

# Universidad Tecnológica Nacional

## Proyecto Final

---

# Digitalización, monitorización, registro y análisis de variables de comportamiento – Caldera Gonella.

---

*Autor:*

- Filgueira Javier

*Director:*

*Proyecto final presentado para cumplimentar los requisitos académicos  
para acceder al título de Ingeniero en Electrónica  
en la*

**Facultad Regional Paraná**

25 de Agosto de 2020



## **Declaración de autoría:**

Yo declaro que el Proyecto Final “Digitalización, monitorización, registro y análisis de variables de comportamiento – Caldera Gonella” y el trabajo realizado es propio. Declaro:

- Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero en Electrónica, en la Universidad Tecnológica Nacional, Regional Paraná.
- Se establece claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pre-grado.
- Siempre que se ha utilizado trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
- Se ha indicado y agradecido correctamente a todos aquellos que han colaborado con el presente trabajo.
- Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, se ha indicado claramente el alcance del trabajo realizado.

Firma:

•

Fecha: 25 de Agosto de 2020



## Agradecimientos:

En los largos años que ha durado este proceso de aprendizaje y formación profesional, tengo que agradecer a cada una de las personas que ha estado a mi lado, mostrando su apoyo y ayuda.

En especial, tengo que agradecer el constante apoyo incondicional de mi familia a lo largo de este camino. A mis padres, por sus consejos y ayuda, apostando siempre por mi futuro. A mis hermanos, por la confianza depositada en mí.

A aquellos amigos que han estado a mi lado en este proceso, apoyándome y escuchando mis preocupaciones. A todos aquellos compañeros con los que he compartido este camino, tanto en los momentos dentro del aula como fuera de la institución. A aquellos profesores con los que he compartido algún tiempo durante estos años de preparación.

Gracias a todos y cada uno de ustedes, por ayudarme a formarme en la persona que hoy soy.

Filgueira Javier



Universidad Tecnológica Nacional

## *Abstract*

Facultad Regional Paraná

Ingeniero en Electrónica

### **Digitalización, monitorización, registro y análisis de variables de comportamiento – Caldera Gonella**

Filgueira Javier

**Abstract:**

The implementation of a system that shows the status of multiple variables and consequently observe the performance of an industrial steam boiler, continuously. This would be possible through the incorporation of a dedicated software, which would receive the data from an acquisition board, and will allow the company to visualize and analyze them.

Likewise, the information obtained may be read from the desktop PC where the program is installed, or from any remote device through an internet server.

The system will use a very low cost Arduino electronic board, which will take information from the sensors located in the boiler sector and from there it will send them to a desktop PC via a communication cable at a distance of approximately 80 meters, to which used the RS-485 protocol.

Upon receiving the data, the software will prepare graphics that will be sent to a external web server, to be read from any device.

Robust and simple software was obtained with intuitive operation and visualization optimum of the variables to observe. Communication is very acceptable, for more that the path of the communication cable runs parallel to conductor's powerful, essential data is not lost.

The data web server is it behaves optimally and I always provide the service for which it is used.

Reading the data from a mobile device through Android software is what ends up giving the project strength. Its simplicity, comfort and clarity, allows a pleasant viewing.

**Keywords:**

Android, Arduino, Backup, Remote device, RS-485, Storage, Visualization.

**Resumen:**

Se realiza la implementación de un sistema que muestra el estado de distintas variables y permite observar el desempeño de una caldera de vapor industrial de manera continua. Esto es posible con la incorporación de un software dedicado, que recibe los datos desde una placa adquisidora, y permite a la empresa la visualización y el análisis de dichos datos. Además, la información obtenida podrá ser leída desde la PC de escritorio donde se encuentre instalado el programa, o desde cualquier dispositivo remoto a través de un servidor en internet.

El sistema utiliza una placa electrónica Arduino de hardware libre de muy bajo costo, que tomará información desde los sensores ubicados en el sector de caldera y desde allí los enviará a una PC de escritorio a través de un cable de comunicación a una distancia aproximada de 80 metros, para lo que se optará por el protocolo RS-485.

El software al recibir los datos, elabora gráficos que serán remitidos a un servidor web externo, para ser leídos desde cualquier dispositivo.

El software obtenido es robusto y simple, con un manejo intuitivo y una visualización óptima de las variables a estudiar. La comunicación es aceptable, y los datos esenciales no se pierden a pesar de la trayectoria que la señal recorre de manera paralela a conductores de alta potencia. El servidor Web de datos se comporta de forma óptima, brindando siempre el servicio para el cual se lo utiliza.

La lectura de los datos desde un dispositivo móvil a través de un software Android es lo que termina dándole firmeza al proyecto. Su simplicidad, comodidad y claridad, permite una agradable visualización.

**Palabras Clave:**

Almacenamiento, Android, Arduino, Backup, Dispositivo remoto, RS-485, Visualización.



### *Reconocimientos:*

Se agradece a la empresa Papelera Entre Ríos S.A. por la confianza y los medios que se me brindaron para poder desarrollar el proyecto final dentro de sus instalaciones.

# Índice:

Capítulo 1: Introducción .....	1
Capítulo 2: Desarrollo .....	6
2.1 Descripción .....	6
2.1.1 Adquisición de señales .....	7
2.1.2 Procesamiento de datos .....	11
A. Investigación de componentes .....	11
I. Análisis y selección.....	11
B. Propuestas de circuitos o esquemas.....	12
I. Pruebas, experiencias.....	12
II. Problemas y soluciones implementadas .....	12
C. Lectura de datos y armado de trama.....	14
2.1.3. Comunicación RS-485 .....	15
A. Tendido del cable a través del campo.....	15
B. Propuestas de circuitos o esquemas.....	16
I. Pruebas, experiencias.....	16
C. Análisis del funcionamiento .....	20
I. Etapas de conversión de datos .....	20
D. Pruebas de recepción de datos.....	24
2.1.4. Interpretación de Datos .....	25
A. Análisis del funcionamiento.....	25
I. Secuencia del software .....	25
II. Problemas y soluciones implementadas .....	28
2.1.5. Almacenamiento diario de datos .....	29
A. Análisis del funcionamiento.....	29
I. Backup de datos continuo.....	29
II. Guardado de graficas en archivos PDF .....	30
B. Desarrollo técnica Backup - Detalle de utilización de archivos .txt .....	32
2.1.6. Comunicación con Servidor Web externo.....	32
A. Investigación de plataformas free .....	32
I. Análisis y selección.....	32
B. Desarrollo del software a medida.....	34
I. Diseño del modo y la forma de la visualización de los datos.....	34
C. Análisis de la aplicación obtenida .....	37
2.2 Diseño Completo.....	37
2.2.1 Diseño final.....	37
2.2.2 Prestaciones.....	37
2.2.3 Fotos del hardware y capturas de pantalla de software.....	38

Capítulo 3: Resultados .....	41
Capítulo 4: Análisis de Costos .....	42
Capítulo 5: Discusión y Conclusión.....	43
5.1 Posibles mejoras.....	43
5.1.1 Hardware .....	43
5.1.2 Software .....	43
5.2 Conclusión .....	43
Capítulo 6: Literatura Citada.....	45
Apéndice A – Hojas de Datos .....	47

## Lista de Figuras:

Figura 1 Diagrama de Bloques.....	6
Figura 2 Recorrido cables de señal .....	7
Figura 3 Caudalimetro de Agua .....	8
Figura 4 Caudalimetro de Vapor .....	8
Figura 5 Ubicación sobre tablero general .....	9
Figura 6 Bandejas portacables .....	9
Figura 7 Interior de tablero - Soporte placa Arduino .....	10
Figura 8 Interior de tablero - Arduino montado.....	11
Figura 9 Circuito conversor diferencial con operacional.....	12
Figura 10 Placa de prueba.....	13
Figura 11 Placa de prueba montada en el tablero .....	13
Figura 12 Verificación de lecturas in situ.....	14
Figura 13 Circuito transmisión y recepción RS-485.....	17
Figura 14 Tablerito de prueba de recepción de datos - PC escritorio .....	18
Figura 15 Placas finales - frontal .....	18
Figura 16 Placas finales - debajo .....	19
Figura 17 Tablerito final de recepción de datos - PC escritorio.....	19
Figura 18 Etapas de conversión de Datos .....	20
Figura 19 Línea no balanceada [4].....	23
Figura 20 Línea balanceada [4].....	23
Figura 21 Diagrama de funcionamiento del software.....	25
Figura 22 Pantalla de bienvenida PC .....	26
Figura 23 Pantalla de Visualización de Datos PC.....	27
Figura 24 Archivo Backup .....	29
Figura 25 Almacenamiento en archivos PDF .....	31
Figura 26 Botones de función .....	31
Figura 27 Logo APP de Diseño.....	34
Figura 28 Edición de Pantallas de visualización Android.....	35
Figura 29 Logo APP de Visualización .....	35
Figura 30 Pantalla Inicial .....	36
Figura 31 Pantallas Caudales y Otros .....	36
Figura 32 Gabinete.....	38
Figura 33 Tablero final.....	38
Figura 34 Almacenamiento en archivo PDF.....	39
Figura 35 Pestañas Caudales y Otros APP.....	39
Figura 36 Desde explorador de internet página thingspeak.....	40

# Lista de Tablas

Tabla 1 Costos.....42

# Lista de Abreviaciones

# Lista de Símbolos

## **Dedicado a:**

Dedicado a mis hijos, a mi mujer que ha estado a mi lado y me ha acompañado en la culminación de este camino.





## Capítulo 1: Introducción

Ante la necesidad de digitalizar mediciones que se realizan diariamente en la caldera de la Papelera Entre Ríos S.A., surge la idea de implementar un sistema que permita ver las variables en tiempo real, y su comportamiento.

Hoy en día la empresa cuenta con un método poco eficaz para relevar datos sobre la medición del caudal de agua consumida (caudalímetro electromagnético de paso total) y el caudal de vapor de salida de la caldera (Transmisor de presión diferencial), los que son ingresados de manera manual en una planilla por el personal a cargo (calderista).

De esta situación nace la idea de implementar un sistema que muestre el estado de múltiples variables y por consecuencia el desempeño de la caldera, de manera actualizada. Esto sería posible a través de la incorporación de un software dedicado, que recibiría los datos desde una placa adquisidora, y permitiría a la empresa la visualización y el análisis de los mismos. La información obtenida podría ser leída desde la PC de escritorio donde se encuentre instalado el programa, o desde cualquier dispositivo remoto a través de un servidor en internet.

Además, el sistema contaría con otras funciones como, el almacenamiento de información por un lapso de 24 horas, generando archivos en PDF que incluyan en su nombre la fecha y hora de su creación para su fácil identificación; la registración del tiempo en que el caudal de vapor de agua (Transmisor de presión diferencial) sea menor a los valores mínimos de producción, logrando conocer así las pérdidas ocasionadas en estos períodos de improductividad; calcular el caudal de agua consumida diariamente (caudalímetro electromagnético de paso total); conocer los tiempos de purgado de la superficie (Controlador BC1100); dar aviso de activación de alarma de la caldera (Burning Control).

El sistema utilizaría una placa electrónica Arduino de hardware libre de muy bajo costo, que tomaría información desde los sensores ubicados en el sector de caldera y desde allí los enviará a una PC de escritorio a través de un cable de comunicación a una distancia aproximada de 80 metros, para lo que se optará por el protocolo RS-485.

El software al recibir los datos, elaborará gráficos que serán remitidos a un servidor web externo, para ser leídos desde cualquier dispositivo.

Se utilizará una fuente de buena calidad para el filtrado de ruido del ambiente industrial y para la comunicación serie RS-485 se realizarán dos placas de conversión confeccionadas en placas de circuitos impresos.

El receptor (PC de escritorio), necesitará un cable de adaptación serie a USB al no contar con este puerto.

### ESTUDIO DE MERCADO

Nuestro producto ha sido pensado para lograr un alto control de variables en la Caldera de Papelera Entre Ríos S.A.. Actualmente, la recolección de estos datos es poco certera, y no acompaña los crecimientos tecnológicos que se necesitan para lograr procesos de producción eficaces.

Ante esto, la instalación de un software que permita dar precisión y rapidez en estos procesos de producción, simboliza un importante aporte para la empresa.

