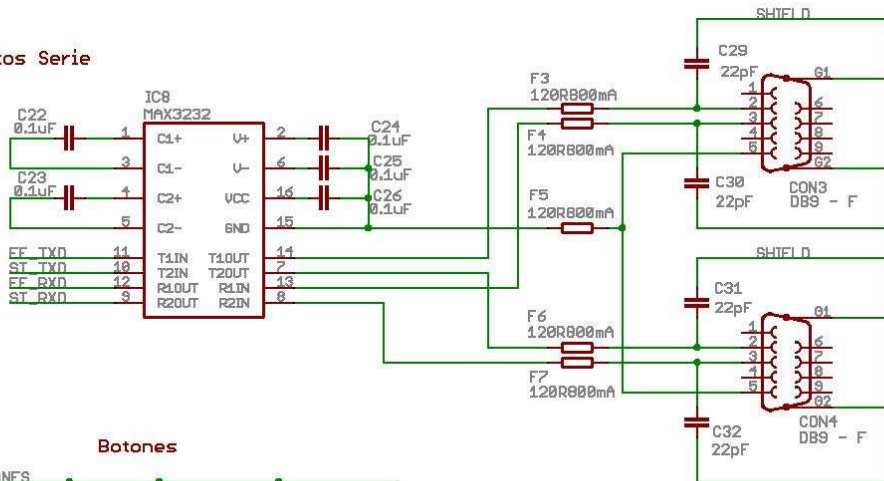
REV:

Date: 05/04/2008 15:04:30

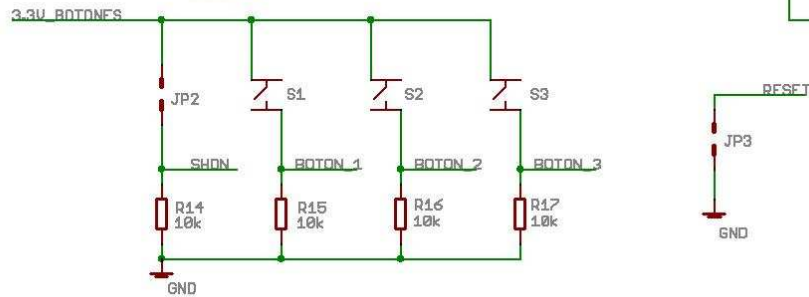
Sheet: 3/8



Puertos Serie



Botones



Interfaces con Usuario

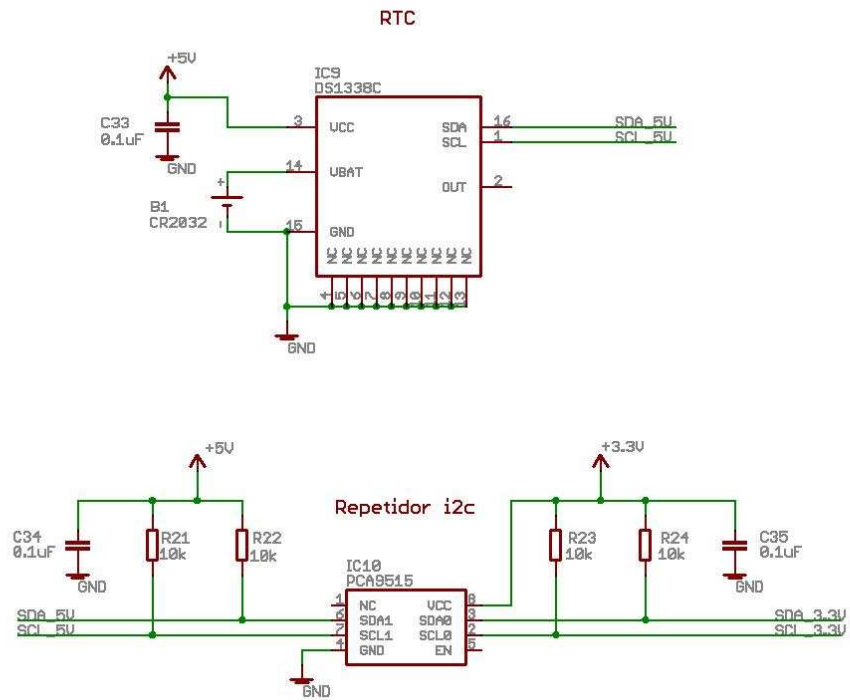
TITLE: adq-vx

Document Number:

REV:

Date: 05/04/2008 15:04:30

Sheet: 4/8



RTC y Repetidor i2c

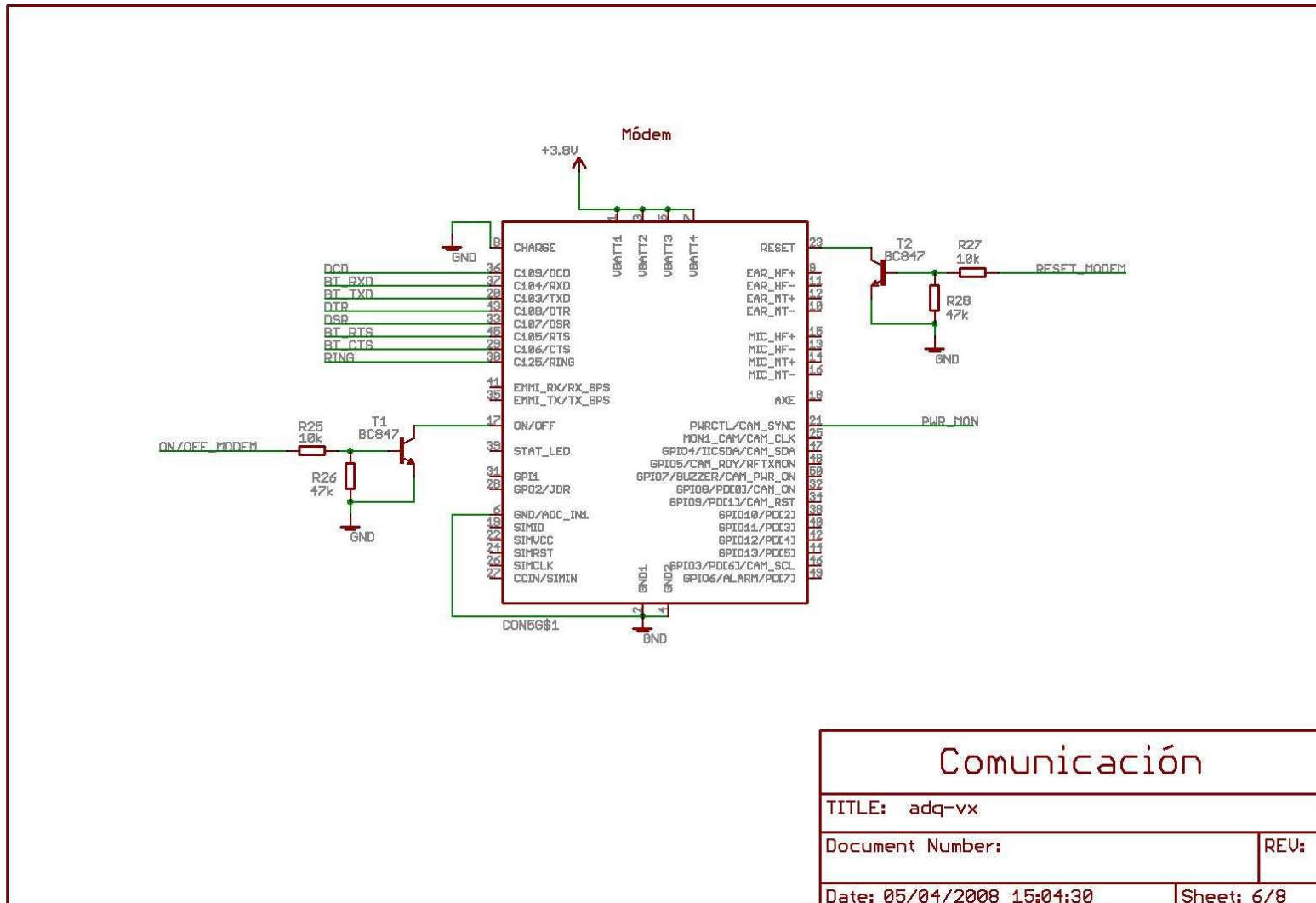
TITLE: adq-vx

Document Number:

REV:

Date: 05/04/2008 15:04:30

Sheet: 5/8



Comunicación

TITLE: adq-vx

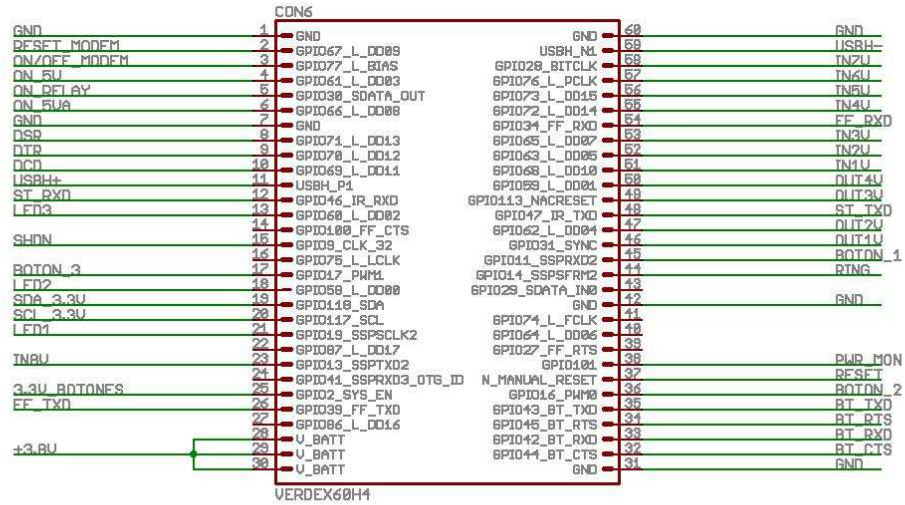
Document Number:

REV:

Date: 05/04/2008 15:04:30

Sheet: 6/8

Conector Verdex



Verdex

TITLE: adq-vx

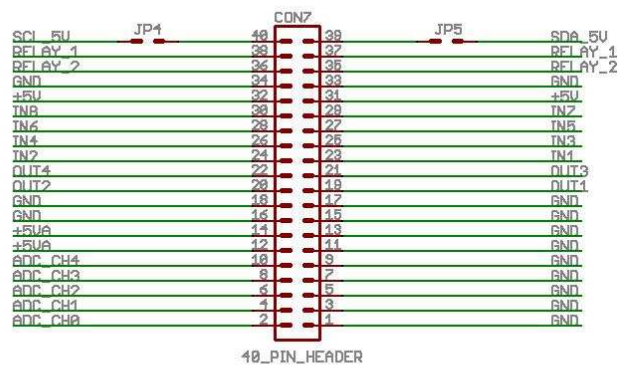
Document Number:

REV:

Date: 05/04/2008 15:04:30

Sheet: 7/8

Conector IDE



Pin Header

TITLE: adq-vx

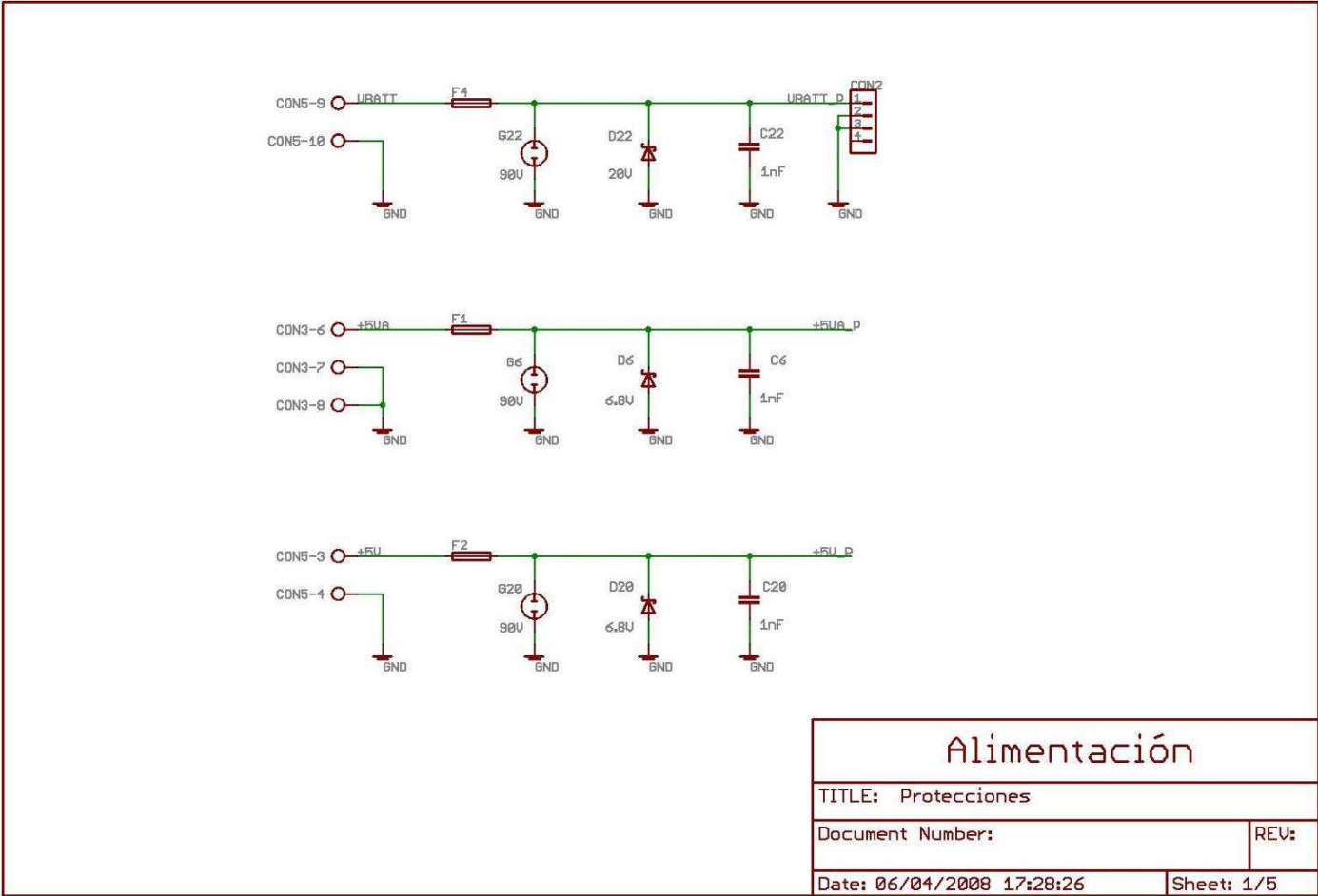
Document Number:

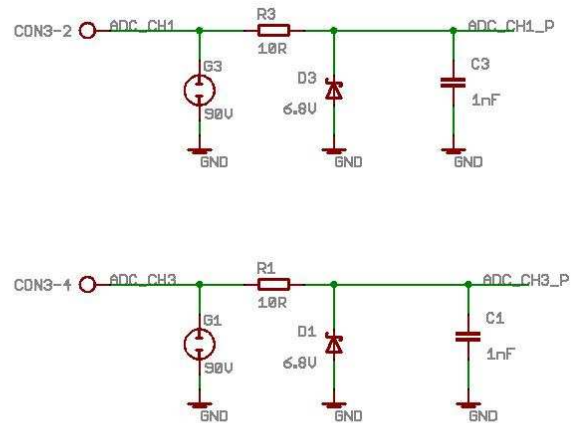
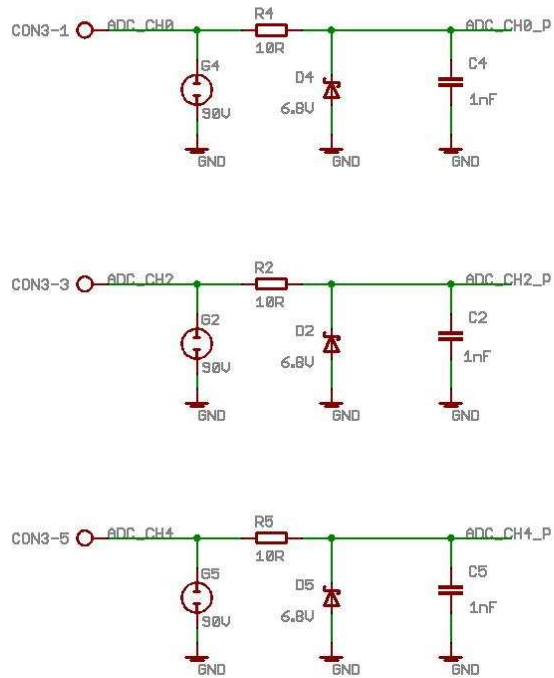
REV:

Date: 05/04/2008 15:04:30

Sheet: 8/8

13.1.2 Esquemáticos del circuito impreso de la Placa de Protecciones





Entradas Analógicas

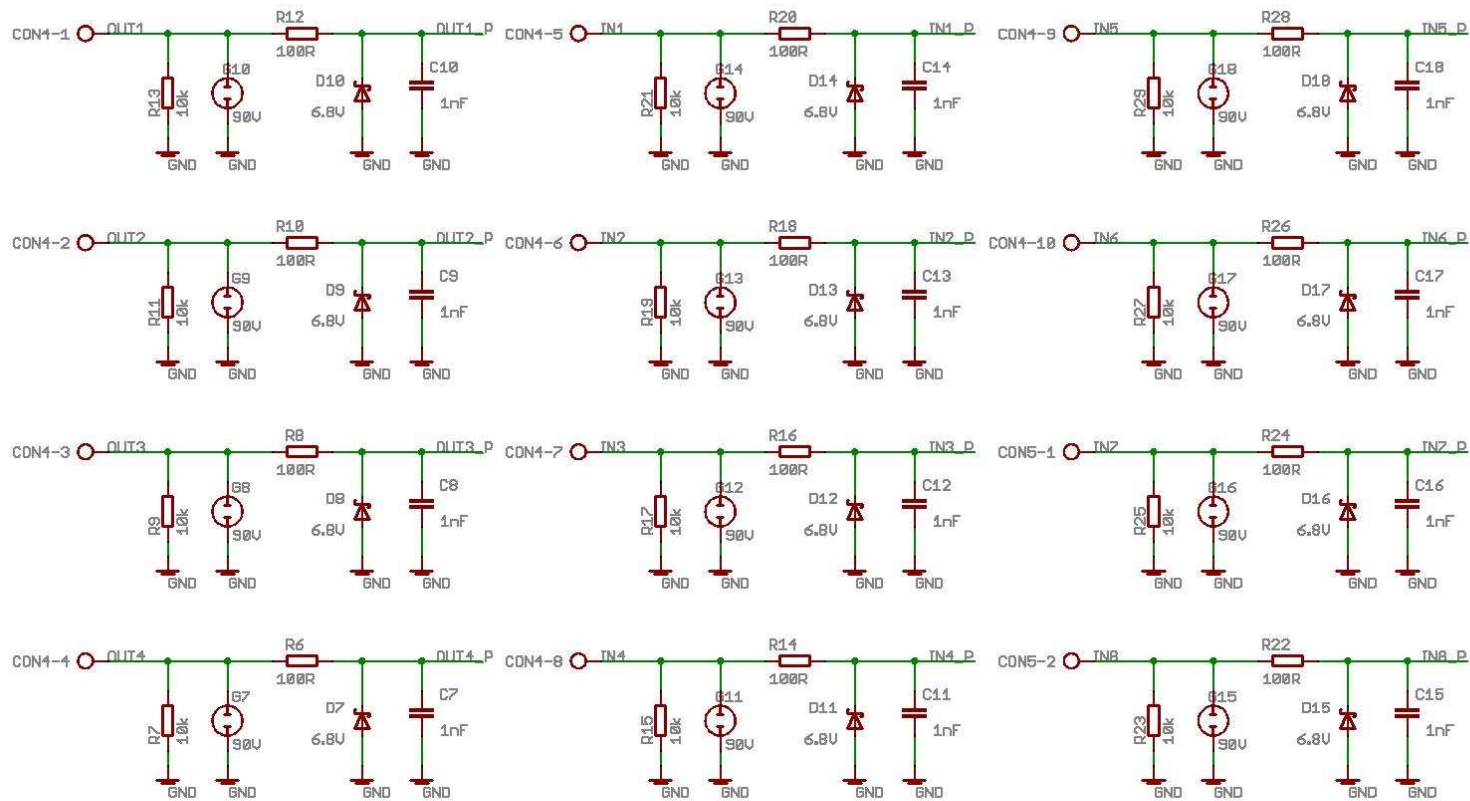
TITLE: Protecciones

Document Number:

REV:

Date: 06/04/2008 17:28:26

Sheet: 2/5



E/S Digitales

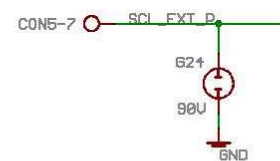
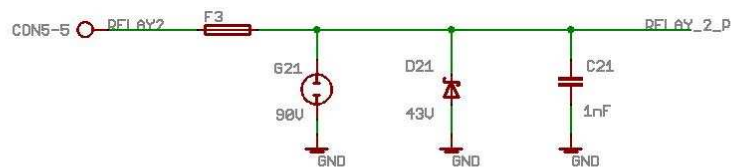
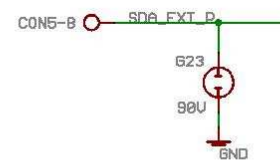
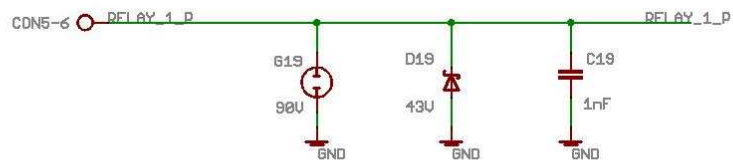
TITLE: Protecciones

Document Number:

REV:

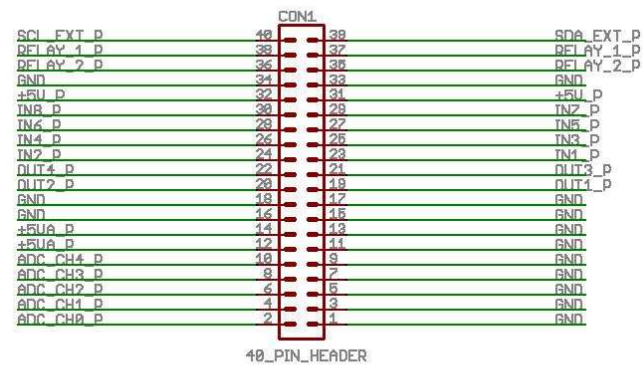
Date: 06/04/2008 17:28:26

Sheet: 3/5



| | |
|-----------------------------|------------|
| <h2>Salida Relay e i2c</h2> | |
| TITLE: Protecciones | |
| Document Number: | REV: |
| Date: 06/04/2008 17:28:26 | Sheet: 4/5 |

Conector IDE



40_PIN_HEADER

Pin Header

TITLE: Protecciones

Document Number:

REV:

Date: 06/04/2008 17:28:26

Sheet: 5/5

13.2 Apéndice B: Mapeo del conector de 60 pines

| Pin | Procesador | adq-vx | Descripción |
|-----|----------------|--------------|--|
| 1 | GND | GND | Tierra |
| 2 | GPIO(67) | RESET_MODEM | Reset del módem |
| 3 | GPIO(77) | ON/OFF_MODEM | Encendido y apagado del módem |
| 4 | GPIO(61) | ON_5V | Encendido y apagado de la fuente de 5V |
| 5 | GPIO(30) | ON_RELAY | Encendido y Apagado de la salida relay |
| 6 | GPIO(66) | ON_5VA | Encendido y apagado del regulador lineal de 5V |
| 7 | GND | GND | Tierra |
| 8 | GPIO(71) | DSR | Señal DSR del módem |
| 9 | GPIO(70) | DTR | Señal DTR del módem |
| 10 | GPIO(69) | DCD | Señal DCD del módem |
| 11 | USBH_P1 | USBH+ | Señal USB D+ |
| 12 | GPIO(46) | ST_RXD | Señal RXD del puerto serie para sensores |
| 13 | GPIO(60) | LED_3 | Encendido y apagado del LED 3 |
| 14 | GPIO(100) | - | - |
| 15 | GPIO(9) | SHDN | Botón SHDN |
| 16 | GPIO(75) | - | - |
| 17 | GPIO(17) | BOTON_3 | Botón 3 |
| 18 | GPIO(58) | LED_2 | Encendido y apagado del LED 2 |
| 19 | GPIO(118) | SDA_3.3V | Señal SDA del bus i2c |
| 20 | GPIO(117) | SCL_3.3V | Señal SCL del bus i2c |
| 21 | GPIO(19) | LED_1 | Encendido y apagado del LED 1 |
| 22 | GPIO(87) | - | - |
| 23 | GPIO(13) | IN8V | Entrada digital 8 |
| 24 | GPIO(41) | - | - |
| 25 | GPIO(2) | 3.3V_BOTONES | Alimentación de los botones |
| 26 | GPIO(39) | FF_TXD | Señal TXD del puerto serie para consola |
| 27 | GPIO(86) | - | - |
| 28 | V_BATT | 3.8V | Alimentación |
| 29 | V_BATT | 3.8V | Alimentación |
| 30 | V_BATT | 3.8V | Alimentación |
| 31 | GND | GND | Tierra |
| 32 | GPIO(44) | BT_CTS | Señal CTS del módem |
| 33 | GPIO(42) | BT_RXD | Señal RXD del módem |
| 34 | GPIO(45) | BT_RTS | Señal RTS del módem |
| 35 | GPIO(43) | BT_TXD | Señal TXD del módem |
| 36 | GPIO(16) | BOTON_2 | Botón 2 |
| 37 | N_MANUAL_RESET | RESET | Botón de reset |
| 38 | GPIO(101) | PWR_MON | Señal PWR_MON del módem |
| 39 | GPIO(27) | - | - |
| 40 | GPIO(64) | - | - |
| 41 | GPIO(74) | - | - |
| 42 | GND | GND | Tierra |
| 43 | GPIO(29) | - | - |
| 44 | GPIO(14) | RING | Señal RING del módem |
| 45 | GPIO(11) | BOTON_1 | Botón 1 |
| 46 | GPIO(31) | OUT1V | Salida digital 1 |
| 47 | GPIO(62) | OUT2V | Salida digital 2 |
| 48 | GPIO(47) | ST_TXD | Señal TXD del puerto serie para sensores |
| 49 | GPIO(113) | OUT3V | Salida digital 3 |

| | | | |
|----|----------|--------|---|
| 50 | GPIO(59) | OUT4V | Salida digital 4 |
| 51 | GPIO(68) | IN1V | Entrada digital 1 |
| 52 | GPIO(63) | IN2V | Entrada digital 2 |
| 53 | GPIO(65) | IN3V | Entrada digital 3 |
| 54 | GPIO(34) | FF_RXD | Señal RXD del puerto serie para consola |
| 55 | GPIO(72) | IN4V | Entrada digital 4 |
| 56 | GPIO(73) | IN5V | Entrada digital 5 |
| 57 | GPIO(76) | IN6V | Entrada digital 6 |
| 58 | GPIO(28) | IN7V | Entrada digital 7 |
| 59 | USBH_N1 | USBH- | Señal USB D- |
| 60 | GND | GND | Tierra |

