



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
ESCUELA DE INGENIERIA

SISTEMA DE MONITOREO DE BATERÍAS EN CABINA PARA UN VEHÍCULO DE TRACCIÓN ELÉCTRICA

JAVIER IBÁÑEZ VIAL

Memoria para optar al título de
Ingeniero Civil Electricista

Profesor Supervisor:
JUAN DIXON R.

Santiago de Chile, 2004



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
ESCUELA DE INGENIERIA
Departamento de Ingeniería Eléctrica

SISTEMA DE MONITOREO DE BATERÍAS EN CABINA DE UN VEHÍCULO DE TRACCIÓN ELÉCTRICA

JAVIER IBÁÑEZ VIAL

Memoria presentada a la Comisión integrada por los profesores:

JUAN DIXON

MARCELO GUARINI

RODRIGO PALMA

Para completar las exigencias del título de
Ingeniero Civil Electricista

Santiago de Chile, 2004

A mi familia

I AGRADECIMIENTOS

Quisiera agradecer, en primer lugar, a Dios, por permitir el éxito de este trabajo y también a todos los que me ayudaron. A Don Juan, a Micah Ortúzar, a Rodrigo Carmi, a los de mi casa, mi familia, a mis padres y hermanos.

INDICE GENERAL

	Pág.
DEDICATORIA.....	ii
AGRADECIMIENTOS	iii
INDICE DE TABLAS.....	v
INDICE DE FIGURAS	vi
RESUMEN.....	vii
ABSTRACT	viii
I. TITULO DEL PRIMER CAPITULO (<i>Heading 1</i>).....	9
1.1 Título del Subcapítulo 1 (<i>Heading 2</i>).....	¡Error! Marcador no definido.
1.1.1 Título de la sección 1 (<i>Heading 3</i>).	¡Error! Marcador no definido.
1.1.2 Título de la sección 2 (<i>Heading 3</i>).	¡Error! Marcador no definido.
1.1.3 Título de la sección 3 (<i>Heading 3</i>).	¡Error! Marcador no definido.
1.2 Título del Subcapítulo 2 (<i>Heading 2</i>).....	¡Error! Marcador no definido.
II. TITULO DEL SEGUNDO CAPITULO (<i>Heading 1</i>); Error!	Marcador no definido.
BIBLIOGRAFIA.....	74
A N E X O S.....	76
Anexo A : Título del Anexo (<i>Heading 1</i>)	¡Error! Marcador no definido.

II INDICE DE TABLAS

Pág.

Tabla 1.1: Comparación de técnicas basadas en modelos.....	3
--	---

III INDICE DE FIGURAS

Pág.

Figura 1.1: Fases del plan de capacidad	2
---	---

IV RESUMEN

Hoy en día el cuidado del medio ambiente y la preferencia por automóviles más silenciosos ha hecho mirar con mayor interés el uso de vehículos eléctricos. La disminución de la contaminación, la simplificación mecánica, limpieza y menos mantención de estos automóviles, junto a la aparición de nuevas tecnologías que permiten un mayor almacenamiento de energía eléctrica, hace necesario investigar más sobre ellos.

El laboratorio de Tracción Eléctrica del Departamento de Ingeniería Eléctrica de la Escuela de Ingeniería cuenta con un vehículo eléctrico movido por dos motores de inducción, los cuales son alimentados por un banco de 24 baterías de plomo ácido de 12 V cada una a través de un inversor de modulación por ancho de pulsos (PWM). Se quiere desarrollar un sistema de monitoreo que muestre al conductor el estado de temperatura y voltaje de cada batería. Este monitoreo se realizará a través de un solo cable de manera secuencial para reducir el número de cables a utilizar.

El vehículo contará con un banco de paneles solares cuya energía y potencia entrante también deberá ser mostrada al conductor a través de una pantalla de cristal líquido. Además, el sistema deberá ser capaz de guardar los datos cada cinco segundos durante 24 horas, los cuales podrán ser traspasados a un computador.

En este trabajo se describirá con detalle el diseño, desarrollo e implementación del sistema de monitoreo descrito en los párrafos anteriores.

V ABSTRACT

A monitoring system for a battery powered electric vehicle (EV) has been implemented and tested. The system allows voltage and temperature monitoring of each one of the 24 batteries. Besides, the system will also allow monitoring the energy delivered by a photovoltaic cell which is being implemented on the roof of the vehicle. The monitoring method uses digital sensors with technology “One Wire”®, which allows transmitting data from all batteries through a serial bus. The information is shown in a LCD screen with 320x340 pixel resolution. The work explains the main characteristics of the system and shows some photographs and some experimental results through the information given by the LCD screen

I INTRODUCCIÓN

En el presente capítulo se definirán los principales objetivos de la memoria, la tecnología utilizada y los plazos en que ella fue desarrollada.

1.1 Objetivos de la memoria

En 2000 la empresa norteamericana “Southern California Edison” donó un vehículo eléctrico Chevrolet E10 para ayudar al desarrollo de la rama de tracción eléctrica de nuestra escuela. Este vehículo no funcionaba y gracias al trabajo de Raúl Tejada [Teje01] se puso en operación y se llevaron a cabo numerosas mejoras en la instrumentación de la cabina del vehículo.

Siguiendo con el trabajo de mejoras de la camioneta este trabajo consistió en agregar elementos para permitir al conductor la visualización en la cabina de la temperatura y del voltaje de cada una de las baterías.

Utilizando la experiencia obtenida en trabajos anteriores resultó necesario utilizar un monitoreo secuencial de las baterías de manera de recorrer las baterías con un solo cable que las fuera sensando una tras otra y así evitarse la gran cantidad de cables que implicaría un monitoreo en paralelo. Esto permitió disminuir problemas de multiplexión, de hardware, de ruido, de conexiones y, a la vez, de ayudar a la característica estética del vehículo.

Un trabajo simultáneo está realizando Alberto Zúñiga, el cual consiste en la implementación de un banco de paneles solares que entregará energía al banco de baterías de la camioneta. Será necesario mostrar al conductor, además, la potencia y energía que está entrando a través de los paneles solares, lo cual se incluyó en este trabajo.

1.2 Elección de tecnologías

En este trabajo se presenta una conjunción de una serie de tecnologías aprovechando los últimos avances electrónicos.

En primer lugar es necesario considerar el alto nivel de ruido que existe en el vehículo, producto de la conmutación realizada por los inversores, además de las resonancias que se producen en la carrocería. El vehículo se puede considerar como una gran antena móvil que recibe señales de distinta frecuencia.

Esta característica de ruido obliga a utilizar cables y conectores blindados en el bus y sensores de buena calidad. Por esta razón se determinó el uso de la tecnología One Wire® de Dallas Semiconductor: sensores digitales que funcionan por un solo bus de nivel 5 V.

Antes de decidirse por esta tecnología se estudió la tecnología de sensores CAN, que están hechos para ambientes con alto ruido y malas condiciones de operación. Sin embargo, la dificultad en adquirirlos, la poca información disponible y el precio hicieron inclinarse por la primera.

Por otro lado, el sensor DS2436 (One Wire®) está diseñado para hacer monitoreo de temperatura y voltaje de baterías, por lo cual, con un solo transductor se midieron los dos datos.

En segundo lugar fue necesario aislar galvánicamente cada una de las tarjetas de sensado, puesto que las baterías están conectadas en serie entre ellas y los voltajes de cada una tienen tierras distintas. Se decidió utilizar una optocupla, más fácil de utilizar y más pequeña que un transformador de señal. Ahora bien, las optocuplas convencionales se usan para aislar sistemas digitales y no para sensar un voltaje analógico, con lo cual, si se mide con ellas, se obtienen errores de linealidad no tolerables. Se buscó una optocupla analógica. La optocupla lineal SLC800 de la empresa “Solid State Optronics” que tiene una linealidad nominal de 0.07%, suficiente para la precisión requerida. Se descartó la posibilidad de utilizar módulos LEM de voltaje debido a su alto consumo de corriente, ya que cada módulo

consumiría cerca de 10 mA por batería en comparación con los 500 uA que la optocoupla lineal SLC800 consume.

En tercer lugar, fue necesario presentarle al conductor de manera satisfactoria los datos que se han recogido, para lo cual se pensó en una pantalla LCD de 320 x 240 pixeles a color. Con el objetivo de evitar botoneras y por la conveniencia del precio se optó por incorporarle un *touch screen* para que el usuario interactúe con el sistema a través de la pantalla. La pantalla y el *touch screen* cuentan con una tarjeta controladora que recibe instrucciones en caracteres ASCII a través de dos puertas serial por las cuales se puede manipular los gráficos de la pantalla recibir las coordenadas del punto que se está presionando en el *touch screen*. El *touch screen* tiene una resolución de 1024 x 1024. Además la tarjeta posee una memoria flash capaz de almacenar algunas imágenes que pueden servir de pantallazos.

En cuarto lugar fue necesario guardar los datos recibidos de los sensores para un posterior estudio en el laboratorio desde un computador. Para ellos se incorporó una memoria no volátil SRAM modelo BQ4016Y, de 1 Mb de capacidad, que guarda los datos de temperatura y voltaje de cada batería y la energía entrante proveniente de los paneles solares cada cinco segundos. La capacidad de esta memoria supera las 24 horas de almacenamiento de datos.

Finalmente todo lo anterior fue coordinado por un microcomputador PIC16F877A que realiza las siguientes tareas:

- a) Obtención de voltaje y temperatura de cada batería mediante el manejo del bus One Wire a través de un bus driver DS2480B.
- b) Recepción de interrupciones del usuario provenientes del *touch screen*.
- c) Despliegue de datos en la pantalla del estado del sistema de monitoreo según requerimientos del usuario.
- d) Almacenamiento y direccionamiento en memoria SRAM si el usuario lo solicita.
- e) Traspaso a computador de los datos almacenados a través de una puerta serial si el usuario lo solicita.
- f) Conversión analógica a digital de señal proveniente del sistema de paneles solares.

(ECHO DE MENOS AQUÍ UNA FIGURA EN DIAGRAMA DE BLOQUES QUE ACLARE AL LECTOR LA CONFIGURACIÓN COMPLETA DEL SISTEMA)

1.3 Tiempo de desarrollo de las actividades

A continuación se muestra una tabla resumen con los trabajos realizados y sus fechas de ejecución:

Tabla 1.1: Cronología de los trabajos realizados

Fecha de ejecución	Trabajo realizado
2° Semestre de 2002	Obtención de información: Estudio del Protocolo One Wire
Diciembre de 2002	Definición final de las tecnologías Diseño y fabricación de la tarjeta de sensado Compra en U.S.A. de sensores y bus master One Wire Compra en U.S.A. de optocouplars lineales
Enero de 2003	Evaluación sobre compra de pantalla y <i>touch screen</i>
Febrero 2003	Llegada y compra de todos los componentes necesarios para la construcción de 26 tarjetas de sensado y una tarjeta controladora.
Marzo a Mayo de 2003	Construcción de 26 tarjetas de sensado, en montaje superficial. Prueba de las 26 tarjetas de sensado
Junio a Agosto de 2003	Construcción de envase de resina y jaula de Faraday para las tarjetas de sensado
Sep. a Nov. de 2003	Programación del microcomputador PIC
Diciembre de 2003	Diseño y construcción de tarjeta controladora definitiva Instalación de 24 tarjetas de sensado en el banco de baterías y calibración
Marzo de 2004	Desarrollo software receptor de datos del sistema a un PC.

	Instalación de la pantalla y de la tarjeta controladora en el vehículo Puesta en operación del sistema definitivo y Marcha Blanca
--	--

1.4 Dificultades surgidas durante el trabajo

Hubo sólo dos problemas importantes que demoraron la implementación del sistema.

El primero fue la aparición de una oscilación en el voltaje de salida del circuito de la optocupla que hacía inestable la medición. Este problema se corrigió con un filtro de condensadores incorporados al circuito de sensado.

El segundo problema fue un cortocircuito que quemó seis tarjetas de sensado por una mala conexión. Se solucionó rehaciendo las tarjetas quemadas. De ahí en adelante se tuvo sumo cuidado en la conexión de las tarjetas, ya que una conexión mal hecha podía significar la destrucción de todo el sistema.

II CARACTERÍSTICAS DE LOS VEHÍCULOS DE TRACCIÓN ELÉCTRICA

2.1 Características generales de un vehículo de tracción eléctrica

Las partes más importantes de un vehículo convencional de tracción eléctrica son un banco de baterías, un inversor y uno o más motores eléctricos. El inversor convierte la corriente continua del banco de baterías a la corriente alterna trifásica requerida por el o los motores. Además, el inversor mediante un control manipulado por el usuario a través del acelerador, regula la frecuencia solicitada por el conductor a los motores. Normalmente este control se hace a través de modulación PWM y electrónica de potencia.

2.2 Características particulares del vehículo de tracción eléctrica utilizado

En este caso particular, el sistema de monitoreo se instaló en una camioneta Chevrolet E10, que es una camioneta S10 adaptada a tracción eléctrica. Es del año 2000. Posee dos motores trifásicos de inducción de jaula de ardilla acoplados mecánicamente mediante correas dentadas de goma. Cada motor tiene una potencia nominal de 12 kW (16 HP), un voltaje nominal de 150 V y una corriente de 85 A. La velocidad nominal es de 7400 RPM y la frecuencia eléctrica nominal es de 250 Hz, por lo que se puede deducir que los motores tienen dos pares de polos por fase. Cada uno está controlado por un inversor individual fabricado por Solectria.

El banco de baterías está formado por 24 unidades de 12 V de plomo ácido, organizadas en dos paquetes de 12 baterías unidas en serie. Los dos paquetes están conectados en paralelo poseyendo un voltaje nominal de 144 V y una capacidad de almacenaje de 50 Ah. Las baterías son de ciclo profundo, es decir, mantienen el nivel de voltaje aunque estén muy descargadas. Sin embargo, el fabricante recomienda que no se descarguen más del 60% para no disminuir su vida útil. Las baterías pueden ser recargadas más de 500 veces sin perder sus características iniciales.

Con estas indicaciones la autonomía del vehículo está en una descarga de 30 Ah lo que equivale a unos 55 kilómetros. Con estas condiciones el vehículo puede alcanzar velocidades de hasta 130 km/h. El inversor controla la corriente entregada a los motores, que en algunos casos puede llegar hasta 230 A.

2.3 Monitoreo de Temperatura y Voltaje del Banco de Baterías

Cuando se le pide mucha corriente a una batería o ésta no está actuando normalmente, ésta comienza a calentarse, a la vez que varía su resistencia interna y su voltaje de salida.

La temperatura de una batería depende de la cantidad de carga que se está aplicando a sus bornes, de la temperatura ambiente y de su funcionamiento interno. En condiciones ambientales donde la temperatura es inferior a los 0° una batería normal no puede operar. Una batería sólo puede alcanzar 45 o 50°C si hay un cortocircuito en su interior o han disminuido notablemente los niveles de ácido.

El voltaje suele disminuir cuando se le pide una corriente elevada o cuando la carga se encuentra baja. Ahora bien, una batería en buenas condiciones no debiera bajar su voltaje a menos de 12 V.

Por las razones anteriores el monitoreo que se implementó mide un rango de temperatura de 0° a 50°C y un rango de voltaje de 10 V a 15 V, con una precisión de 0,1°C y 0,1 V respectivamente.

Debido a la conexión de las baterías, algunas en serie, otras en paralelo, no es posible conectar el monitoreo directamente a las baterías. Cada batería tiene su propia tierra de referencia, constituida por el borne negativo de cada una. El circuito de monitoreo tiene una sola tierra de referencia. Por lo anterior, es necesario que exista una aislación galvánica entre la batería y el circuito de sensado.

Asimismo, la temperatura de una batería puede medirse en el interior del ácido como en algún contacto metálico exterior. Conviene poner el sensor en uno de los bornes de la batería debido a que está en contacto directo con el ácido. En este caso, por el diseño de la tarjeta de sensado, se medirá la temperatura en el borne negativo de cada batería.

III SENSORES ONE WIRE

3.1 Características de la tecnología One Wire de Dallas Semiconductor

La tecnología One Wire® se desarrolló con el objeto de permitir la implementación sencilla de redes con muchos dispositivos tales como sensores, switches, actuadores, potenciómetros, memorias, etc. Se trata de elementos muy pequeños que comparten un bus de datos común. Cada elemento tiene un número de identificación único de 64 bits desde la fábrica.

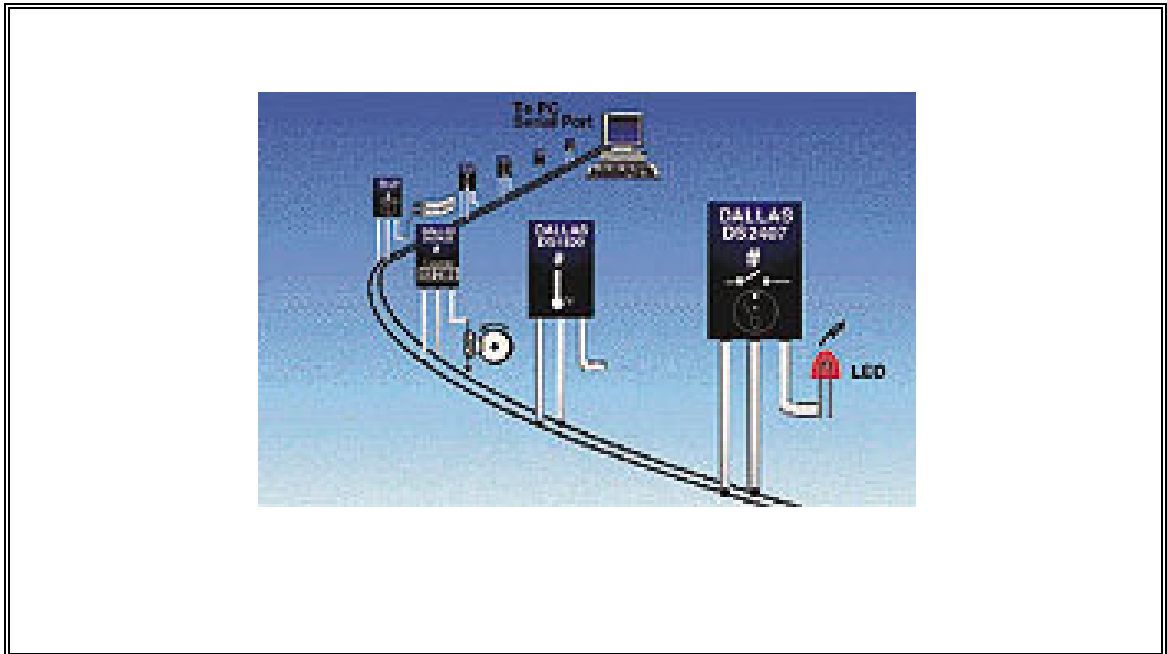


Figura 3.1: Red LAN One Wire®

Esta tecnología tiene una gran cantidad de aplicaciones. Entre ellas se encuentra la identificación (iButtons®), estaciones meteorológicas, monitoreo de baterías, codificación de correo y redes distribuidas de cientos o miles de sensores en un solo bus.