Actualmente en el mercado encontramos sistemas como los Datalogger para control de procesos, adquisición de datos, test automáticos y monitorización en PC. Los modelos incluyen entradas y salidas analógicas y digitales, en cajas compactas para laboratorio, o con una carcasa metálica para montaje en rail o panel, y con software compactos y pocos flexibles lo que impiden un libre acceso al análisis de variables, debiendo contar para ello con el asesoramiento de un agente oficial, incrementando de esta manera los costos del producto. A continuación se detallan modelos y precios de placas:

MagiDeal Módulo de Adquisición de Datos M140	precio 106,99 EUR
OM-USB-1208FS, OM-USB-1408FS and OM-USB-1608FS	precio 229 EUR
CWT-MB serie	precio 200 USD

Fecha del presupuesto: 20 de Abril 2019

Gracias a que Papelera Entre Ríos S.A. cuenta con sensores ubicados en la Caldera, es que se logra tener un avance importante en el proyecto al no emplear tiempo en su colocación, además de tener personal calificado para el manejo del software. Esto nos deja ver que los costos de nuestro proyecto no serán elevados, logrando así hacerlo viable y económico. A través de los datos obtenidos podemos conocer precios aproximados de los insumos tentativos para la elaboración del proyecto:

Placa Arduino UNO	500 pesos
Fuente Switching Regin Power 5VDC 8a 50w	2900 pesos
Fuente 5VDC 2A tipo cargador	1000 pesos
Cable adaptador USB a RS232 (Db9) con cable Mallado Manhattan	1200 pesos
Materiales para placas de conversión serie RS-232 a RS485	400 pesos
Par trenzado UTP CAT6 GLC (100 metros)	5000 pesos
2 Cajas plásticas Roker 71x71x50mm	400 pesos

Fecha del presupuesto: 20 de Abril 2019

Destacamos que se trabajará sobre un software acorde a los avances de la tecnología, provisto de una placa relativamente nueva para el mercado, lo que le permitirá a través de actualizaciones, sumar variables requeridas por la empresa, y adaptarse a los futuros sistemas operativos.

Además al utilizar un servidor web externo orientado a las nuevas tendencias del IOT, tendremos un espacio abierto y sin costos que nos permitirá observar nuestros datos desde cualquier dispositivo.

### Lógica de Funcionamiento

El sistema utilizará una placa electrónica Arduino ubicada en el armario de control de la caldera, que tomará información desde los sensores ubicados en este. Tres de los sensores nos darán una señal de corriente de 4 a 20 mA., estos medirán el caudal de vapor de agua (Transmisor de presión diferencial), el caudal de agua consumida (caudalímetro electromagnético de paso total), y purgado de la superficie (Controlador BC1100); una

cuarta señal nos comunicará la activación de la alarma en la caldera (Burning Control), visualizada mediante una salida relé usada para tal fin.

La placa Arduino que se usará es el modelo UNO, elegida especialmente por la cantidad de entradas análogas, tensión de alimentación, tamaño y disposición con las que cuenta. Posee las siguientes características:

- **Microcontrolador:** ATmega328
- **Voltaje de funcionamiento:** 5 V
- **Pines I/O digitales:** 14 (de los cuales 6 proveen salida PWM)
- **Pines de entradas análogas:** 6
- **Corriente DC por cada pin I/O:** 40 mA
- **Corriente DC en el pin de 3.3 V:** 50 mA
- **Memoria Flash:** 32 KB (ATmega328) de los cuales 0.5 KB son utilizados por el bootloader
- **SRAM:** 2 KB (ATmega328)
- **EEPROM:** 1 KB (ATmega328)
- **Velocidad de reloj:** 16 MHz

Tomando en cuenta la distancia existente entre los sensores y la placa adquisidora (10 metros), se usarán cables mallados para evitar que el ruido industrial se mezcle con nuestros lazos de corriente.

La placa se alimentará con una fuente de buena calidad (Fuente Switching Regin Power Modelo -05SA de 5VDC $\pm$ 10% 8A 50w) que admite una entrada variable entre los 90~264 Vac, ayudándonos a filtrar ruidos del ambiente.

Luego, la placa elaborará una trama de datos que enviará a una PC de escritorio a través de un cable de comunicación serie, ubicada a una distancia aproximada de 80 metros, para lo cual usaremos el protocolo RS-485. El acondicionamiento de la señal para este protocolo necesitará de dos placas de conversión (TTL a RS-485) confeccionadas en placas de circuitos impresos.

Este módulo se conectará desde la salida de comunicación de la placa y será el encargado de convertir las señales seriales TTL a RS-485. Además, se utilizará el integrado SN75176A y elementos pasivos, ahorrando de esta manera la inversión en módulos conversores comerciales que elevarían los costos del proyecto.

El cable que transportará la trama de información será un multiconductor de cobre estañado con blindaje de malla de cobre marca Epuyen que posee las siguientes características:

Multiconductor de cobre estañado flexible, aislación termoplástico (PVC), malla hilo de polipropileno que brinda mayor flexibilidad, blindaje de malla de cobre, cubierta externa PVC, autoextinguible y no propagante de llama.

Radio de montaje: 4 veces el diámetro del cable.  
Tracción máxima: 5 daN/mm<sup>2</sup> sobre conductores de cobre.  
Calibre: diámetro 0.45 mm.  
Formación de conductor: cuerda 7 x 0.15 mm.  
Codigo MCM 04045  
Número de conductores: 4 hilos  
Espesor de aislación 0,25mm  
Espesor de cubierta externa 0,6mm  
Peso nominal 48kg/Km

El receptor (PC de escritorio), necesitará un cable de adaptación serie a USB por no contar con este puerto. Se utilizará un Cable adaptador USB a RS232 (Db9) con cable Mallado Manhattan.

El software al recibir los datos, elaborará gráficos que indicaran la evolución de las señales en función del tiempo y estos serán remitidos a un servidor web externo, para ser leídos desde cualquier dispositivo.

Para la carga de datos en el servidor se utilizará el portal web dweet.io o el Freeboard.io, el primero nos ofrece un servicio para enviar y representar datos en la nube sin necesidad, si quiera, de registrarnos en la plataforma. El segundo nos da una mejor presentación de los datos, es simple, cómodo, listo para funcionar y Open Source.  
Será seleccionado aquel, que al momento de la carga, responda con mayor eficacia al proyecto.

Además, el sistema cuenta con otras funciones como:

- Sincronización del programa con la hora de la PC. El almacenamiento de información por un lapso de 24 horas, (desde las 6:00hs hasta las 6:00hs del otro día) generará archivos en PDF que incluirán en su nombre la fecha y hora del día para su fácil identificación. Además se tendrá la opción de guardar gráficos en cualquier momento del día, cuando se detecte algún tipo de anomalías que se necesiten revisar o por cualquier otro motivo, las que serán archivadas con el mismo formato de nombre.

- Registración del tiempo en que el caudal de vapor de agua sea menor a los valores mínimos de producción. Se tomará un valor mínimo que será equivalente a no tener demanda, los ciclos se irán sumando a lo largo del período y serán visualizados siempre dentro del día, logrando conocer así las pérdidas ocasionadas en estos momentos de improductividad.

- Cálculo del volumen de agua utilizada diariamente. Podrán observarse los intervalos de consumo y al final del día se mostrará el resultado total.

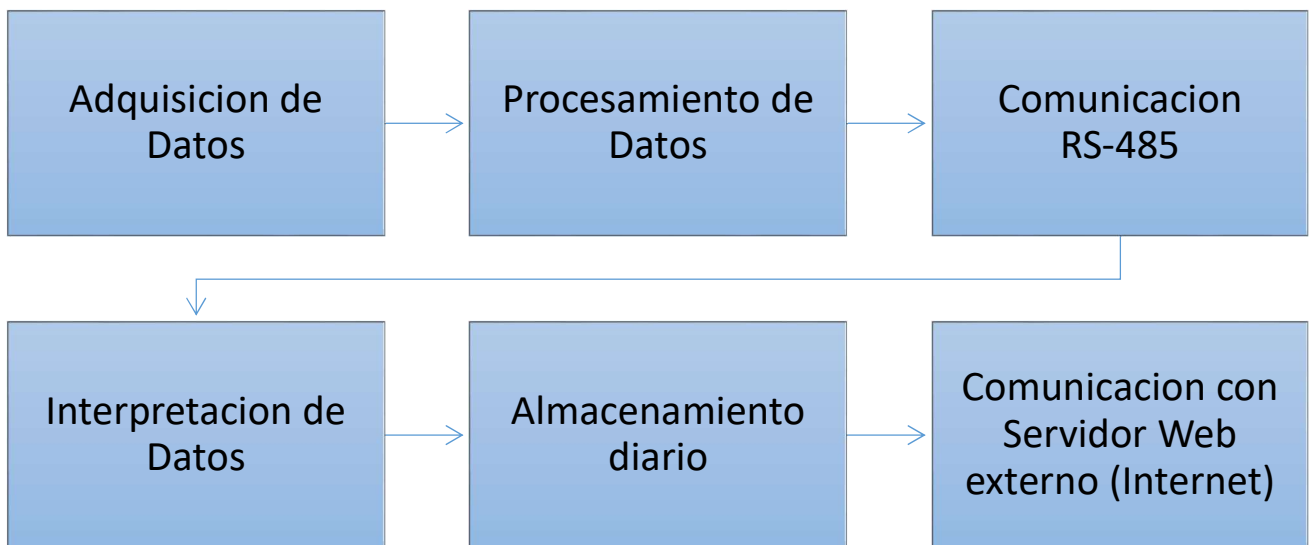
-Conocer los tiempos de purgado de la superficie. El controlador BC1100 de Spirax Sarco es parte de una serie integrada de equipamiento de sala de calderas. El regulador se utiliza conjuntamente con el sensor de conductividad Spirax Sarco y una válvula de purga para registrar y controlar la concentración de los sólidos totales disueltos (TDS) en calderas de vapor. El controlador abre la válvula de purga periódicamente para drenar la línea y permitir que una muestra de agua de la caldera pase por el sensor. La conductividad eléctrica de esta muestra es comparada con el punto seleccionado en el controlador del panel frontal. Si la conductividad es menor que el punto seleccionado, el regulador permite que la válvula de purga se cierre. Si la conductividad es mayor que el punto seleccionado, la válvula seguirá abierta, permitiendo que el agua contaminada de la caldera sea reemplazada y compensada por agua limpia. Observando y analizando esto, vamos a tener una idea del estado de la caldera y el agua consumida.

-Aviso de activación de alarma de la caldera. Se observarán los momentos en que se dispare la alarma con la desventaja de no saber el motivo. Servirá para que queden plasmadas dichas fallas, que luego serán informadas por los turnos respectivos.

## Capítulo 2: Desarrollo

### 2.1 Descripción

El Proyecto puede dividirse en varias etapas:



*Figura 1 Diagrama de Bloques*

### Adquisición de Señales

Cableado desde sensores hasta la placa Arduino. Protección mecánica y contra ruidos. Instalación y armado de la placa, fuente y otros.

### Procesamiento de datos

Programación de la Placa Arduino. Lectura y acondicionamiento de datos. Pruebas y ajustes de parámetros.

## **Comunicación RS-485**

Tendido de cable de comunicación a través de la planta hasta PC. Pruebas y ajustes de parámetros de la trama recibida. Filtrado de errores, etc. Armado e instalación de carcasas con placas conversoras TTL-RS485.

## **Interpretación de Datos**

Programación del software de PC. Pruebas y ajustes gráficos. Cálculos especiales.

## **Almacenamiento diario de datos**

Programación del software de Pc, lógica de nombrado de los archivos PDF.

## **Comunicación con Servidor Web externo**

Programación de comunicación software PC - servidor web externo. Envío, reajuste y visualización de datos.

### **2.1.1 Adquisición de señales**

En esta primera etapa se comienza con la colocación de bandejas porta cables, las que serán utilizadas para el tendido de líneas de comunicación. Estas nos ayudarán en la alimentación de los sensores y la placa adquisidora de datos.

Se evaluó la disposición y recorrido de los cables de señal, para evitar un entrecruzamiento con cables de potencia circundantes y aquellos que están dentro del tablero general de la caldera. Durante una parada en la producción, se hizo una reubicación de cables.



*Figura 2 Recorrido cables de señal*





Figura 3 Caudalimetro de Agua



Figura 4 Caudalimetro de Vapor



*Figura 5 Ubicación sobre tablero general*



*Figura 6 Bandejas portacables*

Uno de los sensores ya instalados releva sus valores en gráficos, a través de un registrador de gráfica circular. Al poseer un cable sin mallado, se procedió a su cambio para lograr un filtro ante los ruidos externos.

