



JIDeTEV

Jornadas de Investigación y Desarrollo Tecnológico
Extensión, Vinculación y Muestra de la Producción



JIDeTEV- Año 2022 -ISSN 2591-4219

Sistema electrónico para red de sensores utilizados en microrredes eléctricas

Gian Madera Nigra ^{a*}, Martin G. Fleita ^a, Fabian H. Álvarez ^a, Juan P. Gross ^b, Fernando Botterón ^b
Guillermo A. Fernández ^b

^a Universidad Nacional de Misiones (UNaM). Facultad de Ingeniería. Misiones, Argentina.

^b Universidad Nacional de Misiones (UNaM)-CONICET. Facultad de Ingeniería. Instituto de Materiales de Misiones (IMAM). Grupo de Investigación y Desarrollo en Electrónica (GIDE). Misiones, Argentina.

giannigra97@gmail.com, MartinFleita4@gmail.com, fabiandroid7@gmail.com, gross@fio.unam.edu.ar,
botteron@fio.unam.edu.ar, guillermo.fernandez@fio.unam.edu.ar

Resumen

Las microrredes son sistemas eléctricos que permiten aprovechar distintas fuentes de energía renovable locales y abastecer con energía eléctrica a diferentes cargas. Para esto, las microrredes deben contar con un sistema de supervisión que gestione sus elementos (fuentes de energía, sistemas de almacenamiento y cargas) ejecutando acciones de control en base a los estados de operación de los mismos, obtenidos estos con mediciones de sensores distribuidos en el sistema. A partir de esto, el presente trabajo muestra el desarrollo de un sistema electrónico que intercomunica de manera efectiva a una unidad maestra con unidades remotas de sensado, utilizando un protocolo de comunicación digital. Las unidades remotas están diseñadas para conectar distintos sensores, procesar su señal y comunicar las variables medidas a demanda de la unidad maestra. Esta última posee un display y pulsadores, a partir de los cuales puede accederse a los valores de las variables sensadas a la distancia. Inicialmente, en el desarrollo del trabajo se seleccionó el protocolo de comunicación, luego se diseñaron y construyeron los circuitos de las unidades mencionadas, como así también fueron desarrollados los programas correspondientes. Se obtuvo una placa maestra con una interfaz de usuario que permite acceder a un menú con diferentes opciones para visualizar las lecturas de las unidades remotas de sensado y controlar las mismas. Entre las conclusiones obtenidas con este trabajo se destaca que, gracias al protocolo de comunicación utilizado, el sistema obtenido resulta versátil, sencillo y robusto. El mismo permite incorporar varios sensores y otras unidades con sólo modificar algunas partes del programa, la comunicación es implementada a través de dos hilos y con características de alta inmunidad al ruido.

Palabras Clave – Microcontroladores, protocolo de comunicación, sensores, menú, comunicación.

1. Introducción

Las microrredes son pequeños sistemas eléctricos en los cuales distintas fuentes de energía y dispositivos de almacenamiento interactúan para suministrar energía eléctrica a un conjunto de cargas [1]. Fuentes de energía renovable como la solar y la eólica, son ampliamente utilizadas en las microrredes. En estos sistemas eléctricos, recursos energéticos como los mencionados, se aprovechan para producir energía eléctrica y abastecer localmente a las cargas que lo requieran. Por este motivo, el uso de las microrredes, no solo tiene la ventaja de aprovechar los recursos energéticos locales, sino que también el hecho de usar la energía eléctrica en el lugar donde se produce, contribuye a reducir las pérdidas y aliviar el uso de la red de distribución eléctrica principal [2].

Las fuentes de energía renovable mencionadas, si bien siempre están presentes, la cantidad en que pueden aprovecharse es intermitente ya que su disponibilidad depende de la época del año y de las condiciones del clima. Atendiendo a esto, para mantener la provisión de energía eléctrica a las cargas, las microrredes incorporan dispositivos de almacenamiento, los cuales entregan su energía almacenada en los momentos en que las fuentes no están disponibles o bien no son suficientes para la demanda de las cargas. Por otra parte, las microrredes también pueden conectarse a la red de distribución eléctrica para intercambiar energía con la misma. En caso de que las cargas no pueden ser abastecidas por las fuentes y dispositivos de almacenamiento disponibles, la microrred importa energía eléctrica de la red; mientras que, si existe un excedente de energía en la microrred, la misma puede inyectar energía eléctrica a la red y con esto obtener algún beneficio económico [3].

Las microrredes cuentan con un sistema de supervisión que gestiona la operación de sus fuentes de energía, dispositivos de almacenamiento y la interconexión a la red de distribución (si existe). El sistema de supervisión mencionado mide variables tales como: corrientes, tensiones, estado de carga de las baterías, temperaturas, radiación solar, velocidad del viento, entre otras, y a partir de sus valores toma decisiones y ejecuta acciones de control sobre el sistema con el fin de obtener la operación adecuada y eficiente de los componentes de la microrred, manteniendo la provisión de energía a las cargas que posee la misma [4].

El funcionamiento del sistema de supervisión mencionado, es posible gracias a las mediciones obtenidas con una red de sensores distribuidos en la instalación. Atendiendo esto, el presente trabajo propone un sistema embebido basado en microcontroladores, que permite la comunicación entre una unidad central y unidades remotas de sensado, para acceder a las variables requeridas por el sistema de supervisión. La unidad maestra, desarrollada en base al microcontrolador ATMEGA 2560, comanda dos unidades remotas de sensado que están construidas con el microcontrolador ATMEGA 328P. La intercomunicación entre estos dos tipos de unidades es realizada mediante el protocolo de comunicación serie denominado CAN (*Controller Area Network*). Las características principales de este protocolo de comunicación de datos digitales, es que posee una topología del tipo bus, utilizando solo dos hilos para establecer las comunicaciones entre los dispositivos que comparten el bus; además, cada dispositivo posee un hardware (controlador) que maneja el intercambio de los mensajes a través del bus con prevención de colisiones y corrección de errores. Otra característica del protocolo CAN es que los mensajes transmitidos utilizan identificadores (ID) basados en el contenido del mensaje en lugar de un ID de dispositivo, lo cual facilita a que los controladores de los dispositivos conectados al bus puedan acceder y filtrar los mensajes que son de interés para el mismo [5].

En el sistema propuesto, las unidades remotas poseen un sensor de temperatura LM35 y tres entradas discretas (del tipo on/off), cuyos estados pueden ser reportados a la unidad maestra. Las unidades remotas poseen un relé que puede activarse cuando la temperatura medida con el sensor indicado, supera un valor determinado el cual es configurable a distancia. Estas unidades también poseen tres salidas digitales de propósito general, que pueden ser comandadas en forma remota.