Una característica de esta red es que ocupa un solo cable para la comunicación de datos, todos los dispositivos envían y reciben por un cable. El bus, por lo tanto es *half dúplex* y utiliza un protocolo orientado a la conexión. Otra característica importante es que es asimétrica, es decir, utiliza una topología maestro-esclavo, en que existe un maestro y varios esclavos.

Una red One Wire puede llegar a medir kilómetros e incluso, existen sensores que no necesitan alimentación, pues se alimentan del mismo bus de datos. La transmisión es serial y existe todo un protocolo de *timings* para operar.

Una red One Wire consta de tres partes: un bus master, un bus de datos y un grupo de elementos One Wire.

3.1.1 Bus Master

El bus master es quien solicita la actividad de los elementos de la red. Nadie transmite si el bus master no lo ha ordenado. Éste puede ser un computador o un microcontrolador, quien debe ser capaz de seguir el protocolo de comunicación estrictamente. La conexión al bus puede hacerse directamente al bus master o a través de un bus driver, el cual se encarga de recibir las señales del computador en protocolo RS232 y transformarlas en protocolo One Wire. De esta segunda forma, se puede operar la red a través de una puerta serial con velocidades de hasta 115200 bps.

En el trabajo implementado en la camioneta se utilizó el bus driver DS2480B para manejar la red desde un microcontrolador PIC.

3.1.2 Bus de datos

El bus One Wire [Dall03A] se diseñó para operar básicamente con cable par trenzado. No está hecho para trabajar en zonas con alto ruido electromagnético, pero puede adaptarse a ellas.

Tiene un estado *idle* o desocupado de 5 V y se conecta a una resistencia *pull up* que se recomienda de 5 k Ω .

Si el bus es muy extenso suele ocurrir que cae el voltaje. Para evitar esto existen repetidores de bus que mantienen el voltaje del bus cada 200 metros. Además, reciben y reenvían la comunicación de un lado para otro, para evitar errores.

El bus de datos consta de dos cables: la tierra y el bus de comunicación. Un tercero debió añadirse a esta aplicación para alimentar el circuito de la aislación galvánica y, por deberse a un ambiente con ruido, se prefirió el cable STP-5 que es blindado.

3.1.3 Elementos One Wire

Existe una gran variedad de sensores, switches y actuadores que pueden colgarse al bus fácilmente. Todos ellos se conectan en serie. Tienen en común la tierra y el bus de comunicación. Existen también conversores A/D por si se quiere incluir datos de sensores analógicos. También pueden agregarse memorias flash si se requiere almacenar datos.

3.2 Protocolo One Wire

El protocolo One Wire permite operar las memorias del bus, solicitar una medición de algún sensor o la conversión de un conversor A/D así como la apertura o cierre de un switch, etc. Todos los comandos que se envían son de 8 bits y los solicita el bus master.

El bus master comienza una comunicación cualquiera con un pulso de reset el cual tiene cierta duración (el bus baja a 0 V). Si existe uno o más dispositivos One Wire, todos responderán con un pulso de presencia (“Presence Pulse”). En este momento el bus master deberá marcar el número de identificación de 64 bits del dispositivo con el que desea comunicarse y algún comando de 8 bits. Este comando dependerá del dispositivo que se trate; por ejemplo, en el caso de un conversor A/D, existe un dispositivo que tiene cuatro conversores: existen cuatro comandos distintos para solicitar una conversión de cada conversor A/D. Luego se espera el tiempo que dura la conversión y finalmente se solicita el dato obtenido con otro comando de 8 bits, después del cual el conversor entrega los datos de forma serial.

El número de identificación tiene la siguiente estructura:

- a) **Código de familia:** 8 bits que indican qué tipo de dispositivo es.
- b) **Número de serie:** 48 bits que indican el número único del dispositivo
- c) **Checksum:** Número CRC de los 56 bits anteriores.

Con respecto al timing de la comunicación las figuras 3.2, 3.3 y 3.4 muestran la forma de escribir un uno, leer un dato, escribir un cero y la forma de un reset pulse y de un presence pulse.

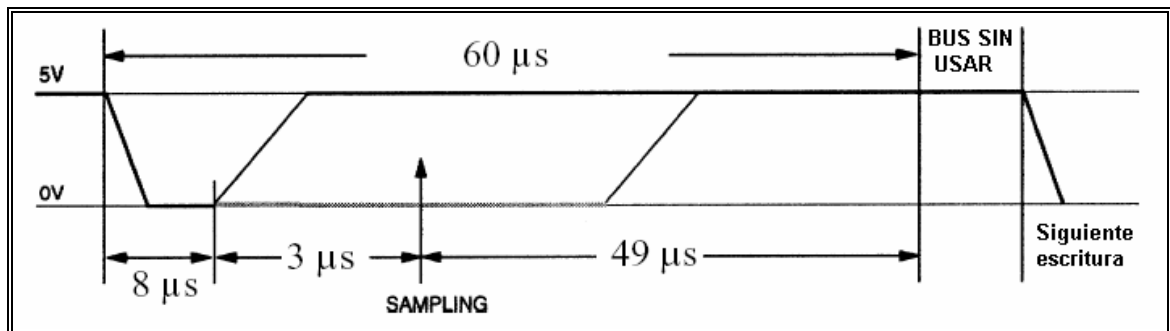


Figura 3.2: Diagrama de escritura de un uno y lectura en el bus One Wire en velocidad de comunicación regular. Tanto para la escritura de un uno como para la lectura los timings son muy parecidos. La gran diferencia es que en la lectura es necesario samplear después de 11 us.

Para escribir un uno el bus master debe seguir la siguiente rutina:

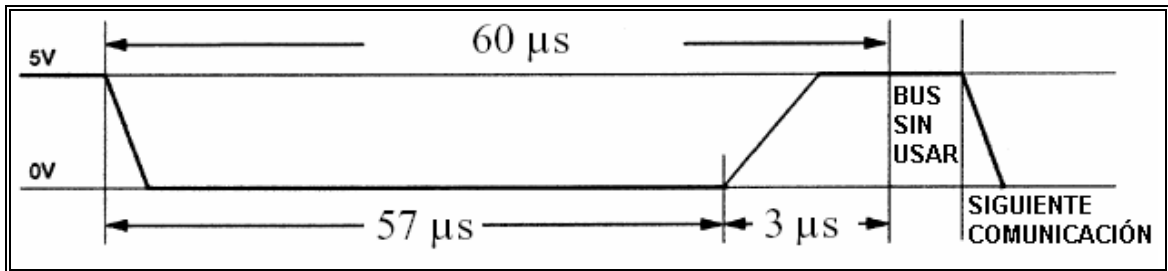


Figura 3.3: Diagrama de escritura de un cero en el bus One Wire a velocidad regular

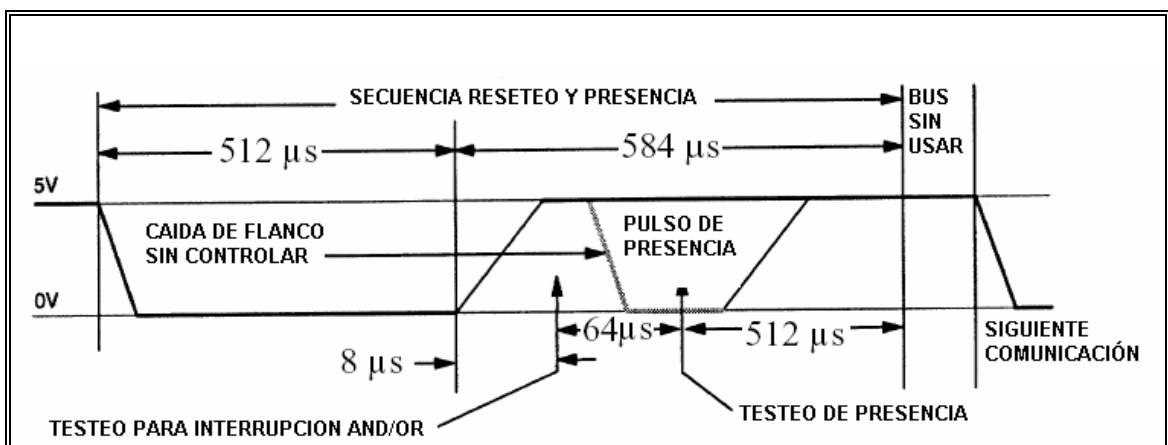


Figura 3.4: Lectura de un bit en el bus One Wire

3.3 Interfaz DS2480B One Wire/UART

El bus driver DS2480B [Dall03C] es un integrado que permite comunicarse con un bus One Wire utilizando una puerta serial. Esto es muy práctico cuando se tiene un microcontrolador que posee una puerta serial incluida. Este es nuestro caso. El microcontrolador PIC16F877A tiene una puerta serial la cual puede ser utilizada para comunicarse con todos los sensores del sistema.

El integrado DS2480B permite una serie de posibilidades de configuración. Puede configurarse para operar de 9600 a 115200 bps entre él y el microcontrolador.

Posee dos modos:

- a) **Command Mode:** Cuando se quiere que el DS2480B haga alguna operación en el bus. Por ejemplo, cuando se quiere resetear el bus, o se quiere cambiar la velocidad de comunicación entre el bus driver y el microcontrolador o cuando se quiere subir el bus a 12V, necesario para algunas aplicaciones con memorias flash.
- b) **Data Mode:** Cuando se quieren enviar comandos directamente a algún dispositivo del bus.

Sin pretender una explicación exhaustiva, el diagrama de flujos de la figura 3.5 resume todos los comandos que, desde el microcontrolador, se le pueden enviar al bus driver:

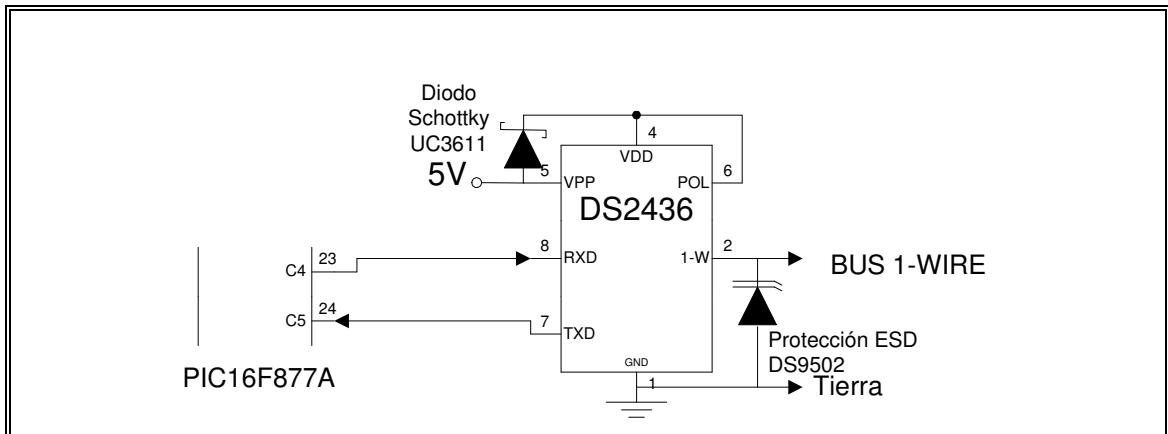


Figura 3.6: Circuito de interfase entre microcontrolador y bus One Wire

3.4 Protocolo One Wire para el sensor de voltaje y temperatura DS2436

En el capítulo I de este trabajo se incluyeron algunas características del sensor DS2436 que lo hacen candidato a ser usado en un monitoreo de baterías.

Metiéndose más en profundidad, este sensor viene a ser un pequeño microcontrolador con una serie de periféricos que le permiten distinguirse por su simplicidad y tamaño. Las partes más importantes del sensor se resumen en la figura 3.7.

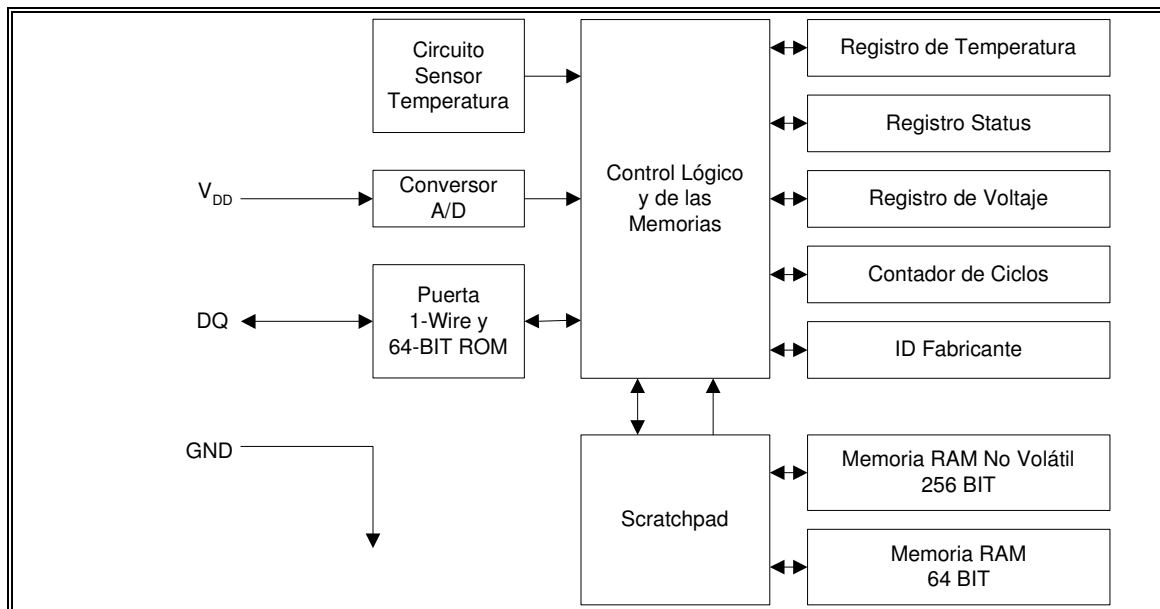


Figura 3.7: Diagrama de la memoria del sensor DS2436

A continuación se explicará alguna de estas partes:

- a) 64-bit lasered ROM ID: Contiene el número de identificación único impreso por el fabricante.
- b) Scratchpad: Permite accesibilidad a la memoria RAM y a la SRAM.
- c) Memoria RAM No Volátil: memoria no volátil de 256 bits disponible para el usuario.
- d) On-board SRAM: memoria de registro de 64 bit para libre uso del usuario
- e) Sensor de Temperatura: sensor de juntura de temperatura con un rango de -40°C a $+85^{\circ}\text{C}$ y una resolución de 0.03125°C .
- f) Convertor A/D de voltaje: mide el voltaje con que se está alimentando el sensor. Puede medir voltajes desde 2,3 V hasta 10,24 V con una resolución de 10 mV.

Para comunicarse con el sensor es necesario utilizar el protocolo de One Wire. Como se explicó anteriormente, este protocolo es manejado por el bus master el cual debe enviar un pulso de reset tras el cual todos los sensores del bus le responderán con un pulso de presencia. En seguida, el bus master debe especificar un comando de marcado y la dirección del sensor con el que quiere comunicarse. Finalmente el bus master, a través de una palabra de 8 bits, solicitará alguna acción que requiere ser hecha por el sensor. Si el bus master pide datos, deberá seguir el comando “leer registros” (B2h) con la dirección de inicio de los registros que se solicitan. El sensor enviará ordenadamente los datos requeridos.

Este sensor tiene múltiples capacidades: tiene un contador de 16 bits en memoria RAM si se quiere utilizar o un número de manufactura que se puede solicitar su impresión a Dallas Semiconductor, además del número de 64 bits. Finalmente 256 bit de memoria no volátil y 63 bit de memoria volátil. Todas estas capacidades no fueron utilizadas en la implementación actual. En este caso sólo se solicitó al sensor hacer conversiones de temperatura, de voltaje y enviarlas al bus One Wire.

La tabla 3.1 resume los comandos utilizados en esta aplicación para el sensor DS2436.

Tabla 3.1: Cuadro resumen con los comandos del sensor 2436 utilizados en esta aplicación

Instrucción	Comando	Operación que se realiza	Estado del Sensor
Convertir temperatura	D2h	Realizar una conversión de temperatura. Los datos se guardan en las posiciones 60h (LSB) y 61h(MSB) de la memoria del sensor	Bit TB del Registro Status es 1 hasta finalizar la conversión
Convertir Voltaje	B4h	Realizar una conversión de voltaje. Los datos se guardan en las posiciones 77h (LSB) y 78h (MSB) de la memoria del sensor	Bit ADB del Registro Status es 1 hasta finalizar la conversión
Leer registros	B2h <dir. Origen>	Leer datos de la memoria flash del sensor (temperatura, voltaje, registro status u otros registros)	Sensor envía los datos en los registros solicitados

3.5 Rutina en protocolo One Wire utilizada

Lo que debe hacer el microcontrolador PIC para obtener la temperatura y el voltaje del sensor es configurar el bus driver DS2480B mediante comandos de 8 bits a 9600 bps para que éste realice la conversión de protocolos y entregue los datos solicitados. La figura 7.2 del capítulo 7 resume cómo el microcontrolador debe ser programado para realizar la rutina de adquisición de datos.

IV TARJETAS ADQUISIDORAS DE DATOS

4.1 Descripción

La tarjeta adquisidora de datos es la que va en cada batería y es la que debe proporcionar la aislación galvánica así como permitir el envío de los datos desde el sensor. Esta tarjeta debe tener ciertas características, debido a que está en contacto directo con las baterías, está a la intemperie y debe soportar el ruido de los inversores que están muy cerca de las baterías.

En el capítulo I se realizó una comparación de las tecnologías disponibles para diseñar una aislación galvánica después de la cual se optó por elegir la optocoupla lineal para realizar esta tarea. No existe gran cantidad de optocouplas lineales en el mercado, puesto que éstas se utilizan esencialmente para aislar circuitos digitales y no para medir un voltaje analógico.