Para dar comienzo al proyecto y ante la necesidad de organizar nuestros dispositivos, se adquirió una caja metálica con grado de IP65, con las dimensiones necesarias para tal fin, lo que nos ayudaría a protegerlos de los factores ambientales del lugar (humedad, polvo, temperaturas), y para una adecuada manipulación de los mismos. Esta caja fue colocada sobre el tablero general de la caldera y se le hicieron perforaciones para tener contacto con el mismo. Al momento de trabajar en la vinculación, se protegió de virutas a todos los componentes en su interior.

Conjuntamente, se puso a tierra el tablero, tanto como medida de seguridad como para absorber sonidos externos. Al momento de realizarlo se observó que el anclaje del tablero principal no estaba en buen estado y se lo reparó.

Para la alimentación de la placa adquisidora de datos se utilizó una fuente de 220vac/5vdc, protegida con un interruptor termo magnético. Además, los sensores o transmisores nos proveen de una señal de corriente (lazo de corriente), que haremos circular por una

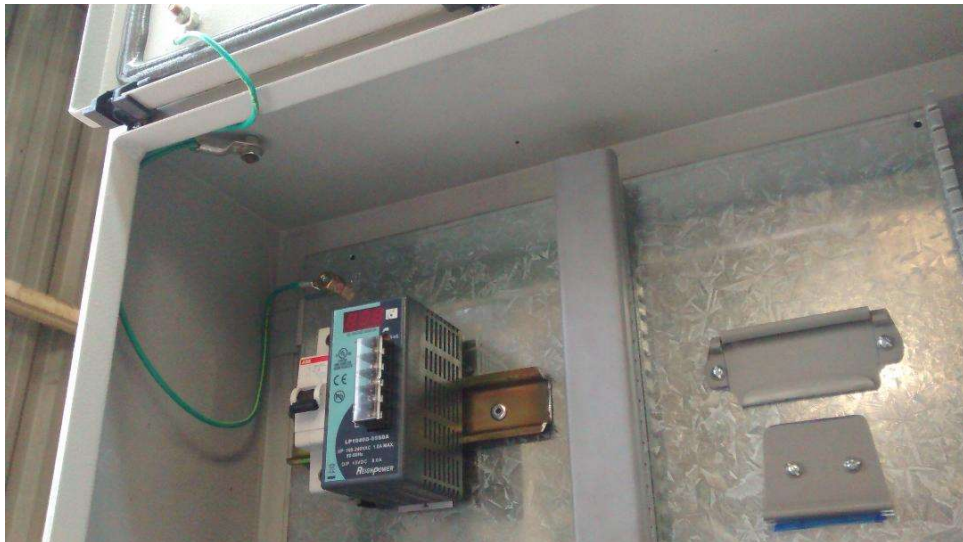
resistencia convirtiéndola en tensión, y como protección para posibles sobre voltajes se conectan diodos Zener en cada entrada digital y analógica de nuestro Arduino.

Los diodos Zener son diodos de silicio fuertemente dopados, los que funcionan como estabilizadores de tensión de la red. Esto nos permite controlar variables de voltaje, conduciendo los posibles excesos de estos a masa.

Por otro lado se colocaron filtros para eliminar ruidos entre masa y tierra y con el fin de proteger la entrada de radiofrecuencia se incorporaron dos diodos en anti paralelo, los que interrumpirán posibles sobretensiones producidas por cargas estáticas.

Teniendo en cuenta los espacios para poder manipularlo, se continuó con el armado del tablero y todos sus elementos, la disposición de los cables y de la placa adquisidora de datos. Para ésta última se armaron borneras especiales tanto para las entradas a utilizar, como para las entradas y/o salidas sin uso. Estos pines deben aterrizar para que no se absorban ruidos externos.

Dentro del tablero, los cables de comunicación fueron puestos en caminos diferentes, evitando de esta manera el contacto entre ellos, y para la transmisión a la PC se utilizó una placa conversor ubicada en su interior. A su vez, sobre este se ha montado un anclaje especial para la placa adquisidora de datos que tiene una protección mecánica individual, con el objetivo de que su colocación y/o extracción sea de fácil manipulación.



*Figura 7 Interior de tablero - Soporte placa Arduino*



*Figura 8 Interior de tablero - Arduino montado*

Siguiendo con este criterio, la placa ha sido dispuesta teniendo en cuenta las dimensiones del cable y la ubicación de la PC y que su funcionamiento sea accesible y práctico.

Al momento del tendido y la conexión de tensión de línea del equipamiento colocado se realizó un corte general de tensión en un lapso no productivo, todo esto con la autorización del Calderista responsable.

## **2.1.2 Procesamiento de datos**

### **A. Investigación de componentes**

#### **I. Análisis y selección**

En esta segunda etapa podemos encontrar las entradas analógicas trabajando entre 0 y 5Vdc. Para esto se elaboró una placa que convierte los lazos de corriente en tensiones de trabajo. En un comienzo como prueba se utilizó únicamente una resistencia para la conversión del lazo de corriente. Pero más tarde y tras una serie de investigaciones se decidió usar un circuito que tiene la característica de aislar la señal de entrada/salida, es decir un aislamiento galvánico, con separación de las partes funcionales de un circuito eléctrico que previene el traspaso de portadores de carga.

Este tipo de aislamiento es utilizado cuando se desean transmitir señales entre las distintas partes funcionales, pero las masas tienen que mantenerse separadas por motivos de seguridad. Al contar con una fuente de 5Vdc, se diseñaron circuitos amplificadores diferenciales de corriente-tensión con operacionales especiales que soportan una alimentación con fuente simple y con la cualidad Rise to rise para poder obtener una conversión 0-20mA a 0-5Vdc.

## B. Propuestas de circuitos o esquemas

### I. Pruebas, experiencias

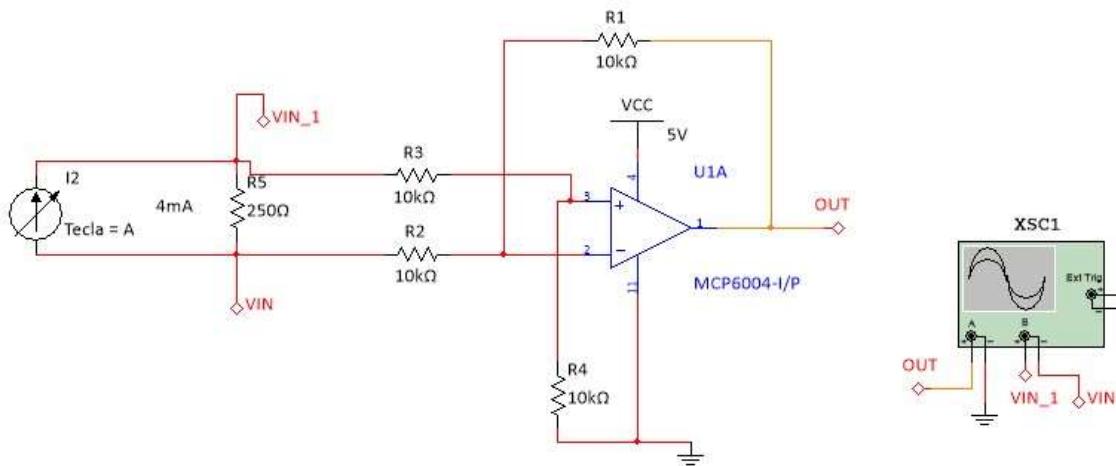


Figura 9 Circuito conversor diferencial con operacional

Para el proyecto se necesitaban realizar tres conversiones independientes. Investigando y analizando las opciones en el mercado, se optó por un integrado que contiene cuatro operacionales, que pueden configurarse en la placa y permiten dejar para una futura ampliación una cuarta opción totalmente funcional. Por otro lado se hicieron las conexiones en la alimentación y se colocaron borneras para dar rigidez a la vinculación de todo lo externo a la placa Arduino.

Para proteger las entradas analógicas, luego de los amplificadores diferenciales, se colocó un diodo zener 5.1V, y para aquellas entradas analógicas sin uso se colocaron a masa a través de una resistencia para asegurar que no ingresen fluctuaciones externas.

Como protección de las entradas digitales se optó por utilizar un optoacoplador el pc817, un dispositivo que emplea luz para conectar dos circuitos eléctricos manteniendo un aislamiento galvánico entre ellos, es decir, nos ayuda a que la carga eléctrica no encuentre un camino para pasar entre ambos circuitos, siendo a través de la luz la única comunicación entre ellos, sin existir contacto eléctrico.

Al contar con un aislamiento galvánico se protege la electrónica de control de cualquier tipo de ruido, sobrevoltaje, pico, armónico, corriente inducida, etc... que podría dañarla. En caso de fallo catastrófico, el optoacoplador podría llegar a ser destruido, pero únicamente tendríamos que reemplazarlo y es un componente económico.

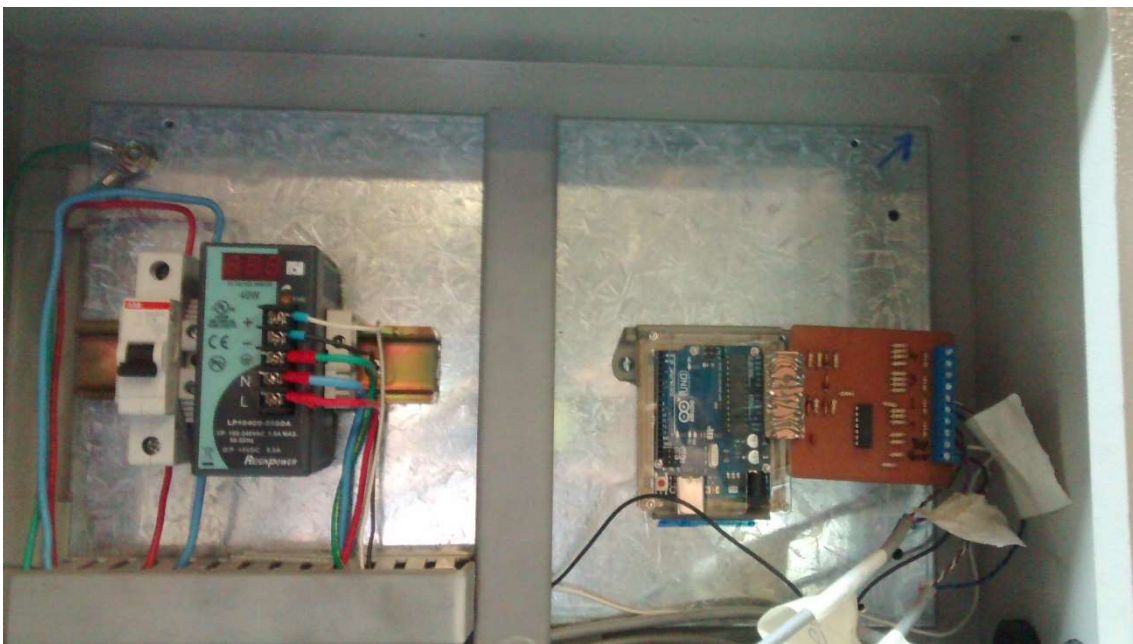
### II. Problemas y soluciones implementadas

El circuito primero se probó en protoboard, para más tarde quemar una placa de prueba y luego ensayar en campo.