La unidad maestra desarrollada para el sistema propuesto, demanda y recibe el resultado de las mediciones, controla las salidas digitales y el relé de las unidades remotas, también cuenta con una interfaz de usuario conformada por un display de cristal líquido y un conjunto de cinco pulsadores con el fin de que el usuario pueda interactuar de forma sencilla con todo el sistema.

El trabajo aquí presentado, si bien tiene por objetivo intercomunicar una serie de sensores remotos con una unidad maestra, su desarrollo en el marco de la asignatura Técnicas Digitales 2 de la carrera Ingeniería Electrónica, está orientado a la integración de los saberes relacionados con el diseño e implementación de sistemas embebidos basados en microcontroladores que son abordados en esta asignatura. Este trabajo también es parte de las actividades de adscripción al proyecto de investigación “Bombeo de agua con energías renovables, almacenamiento de energía y conexión a la red para pequeñas huertas rurales comunitarias: Estudio, diseño y puesta en funcionamiento (16/I1083-PDTS)”, donde se pretende implementar una microrred para abastecer con energía eléctrica a la bomba y otras cargas en una escuela rural de nuestra provincia [6]. En el marco del proyecto mencionado, el sistema obtenido en este trabajo tiene por objetivo constituir un punto de partida para el desarrollo del sistema de supervisión requerido en la microrred mencionada.

A continuación, en el desarrollo de este artículo, inicialmente es presentada la etapa de diseño del sistema propuesto, donde se explican cuestiones relacionadas con protocolo de comunicación utilizado, el circuito y el programa de cada parte que compone al sistema. Posteriormente se muestran los resultados, donde es explicada la operación del menú de funciones que posee el sistema, como así también se presentan las placas desarrolladas para este. Finalmente se expresan las conclusiones obtenidas con el desarrollo del trabajo.

2. Diseño del sistema propuesto

A continuación, se describen cuestiones relacionadas a la selección del protocolo de comunicación y el diseño de los circuitos correspondientes a la unidad maestra y las unidades remotas que conforman el sistema electrónico para la red de sensores utilizados en la microrred.

2.1 Protocolo de comunicación

La microrred correspondiente al proyecto de investigación mencionado está constituida por varios elementos de potencia conectados entre sí (paneles fotovoltaicos, baterías, convertidores electrónicos de potencia y distintas cargas), haciendo esto que su sistema de supervisión deba contar con una importante cantidad de sensores y otras unidades que complejizan el conexionado de los elementos de la instalación. Por otra parte, los convertidores electrónicos de potencia (convertidores CC-CC e inversores) presentes en la microrred, operan con llaves electrónicas que conmutan a altas frecuencias pudiendo generar ruido y causar interferencias en circuitos cercanos, tales como aquellos dedicados a medir las variables requeridas por el sistema de supervisión de la microrred.

Atendiendo a estas cuestiones, en el presente trabajo se selecciona el protocolo CAN para establecer la comunicación entre las diferentes unidades del sistema desarrollado. Este protocolo facilita la conexión de los dispositivos que intervienen en la comunicación, ya que posee una topología tipo bus conformada sólo por dos líneas (CAN-H y CAN-L). Por otra parte, en el protocolo CAN la información digital transmitida se extrae del bus por diferencia de voltaje entre las líneas, reduciéndose así significativamente el ruido en modo común y haciendo que la comunicación adquiera una alta inmunidad al ruido [7], siendo esto necesario en la transmisión de datos dentro de las microrredes.

Por otro lado, a contar con un controlador dedicado, este se encarga de administrar la comunicación por orden de prioridad y filtrar los mensajes interrumpiendo el flujo de trabajo del microcontrolador correspondiente únicamente cuando la trama es de interés para el mismo.

En el protocolo CAN los dispositivos se comunican enviando paquetes de bits denominados tramas de datos, conformadas por distintos campos (grupos de bits) que cumplen con diferentes funciones dentro del proceso de comunicación. Uno de estos campos es el Identificador (ID). El mismo es utilizado en cada dispositivo que recibe la trama para identificar el tipo de trama o bien para identificar al dispositivo que envió la trama de datos. En este protocolo, cada dispositivo conectado al bus posee un controlador CAN dedicado que selecciona las tramas de interés reconociendo su ID. Para efectuar esto, el controlador filtra cada trama de datos recibida y genera una señal de interrupción en el caso de reconocer una que posee el ID deseado. De esta forma, cada vez que se recibe una trama con el ID deseado, el controlador CAN puede interrumpir la operación de otro dispositivo al que está conectado (por ejemplo, un microcontrolador) y es encargado de procesar la información de la trama recibida. Esto resulta en otra ventaja del protocolo CAN, ya que reduce la latencia en la comunicación e incrementa la velocidad de intercambio de datos entre los dispositivos que se comunican.

2.2 Diseño del prototipo de la Unidad Maestra (UM)

2.2.1 Circuito de la UM

Debido a las numerosas librerías de software que posee de soporte y a la disponibilidad, el desarrollo de la unidad maestra está basado en la placa Arduino MEGA que posee el microcontrolador de 8 bits ATMEGA 2560 [8]. La Fig. 1 muestra el esquema del circuito diseñado para esta unidad.

El circuito mencionado posee una interfaz de usuario compuesta de un display LCD de 16 x 2 (dos filas de dieciséis caracteres) y cinco pulsadores. El display permite visualizar el menú de funciones que ofrece el programa desarrollado para esta unidad y también las lecturas obtenidas de las unidades remotas. Los pulsadores posibilitan el desplazamiento entre las diferentes opciones del menú mencionado, y cuentan con una disposición en cruz para implementar las acciones: arriba, abajo, izquierda/-, derecha/+ y seleccionar. El circuito también incorpora tres leds indicadores para observar el estado de comunicación con las unidades remotas. El programa desarrollado para esta unidad enciende estos leds, considerando si el envío y recepción de datos desde las unidades remotas fue o no exitoso.

En la Fig. 1 la placa Arduino MEGA debe conectarse al circuito a través de una tira de pines hembra indicada con H3, donde las líneas GND, VIN, SCK, MOSI, MISO, SDA y SCL tienen que unirse a los pines con el mismo nombre que están predefinidos en la mencionada placa. En cambio, la línea INT de H3 tiene que unirse al pin 2 (PE4/INT4), CS al pin 53 (PB0/SS) y los pulsadores P-ABJ a P-ARR desde el pin 22 (PA0/AD0) al pin 26 (PA4/AD4), en ese orden.