4.2 Aislación galvánica de voltaje

La optocoupla SLC800 [Soli03] tiene una linealidad de 0,07%. Puede operar en modo de operación fotoconductiva o fotovoltaica. Se prefirió utilizar la segunda porque permite un ancho de banda mayor de la señal de entrada. Otra ventaja es que se puede configurar el circuito para que consuma la menor cantidad de corriente.

El funcionamiento de esta optocoupla no es difícil de entender. Consiste en un diodo emisor de luz que emite una corriente proporcional a la corriente que circula por él. En vez de tener un solo fotodiodo a la salida de la optocoupla, tiene dos, los cuales se han construido de manera que sean lo más igualmente posible. Uno de los fotodiodos tiene un circuito de realimentación de manera que tiene como referencia el voltaje de entrada de la batería. Si la corriente recibida por este fotodiodo genera un voltaje en la realimentación mayor que el de la referencia, el LED disminuye su intensidad. Por el contrario, si el voltaje en la salida de la realimentación es menor que el de la referencia, el LED aumenta su intensidad. De esta manera se logra que el voltaje de salida del segundo fotodiodo, asociado a un

circuito que convierte nivel de corriente en voltaje, sea proporcional al voltaje de la batería.

Este sistema puede consumir muy poca corriente. De hecho, pruebas hechas a los circuitos sensores construidos arrojaron un consumo de 0,5 mA, en comparación con los 10 mA o más que consume un módulo LEM.

El circuito es bastante simple y se muestra en la figura 4.1.

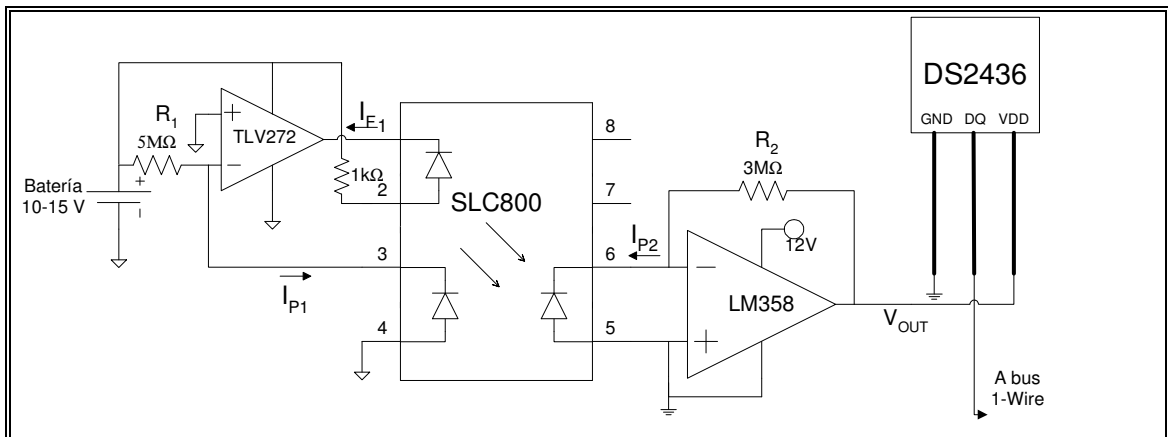


Figura 4.1: Circuito de aislación galvánica

I_F es la corriente que pasa a través del LED. Puede limitarse esta corriente a los niveles que se deseen. Si se quiere que esta corriente no supere los 0,75 mA y si llamamos K_1 a la razón de corrientes entre el LED y el fotodiodo de realimentación y K_2 a la razón de corrientes entre el LED y el fotodiodo de salida se tiene que:

$$I_{P1} = K_1 \times I_{F \text{ MAX}} \quad (4.1)$$

Puede considerarse $K_1 = K_2$ y este valor suele ser del orden de 0,004, luego, $I_{P1} = 6 \times 10^{-6}$ [A]. Ahora bien,

$$R_1 = V_{IN} / I_{P1} \quad (4.2)$$

El voltaje de entrada V_{IN} no sobrepasará nunca los 15 V, con lo cual R_1 queda fijada en $5M\Omega$.

Debido a que el sensor DS2436 que estará en la salida de este circuito no puede recibir más de 10,24 V es necesario escalar la señal. Para dar cierto margen al circuito de aislación se usó una razón de 3/5 entre el voltaje medido y el voltaje de salida del sensor. Esto fija el valor de la resistencia R_2 del circuito ya que,

$$I_{P2} = K_2 \times I_{F\ MAX} \quad (4.3)$$

$$I_{P2} = V_{OUT} / R_2 \quad (4.4)$$

Con lo cual

$$V_{OUT} / V_{IN} = (K_2 * R_2) / (K_1 * R_1) \quad (4.5)$$

Si consideramos $K_1 = K_2$,

$$R_2 = (V_{OUT} * R_1) / V_{IN} \quad (4.6)$$

Es decir, R_2 debe ser de $3 M\Omega$.

La resistencia que hay entre el LED y la alimentación no tiene gran importancia porque la corriente que pasa por ella es controlada por el opamp. Se eligió ponerle una resistencia de $1 k\Omega$.

Debido a la condición de aislación galvánica, todo el circuito de la izquierda de la optocupla en la figura 4.1 debe ser alimentado por la batería individual. Por su parte, el circuito de la derecha debe ser alimentado con la fuente general del monitoreo y con la tierra común. Como el amplificador de la realimentación debe ser alimentado con la misma fuente que mide, la batería, es necesario que sea un amplificador Rail-to-Rail, es decir, que sea capaz de mantener la linealidad incluso cuando el voltaje de entrada sea igual a su voltaje de alimentación. El opamp TLV272 de la empresa Texas Instrument puede aceptar voltajes de hasta 15 V, con lo cual se puede utilizar en esta aplicación.

El amplificador operacional de la salida del circuito no debe tener más características que su linealidad y que soporte voltajes de 10 V o menos. Se eligió el opamp LM358 de Texas Instruments, ya que sólo requiere ser alimentado por 12 V y tierra, operando entre 0 y 10 V.

Diversas pruebas de linealidad se le hicieron al circuito antes de implementarlo en tarjetas PCB, obteniendo un resultado satisfactorio.

4.3 Diseño final de la tarjeta de sensado

La tarjeta de sensado tiene tres partes:

- a) Circuito de alimentación y bus One Wire: el bus posee tres cables: la tierra, la alimentación (13,8 V) y el bus de comunicación. Las tarjetas van en serie, por lo que deben permitir una conexión serial entre ellas. Además el circuito de aislación alimenta la parte de la salida del mismo con la fuente común de la alimentación. Para mejorar el funcionamiento de la tarjeta cada alimentación tiene su alimentación regulada con reguladores LM7812 con condensadores de tantalio conectados en su entrada y en su salida.
- b) Circuito de aislación galvánica: descrito en párrafo 4.1
- c) Montaje físico del sensor: el sensor debe medir voltaje entre sus patas Ground y V_{DD} , pero además debe estar cerca del borne negativo de la batería, para medir temperatura. Para esto se debe conectar el sensor a un cable de tres conectores con la tarjeta de sensado.

Además se le agregó a cada tarjeta una luz de testigo para indicar que el monitoreo está encendido.

La figura 4.1 muestra un esquema del circuito final diseñado para la tarjeta de sensado.

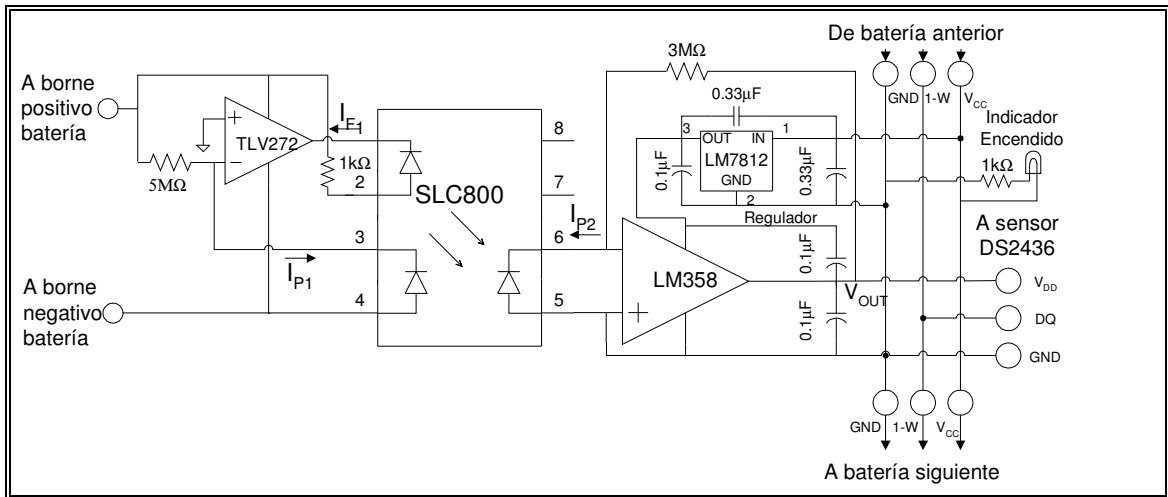


Figura 4.1 a) Diagrama esquemático de la tarjeta de sensado

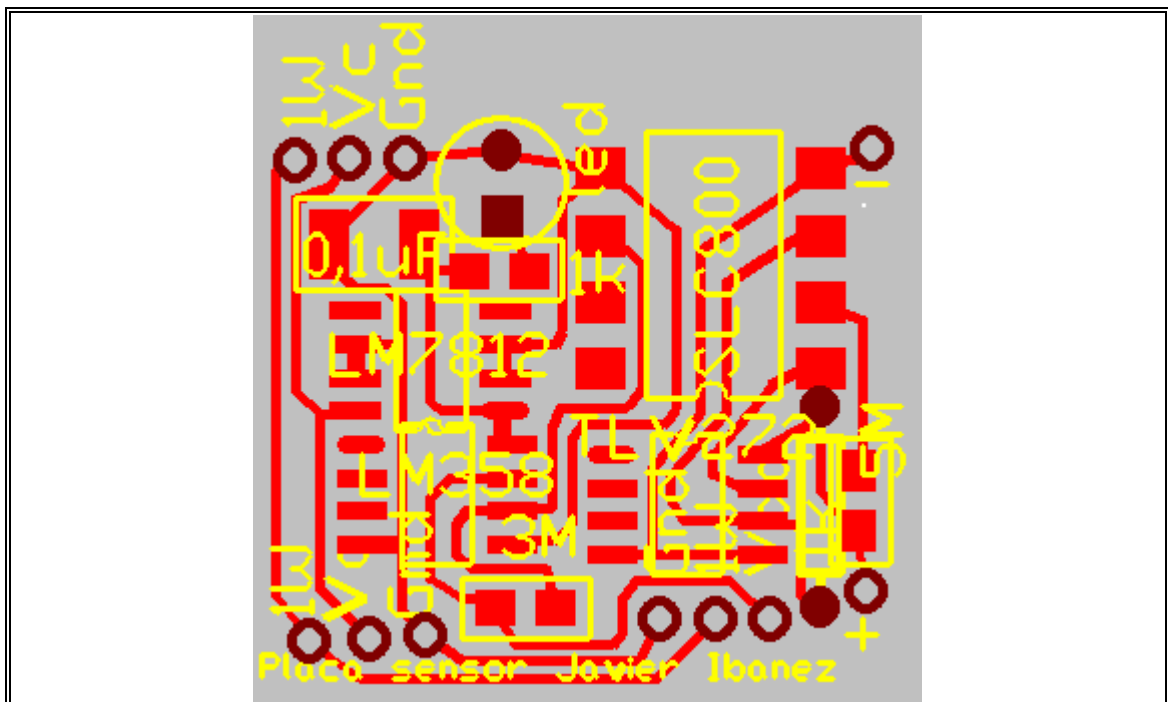
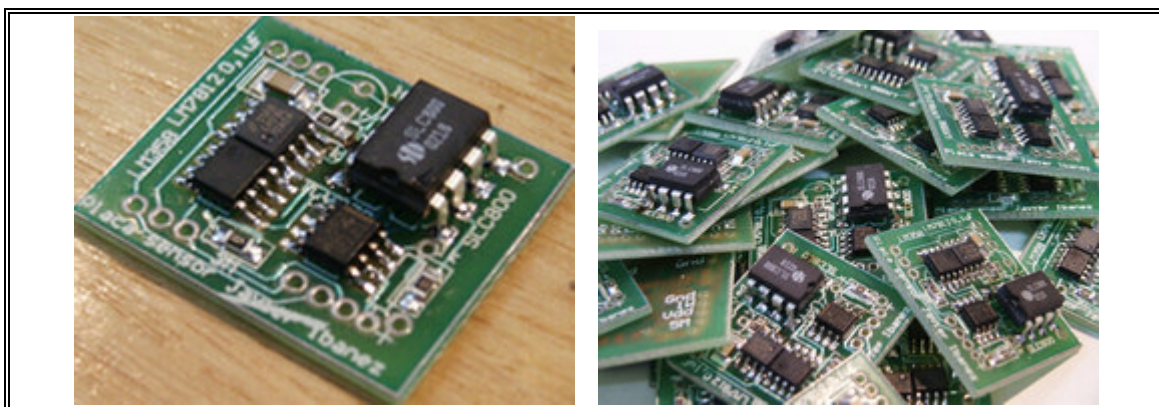


Figura 4.1 b) Dibujo tarjeta PCB de la arjeta de sensado

4.4 Jaula de Faraday y recubrimiento con resina epóxica

Fabricada en PCB la apariencia de la tarjeta se puede observar en la figura 4.2. Se puso especial esfuerzo en que las tarjetas fueran pequeñas y se logró un dimensión de 2.8 x 2.4 cms, con componentes de montaje superficial.



a)

b)

Figura 4.2 a) y b) Fotografías de la placa con todos sus componentes soldados.

El bus se fabricó con un cable apantallado STP-5 que tiene 5 conectores, de los cuales se usaron sólo tres. Debido al ruido electromagnético presente cerca de los inversores del vehículo se debió construir una jaula de Faraday para proteger el circuito de mediciones erráticas. Además, la vibración mecánica del vehículo puede producir que algunos componentes de la tarjeta se desprendan o se trice la soldadura. Para evitar estos posibles problemas se optó por hacer una caja de aluminio de dimensiones un poco mayor a la tarjeta PCB y luego recubrir todo el envase con resina epóxica. La resina epóxica P4 [BASF01] preparada a base de poliéster insaturado tiene una resistencia volumétrica de $10^{16} \Omega/\text{cm}$. y una conductividad térmica entre los 0°C y los 60°C de $0,19 \text{ [W/(m K)]}$. La dilatación térmica es de $1,1 \cdot 10^{-4} \text{ 1/K}$ entre 0°C y 60°C .



Figura 4.3 Resina epóxica P4 de poliéster insaturado.

La Figura 4.3 muestra los distintos componentes que se utilizaron para crear el envoltorio de resina: Resina P4, acelerante de cobalto y Catalizador MEKP, además de los envoltorios de aluminio. La resina es un monómero que polimeriza al estar en contacto con el acelerante de cobalto, endureciéndose en pocos minutos. La reacción se acelera notablemente aplicando un catalizador a la mezcla.

Las tarjetas no debieran calentarse mayormente por la ínfima corriente que consumen y la resistencia volumétrica de la resina es más que suficiente para aislar los circuitos. La dilatación térmica en las dimensiones de las tarjetas es de 0,44 mm en la dimensión más grande considerando una variación de 100°C, lo cual es aceptable.

La figura 4.4 muestra la construcción de la jaula de Faraday y del envoltorio de resina en distintas fases de la construcción.

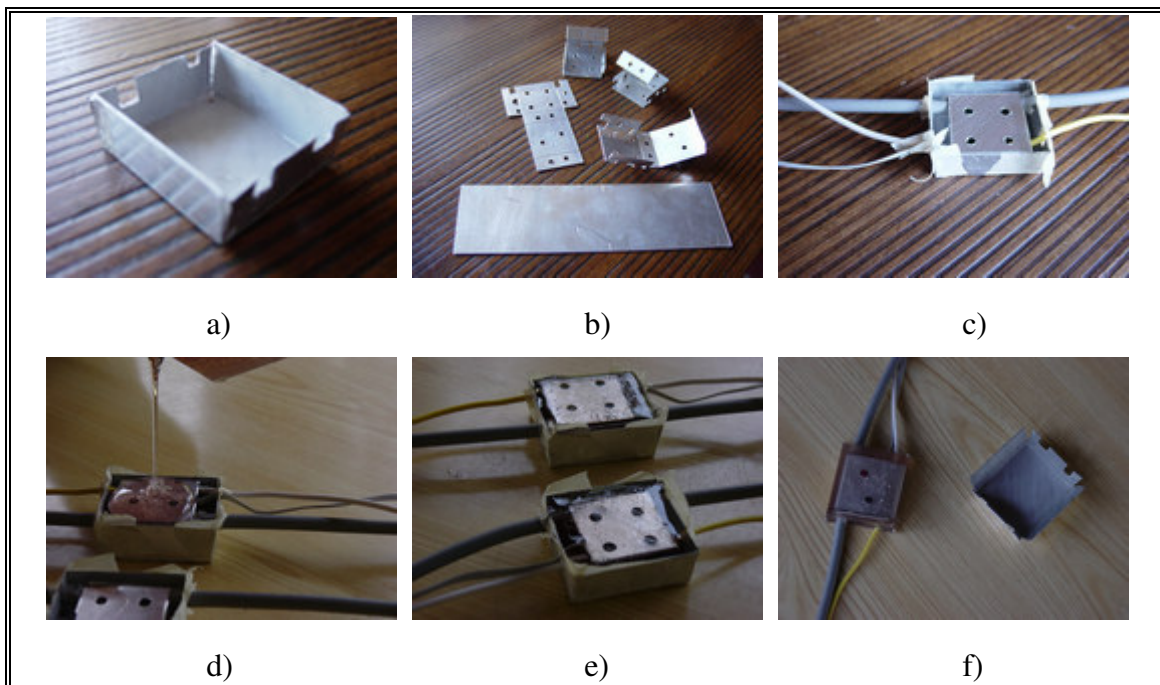


Figura 4.4: a) carcasa de aluminio sola b) jaula de Faraday sola c) Tarjeta con envoltorio antes de ser llenada con resina d) Vertiendo la mezcla e) Mezcla endurecida con carcasa de aluminio f) Tarjeta apantallada, envasada y lista.

4.5 Conectores y cables.

Aparte del cable usado por el bus es necesario especificar el cable que va a cada borne de las baterías y el que conecta el sensor con la tarjeta de sensado.

En el primer caso se usaron cables AWG, sólo para darle cierta consistencia al cable. Con respecto a los conectores se usaron de argolla y se soldaron.