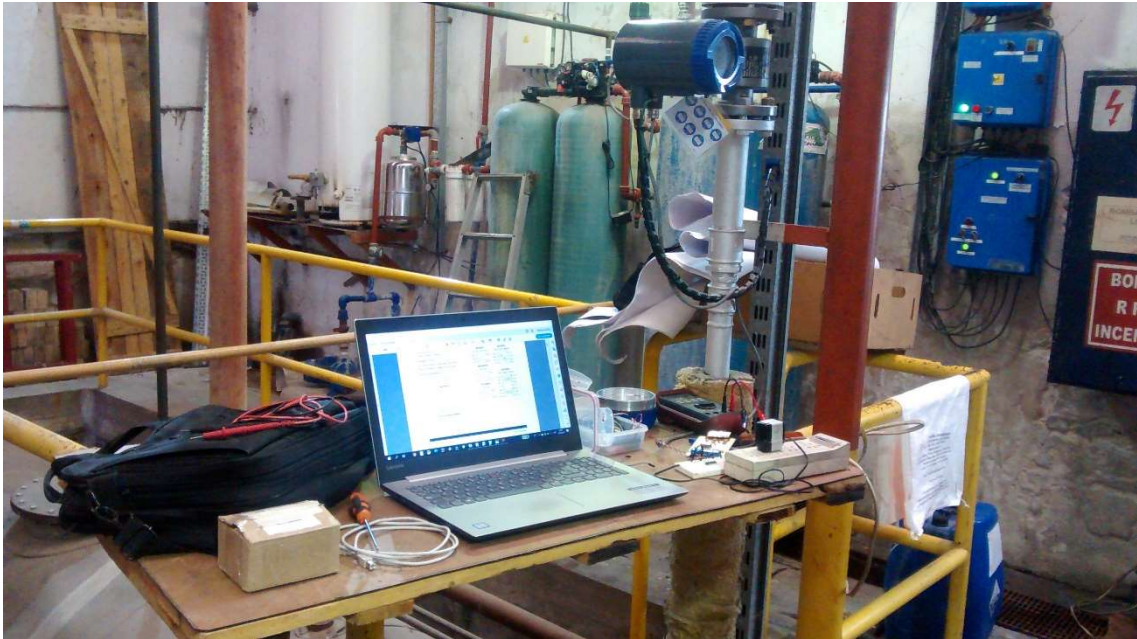
*Figura 10 Placa de prueba*

Superada la etapa de prueba, se rediseñó la placa de conversión (la posición de componentes y otros detalles) y se fabricó la placa final.



*Figura 11 Placa de prueba montada en el tablero*

### C. Lectura de datos y armado de trama



*Figura 12 Verificación de lecturas in situ*

Para programar nuestra placa Arduino Uno el compilador/depurador es el software de la comunidad Arduino, versión 1.8.9. En la programación se declararon las entradas analógicas a utilizar, y a las entradas digitales se les activaron los pull-up internos de la placa para hacer más sencillo el circuito y evitar utilizar resistencias externas. El hardware Arduino tiene incorporado un soporte para la comunicación serie en los pines 0 y 1, utilizados también para llevar la información al ordenador a través de una conexión USB. El soporte nativo serie con el que contamos pasa a través de un componente de hardware (integrado en el chip) llamado UART, que permite al chip ATmega recibir la comunicación serie, incluso mientras trabaja en otras tareas, siempre que haya espacio en la memoria intermedia serie de 64 bytes.

Contábamos además, con la librería "SoftwareSerial.h" que nos permitió lograr la comunicación serie en otros pines digitales del Arduino, usando el software para replicar la funcionalidad. Es posible tener múltiples puertos serie de programas con velocidades de hasta 115200 bps.

Para la depuración se utilizó los pines nativos y para poder observar la trama, contamos con otro puerto de comunicación generado.

Para el armado de la trama, usamos una técnica de detección de errores de comunicación, el Checksum. Es un sistema de almacenamiento que se envía y/o se recibe junto a los datos. Al realizar el tratamiento se lo calcula nuevamente, y se comprueba con el checksum disponible la integridad de los datos recibidos. Existen múltiples funciones de chequeo, con diferentes tasas de detección de fallos y requisitos de cálculo. En esta oportunidad utilizamos el ChecksumFletcher16, un algoritmo de comprobación desarrollado por John G. Fletcher (1934–2012) en el Lawrence Livermore Labs en la década de 1970. El Checksum Fletcher requiere al menos 16 bits y está formado por la concatenación de dos sumatorios. El primer sumatorio es simplemente la suma de los bytes

recibidos (con complemento a 2). El segundo sumatorio es la suma del primero en cada uno de los pasos. Además, este sistema es capaz de detectar intercambios completos de fragmentos de código. El motivo es que el segundo sumatorio acumula cada fragmento una vez que aparece en la secuencia, hasta el final de la misma, a través del primero de estos. Si un fragmento intercambia la posición en los datos, el segundo sumatorio es capaz de detectar el error, aun cuando el primero sea idéntico. Por otro lado, cuanto más largo sea el checksum la fiabilidad es mayor, aunque aumentan las necesidades de comunicación/almacenamiento adicionales para guardarlo. Finalmente, ningún sistema es infalible. Aunque se envíe el mismo mensaje 100 veces (lo que sería una especie de "checksum" enorme e ineficiente), no se podría garantizar que no ocurra el mismo error en las 100 tandas de transmisión. Como casi siempre, es cuestión de llegar a un compromiso entre fiabilidad, longitud, y carga de cálculo [1]. Se realizaron pruebas locales de las tramas que envió la placa y los resultados fueron los esperados.

### 2.1.3. Comunicación RS-485

#### A. Tendido del cable a través del campo

La tercera etapa comenzó con el tendido de cable de comunicación a través de bandejas, donde se tomaron en cuenta las medidas de seguridad adecuadas para el trabajo en altura (superior a los dos metros), desde la placa hasta la PC. Para estas tareas se utilizaron andamios, escaleras, plataformas, vehículos, entre otros. A continuación se enumeran las fases previas al trabajo realizado en altura:

- se identifica el riesgo de caída;
- se hace un control del riesgo;
- Se utiliza un sistema anti caídas (Arnes) en cada punto de las bandejas donde se elevó. Esto debido a la cercanía de cables de potencia;
- se comprueba el estado de la escalera a utilizar (pies antideslizamiento, trabas);
- se usan elementos de protección personal (Anteojos, guantes).
- se señala debidamente con conos baliza la ubicación de la escalera.

El cableado se ubicó en la mejor posición, sobre la bandeja, para estar lo más alejado de cables de potencia que circulan a la par. Se pensó como opción, en el caso crítico, que si estos afectaban mucho a la trama se podría colocar un divisor metálico para apantallar las perturbaciones, pero como los resultados fueron satisfactorios se desistió por esta opción.



## B. Propuestas de circuitos o esquemas

### I. Pruebas, experiencias

Estas placas conversoras TTL a RS485 se diseñaron aplicando el integrado sn75176a (Differential Bus Transceiver), siendo el circuito a utilizar el propuesto por el fabricante.

El transceptor de bus diferencial SN75176A es un circuito integrado monolítico diseñado para la comunicación bidireccional de datos en líneas de transmisión de bus multipunto. Está diseñado para líneas de transmisión balanceadas y cumple con el estándar ANSI EIA / TIA-422-B y la Recomendación V.11 de la UIT.

El SN75176A combina un controlador de línea diferencial de 3 estados y un receptor de línea de entrada diferencial, que funcionan desde una sola fuente de alimentación de 5 V. El controlador y el receptor tienen habilitaciones active-high y active-low, respectivamente, que pueden conectarse externamente para funcionar como control de dirección. Las salidas diferenciales del controlador y las entradas diferenciales del receptor están conectadas internamente para formar puertos de bus de entrada / salida diferencial (E / S) diseñados para ofrecer una carga mínima al bus cuando el controlador está desactivado o  $VCC = 0$ . Estos puertos cuentan con amplios rangos de voltaje de modo común positivo y negativo que hacen que el dispositivo sea adecuado para aplicaciones de línea compartida [2].

En la etapa de prueba de la transmisión como así también la de recepción, se trabajó con jumpers para agregar o quitar las resistencias terminales. La línea debía terminarse en ambos extremos en su impedancia característica ( $R_T = Z_0$ ). También colocamos resistencia de Bias para una polarización externa a prueba de fallas y garantizar así +200 mV en el puerto A-B. Destacamos que la alimentación de la placa es de 5 VDC.

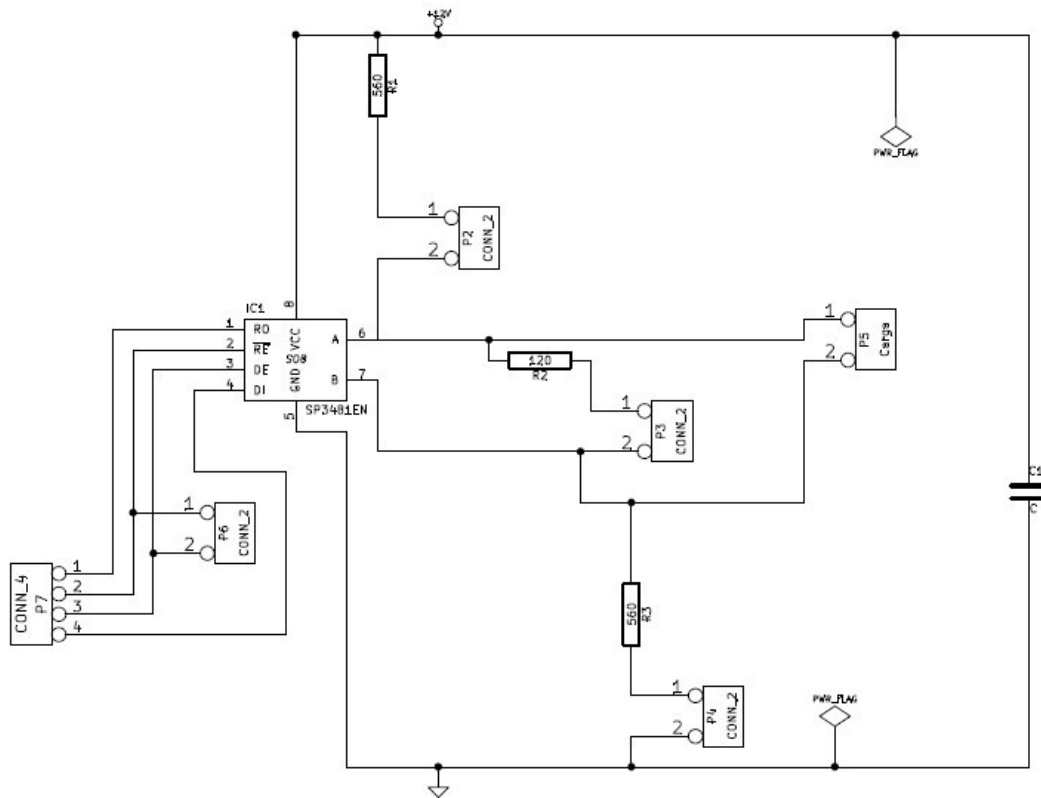


Figura 13 Circuito transmisión y recepción RS-485

Nuestra placa transmisora ubicada en la caldera, es sustentada por la misma fuente que provee a la placa adquisidora de datos. Por otro lado, la placa receptora colocada en la oficina técnica, recibe alimentación de una fuente switching 5VDC 2A tipo cargador. Para finalizar, se acondicionaron los niveles con una placa convertidora de TTL a RS232 db9 y poder entrar a través del adaptador Manhattan por el Usb de la PC de escritorio. Las placas utilizadas para la etapa de prueba tienen la posibilidad de convertirse en transmisor o receptor, y contamos además con el agregado de las resistencias terminales y de Bias, para poder observar su comportamiento.

Para la colocación de la placa receptora RS-485 a TTL, la conversión de TTL a RS-232 y la ficha db9 hembra preparada usamos una caja plástica, y desde allí se pudo conectar exteriormente al convertor Manhattan RS-232 a USB. Sumamos a esto un Jack para Ficha redonda (Diámetro externo: 5.5mm - Diámetro Interno: 2.1mm) donde se conectó la fuente 5VDC.