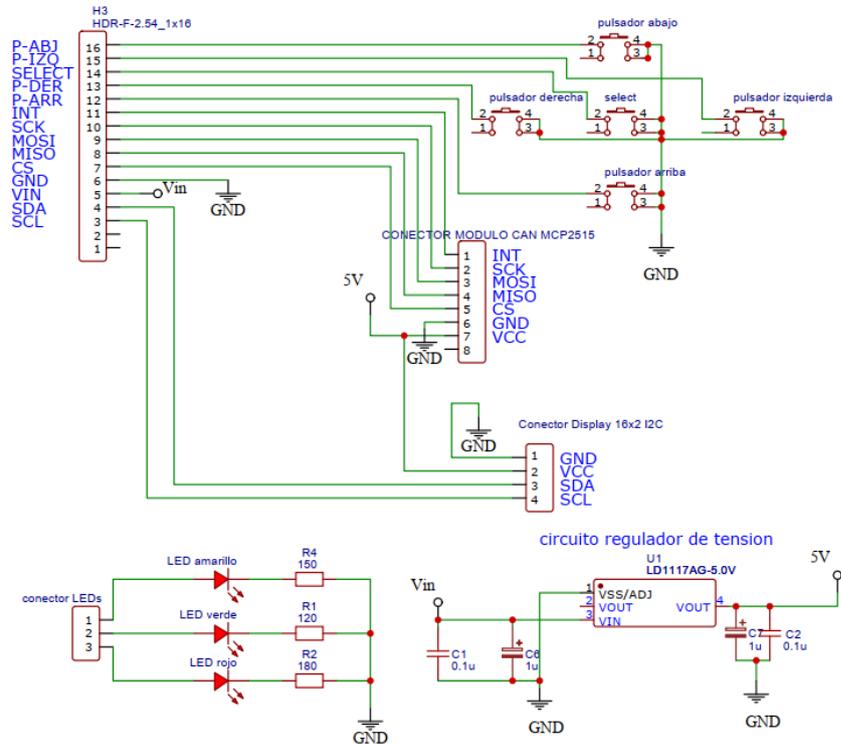


Fig. 1. Circuito del prototipo de la Unidad Maestra.

Los leds indicadores de la Fig. 1 deben conectarse de la siguiente forma a la placa del Arduino MEGA: el LED amarillo al pin 3 (PE5/INT5), el LED verde al pin 4 (PG5/OC0B) y el LED rojo al pin 5 (PE3/OC3A). Las resistencias limitadoras de estos leds son dimensionadas con las ecuaciones (1).

$$R_{LED} = \frac{V_{OH\ MEGA} - V_{DLED}}{I_{LED}} \quad ; \quad P_{R_{LED}} = (I_{LED})^2 \cdot R_{LED} \quad (1)$$

siendo la mínima tensión en estado alto entregada por el microcontrolador $V_{OH\ MEGA} = 4,2\text{ V}$, la caída de tensión en los leds $V_{DLEDa} = 2,2\text{ V}$, $V_{DLEDv} = 1,9\text{ V}$, $V_{DLEDr} = 2\text{ V}$, para una corriente $I_{LED} = 15\text{ mA}$ en cada led, se obtiene los siguientes valores comerciales para las resistencias limitadoras: $R_a = 150\ \Omega$, $R_v = 120\ \Omega$ y $R_r = 180\ \Omega$, todas de $1/4\text{ W}$ de potencia.

Como puede observarse en la Fig. 1, el circuito de la unidad maestra incorpora un regulador de voltaje LD1117AG-5.0, el mismo utiliza la tensión que energiza a la placa Arduino MEGA ($7\text{ V} \leq V_{in} \leq 12\text{ V}$) para proporcionar los 5 V requeridos en la alimentación del display LCD y el módulo controlador CAN utilizado en la comunicación con las unidades remotas. El display es acoplado al circuito de la unidad maestra mediante un conector que vincula al mismo con el microcontrolador ATMEGA 2560 del Arduino®. Estos dispositivos establecen comunicación a través del estándar I²C (*Inter-Integrated Circuit*, líneas SDA y SCL), dado que el display posee un módulo que implementa este tipo de comunicación serie.



Fig. 2. Módulo para comunicación con protocolo CAN.

La comunicación con el bus CAN entre la unidad maestra y las unidades remotas, es implementada utilizando el módulo comercial de la Fig. 2 en cada una de estas unidades. El mismo incorpora un controlador CAN (MCP2515) un *transceiver* (TJA1050). Este módulo permite implementar el protocolo CAN con un tipo de comunicación *half-duplex*, a una velocidad máxima de 10 Mbit/s (alcance < 12 m) y una longitud máxima de cable de 1200 m (con velocidad de hasta 100 kbit/s) [9].

El circuito integrado TJA1050 que posee el módulo de la Fig. 2, es una interfaz adaptadora de niveles de tensión entre el controlador CAN (MCP2515) y los dos hilos (CAN-H y CAN-L) que conforman el bus CAN [10]. Este *transceiver* tiene la capacidad de transmitir y recibir datos manejando la información a través de tensiones diferenciales. El módulo de la figura mencionada, también incorpora el circuito integrado MCP2515, que es un controlador CAN diseñado para la implementación del protocolo CAN (versión 2.0B). Este circuito integrado incorpora en su hardware dos máscaras y seis filtros de aceptación, utilizados para filtrar mensajes no deseados, permitiendo esto reducir la carga por procesamiento de datos al microcontrolador que interactúa con el módulo [11]. Cabe mencionar que, tanto en la unidad maestra como en las remotas, esta interacción es realizada mediante el estándar de comunicación serie SPI (*Serial Peripheral Interface*, líneas SCK, MOSI, MISO y CS).

2.2.2 Programa de la UM

El programa que ejecuta la unidad maestra fue desarrollado en lenguaje C, utilizando el Entorno de Desarrollo Integrado (IDE) para las placas Arduino®. Este entorno permite integrar fácilmente diferentes librerías (como las usadas para el manejo del protocolo CAN) y también realizar la depuración del programa mediante el monitor serie que dispone. La Fig. 3 muestra el diagrama de flujo simplificado del programa realizado para la unidad maestra.

En la Fig. 3a puede notarse que el programa inicia con el menú principal, donde puede accederse a las lecturas y configuraciones de las diferentes unidades remotas.

Cada vez que el usuario accede a uno de los submenús LECTURAS UR-N el programa sigue el diagrama de la Fig. 3b. Para observar el estado de la unidad remota correspondiente en el display de la unidad maestra, son activadas las interrupciones por desbordamiento del temporizador Timer 1 en el microcontrolador ATMEGA 2560 [8]. En cada interrupción la unidad maestra solicita información a la unidad remota correspondiente y cuando la recibe, actualiza el display con esta información.

En el submenú de configuración mostrado en la Fig. 3c, mediante los pulsadores de la unidad maestra alternarse los estados entre 1/0 de las diferentes salidas digitales que poseen las unidades remotas. También puede elegirse la opción para configurar el relé, el cual tiene dos modos de operación; el manual en donde el estado del relé se modifica de igual manera que las salidas digitales y el modo automático en donde se setea un valor de temperatura, el cual indica que a temperaturas igual o mayores el relé debe encenderse.