13.3 Apéndice C: Lista de materiales

| Materiales de la placa adq-vx 1.0 | | | | |
|-----------------------------------|---------------------------------|----------|-------------------|-----------------------|
| Número de Parte Esquemáticos | Componente | Cantidad | Proveedor | N°Parte Proveedor |
| - | Circuito Impreso | 1 | Sunstone Circuits | Quick-turn Q4 Service |
| - | Computadora Verdex XM4 | 1 | Gumstix | GUM270-XM4 |
| - | Kit de tornillos y espaciadores | 1 | Gumstix | KIT0013 |
| CON6 | Conector Hirose 60 pines | 1 | Gumstix | KIT0012 |
| - | Módem GM862 | 1 | Janus | GM862-QUAD |
| - | Antena | 1 | Janus | MC-0169 |
| - | Cable de antena | 1 | Janus | MCE-0056 |
| CON5 | Conector Molex 50 pines | 1 | Janus | MC-0140-G |
| IC2, IC2 | Fuente conmutada 2.25A | 2 | Mouser | 595-PTH08080WAZ |
| IC5 | Convertor analógico digital | 1 | Mouser | 595-ADS7828EB/250 |
| IC3 | Regulador lineal 5.0V | 1 | Mouser | 595-LP2951-50D |
| CON3, CON4 | Conector DB9 hembra | 2 | Digikey | 183-009-213R531 |
| RE1 | Relay de estado sólido | 1 | Digikey | PS710AL-1AA-ND |
| IC9 | RTC | 1 | Digikey | DS1338C-33#-ND |
| IC8 | Max3232 | 1 | Digikey | MAX3232ECAE+-ND |
| C4, C5 | Capacitor 330uF | 2 | Digikey | 565-3081-1-ND |
| CON7 | Conector IDE 40 pines macho | 1 | Digikey | A33170-ND |
| IC6, IC7 | Schmitt Trigger 6 entradas | 2 | Digikey | 296-20706-1-ND |
| IC10 | Buffer i2c | 1 | Digikey | 296-20945-1-ND |
| C17, C18, C20-C26, C28, C33-C35 | Capacitor 0.1uF | 13 | Digikey | PCC1870CT-ND |
| C10-C15 | Capacitor 0.27uF | 6 | Digikey | PCC1888CT-ND |
| C1,C2 | Capacitor 100uF | 2 | Digikey | PCE4023CT-ND |
| JP1-JP5 | Pines de 2 contactos | 5 | Digikey | WM8072-ND |
| M1, M2 | MOSFET | 2 | Digikey | BSS138-FDICT-ND |
| L1 | Inductor 10uH | 1 | Digikey | DN7562CT-ND |
| R14-R17, R21-R25, R27 | Resistencia 10k | 10 | Digikey | P10KECT-ND |
| CON1 | Conector Mate-n-lok macho | 1 | Digikey | A1212-ND |
| B1 | Sócalo de batería | 1 | Digikey | BU2032-SM-HD-GCT-ND |
| IC4 | Regulador lineal 3.3V | 1 | Digikey | 296-17361-1-ND |
| CON2 | Conector USB A Hembra | 1 | Digikey | AE9923-ND |
| S1-S3 | Pulsadores | 3 | Digikey | CKN9084CT-ND |
| C7-C9, C16, C19 | Capacitor 10uF | 5 | Digikey | 493-2351-1-ND |
| C29-C32 | Capacitor 22pF | 4 | Digikey | 399-1197-1-ND |
| CH1 | Choke 90R 400mA | 1 | Digikey | 445-2207-1-ND |
| R1, R4, R11, R13, R26, R28 | Resistencia 47k | 6 | Digikey | P47KECT-ND |
| R5-R9 | Resistencia 100R | 5 | Digikey | P100ECT-ND |
| F3-F7 | Ferrita 120R 800mA | 5 | Digikey | 240-2401-1-ND |
| F1, F2 | Ferrita 600R 1A | 2 | Digikey | 240-2405-1-ND |
| R18-R20 | Resistencia 750R | 3 | Digikey | P750ECT-ND |
| - | Batería 3.3V | 1 | Digikey | P189-ND |
| - | Jumpers | 3 | Digikey | S9001-ND |