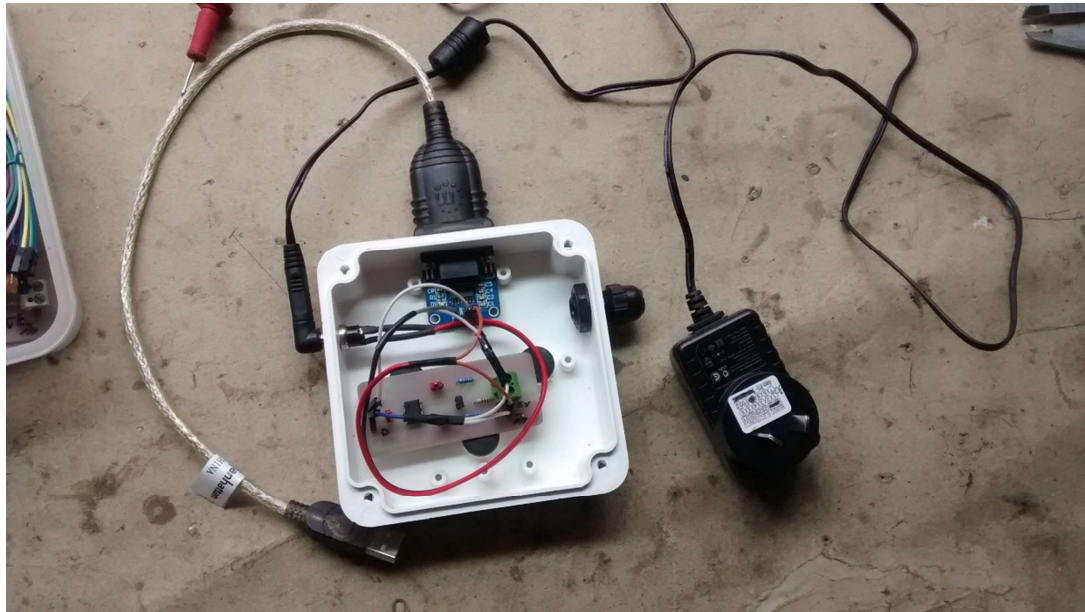


Figura 14 Tablerito de prueba de recepción de datos - PC escritorio

Las placas finales son de pertinax, simple faz, máscara antisoldante y serigrafía frontal.

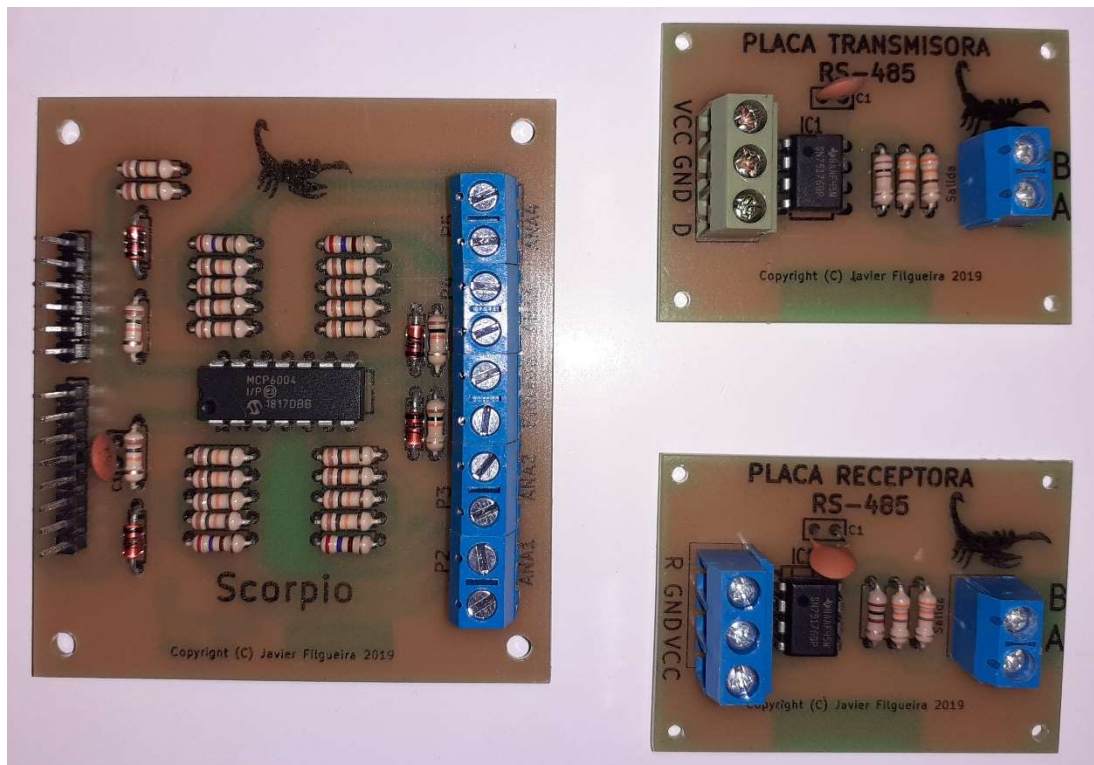


Figura 15 Placas finales - frontal

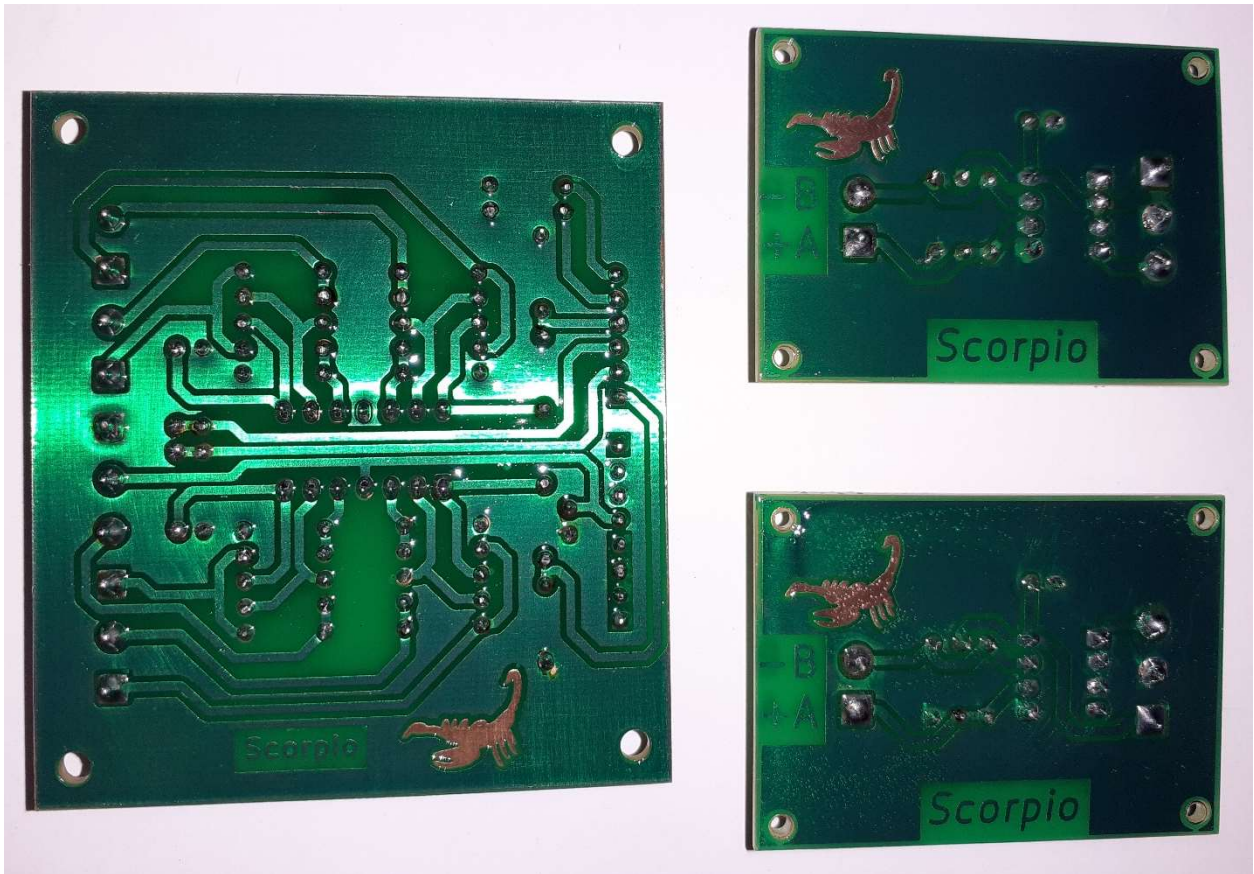


Figura 16 Placas finales - debajo

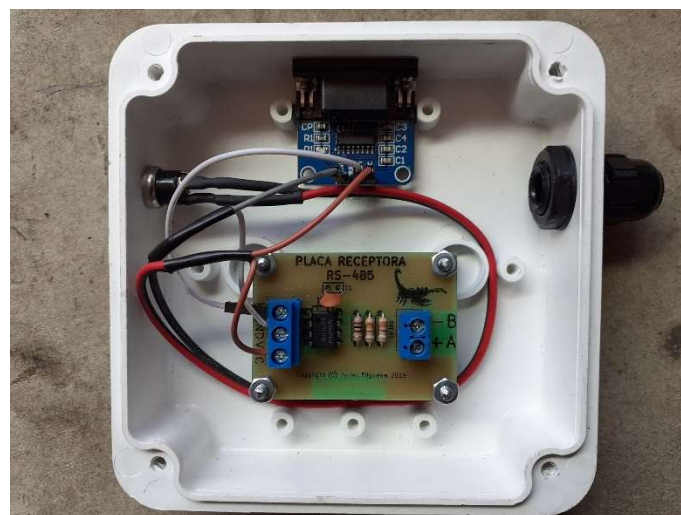


Figura 17 Tablerito final de recepción de datos - PC escritorio

## C. Análisis del funcionamiento

### I. Etapas de conversión de datos

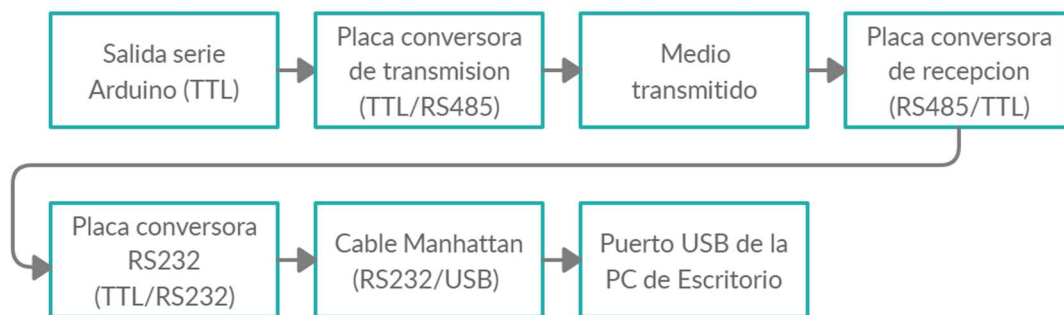


Figura 18 Etapas de conversión de Datos

Prácticamente todas las placas Arduino disponen al menos de una unidad UART. La placa Arduino UNO dispone de una unidad UART que opera a nivel TTL 0V / 5V.

La UART (universally asynchronous receiver/transmitter) es una unidad que incorporan ciertos procesadores, encargada de realizar la conversión de los datos a una secuencia de bits y transmitirlos o recibirlos a una velocidad determinada.

El término TTL (transistor-transistor logic) significa que la comunicación se realiza mediante variaciones en la señal entre 0V y Vcc (donde Vcc suele ser 3.3V o 5V). Por el contrario, otros sistemas de transmisión emplean variaciones de voltaje de -Vcc a +Vcc.

Aunque lo normal dentro del mundo de la electrónica digital es utilizar tensiones dentro del rango de 0 a 5 voltios, la conexión RS-232-C define su propio entorno eléctrico, el cual utiliza el rango -15 a +15 voltios.

En la figura 18 se observan las etapas de conversión de la información, desde la salida Serie Arduino, pasando por las distintas fases, hasta su ingreso a la Pc de escritorio.

La información en forma de secuencia de bits sale de nuestra placa Arduino con niveles de tensión TTL y luego se transforma al protocolo RS485 por la cualidad que posee de alcanzar grandes distancias. La distancia de la placa a la PC de escritorio es de aproximadamente 80 metros a través de bandejas de cables de potencia. Junto a la PC de escritorio se encuentra la carcasa que contiene el receptor RS485 que toma la trama y la convierte nuevamente a niveles de tensión TTL, para luego volver a reconvertirlo al protocolo RS232 que es con el que trabaja nuestro cable de comunicación Manhattan. Por último este realiza la conversión RS232 a USB para ingresar por el puerto de la Pc.

### El Protocolo RS 232

La conexión serie RS-232-C utiliza varias líneas para realizar la comunicación. Unas líneas son entradas y otras son salidas. No existen líneas bidireccionales, por tanto las conexiones entre las líneas siempre han de ir de una entrada a una salida.

Las líneas que son entradas tienen una tensión prácticamente cero con respecto a la puesta a tierra, aunque no todas las líneas que tengan tensión cero son entradas, ya que las líneas

sin conectar cuentan con la misma tensión. Las líneas de salida pueden ser de dos tipos, bien de transmisión o bien de control de la comunicación. Ambas se caracterizan por tener tensión diferente de cero.