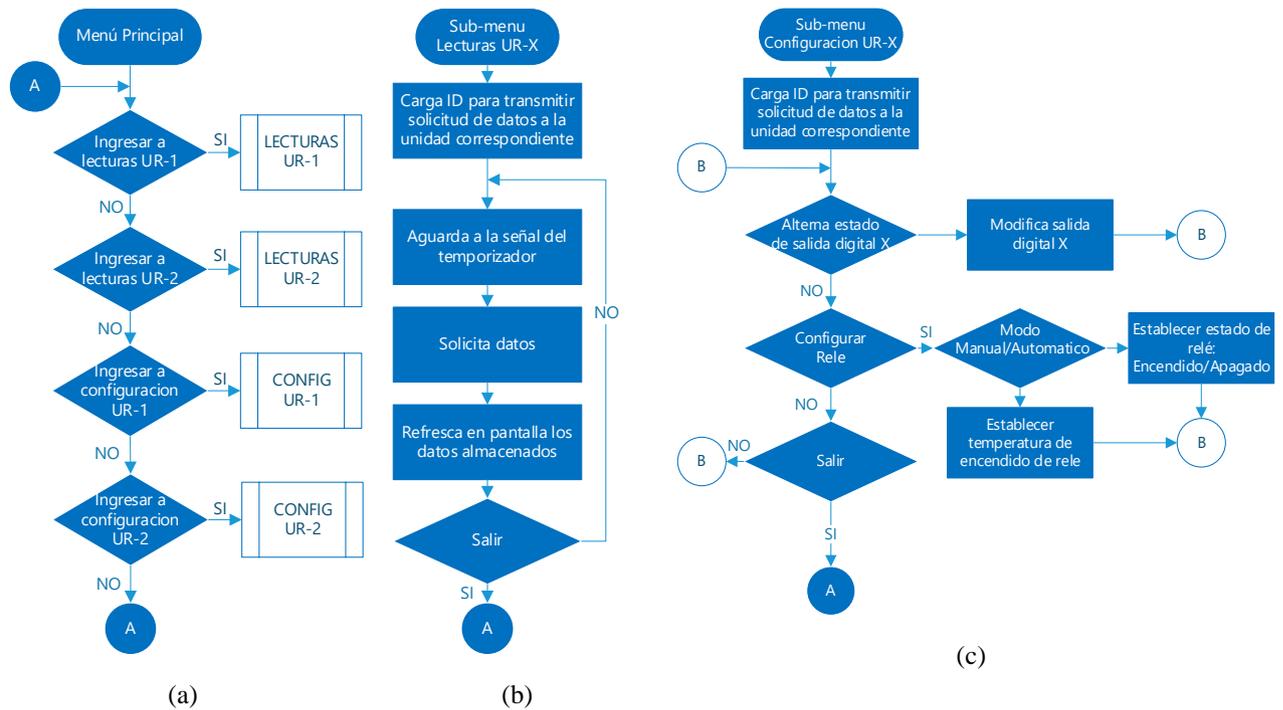


Fig. 3. Diagrama de flujo del programa ejecutado en la unidad maestra: (a) Rutina principal; (b) Submenú para leer las unidades remotas; (c) Submenú para configurar las unidades remotas.

En el programa desarrollado, el temporizador Timer 1 es configurado para operar en modo *free-running* y las interrupciones mencionadas son producidas con un intervalo de tiempo $Temp$, el cual puede calcularse a través de la ecuación (2), donde $VPC1$ es el valor cargado en el registro TCNT1 del temporizador con cada interrupción.

$$Temp = \left[(2^{16} - 1) - VPC1 \right] \cdot N / f_{osc} \quad (2)$$

Para que la solicitud de información a la unidad remota correspondiente cuando se accede a alguno de los submenús LECTURAS UR-N se efectúe cada $Temp = 100$ ms, considerando que el *prescaler* del Timer 1 es configurado con $N = 64$ y la frecuencia del cristal oscilador que posee el microcontrolador es $f_{osc} = 16$ MHz, a partir de la ecuación (2) resulta $VPC1 = 40535$.

Por otra parte, en la configuración del Timer 1, el programa carga el registro TIMSK1 con 0x01 a los efectos de activar la interrupción por desbordamiento. Mientras que el registro TCCR1B se carga

con el valor 0x03 (únicamente cuando se requiere activar dicho temporizador) para seleccionar el *prescaler* correspondiente y la señal de reloj desde la CPU del microcontrolador [12].

2.3 Diseño de las Unidades Remotas (UR)

2.3.1 Circuito de las UR

El circuito de las unidades remotas cuenta con tres entradas/salidas digitales, un sensor de temperatura LM35 y también posee una salida de potencia implementada mediante un relé. Todas las salidas cuentan con sus respectivos leds indicadores.