| Materiales de la placa adq-vx 1.0 /cont. | | | | |
|--|----------------------------------|----------|--------------------|-----------------------|
| C3 | Capacitor 1uF | 1 | Digikey | 495-2279-1-ND |
| T1, T2 | Transistor BJT | 2 | Digikey | BC847BW-TPCT-ND |
| LED2 | LED amarillo | 1 | Digikey | 475-1181-1-ND |
| LED3 | LED verde | 1 | Digikey | 475-1179-1-ND |
| LED1 | LED rojo | 1 | Digikey | 475-1180-1-ND |
| C6, C27 | Capacitor 0.01uF | 2 | Digikey | 311-1174-1-ND |
| R3 | Resistencia 1.27k | 1 | Digikey | P1.27KFCT-ND |
| R10 | Resistencia 124k | 1 | Digikey | P124KFCT-ND |
| R12 | Resistencia 261R | 1 | Digikey | P261FCT-ND |
| R2 | Resistencia 348R | 1 | Digikey | P348FCT-ND |
| Materiales de la placa de protecciones | | | | |
| Número de Parte Esquemáticos | Componente | Cantidad | Proveedor | N° Parte Proveedor |
| - | Circuito Impreso | 1 | Sunstone Circuits | Quick-turn Q1 Service |
| G1-G24 | Descargadores Gaseosos 90V | 24 | Digikey | 2035-09-BLF-ND |
| D1-D18, D20 | Zener 6.8V | 19 | Digikey | 1N6267ARL4GOSCT-ND |
| CON4, CON5 | Conector Phoenix Contacts 10 pos | 2 | Digikey | 277-1244-ND |
| F1-F4 | Porta Fusibles | 4 | Digikey | F1471-ND |
| CON3 | Conector Phoenix Contacts 8 pos | 1 | Digikey | 277-1242-ND |
| CON1 | Conector IDE 40 pines macho | 1 | Digikey | A33170-ND |
| R6, R8, R10, R12, R14, R16, R18, R20, R22, R24, R26, R28 | Resistencia 100 ohm | 12 | Digikey | P100W-1BK-ND |
| R7, R9, R11, R13, R15, R17, R19, R21, R23, R25, R27, R29 | Resistencia 10k ohm | 12 | Digikey | P10KW-1BK-ND |
| C1-C22 | Capacitores 1nF | 22 | Digikey | BC1072CT-ND |
| - | Fusible 0.5A | 3 | Digikey | F2450-ND |
| D19, D21 | Zener 43V | 2 | Digikey | 1.5KE43AGOS-ND |
| CON2 | Conector Mate-n-lok macho | 1 | Digikey | A1212-ND |
| R1, R2, R3, R4, R5 | Resistencia 10R | 5 | Digikey | P10W-1BK-ND |
| D22 | Zener 20V | 1 | Digikey | 1.5KE20AGOS-ND |
| - | Fusible 3A | 1 | Digikey | F2454-ND |
| Materiales para gabinete y alimentación | | | | |
| Número de Parte Esquemáticos | Componente | Cantidad | Proveedor | N° Parte Proveedor |
| - | Gabinete | 1 | Fivisa | AGP 4232 P1 |
| - | Bandeja | 1 | Tefix | - |
| - | Panel solar 10W y soporte | 1 | Terko | - |
| - | Regulador de carga 5A | 1 | Terko | - |
| - | Batería 12V 17Ah | 1 | La Casa de la Pila | - |
| - | Cables y terminales | - | Fivisa | - |
| - | Tornillería | - | Ferretería | - |
| - | Adhesivos | 3 | 4Tintas | - |
| - | Conector IDE 40 pines hembra | 2 | Digikey | MKC40A-ND |
| - | Cable plano 40 conductores | - | Eneka | - |
| - | Conector Mate-n-lok hembra | 2 | Digikey | A25436-ND |

13.4 Apéndice D: Contenido de CD

- Documentación_Proyecto_Adquisidor_Remoto.pdf
- Manual_de_Usuario.pdf
- Manual_de_Fabricación_y_Desarrollo.pdf
- Carpeta user.collection: Contiene todos el código del software
- Imagen Precompilada:
 - uboot.bin
 - adq-vx-image-gumstix-custom-verdex.jffs2
 - ulmage-2.6.21-r1-gumstix-custom-verdex.bin
- Archivos de sitio Web

13.5 Apéndice E: Manual de usuario

13.6 Apéndice F: Manual de fabricación y desarrollo