En el segundo caso se usó cable de bus (el mismo que se usa para discos duros). Cada pata del sensor se soldó y se recubrió con un tubo termorretráctil. Para fijar el sensor al borne negativo, se abrió un conector de argolla y se rodeó el sensor, apretándolo con un tubo termorretráctil. De esta manera la argolla del conector se puede atornillar al borne de las baterías con una tuerca.

Mayores detalles de estas conexiones se aprecian en la figura 4.5.

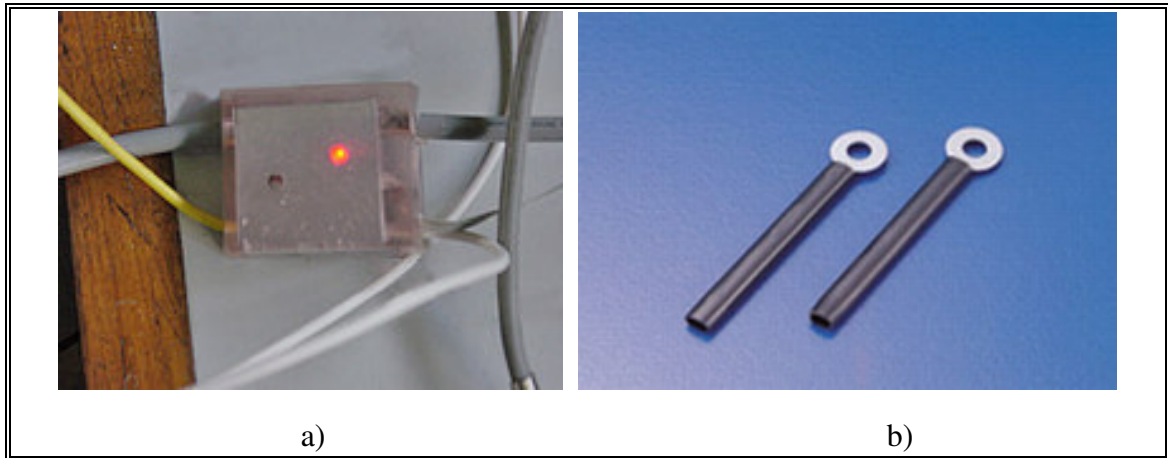


Figura 4.5: a) Cables de la tarjeta de sensado y b) Conectores a las baterías de la tarjeta de sensado

V BUS SERIAL DE DATOS

5.1 Aspectos importantes

Pese a que puede parecer un mero detalle se ha de tener sumo cuidado en elegir el cable y los conectores adecuados para que la transmisión de los datos sea la correcta. Además, debido a la cantidad de tarjetas es importante buscar una solución económicamente conveniente. También debe considerarse que mientras más conectores hayan en el bus, más posibilidades de desconexión accidental existen y más ruido se introducirá en el bus.

Debido a lo anterior se utilizó cable STP-5 doblemente apantallado, con malla de tierra y con recubrimiento de aluminio. Los conectores se pusieron de manera de interconectar una tarjeta a otra. De cada tarjeta sale a ambos lados el cable con una cierta dimensión. Entre medio de dos tarjetas existe un solo conector apantallado tipo Mini DIN 4. La gracia de este conector es que es barato, posee un cierto apantallamiento, es fácil de enchufar y permite la conexión de varios conectores a la vez sin hacer cortocircuito (como lo haría un conector jack).

La figura 5.1 muestra el tipo de conexión utilizado en el bus.

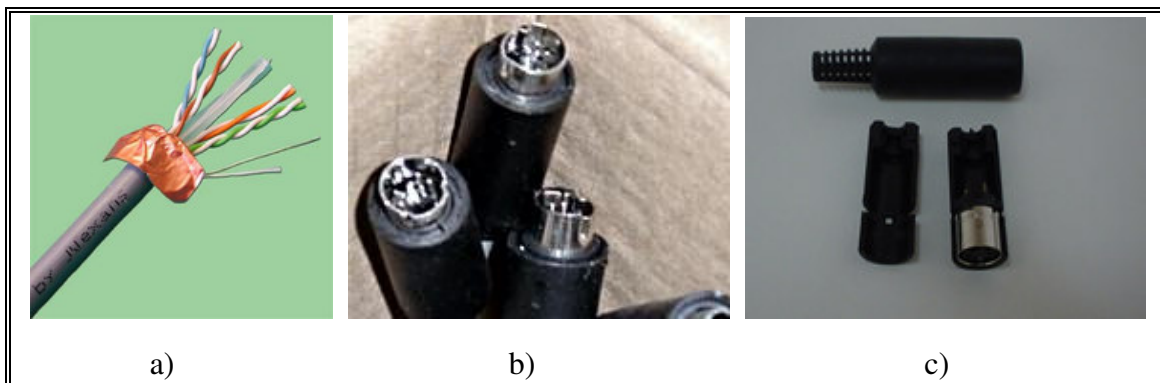


Figura 5.1: a) Cable bus 1-Wire, b) y c) conector utilizado para el bus serial

Para que la jaula de Faraday de las tarjetas funcione, debe estar conectada a la tierra del sistema. Se debió tener cuidado de hacer esta conexión en el momento del vaciamiento de la resina en cada una de las tarjetas.

Debido a que unos de los cables del bus es de alimentación, pese a que no consume mucha corriente, se alimentó por ambos lados, es decir, desde la batería 1A y desde la batería 12B (hay dos paquetes de baterías en paralelo, A y B, cada uno con 12 baterías en serie). El cable del bus que comunica la señal digital de los sensores (cable One Wire) se conectó al bus master por un extremo y se dejó libre por el otro.

En total, el bus construido tiene aproximadamente 20 metros.

5.2 Instalación definitiva en el vehículo

La instalación del bus parece sencilla, pero es necesario tener sumo cuidado porque una mala conexión puede significar el perder una de las tarjetas de sensado. Concretamente si los conectores que van a los bornes tocan la carrocería del vehículo y ya tienen un borne conectado, la tarjeta se quema porque el opamp TLV272 no aguanta voltajes negativos.

Debido a lo anterior es necesario envolver con huincha aisladora los dos bornes y los terminales de los conectores del bus One Wire en el momento de instalar los sensores en el banco.

Las tarjetas van pegadas a cada batería con Velcro.

La forma de conexión del bus no importa para el funcionamiento del monitoreo. Sólo debe haber cuidado en ver qué batería es monitoreada por qué sensor, pero aunque se conecte el bus en forma aleatoria los datos se presentarán ordenadamente porque el PIC marcará los números de los sensores por orden. Una conexión lógica es seguir la conexión de las baterías 1A a 12A, para luego proseguir con la batería 1B siguiendo hasta la batería 12B.

La figura 5.2 muestra un esquema de conexión final de los sensores.

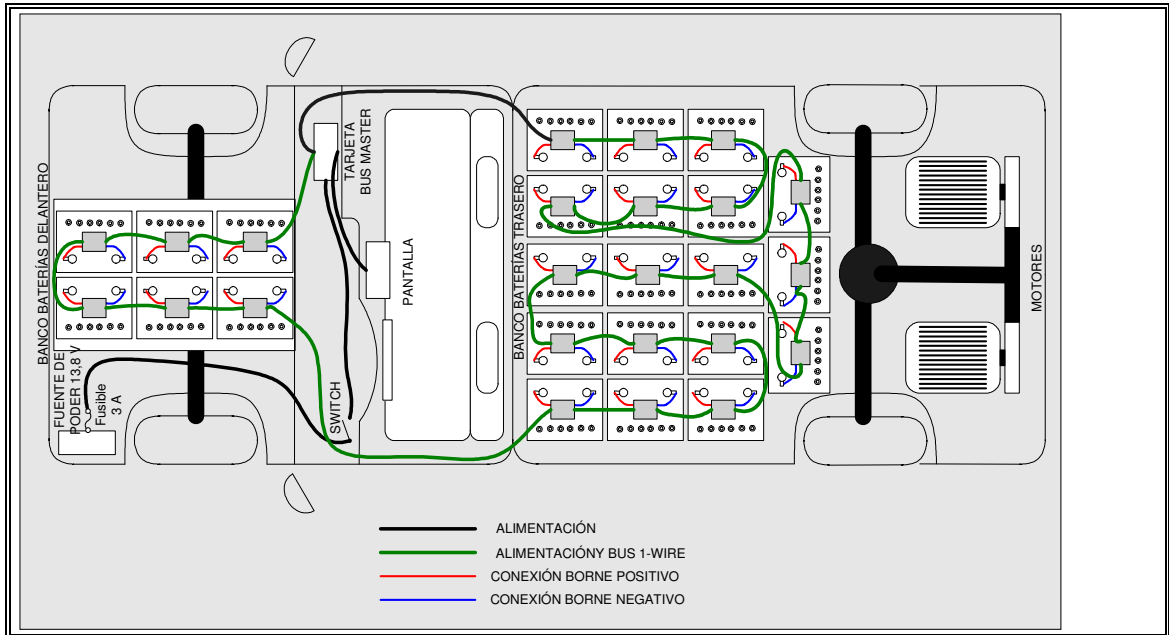


Figura 5.2: Diagrama general de conexión del bus One Wire junto con las fuentes de alimentación y la posición de la pantalla y de la tarjeta controladora.

VI PANTALLA

6.1 Descripción general

Tiene poco sentido adquirir datos, aún de forma eficiente, si no se muestran en forma cómoda para el usuario. Es por eso que se utilizó una interfaz gráfica de excelente calidad, que a la vez puede permitir ampliaciones futuras.

La empresa Apollo Display Technologies LLC ofrece una gama variadísima de pantallas de distintos tipos, tamaños y calidades. Después de hacer una búsqueda minuciosa se optó por adquirir una pantalla RGB de 320 x 240 pixeles, modelo DMF51167, la cual puede generar 256 colores.

Además de la pantalla se incorporó un touch screen marca Gunze Electronics USA Corporation modelo R-22.

Tanto la pantalla como el touch screen son controlados por una tarjeta de vídeo CDS1375, la cual permite conectarse con ambos dispositivos a través de una puerta serial. Todos los voltajes de alimentación requeridos son generados en este controlador. Además el contraste de la pantalla puede ser controlado por un potenciómetro digital, por lo que éste también se puede variar desde el puerto serial. La tarjeta cuenta con una memoria flash que puede almacenar imágenes de manera de agilizar el refresco de pantallas. Permite la realización de 10 tipos distintos de letras, así como dibujar círculos, cuadrados, líneas y elipses. Finalmente tiene incorporado un pequeño parlante que puede hacer sonar un beep si se requiere.

La figura 6.1 muestra una foto externa de la pantalla

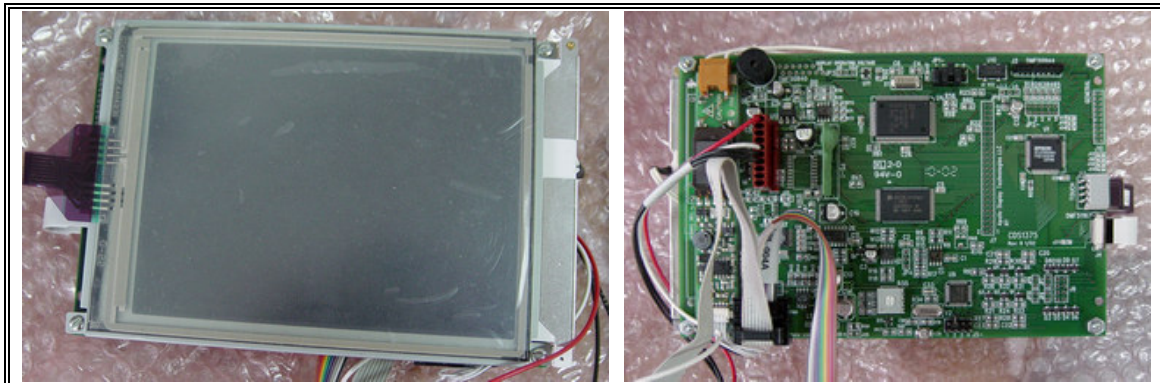


Figura 6.1: Fotografía de la pantalla DMF51167 y de su controlador

6.2 Touch Screen

Con el objeto de evitarse botoneras y hacer más eficiente la comunicación entre el usuario y el sistema se incorporó a la pantalla un touch screen.

El touch screen es una delgada lámina transparente que se pega frente a la pantalla y consiste en un arreglo de resistencias colocadas horizontal y verticalmente. Al tocar una zona de la lámina ambas resistencias, la vertical y la horizontal, varían su valor generándose dos voltajes de salida, un valor para la posición x y otro para la posición y , los cuales son posteriormente transformados por conversores A/D de 10 bits, en señales digitales.

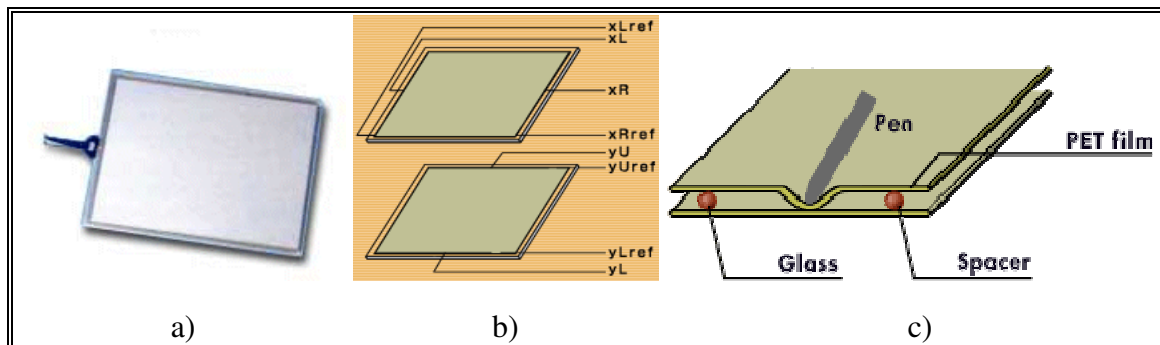


Figura 6.2: a) Aspecto del touch screen b) Arreglo de resistencias del touch screen c) Principio de funcionamiento

6.3 Despliegue de datos final de pantallas

Para facilitar la visualización de los datos la información obtenida se separó en cinco pantallas distintas, las cuales se pueden ir cambiando presionando en la parte inferior de la pantalla gracias a la presencia del touch screen.

Para poder realizar esta configuración debió grabarse previamente algunas imágenes en la memoria flash del controlador de la pantalla y el PIC, a través de la puerta serial, le va indicando qué imagen debe llevarse a la pantalla y en qué coordenadas, de manera que un solo pantallazo puede tener tres o cuatro imágenes pequeñas desplegadas.

La grabación de las imágenes se hizo mediante el terminal *hyperterminal* de un computador y las imágenes, previamente dibujadas en *Paintbrush* se transformaron a hexadecimal con un software especial que venía con la pantalla.

6.3.1 Pantalla de Voltaje

Cuando el usuario desea ver esta pantalla el controlador de la pantalla desplegará el siguiente dibujo en bitmap.

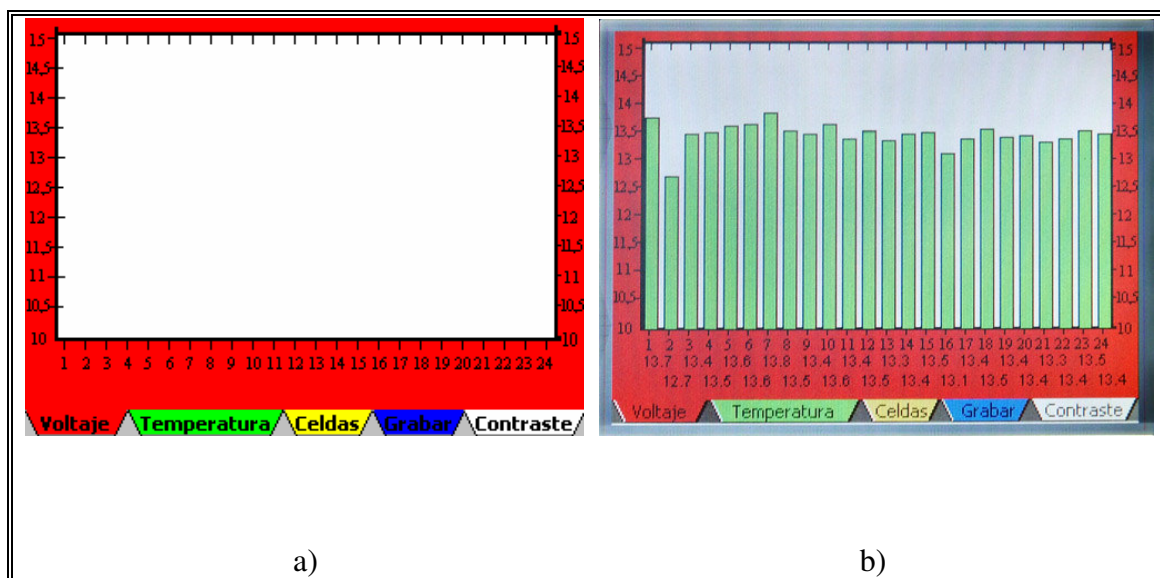


Figura 6.3: a) Dibujo bmp permanente en la pantalla “Voltaje” b) Apariencia final de la pantalla “Voltaje”

Las barras de voltaje son cuadradas y el valor del voltaje se muestra con un decimal bajo el número de la batería en dos filas. Las barras serán de color verde si están entre 12 y 13 V, amarillas si son menores que 12 V y azules si son mayores que 13 V. Si el conductor desea cambiar a otro menú debe apretar la pantalla con el dedo o con un lápiz para *Palm* en las etiquetas inferiores.

6.3.2 Pantalla de Temperatura

Cuando el usuario presione la etiqueta Temperatura, el PIC enviará los comandos al controlador de la pantalla para que ponga un color de fondo verde, y pegue los dibujos específicos de la memoria flash que correspondan a este pantallazo. Al igual que en el caso del voltaje se dibujará un rectángulo de altura proporcional a la temperatura de cada batería y se escribirá el valor de la temperatura con un decimal debajo de la barra correspondiente. Si la temperatura está entre 15°C y 30°C las barras son de color verde. Si la temperatura sobrepasa los 30°C las barras se ponen rojas y si están por debajo de los 15°C las barras se ponen amarillas.

La apariencia se muestra en la figura 6.4.