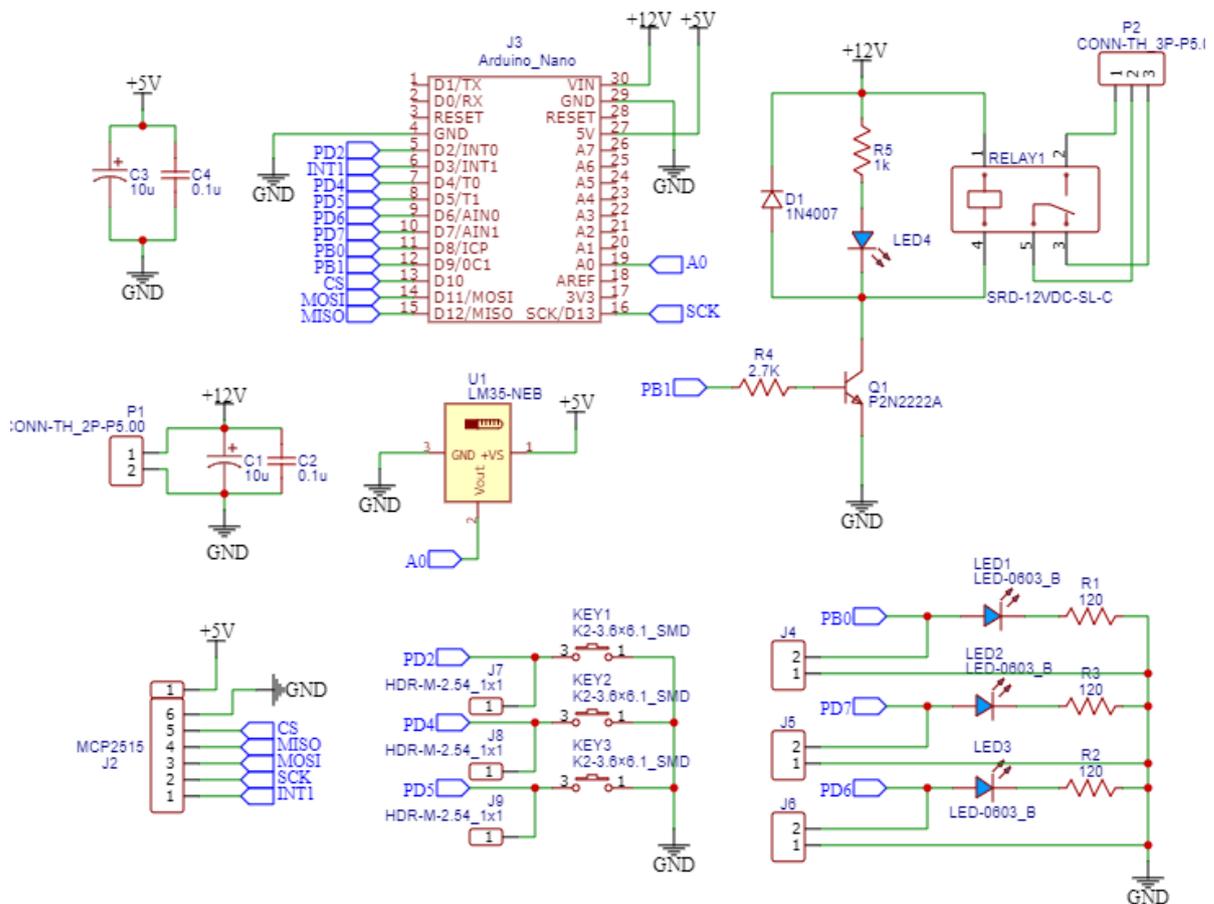


Fig. 4. Circuito del prototipo de las Unidades Remotas.

La Fig. 4 presenta el circuito que posee cada unidad remota construida. El mismo está basado en la placa Arduino NANO, desarrollada con un microcontrolador de 8 bits ATMEGA 328P [13]. Este circuito, al igual que en la unidad principal, utiliza el módulo comercial de la Fig. 2 para implementar la comunicación a través del bus CAN. Dicho módulo puede conectarse al circuito mediante la tira de pines macho J2 mostrada en la Fig. 4.

El circuito de las unidades remotas incorpora un sensor de temperatura LM35 que está conectado en su configuración básica, permitiendo medir temperaturas de 2 a 150 °C y proporcionado en su

salida una señal de tensión analógica con un factor de escala de 10 mV/°C [14]. Esta señal analógica es inyectada al pin A0 (PC0) de la placa Arduino NANO.

El circuito de cada unidad remota también posee tres pulsadores y tres leds rojos indicadores. Los pulsadores están conectados a los pines D2 (PD2), D4 (PD4) y D5 (PD5) de la placa mencionada y hacen uso de las resistencias *pull-up* internas que posee su microcontrolador. Estos pulsadores son usados para simular el estado de las entradas digitales en las unidades remotas. Los tres leds que posee cada unidad remota, son usados para indicar el estado de las salidas digitales, asociadas a los pines D8 (PB0), D7 (PD7) y D6 (PD6), los cuales pueden comandarse a distancia con la unidad maestra. Las resistencias limitadoras de corriente que tienen estos leds, son calculadas a través de las ecuaciones (1) obteniéndose para todas ellas 120 Ω y 1/4 W.

Las unidades remotas están provistas de un relé SRD-12VDC-SL-C que puede ser gobernado a distancia por la unidad maestra. Este dispositivo permite que el sistema pueda manejar cargas de corriente continua o de corriente alterna como ser motores, lámparas, entre otras. Como puede apreciarse en la Fig. 4, el relé está comandado por el pin D9 (PB1) del Arduino NANO. Esto se hace a través del transistor P2N2222A, que actúa como *driver* de corriente para el pin mencionado. Las resistencias limitadoras de corrientes asociadas al circuito de dicho transistor pueden dimensionarse con las ecuaciones (3) y (4).

$$R_{LED} = \frac{V_{CC} - V_{LED} - V_{CEsat}}{I_{LED}} ; P_{R_{LED}} = I_{LED}^2 R_{LED} \quad (3)$$

$$I_C = I_{LED} + I_R ; I_B = 10 \cdot (I_C / h_{FEmin}) ; R_B = \frac{V_{OH} - V_{BEsat}}{I_B} ; P_{R_B} = I_B^2 R_B \quad (4)$$

A partir de la hoja de datos del transistor mencionado, la caída de tensión colector-emisor de saturación es $V_{CEsat} = 0,3$ V, la caída de tensión base-emisor de encendido $V_{BEsat} = 1,2$ V y la ganancia de corriente $h_{FEmin} = 100$. Considerando que las corrientes en la bobina del relé es $I_R = 30$ mA y en el led indicador el $I_{LED} = 10$ mA, para $V_{CC} = 12$ V y una salida en estado lógico alto del microcontrolador $V_{OH} = 4,1$ V, los valores comerciales para las resistencias limitadoras obtenidas con las ecuaciones anteriores son $R_{LED} = 1$ kΩ y $R_B = 2,7$ kΩ, ambas de 1/4 W.

El sistema desarrollado posee dos unidades remotas que se comunican con la unidad maestra utilizando el protocolo de comunicación CAN. En esta comunicación, las unidades remotas reciben ordenes de la unidad maestra, que solicita información sobre las entradas digitales o la temperatura.

2.3.2 Programa de las UR

El programa ejecutado por cada unidad remota fue desarrollado en lenguaje C, utilizando el Visual Studio Code, que es un editor de código fuente gratuito y de código abierto que permite incorporar el IDE de las placas Arduino®. Esta herramienta fue utilizada debido a que presenta ventajas relacionadas con la escritura del código, como ser: soporte para depuración de diferentes dispositivos, resaltado de sintaxis, finalización inteligente de código, entre otros. La Fig. 5 muestra el diagrama de flujo simplificado de este programa, el cual gestiona las tramas de datos recibidos mediante el módulo MCP2515 y actúa en consecuencia de ello. Para realizar esta tarea, el programa configura los

respectivos puertos del microcontrolador como entradas y salidas, también configura su convertor analógico-digital (ADC) para procesar la señal de salida del sensor LM35 y el Timer 1 (de 16 bits) empleado en las temporizaciones utilizadas.