La línea de transmisión está normalmente a tensión negativa, cuando no transmite, en cambio las líneas de control pueden estar tanto a tensión negativa como positiva.

Una característica muy tranquilizadora de la conexión serie RS-232-C es que cualquier línea puede soportar una tensión comprendida entre -15v y +15v sin dañarse, por tanto aunque conectemos líneas equivocadamente, o se produzca algún corto en el conector, los equipos no deben estropearse. Esto las protege frente cortocircuitos en la línea y permite probar diferentes tipos de conexiones con absoluta tranquilidad.

Existen dos tipos de lógica posibles, la lógica positiva y la lógica negativa. La lógica positiva es la que asigna al valor lógico "1" un voltaje mayor que al que asigna al valor lógico "0" y la lógica negativa es la que asigna los valores a la inversa. Este tipo de lógica, más compleja, es muy utilizada, ya que presenta otro tipo de ventajas a nivel de diseño de circuitos electrónicos. La lógica usada por la norma RS-232-C es la lógica negativa, es decir, que al bit "1" le asigna la tensión de -15 voltios y al bit "0" le asigna la tensión de +15 voltios.

Las asignaciones de tensiones no son estrictas, sino que pueden variar dentro de un amplio margen sin que se produzcan errores. La salida serie para el bit "0" puede tomar un valor entre +5 y +15 sin que haya problemas. De hecho, casi ningún dispositivo utiliza los +15, sino que lo normal es usar +12 o incluso +8. La salida para el bit "1" ha de estar entre -5 y -15 voltios. Cuando el voltaje está entre -5 y +5 voltios, resulta imposible determinar el valor lógico que se ha enviado.

Las entradas de una conexión serie RS-232-C son aún menos estrictas, ya que el receptor entiende que ha llegado un "0" cuando el voltaje está entre +3 y +15 voltios y entiende que ha llegado un "1" cuando el voltaje está entre -3 y -15 voltios. Cuando llega un voltaje entre -3 y +3 voltios el valor lógico que el receptor entenderá es impredecible.

Así pues, la norma de conexión RS-232-C es bastante tolerante frente a variaciones de voltaje y ello es debido a la elección del rango de tensiones de -15 a +15 voltios, en vez del habitual de +5 a 0 voltios, que habría dado menos margen de tolerancia.

### Margen de ruido

Se conoce como margen de ruido a la amplitud máxima de la perturbación que puede producirse en la salida de una señal sin que afecte en la entrada del siguiente circuito. Para medir este parámetro se toma el caso menos favorable, tal es así, que en la conexión serie RS-232-C, se da cuando la salida está emitiendo con el valor más crítico, que es 5 voltios (un "0"). Como la entrada lee un "0" hasta 3 voltios, entonces el margen de ruido para esta conexión es de 2 voltios. Esto quiere decir que es inmune a ruidos de 2 voltios, o incluso mayores en los casos normales de utilización. Esta característica es extremadamente valiosa cuando los cables han de pasar cerca de dispositivos que generan interferencias eléctricas: líneas de alta tensión, motores eléctricos, alumbrado fluorescente etc.

Además, el margen de ruido también da seguridad frente a caídas de voltaje por la resistencia óhmica del cable, aunque éstas en general no suelen ser significativas.

## La longitud de la línea

La longitud de la línea tiene influencia en las señales eléctricas que viajan por ella.

Hay que tener en cuenta que el equivalente eléctrico de un cable es una resistencia y una bobina en serie, más un condensador en paralelo, cuyo valor es proporcional a la longitud del primero.

El valor de la resistencia, que depende sólo de la longitud de la línea, no es el que limita su longitud máxima, sino que es la capacidad y la inductancia, que dependen también de la velocidad de transmisión, las que determinan esta longitud. Esto es debido a que la señal transmitida se puede asemejar a una corriente alterna, cuya frecuencia aumenta al aumentar la velocidad de transmisión. En consecuencia la frecuencia también aumenta la capacidad e inductancia de la línea, dado que estos parámetros dependen de ella, y llegado el momento, la atenuación que experimenta la señal causa errores en los datos que se transmiten. La interfaz RS-232 está diseñada para imprimir documentos en distancias cortas, de hasta 15 metros según la norma, y para velocidades de comunicación bajas, de no más de 20 kbps [3]

## El Protocolo RS 485

Esta interface tiene muchas ventajas con respecto a RS 232, entre las cuales se mencionan:

### a) Bajo costo

Los Circuitos Integrados para transmitir y recibir son baratos y solo requieren una fuente de +5V para poder generar una diferencia mínima de 1.5v entre las salidas diferenciales. En contraste con RS-232 que en algunos casos requiere de fuentes dobles para alimentar algunos circuitos integrados.

### b) Capacidad de interconexión:

RS-485 es una interface multi-enlace con la capacidad de poder tener gran cantidad de transmisores y receptores. Con una alta impedancia receptora, los enlaces con RS-485 pueden llegar a tener a lo máximo hasta 256 nodos.

### c) Longitud de Enlace:

En un enlace RS-485 puede tener hasta 1300 metros de longitud, comparado con RS-232 que tiene como límite típico 15 metros.

### d) Rapidez:

La razón de bits puede ser tan alta como 10 Mega bits/ segundo.

## Balanceo y Desbalanceo de Líneas

La razón por la que RS-485 puede transmitir a largas distancias, es porque utiliza el balanceo de líneas. Cada señal dedica un par de cables, sobre uno de ellos se encontrará un voltaje y en el otro estará su complemento, de esta forma, el receptor responde a la diferencia entre voltajes.

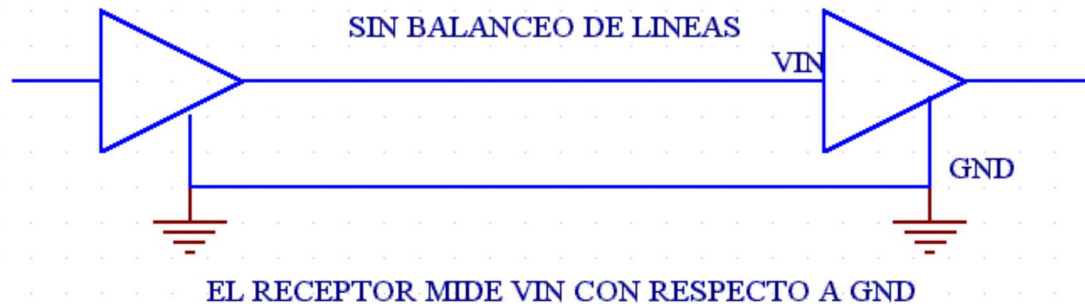


Figura 19 Línea no balanceada [4]

La ventaja de las líneas balanceadas es su inmunidad al ruido

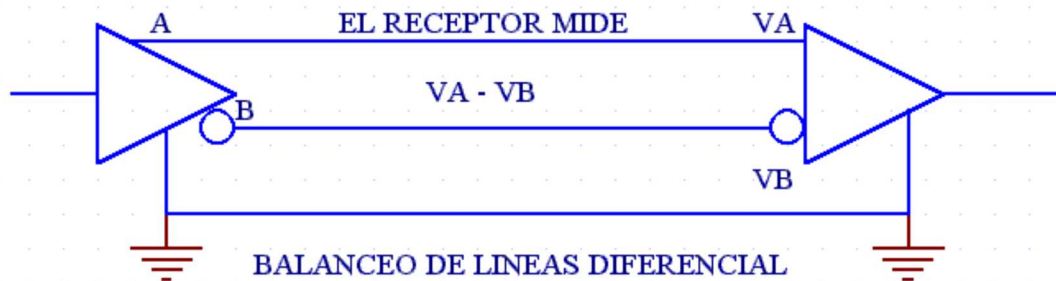


Figura 20 Línea balanceada [4]

En cuanto a las líneas balanceadas (Figura 20) la **TIA/EIA-485** designa a estas dos líneas como **A y B**. En el controlador TX, una entrada alta TTL causa que la línea A sea más positiva (+) que la línea B, mientras que un bajo en lógica TTL causa que la línea B sea más positiva (+) que la línea A. Por otra parte en el controlador de recepción RX, si la entrada A es más positiva que la entrada B, la salida lógica TTL será "1" y si la entrada B es más positiva (+) que la entrada A, la salida lógica TTL será un "0".

### Requerimientos de Voltaje

Las interfaces típicas RS-485 utilizan una fuente de +5 Volts, pero los niveles lógicos de los transmisores y receptores no operan a niveles estándares de +5V o voltajes lógicos CMOS. Para una salida válida, la diferencia entre las salidas A y B debe ser al menos +1.5V. Si la interfase está perfectamente balanceada, las salidas estarán desfasadas igualmente a un medio de la fuente de Voltaje.

En el receptor RS-485, la diferencia de voltaje entre las entradas A y B necesita ser 0.2V. Si A es al menos 0.2V más positiva que B, el receptor ve un 1 lógico y si B es al menos 0.2v más positivo que A, el receptor ve un 0 lógico. Si la diferencia entre A y B es menor a 0.2v, el nivel lógico es indefinido. Si esto ocurre habría un error en la transmisión y recepción de la información.



La diferencia entre los requerimientos del Transmisor y el Receptor pueden tener un margen de ruido de 1.3V. La señal diferencial puede atenuarse o tener picos de largo como de 1.3v, y aun así el receptor vera el nivel lógico correcto. El margen de ruido es menor que el de un enlace RS-232, no hay que olvidar que RS-485 maneja señales diferenciales y que cancela la mayoría del ruido a través de su enlace.

El total de corriente utilizada por un enlace RS-485 puede variar debido a las impedancias de los componentes, incluyendo los Transmisores, Receptores, cables y la terminación de los componentes. Una baja impedancia a la salida del Transmisor y una baja impedancia en los cables, facilitan los cambios de nivel y aseguran que el receptor vea la señal, no importa cuán larga sea la línea de transmisión. Una alta impedancia en el receptor decrementa la corriente en el enlace e incrementa la vida de la fuentes de voltaje.

La terminación en los componentes, cuando se utilizan, tienen un gran efecto sobre la corriente en el enlace. Muchos enlaces con RS-485 tienen una resistencia de 120 ohms a través de las líneas A y B en cada extremo de la línea. Por lo tanto cada enlace tiene dos terminales [4].

#### Cable de comunicación Manhattan

Convierte rápidamente un puerto USB a un puerto Serial/RS232/COM/DB9.

Su chip Prolific PL-2303RA adaptable y compatible lo convierte en la solución ideal para muchos sistemas operativos y plataformas.

Se conecta fácilmente en la mayoría de equipos DB9 como módems, cámaras digitales, impresoras de etiquetas, escáners de códigos de barras, palms, GPS, telescopios, máquinas CNC, bordadoras, sistemas de control automotriz, organizadores digitales, entre otros.

Instalado como un puerto COM/RS-232 estándar para brindar conectividad serial en dispositivos tradicionales y de alta tecnología [5].

#### D. Pruebas de recepción de datos

En las primeras pruebas de recepción de tramas, se pudo ver que la cantidad de datos aportados a través de una comunicación directa con la notebook, en un tiempo de 72 segundos, se correspondía a los valores recibidos sin errores, promedio que utilizaríamos para trazar la gráfica. Luego se observó la cantidad de datos tomados, a través del cable de comunicación tendido por la Pc de escritorio, y se registraron los siguientes datos de ambas tomas:

Comunicación directa = 960 tramas de datos

Comunicación cableada = 930 tramas de datos

Velocidad de transmisión = 9600 Baudios

Trama = [Start][vapor][agua][purgado\_superf][alarma][Checksum\_0][Checksum\_1][Stop]...