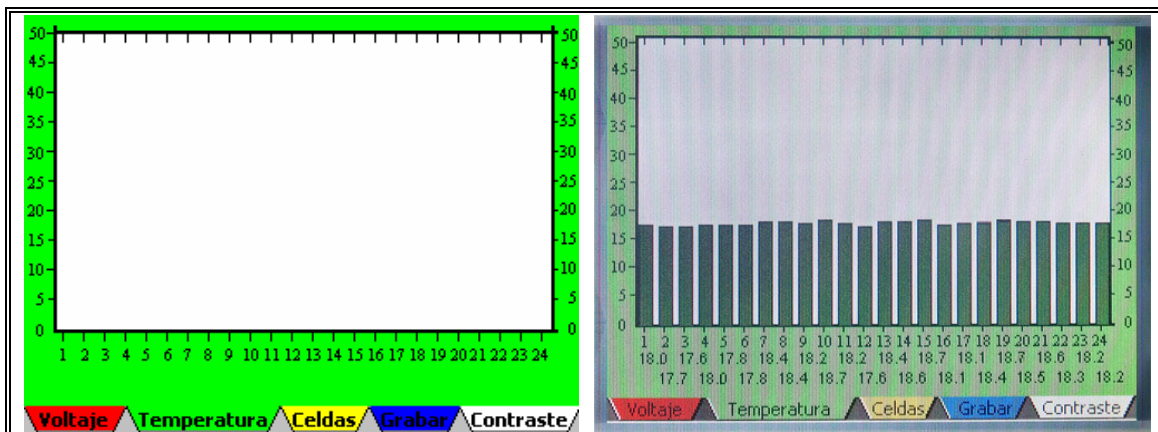


Figura 6.4: a) Dibujo bmp permanente en la pantalla “Temperatura” b) Apariencia final de la pantalla “Temperatura”

6.3.3 Pantalla de Celdas Solares

Esta pantalla muestra el estado de los paneles solares que se incorporarán a la camioneta. Por indicaciones de la persona que está trabajando en estos paneles la capacidad máxima de potencia entregada por el panel solar será de 1kW. El PIC, como se verá más adelante, deberá ser capaz de recibir la información que los circuitos del panel le entreguen a través del convertor A/D1 (pin 3) del PIC y él deberá presentar el dato en la pantalla, así como calcular y presentar la energía que en cierto tiempo se ha recibido por esta fuente de energía.

La figura 6.5 presenta la parte estática de la pantalla que se saca, como en los casos anteriores, de la memoria flash del controlador de la pantalla y una foto con la presentación final. Nótese que también se entrega el intervalo de tiempo en el que se ha calculado la energía entrante. Este tiempo puede reiniciarse si se oprime el botón “Reiniciar tiempo”.

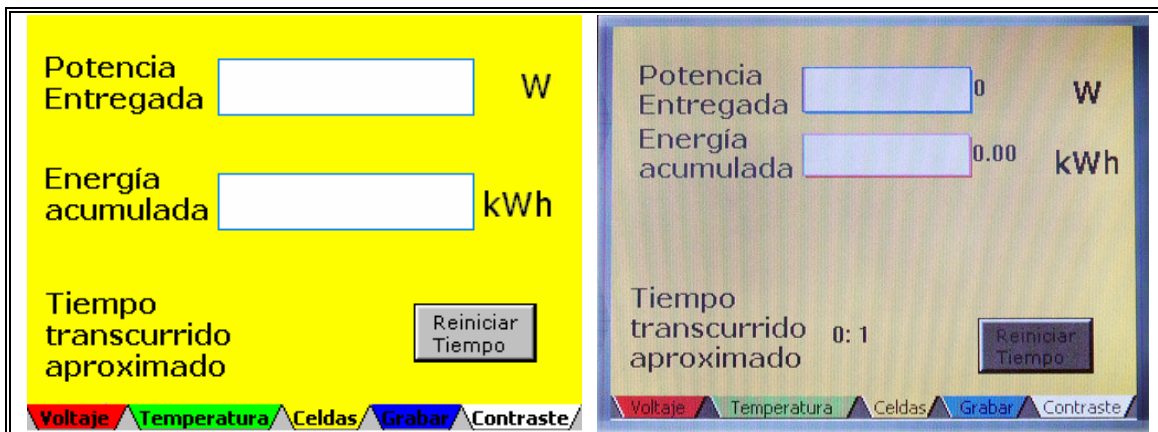


Figura 6.5: a) Dibujo en bitmap de la parte fija de la pantalla “celdas” b) Fotografía de la presentación final de la pantalla “celdas”.

6.4 Pantalla “Grabar”

La interfaz de grabación maneja los datos que se están guardando en la memoria flash BQ4016Y. Además, desde ella, se permite al usuario traspasar los datos de esta memoria a un PC. Además en la parte inferior de la pantalla aparece en letras rojas el estado de grabación del sistema de monitoreo. Existen cinco estados posibles:

- a) Sin grabar: el sistema está adquiriendo los datos desde el sensor, pero no los está grabando en la memoria flash.
- b) Grabando: el sistema está adquiriendo los datos desde el sensor y los está grabando en la memoria flash.
- c) En Pausa: el usuario puede detener la grabación de los datos por un momento. Para seguir grabando deberá apretar en la pantalla el botón “Grabar/Seguir grabando”.
- d) Memoria Borrada: Con el botón “borrar” se borran todos los datos de la memoria.

- e) Enviando: El sistema está enviando los datos a un PC. Cuando termina queda en el estado “sin grabar”.

Finalmente una barra muestra el porcentaje de memoria utilizada en ese momento.

La apariencia de la interfaz se muestra en la figura 6.6.

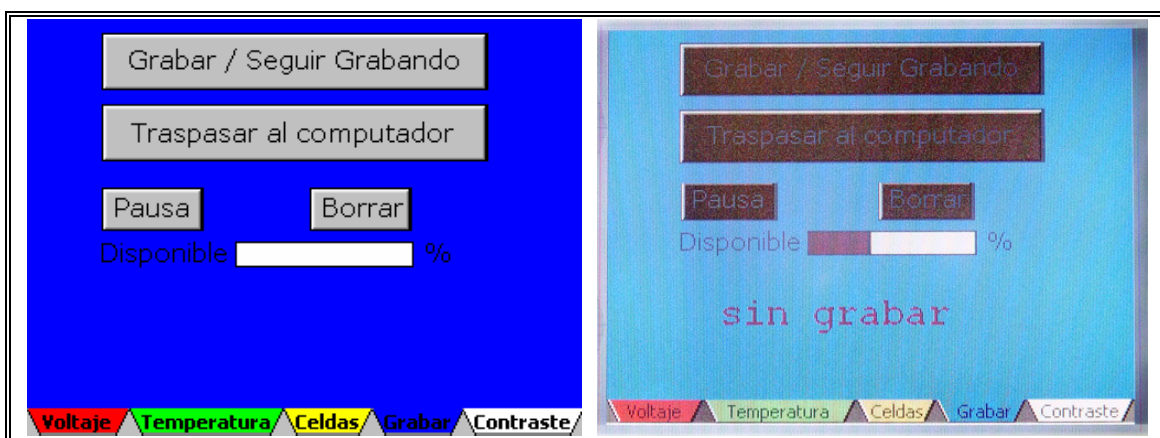


Figura 6.6: a) Dibujo de la parte fija de la pantalla “Grabar” b) Apariencia de la pantalla en la realidad.

La memoria del controlador de la pantalla es pequeña por lo que es necesario optimizar al máximo las imágenes. En este caso, sólo los botones grises y la barra de menú inferior están guardados en la memoria.

6.5 Pantalla “Contraste”

Esta pantalla corresponde a la presentación inicial del monitoreo. Contiene el escudo de la Universidad Católica y la descripción del sistema. Además se puede controlar el contraste de la pantalla con dos botones, aprovechando el control de contraste digital que permite el controlador de la pantalla.

La apariencia de esta pantalla se muestra en la figura 6.7.

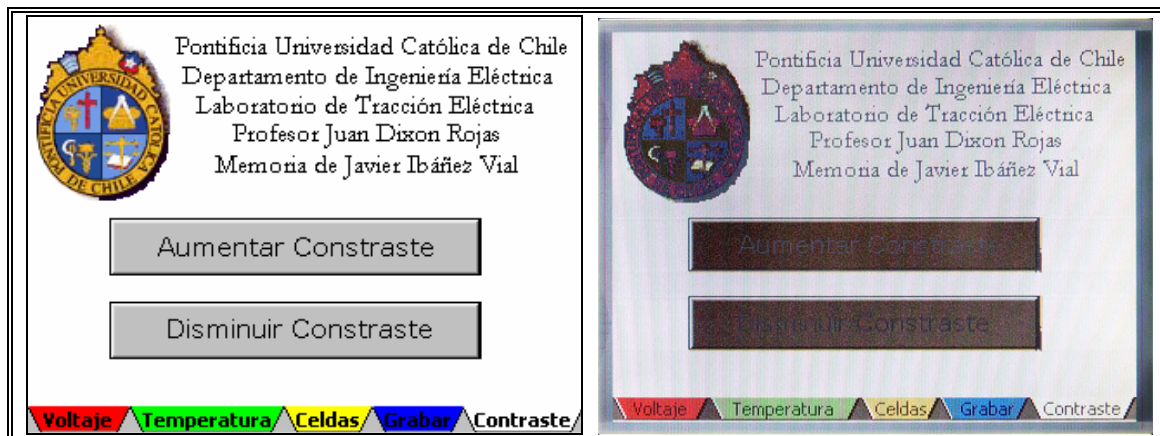


Figura 6.7: a) Apariencia en bitmap de la pantalla “Contraste” b) Fotografía definitiva de la pantalla “Contraste”

6.6 Otras características de la presentación en pantalla.

No basta con presentar los datos en la pantalla. Además es necesario avisarle al usuario cuándo se ha presionado realmente el touch screen. Esto se realiza haciendo sonar el beep por 2 ms.

Cuando se toca el touch screen el controlador de la pantalla -que a la vez controla el touch screen- envía las coordenadas a través de una puerta serial al PIC. El PIC, detecta una interrupción por puerta serial y lee lo que ha llegado. El formato de los datos llegados se muestra en la figura 6.8. Si se toca la pantalla en el extremo inferior izquierdo, la transmisión será T0000;0000←. Si se toca la pantalla en el extremo superior derecho la transmisión será T1024;1024←.

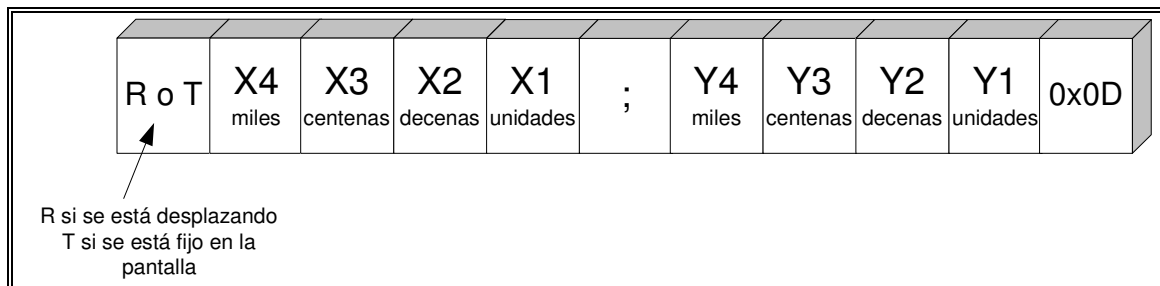


Figura 6.8: modo de envío de datos de las coordenadas del touch screen

El PIC graba estos 11 bytes y los analiza. Si los datos llegados son válidos se ve si corresponden a un punto de la pantalla que tenga dibujado un botón. Si es así se hace sonar el beep y se ejecuta la acción correspondiente a ese botón.

6.7 Contenedor y ubicación definitiva de la pantalla en la cabina

La pantalla se instaló dentro de un pequeño contenedor y se puso en el techo de la cabina del vehículo entre las dos viseras. Se le dio cierta inclinación de su parte inferior para que fuera más fácil verla desde el sitio del conductor.

Con respecto al contenedor de la pantalla, se barajaron muchas posibilidades, entre ellas, la misma resina P4, fibra de vidrio, plástico, acrílico y policarbonato. La manera más sencilla y barata, fue el acrílico. Las otras posibilidades son quizá más atractivas, pero es difícil hacer un molde en el que quepa exactamente la pantalla. Ahora bien, se cuidó de que el grosor del acrílico resistiera temperaturas como las que puede alcanzar una cabina de vehículo. Haciendo las consultas a expertos en el tema se decidió por hacer una caja de acrílico negro, con perforaciones para ventilación de circuitos y de un grosor de 6 mm. La caja es rectangular y tiene 2,5 cms. de profundidad. Debido a que pegar directamente este chasis en el techo es poco seguro y, además, tapa el espejo retrovisor hubo que hacer un corte en el recubrimiento interior del techo de la cabina y meter allí el contenedor. Sólo se cortó la parte de arriba del techo con lo que la pantalla adoptó una mejor verticalidad.

Para fijar el contenedor al techo se pusieron dos barras de aluminio perforadas a las cuales se aperraron unos tornillos que afirmaban la parte trasera de

la caja de acrílico. Las tuercas donde atornillan los tornillos que fijan la parte delantera de la caja de acrílico se fijaron a la tapa trasera con adhesivo poxipol. Finalmente, se puso un perfil de aluminio en la parte más alta de la caja de la pantalla, el cual va por dentro del techo e impide que se caiga. Todo lo anterior quedó muy firme.

El esquema de la figura 6.9 ilustra lo descrito en el párrafo anterior.

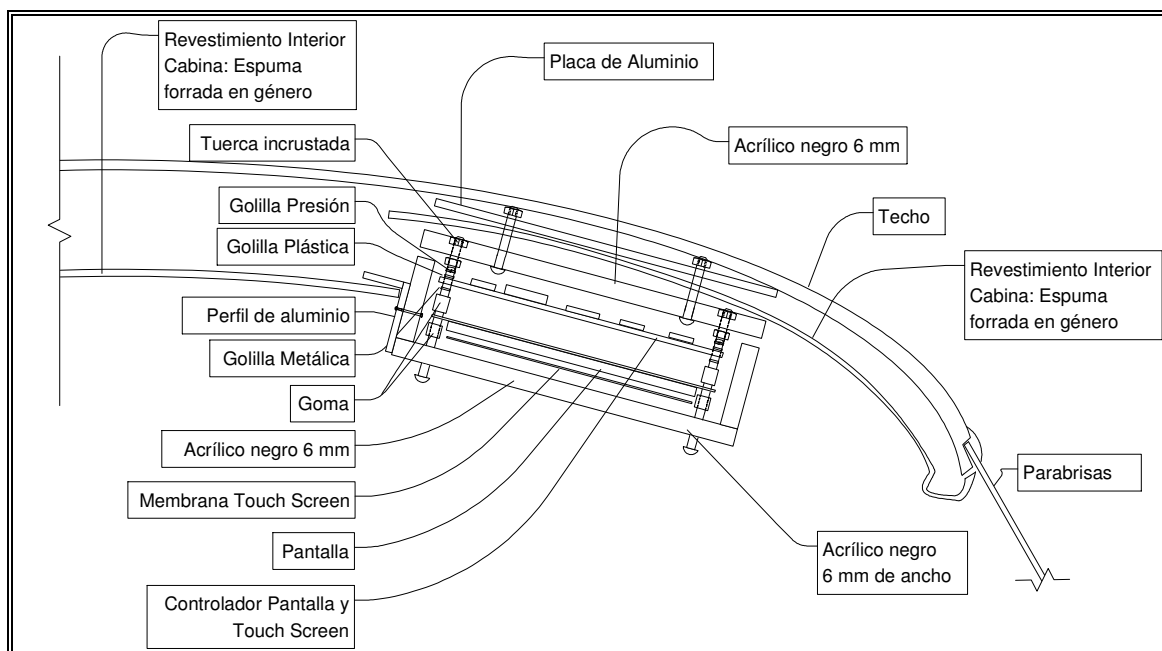


Figura 6.9: sistema de anclaje de la caja de la pantalla a la cabina.

La figura 6.10 muestra algunas fotos de la posición definitiva de la pantalla.



Figura 6.10: Fotografías de la instalación de la pantalla en la cabina.

VII TARJETA CONTROLADORA

7.1 Funciones Generales

Es interesante considerar que el microcontrolador PIC deberá controlar muchas cosas a la vez. Deberá realizar la rutina de adquisición de datos del bus One Wire, la recepción de las coordenadas del touch screen cuando éste sea presionado, la grabación, lectura y direccionamiento de la memoria flash y los comandos que permitirán observar los datos en la pantalla.

7.2 Microcontrolador

El microcontrolador PIC16F877A es el cerebro del sistema de monitoreo. Si bien la pantalla tiene un controlador propio y el touch screen también, el PIC pondrá de acuerdo todas las actividades del monitoreo y le dará las prioridades necesarias. En la figura 7.1 se muestra un esquema que aclara todas las tareas que coordina el microcontrolador.

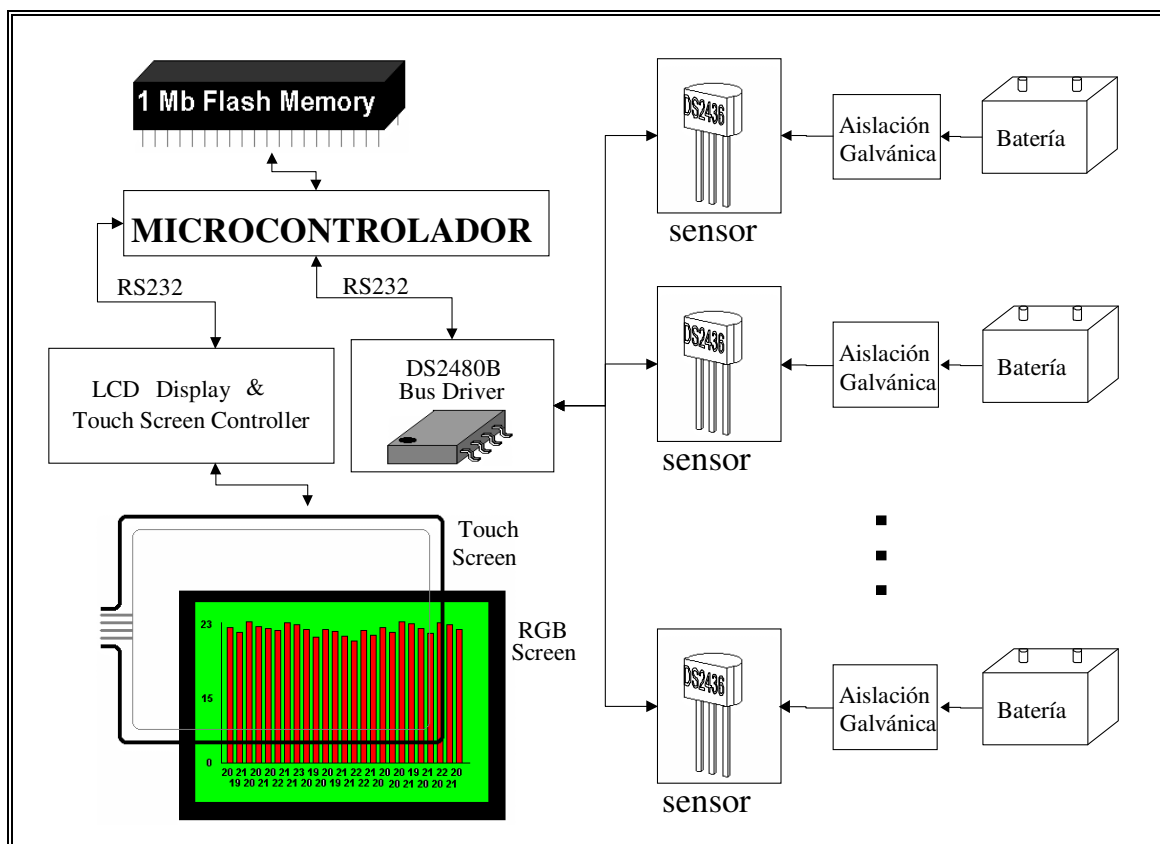


Figura 7.1: Esquema total del sistema de monitoreo.