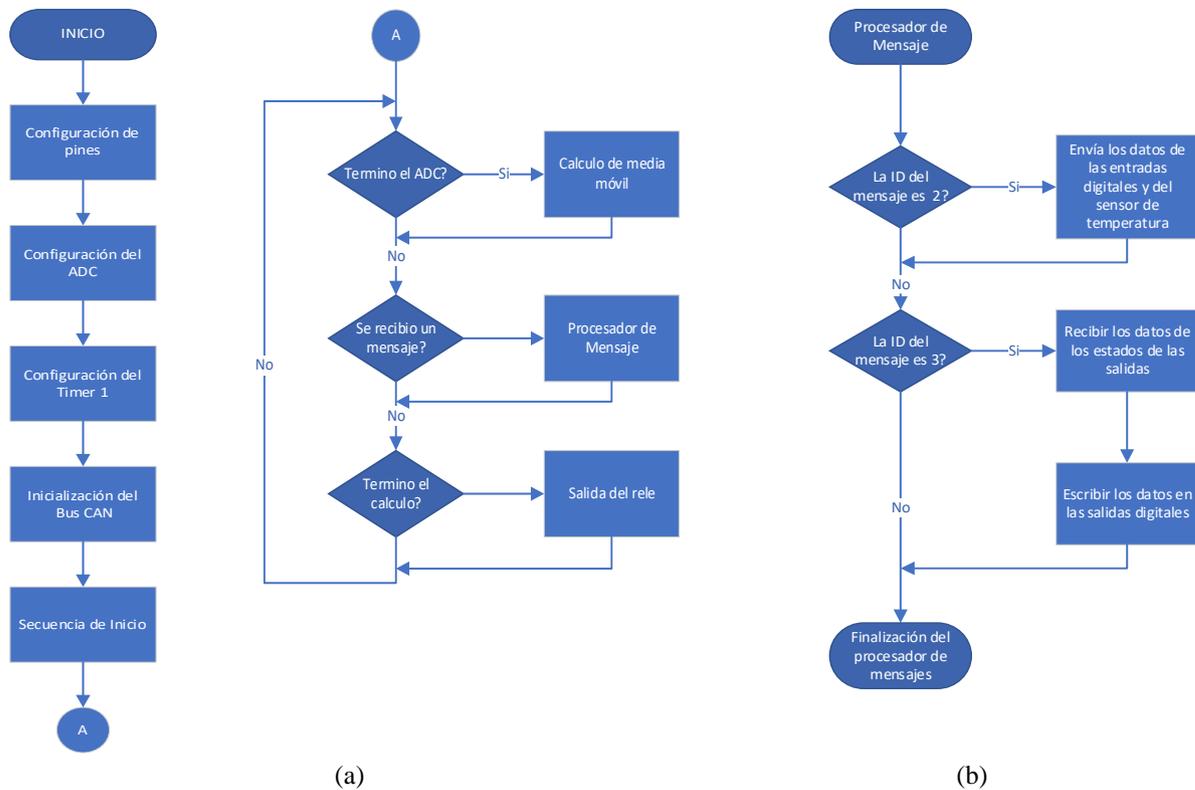


Fig. 5. Diagrama de flujo del programa ejecutado en las unidades remotas: (a) Rutina principal; (b) Rutina de servicio a la interrupción correspondiente a un nuevo mensaje coincidente.

El ADC opera en forma temporizada, para muestrear la señal de salida del sensor de temperatura. Para esto el Timer 1 es configurado en modo CTC, reseteándose cada un segundo. En este modo de operación, el contador del Timer 1 es borrado cuando alcanza el valor de tope cargado en el registro $OCR1A$ [12], el cual es obtenido con la ecuación (5).

$$Temp = (1 + OCR1A) \cdot N / f_{osc} \quad (5)$$

De esta forma, siendo el periodo de muestreo $Temp = 1$ s y el *prescaler* del Timer 1 configurado con $N = 256$ y considerando que la frecuencia del cristal oscilador que posee el microcontrolador es $f_{osc} = 16$ MHz, el valor de tope cargado en el registro $OCR1A$ es 62499.

Para que el ADC del microcontrolador funcione correctamente, la frecuencia de su reloj debe estar entre 50 kHz y 200 kHz [13]. Atendiendo a esto, el *prescaler* de la señal de reloj del ADC es configurado con $N_{ADC} = 128$. De esta forma, queda con $f_{clkADC} = 125$ kHz, haciendo que la conversión de cada muestra demore $T_{CONV} = 116$ μ s, ambos valores obtenidos con las ecuaciones (6).

$$f_{clkADC} = \frac{f_{osc}}{N_{ADC}} ; T_{CONV} = \frac{(1,5+13)}{f_{clkADC}} \quad (6)$$

El inicio de la conversión del ADC cada $Temp = 1$ s, cuando el contador del Timer 1 es reseteado, se efectúa gracias al uso de la función *auto-triggering* que posee el ADC del microcontrolador.

En el diagrama de la Fig. 5, luego de obtenerse la conversión con el ADC el programa activa una bandera para indicar que debe efectuarse el cálculo de la media móvil de las mediciones realizadas. Mediante este cálculo se busca eliminar el ruido captado en las mediciones de temperatura. Una vez realizado el cálculo y la conversión correspondiente, dependiendo de la configuración recibida desde la unidad maestra y de la temperatura media, el programa activa el relé si esto corresponde.

En el sistema desarrollado, la unidad maestra puede enviar un mensaje con el identificador (ID) correspondiente a una de las unidades remotas y así solicitar de la misma el estado de sus entradas digitales presente en ese momento, como así también el valor de la temperatura obtenida en el último cálculo de la media móvil, el cual es realizado cada un segundo.

2.3.3 Formato de la trama de datos para la comunicación entre las unidades

Todas las funciones que posee el sistema electrónico desarrollado, se logran mediante la comunicación de la unidad maestra a las unidades remotas. Es decir, la primera siempre es la encargada de iniciar la comunicación, utilizando el protocolo CAN con el cual se intercambian tramas de datos con los formatos mostrados en la Fig. 6. Cabe aclarar que en el protocolo CAN las tramas de datos transmitidas constan de más partes, la figura mencionada resume los campos de la trama para facilitar la explicación.



Fig. 6. Trama de datos CAN para: (a) Configuración de salidas; (b) Solicitud de datos; (c) Recepción de datos.

Cuando la unidad maestra requiere una acción de configuración, envía una trama de datos como la indicada en la Fig. 6a, donde el campo de datos (marcado con el recuadro rojo) posee una longitud de cuatro bytes. El primer byte es para indicar el modo de operación del relé (manual o automático) y también el estado del mismo (encendido, apagado o la temperatura de activación), en los demás bytes se indican el estado (encendido o apagado) de las salidas digitales de la unidad remota correspondientes. Para realizar la lectura de la temperatura medida y las entradas digitales de las unidades remotas, la unidad maestra envía una petición de datos a la unidad correspondiente con una trama con el formato indicado en la Fig. 6b.

Ante una petición de datos, la unidad remota correspondiente responde con una trama donde el campo de datos posee cuatro bytes, como se muestra en la Fig. 6c. Uno de los bytes contiene el valor de temperatura medido y el resto representa el estado de cada entrada digital de la unidad remota que envía la trama. Para atender al envío de este tipo de trama, la rutina principal del programa ejecutado en la unidad maestra, es interrumpido y su rutina de servicio a la interrupción recibe y almacena el campo de datos de la trama recibida.