### 2.1.4. Interpretación de Datos

#### A. Análisis del funcionamiento

##### I. Secuencia del software

En esta atapa nos focalizamos en la programación del software, cuyas fases describimos a continuación:

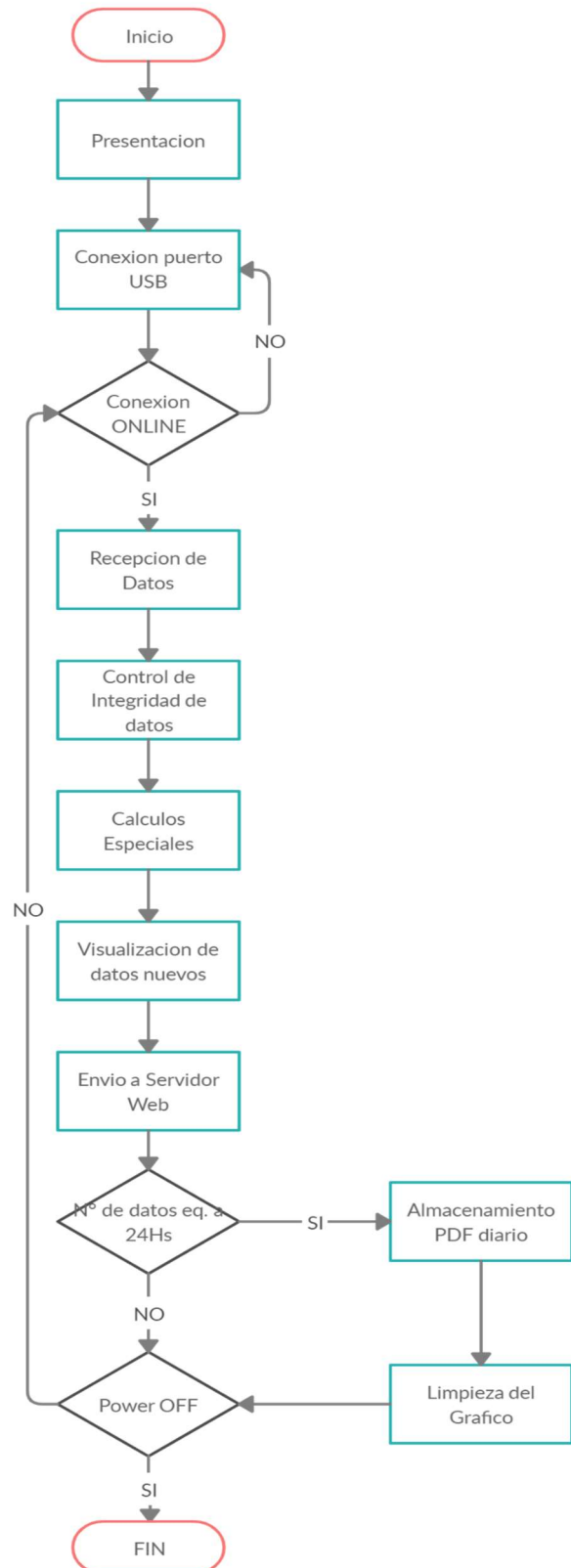


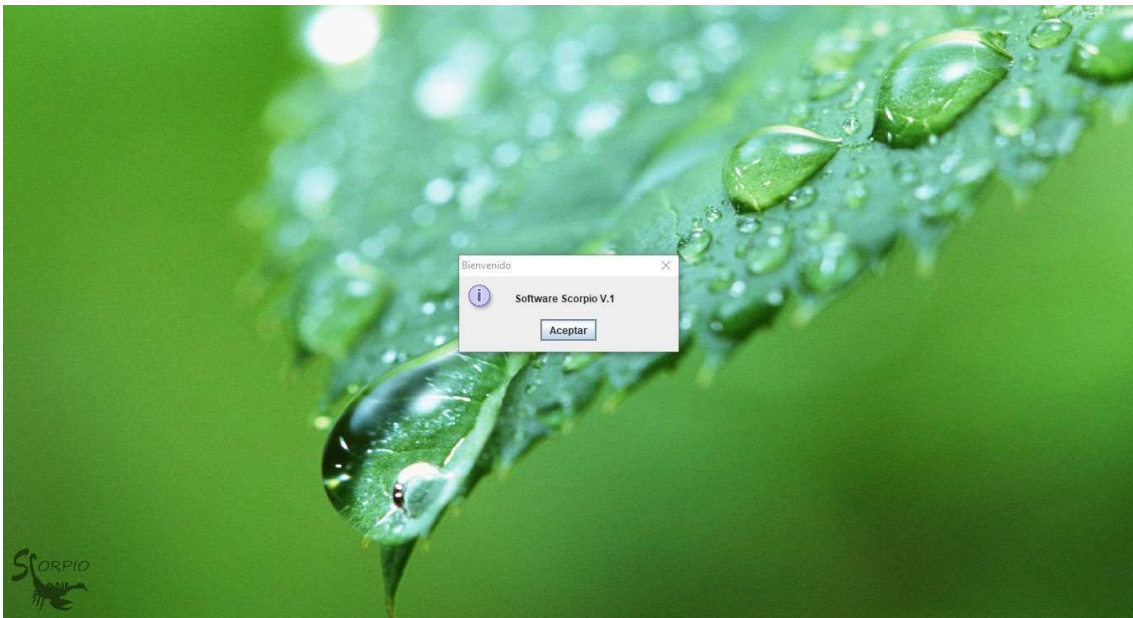
Figura 21 Diagrama de funcionamiento del software

Para la programación se utilizó el software Processing 3.5.3., de adquisición gratuita y código abierto, y ejecutable en las plataformas Mac, Windows y GNU / Linux.

El procesamiento continúa siendo una alternativa a las herramientas de software patentado con licencias restrictivas y costosas, lo que lo hace accesible para las escuelas y los estudiantes individuales. Su estado de código abierto fomenta la participación y colaboración de la comunidad que es vital para el crecimiento de Processing. Los colaboradores comparten programas, contribuyen con códigos y crean bibliotecas, herramientas y modos para ampliar las posibilidades del software. La comunidad de Processing ha escrito más de cien bibliotecas para facilitar la visión por computadora, la visualización de datos, la composición de música, la creación de redes, la exportación de archivos 3D y la programación de la electrónica [20].

## Presentación

En la primera pantalla se da la bienvenida al usuario. El usuario debe presionar el botón de ACEPTAR para continuar.



*Figura 22 Pantalla de bienvenida PC*

## Conexión USB – Recepción de Datos – Integridad de Datos

El software establece una conexión a través del puerto USB con la PC de escritorio, por donde se reciben las tramas de datos que más tarde se procesarán aplicando la técnica checksum para detectar posibles errores. En caso de que la conexión con el puerto se pierda, el software buscará reconectar automáticamente, mientras continúa graficando valores nulos.

En la pantalla se podrá observar el estado de la conexión con las lecturas "ONLINE", o en caso contrario "OFFLINE".

Una vez que la trama pasa el chequeo, los datos serán separados y almacenados en la memoria, para más tarde ser re-escalados según el tipo de variables al que correspondan.

## Visualización

Estos nuevos valores ocuparán un lugar espacial en la pantalla dentro de los rangos de valores, como puntos en la gráfica, por ejemplo el caudal de agua. Las dimensiones de las gráficas varían en altura, según la importancia de los datos a analizar, pero el ancho es común a todas, el que fue establecido en 1200 pixeles o puntos (siendo 1366 pixeles el ancho de la pantalla total). El eje horizontal de las gráficas nos indica el tiempo en que se realizó la medición, de ahí que para representar 24 horas ( $24 \times 60 \times 60 = 86400$  segundos) en 1200 puntos nos corresponde marcar cada punto cada 72 segundos. Todos los datos que se obtengan dentro de este lapso, serán promediados y agregados a la gráfica en su posición correspondiente.

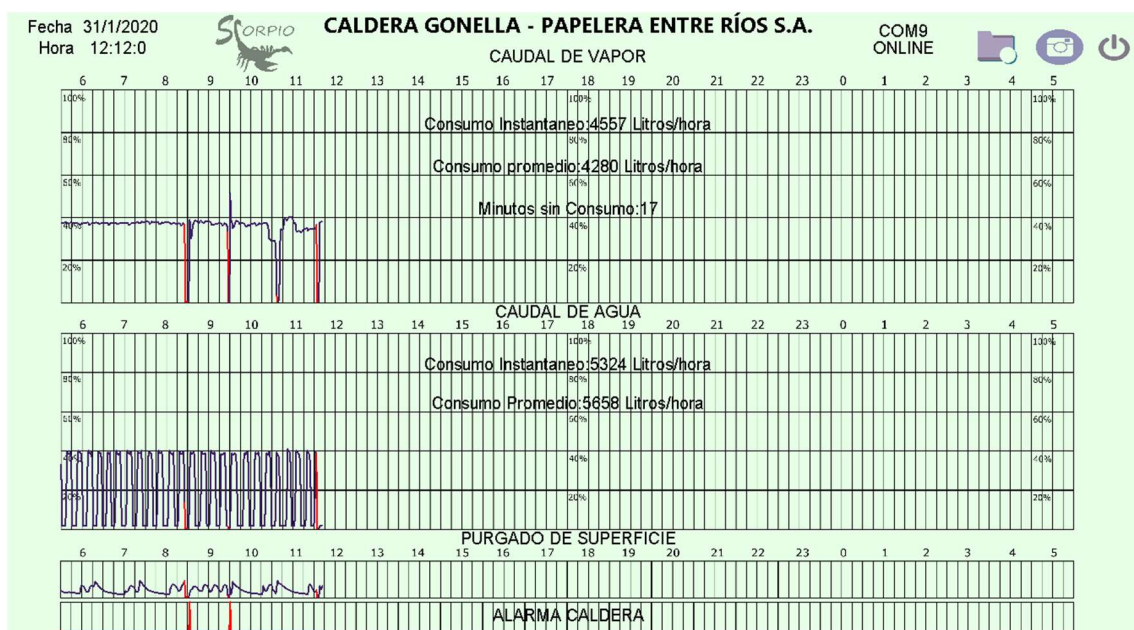


Figura 23 Pantalla de Visualización de Datos PC

Al completar los 1200 pixeles horizontales (equivalente a las 24 hs. cumplidas), los datos son guardados en PDF y eliminados de la gráfica, para poder dar comienzo a un nuevo día. En los extremos de la gráfica se indica el porcentaje de lectura y la hora. La franja horaria revelada inicia a las 6:00 AM y finaliza a las 6:00 AM del otro día. Debemos aclarar que al recibir los valores provenientes de una lectura de tensión analógica, resultante de la conversión de un lazo de corriente (4 a 20mA), si a este valor le corresponde una lectura menor a un volt (equivalente en tensión a valores entre 0 a 4mA => valores erróneos), se producirá un error de lazo, el que será detectado por el software graficando un punto en cero y de color rojo.

## Acondicionamiento de datos

El valor del caudal de vapor que se recibe es escalado de acuerdo a la unidad que vamos a manejar y contrastado con el Graficador circular Honeywell y el controlador UDC 3300, dos elementos de medición instalados desde hace tiempo utilizados como indicadores del comportamiento de la caldera.

En el caso del caudal de agua, los valores que entrega el sensor pueden variar por programación local, según lo decida el usuario. En nuestro caso fue configurado para que el valor máximo sea aproximadamente un 120 % del valor de medición de consumo normal del agua que envían las bombas a la caldera.

Otro valor es el del sensor de purgado de superficie que nos da idea de la concentración de sólidos totales disueltos (TDS) en la caldera.

El valor de la alarma nos dará indicios de errores que se produzcan en el control de la caldera (Burning Control). Es un indicador digital que varía entre cero o uno. El valor uno significa alarma de error activada, y el valor cero, un buen funcionamiento.

Además el sistema muestra mediciones instantáneas y un promedio diario de los datos del Caudal de agua y de Vapor. Estos valores son escalados para que se visualicen en unidades de Litros/hora.

Aclaremos que para medir "Minutos sin consumo" realizamos un cálculo especial, donde se contabilizan los minutos en los que los valores no superan un límite de consumo de vapor. Esta demarcación se fijó observando los valores de consumo en los momentos donde la producción de papel estaba detenida. Estos minutos acumulados indicarían de manera indirecta los lapsos de "No Producción" durante el día.

## II. Problemas y soluciones implementadas

El problema que se tuvo fue que si la conexión USB se perdía por desconexión física del cable Manhattan el programa no volvía a reconectar con el puerto que generaba el conversor. Se investigó alguna técnica y se encontró una forma estrictamente del lenguaje Java para detectar excepciones.

Una excepción en Java es un evento no deseado o inesperado, que ocurre durante la ejecución de un programa, es decir, en tiempo de ejecución, e interrumpe el flujo normal de las instrucciones del programa.

Un error indica un problema grave que una aplicación razonable no podría detectar, en cambio la excepción brinda condiciones para poder hacerlo.