Se eligió este controlador por muchas razones. Es más, en un principio se utilizaría el DSP TMS320F241FN, pero la capacidad del PIC de manejar varias puertas seriales configuradas por software, lo hizo más conveniente.

El PIC debe comunicarse con el bus One Wire a través de una puerta serial de 9600 bps. Al mismo tiempo debe comunicarse con la pantalla por una segunda puerta serial de 57600 bps. Esta comunicación debe incorporar un control de flujo para no sobrecargar el buffer de entrada de la pantalla. Debe comunicarse también con una tercera puerta serial al touch screen con una velocidad de 19200 bps. Finalmente, debe comunicarse con una cuarta puerta serial al computador a 115200 bps.

Se eligió el PIC16F877A porque tiene una memoria de programa de 8 Kb y una memoria eeprom de 256 bytes. El programa que maneja todos los elementos es largo, sobre todo considerando que se utilizó un compilador en C, el cual, aunque facilita la programación es menos eficiente que el assembler. En todo caso, el compilador vale la pena debido a la complejidad del software diseñado. Debido a esto se utilizó un 99% de la memoria de programa y el uso máximo de la memoria de registros fue de 100%, por lo que el microcontrolador apenas cumplió con los requerimientos del sistema.

Se utilizó un cristal de 19,6 MHz, con lo que la velocidad del controlador es de 4,9 MHz. Esto quizá supone una desventaja con respecto al DSP, ya que éste funciona a 20 MHz, pero como en esta aplicación lo lento son los periféricos y la comunicación con ellos, no tiene mayor relevancia.

La elección del paquete de 40 pines del PIC se debe a la gran cantidad de patas que se requieren para los distintos dispositivos. El PIC tiene una disponibilidad de 33 puertos I/O: 12 se utilizaron para manejar la memoria, dos para la comunicación con el bus One Wire, tres para la comunicación con la pantalla, tres para la comunicación con el touch screen y dos para la comunicación con el computador. Se dejaron seis conversores A/D para futuras aplicaciones, una de las cuales será la de los paneles solares (A/D 1), con lo cual sólo quedan 3 patas libres.

Se utilizó la interrupción externa de la puerta B0 para recibir las interrupciones del usuario a través del touch screen.

Para calcular la energía entrante de las celdas solares y para refrescar los gráficos de la pantalla se usó la interrupción por *timer0*, pero sólo utilizando el flag, ya que la primera prioridad, después del usuario, debe tenerla la pantalla.

Se utilizó el compilador CCS para programar el PIC.

7.3 Control del bus de datos One Wire y calibración de sensores

El bus One Wire se controla a través de los pines C4 (salida) y C5 (entrada) del PIC. Estos están configurados como puerta serial siguiendo el protocolo RS232 a una velocidad de 9600 bps.

A través de estas dos puertas se envían los comandos de ocho bits en forma serial que el bus driver requiere para obtener la temperatura y el voltaje de cada sensor.

Debido a que cada sensor tiene un número único, en alguna parte del PIC deben guardarse estos datos. Para esto se utilizó la eeprom del PIC.

Cuando se tuvo acceso a los sensores se creó un rutina que mostrará el número de identificación de cada sensor. El primer byte es común a todos los sensores porque todos son de la familia DS2436, cuyo byte común es 1Bh. Los tres bytes siguientes eran distintos, propios de cada sensor. Luego venían 3 bytes de valor 00h y finalmente el checksum, obviamente distinto en cada uno. En conclusión, es necesario guardar sólo 4 bytes de identificación. Considerando 24 baterías suman 96 bytes que deben guardarse en la eeprom del PIC.

Además, debido al circuito de aislación galvánica, es necesario calibrar el dato de voltaje de los sensores. Se consideró suficiente guardar un datos de pendiente de 16 bits con la coma al medio y la misma precisión para el dato de offset o constante a sumar. De esta manera, por cada sensor deben guardarse 4 bytes para la calibración. La temperatura no necesita calibración porque los sensores vienen calibrados.

En conclusión los números ID de los sensores ocupan desde la posición 00h hasta la posición 5Fh de la eeprom y los datos de la calibración usan desde la posición 60h hasta la posición BFh. Durante la rutina de sensado se irán leyendo estos datos a medida que se requieran.

Utilizando los comandos de la tabla 1.1 y de la figura 3.4, la figura 7.2 muestra el diagrama de flujos de la comunicación con el sensor.

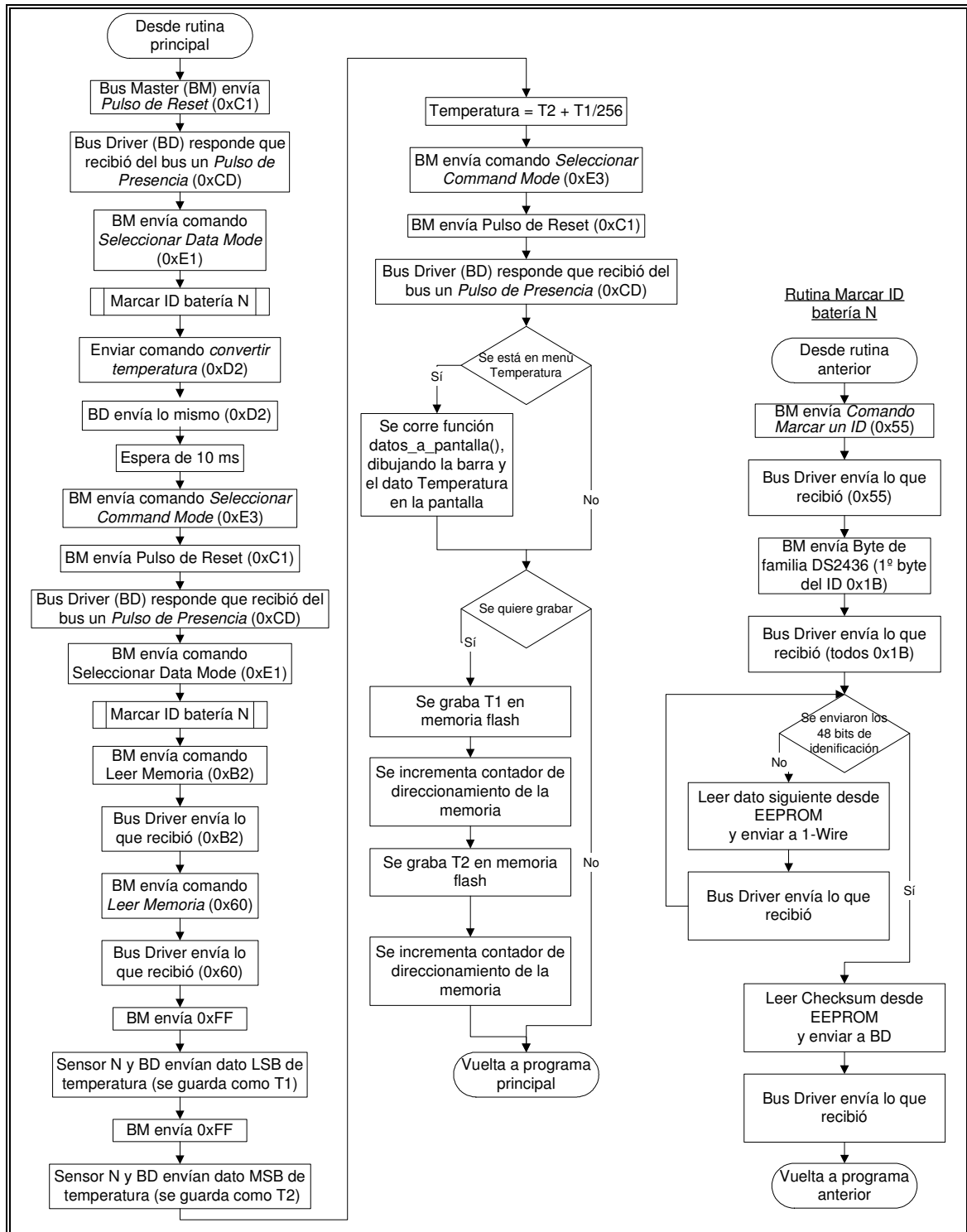


Figura 7.2 a) Diagramas de flujo de la rutina necesaria para solicitar temperatura.

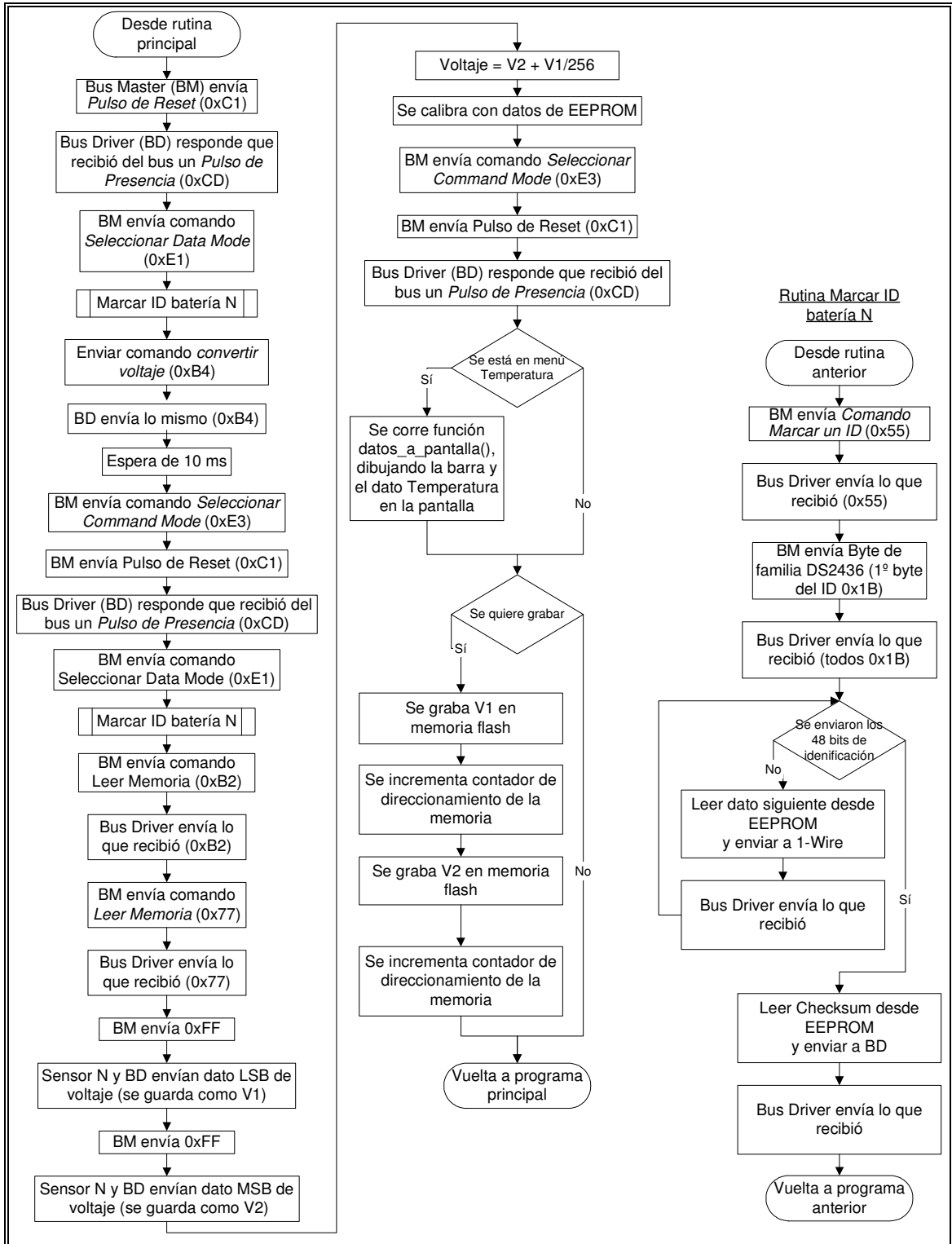


Figura 7.2 b) Diagrama de flujo de la rutina necesaria para solicitar voltaje.

En el párrafo 7.9 se presenta un cuadro resumen con las características más importantes de las funciones del programa completo y el Anexo I contiene el programa completo del microcontrolador con sus comentarios.

La función volt(bateria) con la ayuda de la función marcar_ID1(bateria) solicita al bus One Wire, mediante la rutina descrita en la figura 7.2, el dato de voltaje de la batera n° bateria y lo guarda como una variable de tipo float. Además, se calibra el dato, se imprime en la pantalla si el menú “Voltaje” está activo y se guarda en la memoria Flash externa si se están guardando los datos.

La función temp(bateria), realiza el mismo trabajo que volt(bateria), pero con la temperatura y sin calibrar.

7.4 Control de la pantalla

La mayor parte de los recursos del PIC se utilizaron en manejar de manera adecuada los datos que irán a la pantalla. La puerta C2 y C3 del PIC se utilizaron como puerta serial para salida y entrada de datos respectivamente. Además se instaló un control de flujo en la puerta B0. La pantalla envía un pulso de 5V cuando el 80% de su buffer de entrada está lleno. El PIC espera 400 ms antes de enviar algún otro dato.

Como se explicó en el capítulo 6, las partes inmóviles de las cinco pantallas están almacenadas en la memoria flash del controlador de la pantalla. El controlador permite escribir texto, figuras geométricas y líneas. La tabla 7.1 resume y describe los comandos que se pueden enviar en código ASCII a la pantalla.

Tabla 7.1: Resumen de comandos más importantes permitidos por el controlador de la pantalla.

Comando	Descripción
Backlight_off;	Apaga la luz de fondo de la pantalla
Backlight_on;	Enciende la luz de fondo de la pantalla
Beep 10;	Hace sonar el beep por 10 ms
Bmpfile x0 y0 "texto"	Pega una imagen desde la posición x0 y0
EF;	Borra memoria flash
LF x0 y0 x1 y1	Graba en la memoria flash el contenido en la pantalla comprendida entre x0,y0 y x1,y1. El controlador devolverá un número de identificación de la imagen
SF n x0 y0	Despliega imagen n en coordenada x0,y0
Clearscreen;	Deja la pantalla en blanco
Circle x y r lc lw fc;	Dibuja un círculo con centro en x,y, radio r, color de línea lc, ancho de línea lw y color de llenado fc
Line x0 y0 x1 y1 lc lw	Línea desde x0,y0 hasta x1,y1 color lc y grosor lw.
Recyangle x0 y0 x1 y1 lc lw fc	Rectángulo con vértices opuestos x0,y0 y x1,y1 color de línea lc, ancho de línea lw y relleno de color fc
Setbackground n;	Elige color de fondo n. Se pueden elegir 256 colores
Textring x y "texto" ty tc fc	Escribe desde coordenada x,y "texto" con tipo de texto ty (10 tipos), color de línea del texto tc (256 posibilidades) y llenado de color fc (256 posibilidades)
Vee_down n;	Bajar contraste n niveles. El total de niveles es 256
Vee_up n;	Sube contraste n niveles.

La función `mostrar_menu()` muestra la parte estática de las pantallas usando la memoria flash del controlador de la pantalla. Pregunta al touch screen si el usuario solicitó cambio de menú.

La función `datos_a_pantalla(dat_comp, bateria, vot)` permite a las funciones `volt()` y `temp()`, una vez obtenidos los datos del bus One Wire, desplegar los datos en la pantalla si se está en el menú “Voltaje” o “Temperatura”. Esta función imprime en la pantalla la barra proporcional al dato obtenido (`dat_comp`) y escribe su valor bajo la barra. La posición de la barra estará dada por el segundo argumento de la función (`bateria`) y el tercer argumento indica si es un dato de voltaje o temperatura.

En el párrafo 7.9 se presenta un cuadro resumen con las características más importantes de las funciones del programa completo y el Anexo I contiene el programa completo del microcontrolador con sus comentarios.

7.5 Control del touch screen

La comunicación con el touch screen se hace mediante las puertas C6 y C7 del PIC para enviar y recibir respectivamente. Estas puertas corresponden a la puerta serial del PIC, con lo que se puede activar la interrupción por puerta serial. La rutina de interrupción `serial_isr()` va guardando los datos llegados en el arreglo `buffer[]`. Aquí ya se empieza a hacer un chequeo de los datos que llegan. Por ejemplo, si se recibe una “T” o una “R” quiere decir que se trata de la primera letra de envío, por lo tanto este dato se guardará en la posición 0 de `buffer[]`.

La interrupción va recibiendo los datos de a uno, de manera que no se queda esperando todos los datos que faltan por llegar. Así, la interrupción de debiera durar más que 10 ciclos.

La función `datook()` entrega un 1 si el dato de `buffer[]` es válido o un 0 en caso contrario. Se considera un dato válido cuando en la posición 0 hay una “T” o una “R”, cuando en las posiciones 1 a 4 hay números que equivalen a códigos ASCII de números (del 42 al 52), la posición 5 es una coma, las posiciones 6 a 10 son

números, al igual que en las posiciones 1 a 4, y la posición 11 es un 13 (“return” o 0Dh).

Para hacer los cambios de pantalla una vez que la función `datook()` ha validado el dato de entrada, es necesario ejecutar las operaciones que el usuario pide. Como se deben respetar los cálculos que exigen un tiempo determinado y para no estorbar la recepción de los datos desde el touch screen, se usa una interrupción por timer, pero oculta, es decir, la interrupción está deshabilitada, pero se utiliza su flag.