La unidad maestra distingue a las diferentes unidades remotas mediante el valor contenido en el campo identificador (ID) de las tramas mostradas en la Fig. 6. Cuando esta unidad realiza una transmisión, el campo ID es cargado con un valor predeterminado a través del cual se indica la unidad remota con la que desea comunicarse. En el caso de la recepción, cuando la unidad maestra solicita a alguna de las unidades remotas los valores de temperatura medidos y los estados de sus entradas digitales, al recibir una trama, el valor de su identificador es almacenado en una variable "IDreciv", para luego determinar a qué unidad remota corresponde la información recibida. Esto es realizado de esta manera para evitar que el usuario pueda confundirse entre las lecturas de las distintas unidades remotas. Por otro lado, todas las ID utilizadas en los programas de las unidades, están configurados desde un principio y en forma fija. De esta forma, si la ID de la transmisión que recibe la unidad maestra no coincide con alguna de las ID configuradas, en su rutina principal no es interrumpida y la trama correspondiente es desestimada.

Finalmente cabe mencionar que cuando al iniciar una transmisión, en la unidad maestra se enciende el led amarillo "send", posteriormente se apaga y se enciende el led verde o rojo para avisar que la transmisión se realizó con éxito o bien hubo un error respectivamente.

3. Resultados

Para que el usuario pueda interactuar con el sistema construido, se ha desarrollado un menú de funciones en el programa de la unidad maestra. La Fig. 7 presenta un diagrama a través del cual puede explicarse la operación del menú de funciones mencionado, el cual puede visualizarse en el display de la unidad maestra.

Con el diagrama de la Fig. 7 puede comprenderse el manejo general de la unidad maestra. Como, por ejemplo, puede notarse que las salidas digitales de las unidades remotas son activadas directamente por el usuario de forma manual mediante el submenú CONTROL DE UR-N y que también puede comandarse el relé. Este último presenta dos modos de operación. El primero es de forma manual, teniendo un manejo similar al resto de salidas digitales de cada unidad remota. El segundo modo es automático, donde puede configurarse un valor "X" de temperatura y cuando la unidad remota correspondiente mida una temperatura igual o superior a esta, el relé se activará automáticamente. Por otro lado, la Fig. 7 también muestra que a través del submenú LECTURAS UR-N puede observarse en el display el estado de las entradas digitales y el valor de temperatura medido en la unidad remota correspondiente. Este último permite corroborar el funcionamiento correcto de la configuración del relé que posee la unidad remota correspondiente.

El menú de funciones desarrollado posee un cursor el cual permite al usuario recorrer las diferentes opciones mediante los pulsadores "arriba" y "abajo" indicados en la placa de la Fig. 8a. Para seleccionar la opción sobre la que está ubicado el cursor, debe presionarse el pulsador "select". Con

este pulsador, a través del submenú CONTROL DE UR-N, también puede modificarse (alternado su estado) las salidas digitales correspondientes en alguna de las unidades remotas. Para modificar la temperatura “X” en un rango de valores, pueden presionarse los pulsadores de “izquierda” o “derecha”, los cuales decrementan e incrementan respectivamente dicha variable.

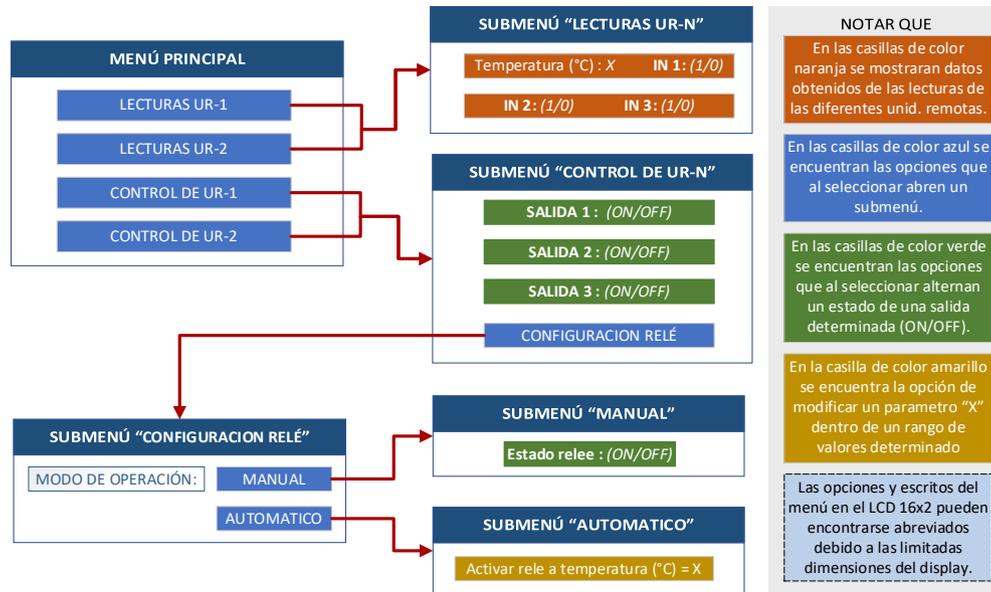


Fig. 7. Diagrama explicativo del menú de funciones de la unidad maestra.

En cuanto al diseño del circuito impreso (PCB) de las placas que posee el sistema, fue realizado con el software EasyEDA. Luego de realizar el diseño mencionado, para la construcción de las placas el PCB fue impreso sobre papel fotográfico; luego de esto el tóner de la impresión fue transferido a las placas de cobre virgen aplicando calor mediante una plancha; se retiró el papel fotográfico y la placa introducida en cloruro férrico para realizar el proceso de grabado del circuito. Una vez finalizado este proceso, cada placa fue limpiada con una virulana y posteriormente se realizó las perforaciones con una mecha de 1mm para luego montar y soldar todos los componentes utilizados. La Fig. 8 muestra los resultados obtenidos mediante los procesos indicados.

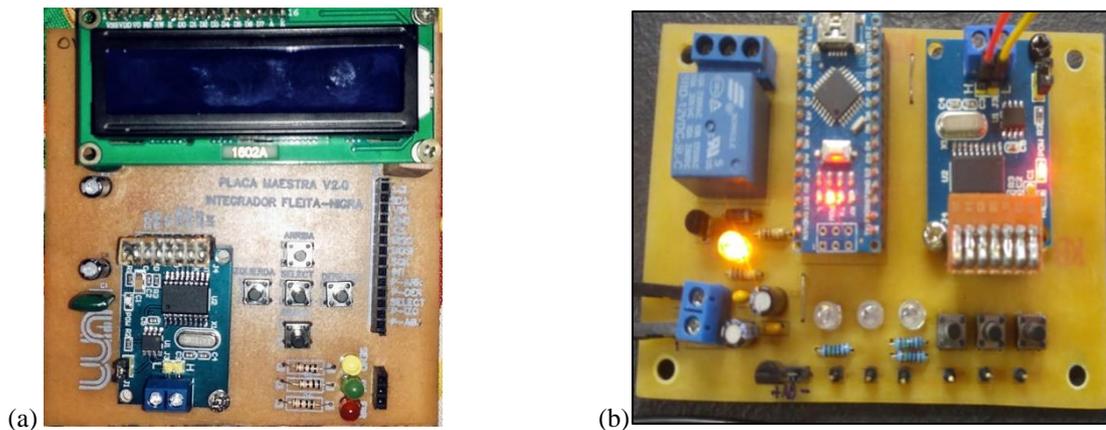


Fig. 8. Placas construidas: (a) Unidad maestra; (b) Unidad remota.