Existen algunos momentos en que es necesario esperar algunos milisegundos; por ejemplo, cuando se espera recibir un dato o cuando se está haciendo una conversión o cuando se están graficando los datos en la pantalla. En vez de esperar sin hacer más, se aprovechan esos momentos para revisar si hay un dato correcto en `buffer[]` o si es tiempo de actualizar el cálculo de la energía de las celdas solares y el reloj o si es necesario esperar 400 ms antes de seguir enviando datos a la pantalla. Esto es lo que realiza la función `wait(t)`; espera `t` milisegundos, pero al mismo tiempo está viendo si conviene actualizar las rutinas recién mencionadas.

7.6 Control y direccionamiento de la memoria

Cuando el usuario está en el menú “Guardar” y presiona el botón “Grabar/Seguir Grabando” un bit de control llamado *guardando* pasa de 0 a 1. Así se activan todos los procesos que permiten el guardado y direccionamiento de datos desde el bus One Wire a la memoria flash externa.

La memoria BQ4016Y [Benc96] tiene una capacidad de almacenaje de 1 MB, de 8 bit por byte. Esto implica que tiene 20 entradas de direccionamiento (A0 a A19), 8 entradas de datos (D0 a D7). Estas últimas pueden servir de salida, cuando se requiere leer un dato de la memoria, o de entrada, si se quiere escribir en la memoria. Además, la memoria tiene 3 pines de control: CE o Chip Enable, OE o Output Enable y WE o Write Enable. Como no se dispone de 31 puertas del PIC para controlar sólo la memoria, se utilizaron contadores de 12 bits en cascada para el direccionamiento. De esta forma se utilizan 12 puertas: 8 para los datos (banco D del PIC), una para incrementar el clock de los contadores (puerta E0 del PIC), una para

resetear los contadores (puerta E1), una puerta para activar las salidas de la memoria con el controlador OE (puerta C1) y una puerta para permitir la escritura en la memoria (puerta C0).

La figura 7.3 muestra un esquema del hardware implementado para permitir el almacenamiento de los datos en la memoria flash externa.

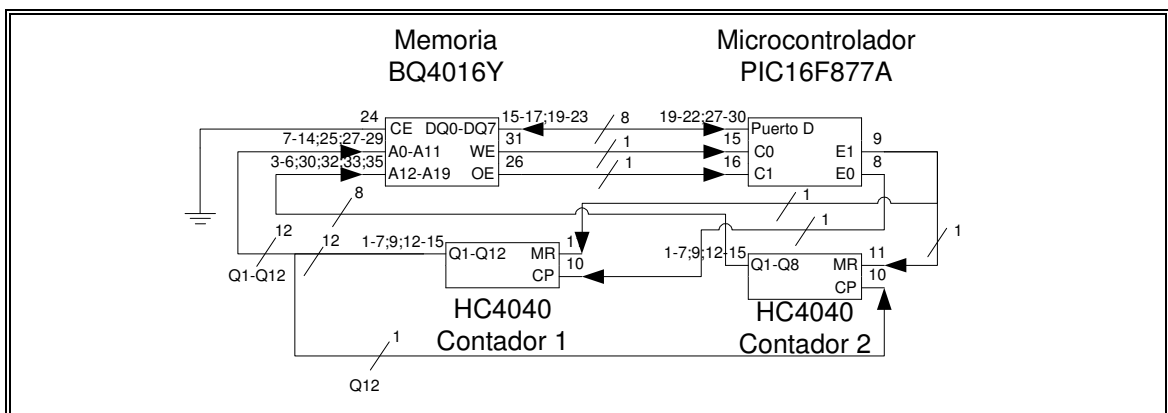


Figura 7.3: Diagrama del hardware para el direccionamiento de la memoria flash externa

El timing para escritura y lectura de la memoria lo realizan las funciones `a_memoria(dato)` y `de_memoria()`, con ayuda de las funciones `incrementar_dirección()` y `ciclo(i,j)`.

Debido a que los datos se guardan uno detrás de otro, no es necesario especificar una dirección de memoria específica de guardado. Cada vez que se graba un dato con `a_memoria(dato)` esta función incrementa los contadores en 1 con la función `ciclo(i,j)` e internamente se lleva un registro de 20 bits que maneja la función `incrementar_dirección()` para saber en qué lugar se guardó el dato.

La función `ciclo(i,j)` puede ser usada para incrementar `i` veces el contador. El parámetro `j` se usa en uno si se quiere resetear los contadores y cero si se quiere lo

contrario. Para grabar datos, uno detrás de otro, debe ocuparse esta función en ciclo(1,0).

La memoria se demora 70 ns en escribir o entregar un dato. Considerando que un ciclo del PIC demora 204 ns. no es necesario esperar a la memoria. La función `a_memoria(dato)` pone el dato a guardar en el banco D, ejecuta ciclo(1,0), baja el flag WE y lo sube.

Ahora bien, se dijo que los datos de temperatura son de 16 bits, 8 para el entero y 8 para los decimales. Asimismo ocurre con el voltaje y con la potencia de las celdas solares. Para guardar ordenadamente los datos primero se graba un número que indica qué dato se guardará en la dirección siguiente y subsiguiente (decimal y entero respectivamente). Este número va de 1 a 49. Será 49 si los datos que se guardarán a continuación serán de la potencia de las celdas solares. Si el número es 1, 3, 5 hasta 47, los datos siguientes corresponden a temperatura de la batería 1, 2, 3 hasta 24. Si el número es 2, 4, 6 hasta 48, los datos siguientes corresponden a voltaje de la batería 1, 2, 3 hasta 24. De esta manera, cuando el computador quiera leer los datos no habrá confusión.

La figura 7.4 resume la forma de almacenamiento de los datos en la memoria flash.

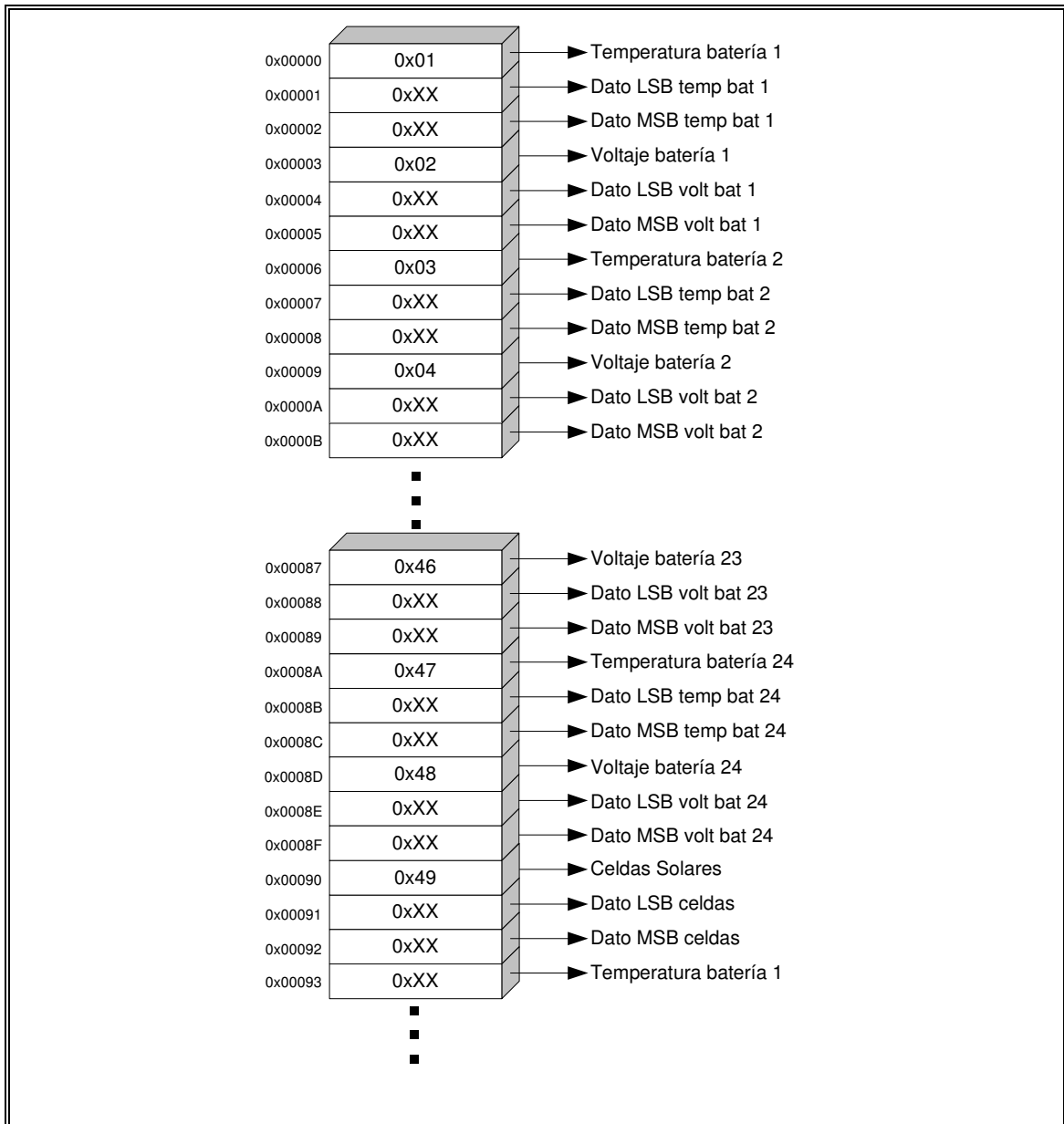


Figura 7.4: Forma de almacenamiento de los datos en la memoria flash externa.

Para leer los datos se utiliza la función `de_memoria()` que, en coordinación con las funciones `ciclo(1,0)` e `incrementar_direccion()`, entrega los datos uno detrás de otro.

Cuando el usuario presiona el botón “Traspasar al Computador” del menú “Guardar” se ejecuta la función `enviar_datos()`, que partiendo desde la dirección 00h hasta la dirección que contenga `incrementar_direccion()`, envía los datos al computador. La forma de envío se explicará en el párrafo 7.7.

Un detalle que vale la pena mencionar es que la función `incrementar_dirección()` va guardando la dirección en la que se está guardando un dato. Si por accidente o por otro problema, se apagara el monitoreo, para no perder el lugar en que quedó la última grabación, la dirección de 20 bits se va guardando en las direcciones FDh, FEh y FFh de la eeprom del PIC. Cada vez que se enciende el monitoreo lo primero que se hace es cargar desde estas direcciones el valor de la dirección. La función `recuperar_punteros()` se preocupa de esta operación al inicio del programa `main()`.

7.7 Comunicación con el computador

7.7.1 Hardware

La comunicación con el computador se hace a 115200 bps a través de las puertas B5 y B4 para enviar y recibir respectivamente. Esta comunicación debe seguir el orden con que se han guardado los datos previamente en las funciones `temp()` y `volt()` gracias a la función `a_memoria(dato)`.

7.7.2 Software del PIC

La figura 7.4 muestra la forma de almacenamiento de los datos en la memoria. Cuando el usuario presiona “Traspasar al computador” se establece una comunicación con el PC. La figura 7.5 muestra un diagrama de flujo de la comunicación entre microprocesador y computador.

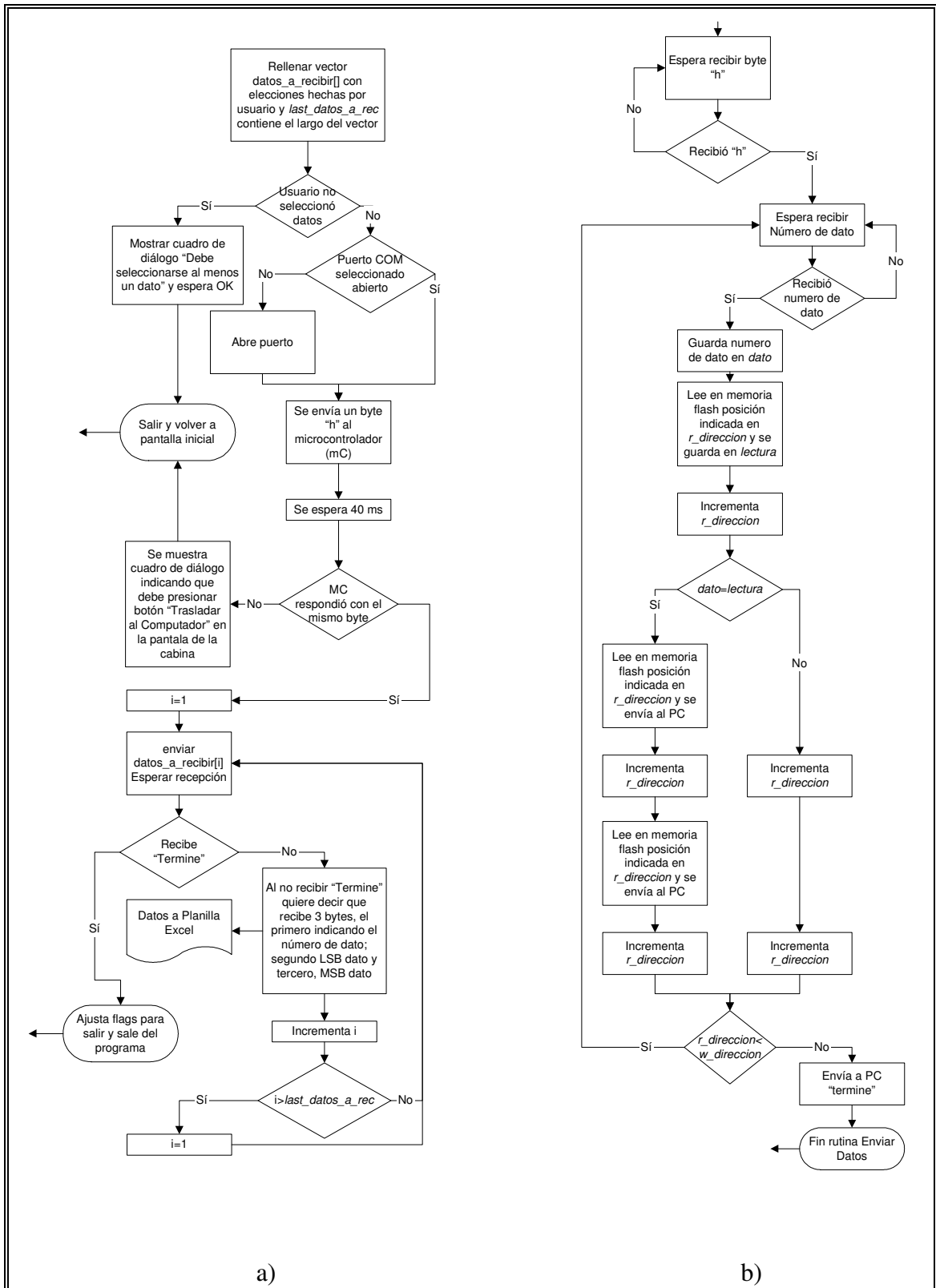


Figura 7.5: Comunicación PIC y PC. a) Rutina realizada por el computador PC.
b) Rutina realizada por el PIC.

7.7.3 Software de Interfaz desarrollado en el computador

El software que se diseñó en el computador debe ser capaz de recibir los datos ordenadamente según el esquema de la figura 7.5. Este software se diseñó en Visual Basic. La idea central es tratar de cargar lo menos posible de trabajo al PIC, ya bastante sobrecargado, y aprovechar los recursos del PC. Es por eso que el control de la comunicación lo lleva siempre el PC.

Como se puede ver en la figura 7.5 el computador tiene un arreglo de porte máximo 49 llamado `datos_a_recibir[]`, donde se guardan los datos que el usuario del PC solicita. Se pueden elegir únicamente la transmisión de algunos voltajes o de algunas baterías, o sólo el dato de las celdas solares, así como todos los datos. ¿Cómo se hace esto? El PC enviará al PIC el número único de dato que requiera (el número de dato es un número del 1 al 49; 1, 3, 5 ... son datos de temperatura de las baterías 1, 2, 3, ...; 2, 4, 6 ... son datos de voltaje de las baterías 1, 2, 3,...; y 49 es el número de dato de la energía de las celdas solares). El PIC buscará cada tres direcciones el número requerido y cuando lo encuentre mandará al PC el número de dato y los dos bytes de datos. A continuación el PC enviará otro número de dato requerido. Si sólo se requiere un dato el PC enviará siempre el mismo dato. Cuando ya no queden más datos el PIC enviará una cadena ASCII "termine" para avisar que terminó, con lo cual el PC no enviará más solicitudes de datos y presentará los datos recibidos en una planilla Excel que será susceptible de guardar si se requiere.

Explicar el funcionamiento acabado del programa en Visual Basic resultaría extenso. El Anexo J contiene el código comentado de este programa.

En la figura 7.6 se muestran fotografías de la interfaz desarrollada para el computador.

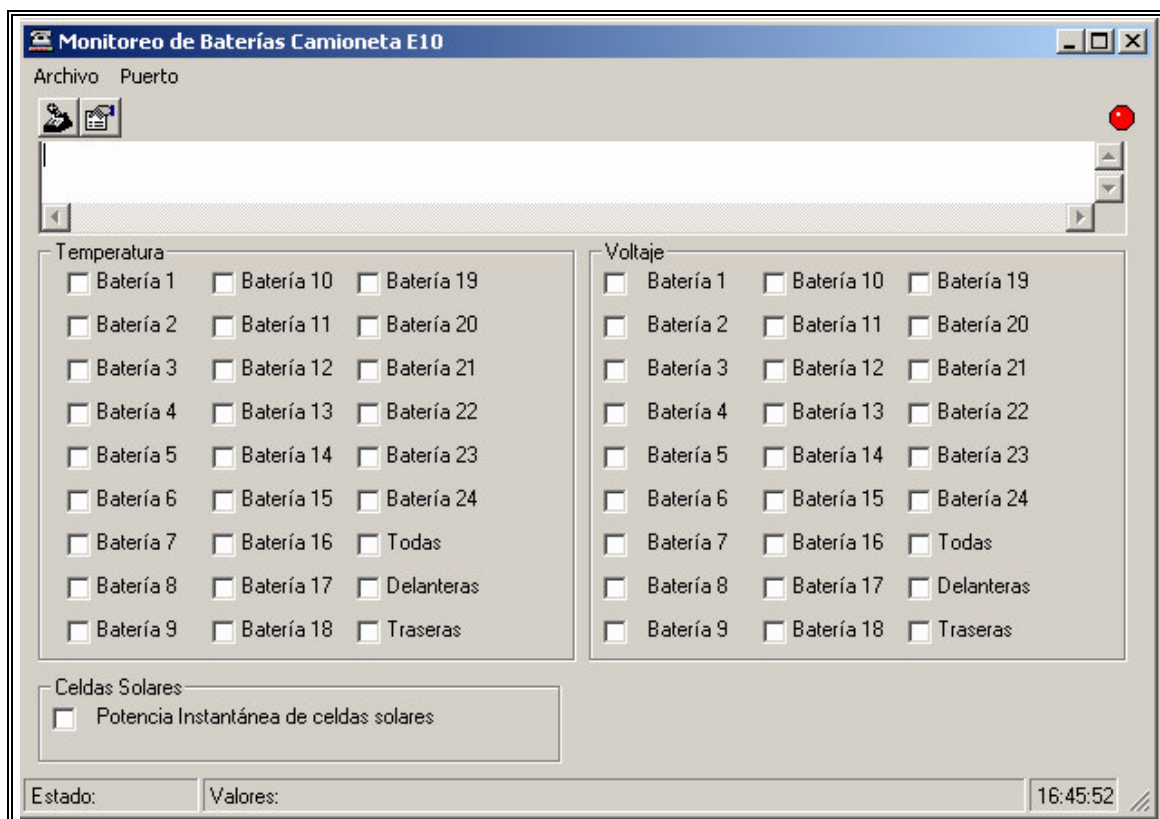


Figura 7.6: Apariencia de la Interfaz para recepción de datos desde el PC

Con respecto al programa, primero es necesario seleccionar qué datos se quieren recibir. Si no se ha seleccionado ninguno, aparece un cuadro de diálogo advirtiéndole que debe elegirse al menos un dato del monitoreo. Después se debe presionar el botón “Traspasar al Computador” en la pantalla en el menú “Guardar”. Luego se hace click en el ícono “Recibir datos”. Si aún no se ha presionado el botón “Traspasar al Computador” desde la pantalla aparecerá un cuadro de diálogo advirtiéndole la necesidad de hacerlo. La interfaz comenzará a mostrar los números de las baterías que está recibiendo en un cuadro de texto en la parte superior de la pantalla. Cuando termine, lo que ocurre cuando el PIC envíe una cadena “termine”, se cerrará el cuadro de diálogo y los datos estarán presentados en una planilla Excel, ya calibrados.

7.8 Adquisición de datos de las celdas solares

Las celdas solares son un proyecto que está desarrollando Alberto Zúñiga, que consiste en un arreglo de paneles solares que aportarán energía equivalente a 1kW. Para observar el comportamiento de la celda se dejó la puerta del conversor A/D 1 (puerta I/O A1). La señal de entrada debe tener una impedancia máxima de entrada de 10 k Ω y un nivel de voltaje de 0 a 5 V. Una lectura de 5V equivaldría a una medición de 1024 W. Este conversor no está protegido, por lo que el proyecto de las celdas debe interponer un buffer de protección entre el sistema que entrega la medición y el conversor A/D, de manera de no quemar el PIC si se pide un exceso de corriente.

La medición es realizada en el programa por la función `adc(incr)`. El parámetro `incr` será cero si se pide que se actualicen los relojes de la pantalla “Celdas” sin tener que hacer una conversión. En el caso contrario se hará una conversión, la cual es de 10 bits, y se calculará la actualización de los relojes así como el cálculo de la energía entrante.

Cada cinco segundos se recalculan los relojes y la energía desde la función `wait()`. La energía se calcula integrando la potencia recibida por el tiempo. Se supone que durante los cinco segundos el valor de potencia permanece constante.

La hora se actualiza cada un minuto y es un tiempo aproximado.

En el menu “Celdas” se muestra la energía, la potencia y la hora desde que comenzó la integración de la potencia. Si se quiere comenzar una nueva integración se presiona el botón “Reiniciar tiempo” desde el touch screen de la pantalla.

7.9 Cuadro resumen de las funciones más importantes del programa del microcontrolador

A continuación se presenta un cuadro resumen con todas las funciones del programa del PIC y su relación con las pantallas.

Tabla 7.2: Cuadro resumen con funciones del programa monitoreo del PIC

Nombre de la función	Descripción	Dispositivo que maneja
Void a_pantalla(char b);	Envía byte <i>b</i> a la pantalla	Pantalla
Void a_OW(char a);	Envía byte <i>a</i> al bus driver One Wire	One Wire
Char de_OW();	Recibe un byte del bus driver One Wire	One Wire
Void marcarID1(int bat_n)	Marca número ID del sensor correspondiente a labatería <i>bat_n</i> (1 a 24)	One Wire Eeprom PIC
Void datos_a_pantalla(float dat_comp, int pos, int vot)	Dibuja en la pantalla las barras de voltaje o temperatura (uno u otro depende de bit <i>vot</i>), en la posición según el número de batería dado por <i>pos</i> . El dato corresponde a <i>datcomp</i> . Selecciona color de barra y escribe valor en la pantalla	Pantalla
Void temp(int bateria)	Obtiene valor de temperatura de batería <i>bateria</i>	One Wire Memoria Flash Ext
Void volt(int bateria)	Idem <i>temp()</i> para voltaje	One Wire Memoria Flash Externa
Void adc(short int incr)	Si <i>incr=1</i> hace una conversión y recalcula valores del reloj de los paneles solares. Si no, sólo actualiza el reloj	Paneles Solares
Void serial_isr()	Rutina de interrupción externa. Guarda el dato que llegó desde el touch screen en <i>buffer[]</i> en el lugar que corresponde	Touch Screen
Void ciclo(long int)	Incrementa clock de los contadores de direccionamiento de la memoria <i>i</i> veces.	Memoria

i, short int j)	Si $j=1$, resetea contadores. Si no, sólo los incrementa	Flash Externa
Void esperar_touch()	Observa si se están recibiendo datos desde el touch. Si es así, espera hasta que se reciban los once caracteres.	Touch Screen Coordinación de todo el sistema
Void mostrar_menu()	Observa si han llegado datos desde el touch screen y hace los cambios de flags y menús en ese caso	Touch Screen Pantalla
Void Wait(byte ms)	Espera ms milisegundos mientras observa si han llegado datos desde el touch screen. Actualiza los clocks del sistema. También controla el flujo hacia la pantalla	Touch Screen Celdas Solares Pantalla
Void ctrl_mem(int select)	Si $select = "w"$ setea la memoria flash externa en modo escritura. Si $select = "r"$ la setea en modo lectura. Si $select = "o"$ apaga las salidas de la memoria	Memoria Flash Externa
Void a_memoria(byte dat)	Graba dat en la dirección siguiente disponible en la memoria externa	Memoria Flash Externa
Void a_PC(byte a)	Envía a al PC	PC
Byte de_memoria()	Lee un byte en la dirección siguiente de la memoria flash externa	Memoria Flash Externa
Void enviar_datos()	Envía datos a PC según protocolo establecido	PC
Short int datook()	Devuelve 1 si contenido de $buffer[]$ es válido y 0 en caso contrario	Touch Screen
Void incrementar_direccion(char a)	Si $a="r"$ incrementa puntero lectura de la memoria externa. Si $a="w"$ incrementa puntero de escritura de la memoria externa	Memoria Flash Externa
Void fondo(byte color)	Cambia el fondo de la pantalla según $color$	Pantalla

Void textstring()	Envia cadena <i>texstring</i> a la pantalla	Pantalla
Void rectangle()	Envia cadena <i>rectangle</i> a la pantalla	Pantalla
Void grabar_punteros()	Graba en la Eeprom del PIC en las posiciones FDh, FEh y FFh los punteros de escritura de la memoria flash	Memoria Flash Externa Eeprom del PIC
Void recuperar_punteros()	Recupera puntero de escritura de la memoria externa desde la memoria Eeprom del PIC cuando se inicia el monitoreo	Eeprom del PIC Memoria Flash Externa
Void dos (long int y1, long int y2)	Envía dos números, <i>y1</i> e <i>y2</i> separados con espacios a la pantalla	Pantalla
Void tres(long int z1, long int z2, long int z3)	Idem <i>dos(y1,y2)</i> , pero con <i>z1</i> , <i>z2</i> y <i>z3</i> . Al final pone un “;”	Pantalla
Void vee(short int updown)	Si <i>updown</i> =0 envía comandos a la pantalla para bajar contraste en 10 niveles. Si <i>updown</i> =1 sube el contraste 10 niveles	Pantalla
Float calib(int bateria, short int elegir)	Entrega los datos guardados en la eeprom del PIV de pendiente (elegir=0) u offset (elegir=1) de calibración	Eeprom del PIC

7.10 Construcción, chasis y ubicación final de la tarjeta controladora

El esquema final de la tarjeta controladora, con todas sus partes se muestra en la figura 7.7. En la figura 7.8 se muestran fotografías de la placa PCB final de la tarjeta controladora.

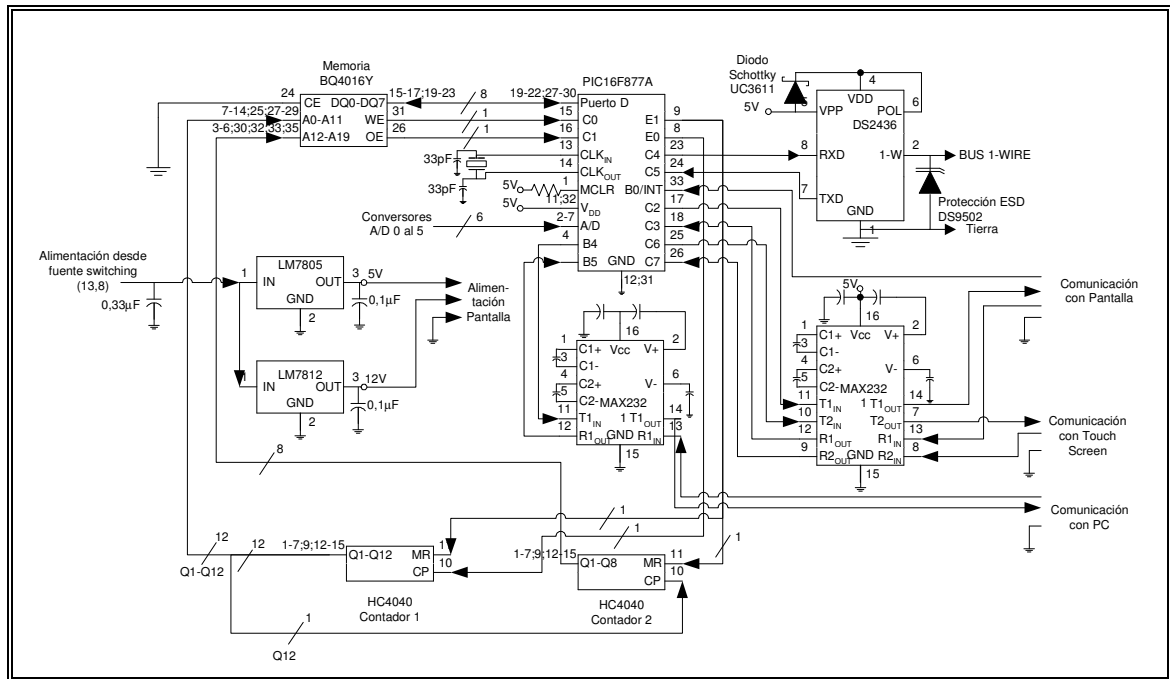


Figura 7.7: Esquema del diseño final de la tarjeta controladora

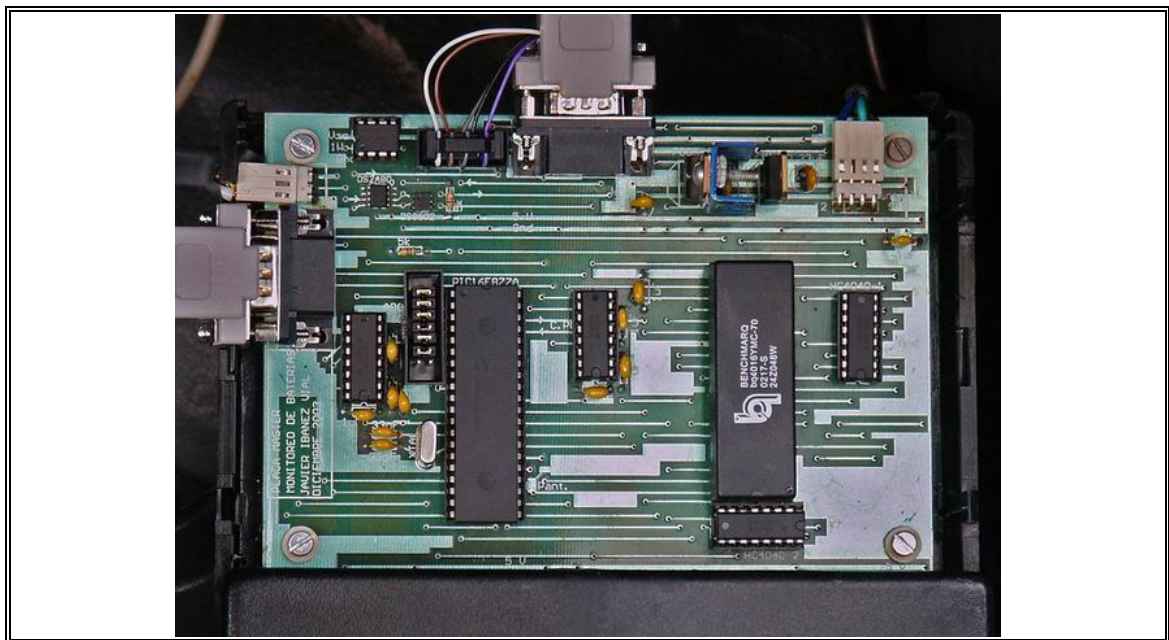


Figura 7.8 a) Fotografía de la tarjeta controladora final.

8. RESULTADOS DE LA IMPLEMENTACIÓN Y CONCLUSIONES

Este trabajo se hizo con la idea de aportar a la investigación que se hace en el laboratorio de tracción eléctrica de la Universidad Católica. Debido a esto se puso esfuerzo en que el sistema fuera de larga duración. Pese a la cantidad de trabajo que demandó la realización de cada una de las tarjetitas de sensado, su posterior embarrilamiento con resina, la confección de la tarjeta controladora y todo el aprendizaje que implica enfrentarse a nuevas tecnologías como la pantalla LCD, sensores digitales One Wire, el manejo de la memoria Flash y el touch screen, el monitoreo está funcionando correctamente.

Durante las pruebas realizadas con el sistema de monitoreo instaladas en el vehículo, los sensores se equivocan en muy pocas ocasiones. Sí lo hacen y sería óptimo que no lo hicieran.

A la vez, la calibración de los sensores podría haber sido más precisa.

El autor de esta memoria está satisfecho por lo aprendido en este trabajo, ya que lo que más se valoró fue el enfrentarse con un problema práctico de cierta envergadura, en que no sólo existe la complicación de saber soldar componentes de montaje superficial, sino en la elección correcta de los dispositivos a utilizar.

Como todo proyecto de ingeniería el monitoreo acepta múltiples mejoras: podría hacerse más eficiente el programa del microcontrolador PIC, se podría mejorar el apantallamiento de los cables del bus, se podría mejorar la interfaz que recibe los datos en el computador, etc.

La precisión de los sensores cumple con lo necesario, la presentación de los datos es muy cómoda y buena, la grabación de los datos es fiel.

El autor de esta memoria quiere destacar la dificultad en hacer un trabajo tan práctico en que una mala conexión puede significar meses de retraso en el trabajo y, por lo demás, un gasto importante.

En conclusión, el trabajo cumple con las expectativas satisfactoriamente.

BIBLIOGRAFIA

- [Teje01] TEJEDA BERTENS RAUL **Diseño e Implementación de Instrumentación Digital Para Vehículo Eléctrico**. Memoria para optar al título de Ingeniero Civil Electricista, Santiago de Chile, 2001.
- [Kies94] KIESSLING, R., **A battery model for monitoring of and corrective action on lead-acid EV batteries**. Battery Conference on Applications and Advances, Proceedings of the Ninth Annual Meeting, Long Beach, CA, USA January 11-13, 1994.
- [Texa96] TEXAS INSTRUMENTS **Printed_Circuit_Board Layout for Improved Electromagnetic Compatibility**. October 1996, Application Report, Texas Instrument literature number SDYA011.
- [Benc96] BENCHMARQ **BQ4016Y 1024Kx8 Nonvolatile SRAM** September 1996, Benchmarq Data Sheet.
- [Soli03] SOLID STATE OPTRONICS **SLC800 Linear Optocoupler In An Isolation Amplifier Circuit**. April 16, 2003, Solid State Optronics Application Note
- [Dall03A] DALLAS SEMICONDUCTOR **MicroLAN Design Guide**. September 26, 2003, Dallas Semiconductor Tech Brief.
- [Dall03B] DALLAS SEMICONDUCTOR **Using the DS2480B Serial 1-Wire Line Driver**. March 31, 2003, Dallas Semiconductor Application Note
- [Dall03C] DALLAS SEMICONDUCTOR **DS2480B Serial 1-Wire Line Driver**. July 1, 2003, Dallas Semiconductor Data sheet
- [Dall02A] DALLAS SEMICONDUCTOR **DS2436 Battery ID/Monitor Chip**. June 8, 2002, Dallas Semiconductor Data sheet
- [BASF01] BASF CHILE S.A. **Palatal P4**. Marzo de 2001. Información Técnica BASF Chile S.A.

A N E X O S

ANEXO A: ESPECIFICACIONES PIC 16F877A

ANEXO B: PANTALLA APOLLO DISPLAY TECHNOLOGIES CDS1375

ANEXO C: ESPECIFICACIONES MEMORIA FLASH BQ4016Y

ANEXO D: ESPECIFICACIONES SENSOR DE TEMPERATURA Y VOLTAJE DS2436

ANEXO E: ESPECIFICACIONES OPTOCUPLA LINEAL SOLID STATE OPTRONICS SLC800

ANEXO F: ESPECIFICACIONES DRIVER BUS ONE WIRE

ANEXO G: ESPECIFICACIONES CAMIONETA CHEVROLET E10

ANEXO H: CARACTERÍSTICAS ELÉCTRICAS DE LAS BATERÍAS

ANEXO I: PROGRAMA DESARROLLADO PARA EL MICROCOMPUTADOR

ANEXO J: PROGRAMA DESARROLLADO PARA LA RECEPCIÓN DE DATOS DE UN COMPUTADOR A TRAVÉS DE LA PUERTA SERIAL