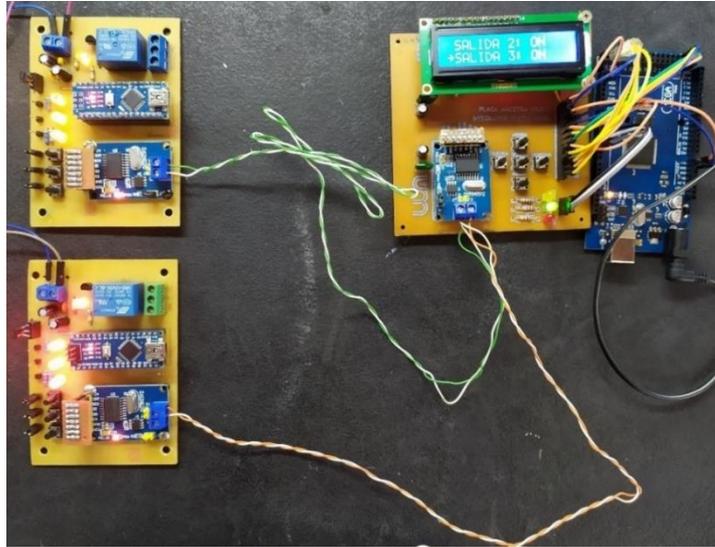


Fig. 9. Sistema electrónico interconectado, pruebas de laboratorio.

Una vez finalizada la programación y construcción del sistema, la unidad maestra fue interconectada con las dos unidades remotas como lo muestra la Fig. 9. A partir de esto, con las realizas pudo notarse que la unidad maestra logro con éxito intercomunicarse con las unidades remotas las cuales informaban del estado tanto de sus sensores como de sus entradas digitales. Estas también respondían correctamente a la configuración de salida y relé realizadas por el usuario, poniendo en alto las salidas cuando la unidad maestra así lo requería o accionando el relé a la temperatura definida previamente en el menú. También pudo corroborarse que las mediciones de temperatura realizadas a través de las unidades remotas, en algunos casos tenían una diferencia de 1 a 2 °C con respecto a las mediciones obtenidas en unos multímetros con sensores de temperaturas usados para comparar.

La realización de este trabajo permitió comprender a fondo la operación del protocolo CAN utilizado para intercomunicar las distintas unidades. Con el mismo se adquirió experiencia en el manejo de librerías, el desarrollo de programas y la corrección de problemas que pueden presentarse al utilizar los módulos comerciales para la comunicación a través del protocolo CAN.

4. Conclusión

Mediante este trabajo se logró establecer con éxito la comunicación bidireccional entre dos unidades remotas y la unidad maestra. Pudiendo esta última recibir con éxito información en tiempo real del estado de las entradas y comandar el estado de las salidas de las unidades remotas. Para alcanzar estos resultados hubo que interiorizarse con funcionamiento del protocolo de comunicación CAN, comprendiendo su funcionamiento y las ventajas que el mismo ofrece. Por otra parte, el desarrollo de los circuitos y programas, permitió poner en juego conocimientos relacionados con circuitos de polarización de leds, transistores y relés, como así también el diseño de circuitos con microcontroladores y la programación de estos dispositivos mediante lenguaje C. Esto demuestra que, además de cumplir con las especificaciones requeridas en la consigna de la actividad integradora

propuesta en la asignatura, la realización de este trabajo permitió integrar diferentes saberes propios de la asignatura como también otros adquiridos en asignaturas previas.

Los resultados obtenidos pueden utilizarse en diferentes áreas en las cuales son necesarias el control de la temperatura y a su vez la activación de diferentes elementos a distancia. Como, por ejemplo, la calefacción de diferentes recintos de forma individual, los cuales son comandados por una unidad central, que permite monitorear el estado de las mismas.

Esta actividad nos brindó la posibilidad de comprender la operación de un protocolo de comunicación muy potente, que es ampliamente usado en la industria, debido a su alta inmunidad al ruido. El sistema desarrollado puede utilizarse en diferentes proyectos que requieran una transmisión de datos cableada, como por ejemplo el proyecto de investigación del cual también forma parte este trabajo y donde debe realizarse el tratamiento y transmisión de variables asociadas a los generadores fotovoltaicos y baterías empleadas en una microrred eléctrica. Es por esto que puede afirmarse que los resultados obtenidos en este trabajo constituyen una base para el desarrollo del sistema de supervisión de la microrred mencionada.

Referencias

- [1] N. Hatziargyriou. Microgrids: Architectures and Control. 2014.
- [2] Asian Development Bank, A. D. B. Handbook on Microgrids for Power Quality and Connectivity. 2020.
- [3] R. K. Chauhan, K. Chauhan. Distributed Energy Resources in Microgrid. Integration, Challenges and Optimization. 2019.
- [4] M. S. Mahmoud. Microgrid: Advanced Control Methods and Renewable Energy System Integration. 2017.
- [5] S. Corrigan. Introduction to the Controller Area Network (CAN), Application Report SLOA101B. Texas Instruments. 2016.
- [6] A. G. Maxit, G. A. Fernández, F. Botterón. Avances en el desarrollo de un sistema de supervisión de una microrred utilizada para el bombeo de agua en huertas comunitarias rurales. JIDeTEV 2021.
- [7] M. Di Natale. Understanding and Using the Controller Area Network Communication Protocol CAN. 2016.
- [8] Atmel. ATmega640/V-1280/V-1281/V-2560/V-2561/V. Hoja de datos disponible on-line: https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf
- [9] Módulo CAN MCP2515. Disponible on-line: <https://naylorlampmechatronics.com/alambrico/253-modulo-can-mcp2515.html>
- [10] Philips Semiconductors. TJA1050 High speed CAN transceiver. Hoja de datos disponible on-line: <https://www.nxp.com/docs/en/data-sheet/TJA1050.pdf>
- [11] Microchip Technology. Stand-Alone CAN Controller with SPI Interface. Hoja de datos disponible on-line: <https://ww1.microchip.com/downloads/en/DeviceDoc/MCP2515-Stand-Alone-CAN-Controller-with-SPI-20001801J.pdf>
- [12] Apuntes de la asignatura Técnicas Digitales 2, Universidad Nacional de Misiones, Facultad de Ingeniería. 2020.
- [13] Atmel. ATmega328P: 8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash. Hoja de datos disponible on-line: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf
- [14] Texas Instruments. LM35 Precision Centigrade Temperature Sensors. Hoja de datos disponible on-line: <https://www.ti.com/lit/ds/symlink/lm35.pdf>