Adaptando esta técnica a mi necesidad, la utilicé para realizar una reconexión, o sea, si en medio de la rutina de conexión se produce un evento inesperado (desconexión del puerto) buscará reconectarse automáticamente.

## 2.1.5. Almacenamiento diario de datos

### A. Análisis del funcionamiento

#### I. Backup de datos continuo

En esta quinta etapa nos ocupamos en primer lugar del backup de datos en el rango horario. Ante la posibilidad de eventuales cortes de energía, y la consecuente pérdida de datos, se realizó una lógica utilizando archivos de texto (.txt), es decir, un archivo informático que contiene texto formado únicamente por caracteres que son legibles por las personas, careciendo de cualquier tipo de formato tipográfico y su destacado tamaño ligero. El texto almacenará datos como día, mes, número de ubicación temporal (0 a 1200), los datos de caudal de vapor, agua, purgado de superficie y alarma, y se ejecutará de manera automática cada 15 minutos o cuando activemos el icono de power off (cerrado del programa).



```

base: Bloc de notas
Archivo Edición Formato Ver Ayuda
-1,31,1;;
0,425.0,148.0,0.0,0.0;;
1,425.0,148.0,0.0,0.0;;
2,425.0,148.0,0.0,0.0;;
3,425.0,148.0,0.0,0.0;;
4,425.0,148.0,0.0,0.0;;
5,425.0,148.0,0.0,0.0;;
6,425.0,148.0,0.0,0.0;;
7,425.0,148.0,0.0,0.0;;
8,425.0,148.0,0.0,0.0;;
9,425.0,148.0,0.0,0.0;;
10,425.0,148.0,0.0,0.0;;
11,425.0,148.0,0.0,0.0;;
12,425.0,148.0,0.0,0.0;;
13,425.0,148.0,0.0,0.0;;
Windows (CRLF) Línea 14, columna 25

```

Figura 24 Archivo Backup

Al iniciar nuestro programa se observará el archivo de backup, y si los datos son válidos dentro de los campos a verificar como el día, el mes y número de ubicación temporal, estos se agregarán al gráfico correspondiente, para luego recibir nuevos datos a través del puerto serie.

Observando el gráfico podemos distinguir la línea -1,31,1;;

En ella las comas son separadores de los datos, el punto y coma indica fin de trama o línea. El -1 nos indica que en esta línea está la fecha de los datos relevados (-1, día, mes;;).

Las siguientes líneas corresponden a:

0,425.0,148.0,0.0,0.0;;

13,425.0,148.0,0.0,0.0;;

Ubicación temporal (0 a 1200), caudal de vapor, caudal de agua, purgado de superficie y alarma.

## II. Guardado de graficas en archivos PDF

Luego se trabajó en el guardado automático del rango horario, lógica de nombrado y almacenamiento de los archivos PDF. Al completar los 1200 pixeles horizontales (equivalente a las 24 hs cumplidas), los datos son guardados para la llegada de nueva información. Los gráficos se limpian y se comienza nuevamente. La franja horaria relevada comienza a las 6:00 AM y termina a las 6:00 AM del otro día. Una vez que las gráficas están completas se guardan en un formato de documento portátil (PDF), formato utilizado para presentar e intercambiar documentos de manera fiable, independientemente del software, el hardware o el sistema operativo. El PDF es actualmente un estándar abierto, reconocido por la Organización Internacional para la Estandarización (ISO). Los documentos PDF pueden contener vínculos y botones, campos de formulario, audio, vídeo y lógica empresarial. Pueden ser firmados de manera electrónica, y se pueden ver con facilidad en Windows o Mac OS utilizando el software gratuito Acrobat Reader DC. [6]

Los archivos siguen las siguientes reglas para la lógica de nombrado y enrutamiento:

**(Ruta software)\Almacenamiento\"Año\" \"Dia\"-\"Mes\"-**

**\"Año\"\_\"Hora\"hs\"Minutos\"min\_datos.pdf**

**Ruta software: ruta donde se ubica el ejecutable del programa.**

**Por ejemplo: C:\CalderaGonella\Software**

**Año (actual)**

**Día (actual)**

**Mes (actual)**

**Hora (actual)**

**Minutos (actual)**

**Todas estas variables darán nombre al archivo en cuestión.**

**Por ejemplo C:\CalderaGonella\Software\Almacenamiento\2020\26-1-**

**2020\_16hs20min\_datos.pdf**

torizacion de variables\_Caldera > Software proyecto final > Depuracion de comunicacion y grafica > Ensambleoctavo > Almacenamiento > 2020

Nombre	Fecha de modifica...	Tipo	Tamaño
19-1-2020_19hs48min_datos	19/1/2020 19:48	Adobe Acrobat D...	306 KB
19-1-2020_19hs56min_datos	19/1/2020 19:56	Adobe Acrobat D...	306 KB
19-1-2020_19hs57min_datos	19/1/2020 19:57	Adobe Acrobat D...	306 KB
19-1-2020_19hs58min_datos	19/1/2020 19:58	Adobe Acrobat D...	307 KB
19-1-2020_19hs59min_datos	19/1/2020 19:59	Adobe Acrobat D...	306 KB
19-1-2020_20hs0min_datos	19/1/2020 20:00	Adobe Acrobat D...	306 KB
27-1-2020_22hs28min_datos	27/1/2020 22:28	Adobe Acrobat D...	306 KB
27-1-2020_22hs33min_datos	27/1/2020 22:33	Adobe Acrobat D...	306 KB
27-1-2020_22hs34min_datos	27/1/2020 22:34	Adobe Acrobat D...	306 KB
27-1-2020_22hs37min_datos	27/1/2020 22:37	Adobe Acrobat D...	306 KB
27-1-2020_22hs42min_datos	27/1/2020 22:42	Adobe Acrobat D...	306 KB
27-1-2020_22hs44min_datos	27/1/2020 22:44	Adobe Acrobat D...	305 KB
27-1-2020_22hs48min_datos	27/1/2020 22:48	Adobe Acrobat D...	306 KB
28-1-2020_19hs24min_datos	28/1/2020 19:24	Adobe Acrobat D...	303 KB

Figura 25 Almacenamiento en archivos PDF

Para obtener una impresión instantánea de la pantalla ubicamos un ícono con esta función, que permitirá contar con un archivo PDF antes de que se cumpla la franja horaria donde se hace el almacenamiento automático.

Conjuntamente, se colocó otro ícono que proporciona un rápido y fácil acceso a los archivos ya guardados, sin necesidad de conocer la ruta que nos lleve a su ubicación. Al presionar el botón el programa nos conducirá directamente a la carpeta donde se encuentran almacenados.

El botón de power off, primero actualizará el archivo de backup, seguidamente cerrará la comunicación con la placa adquisidora de datos y finalizará la interface gráfica de visualización de datos.

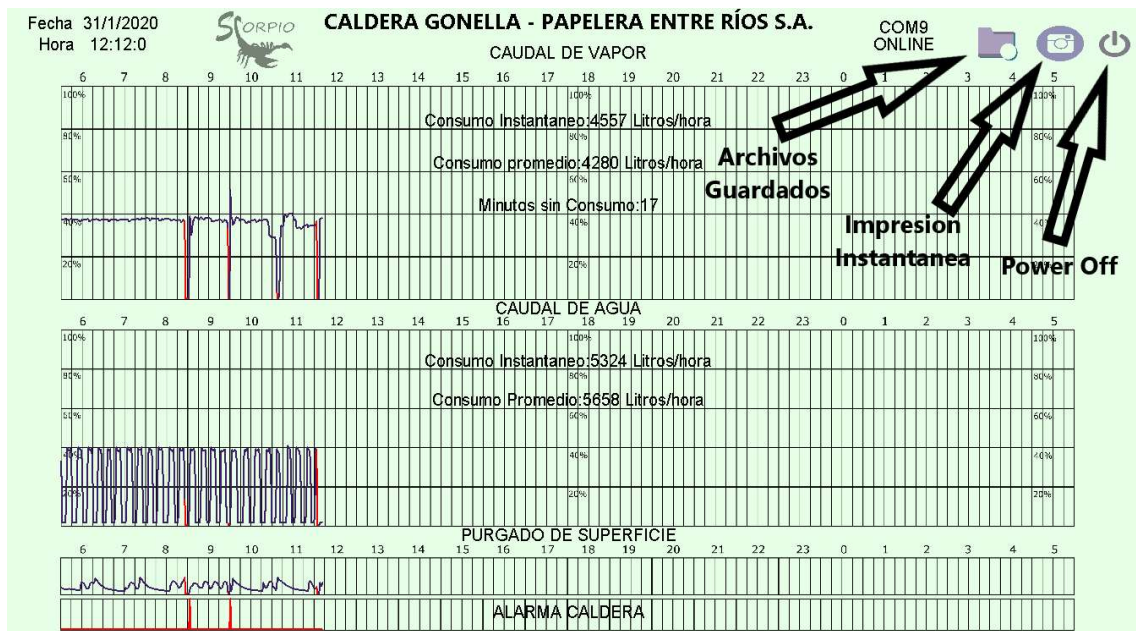


Figura 26 Botones de función



## **B. Desarrollo técnica Backup - Detalle de utilización de archivos .txt**

Al crear o abrir un archivo creado, este debe permanecer abierto para poder cargarlo con información, si el programa se apaga inesperadamente y ese archivo no se cerró por línea de programa todo el contenido será perdido. Ante esto se pensó en la siguiente práctica: La técnica utiliza dos archivos de texto que trabajan de la siguiente manera.

El archivo base.txt es el que guarda los datos de forma esporádica.

El archivo flash.txt es el que guarda los datos del anterior y los siguientes datos entrantes hasta que se actualiza el archivo base.txt con estos datos.

Al iniciar nuestro programa se abrirán ambos archivos, se leerá el archivo base.txt, y si los datos son válidos dentro de los campos a verificar como el día, el mes y número de ubicación temporal, estos se agregarán al gráfico correspondiente, para luego recibir nuevos datos a través del puerto serie.

El contenido del archivo base.txt se colocará dentro del archivo flash.txt (inicialmente vacío). El archivo base.txt se cerrará y dejará abierto el archivo flash.txt para ir cargando los nuevos datos.

Cada 15 minutos se realiza la siguiente rutina:

Se cierra y abre el archivo flash.txt, de esta manera queda seguro su contenido.

Continuamente se abre el archivo base.txt y se guarda dentro el contenido del flash.txt. Por último se cierra el archivo base.txt.

De esta manera el máximo lapso que se puede perder es de 15 minutos. Se decidió ese tiempo para no sobrecargar la memoria de la pc que corre el software.

Queda por mencionar la rutina que lleva a cabo antes de cerrar el programa principal (presionando botón power off):

Se cierra y abre el archivo flash.txt, de esta manera queda seguro su contenido.

Luego se abre el archivo base.txt y se graba dentro el contenido del flash.txt. Se cierra el archivo base.txt. Se deja en blanco el contenido de flash.txt y el archivo se cierra.

### **2.1.6. Comunicación con Servidor Web externo**

#### **A. Investigación de plataformas free**

##### **I. Análisis y selección**

En esta última etapa nos ocupamos de la carga de datos en el servidor web externo. En un primer momento se pensó en utilizar el portal web dweet.io que nos ofrecía un servicio de envío y representación de datos en la nube sin la necesidad de registrarnos en la plataforma, sin embargo las gráficas que realizaba eran demasiado básicas y la respuesta temporal del flujo de datos no coincidían con los resultados que deseábamos obtener.

Otra opción era Freeboard.io, un portal que nos daba una mejor presentación de los datos, pero continuaba sin cumplir con nuestras expectativas.

Por último nos decidimos por el servidor Web [www.Thingspeak.com](http://www.Thingspeak.com) que entre otras opciones nos permitía agregar, visualizar y analizar flujos de datos de forma directa en la nube. Al mismo tiempo contaba con características claves como, configurar fácilmente los dispositivos y de esta manera enviar datos al portal utilizando los protocolos de IO más populares; la visualización de datos de los sensores en tiempo real; el agregado de los datos bajo demanda de fuentes de terceros; la utilización de la potencia de MATLAB para proveer de sentido a los datos de IO; la ejecución de análisis de IO de manera automática en base a horarios y eventos. Además, admitía el armado de prototipos y la construcción de sistemas de IO sin necesidad de crear servidores o desarrollar software web.

Debemos aclarar que la Internet de los objetos (IO) es un sistema de “cosas conectadas”, que generalmente comprenden un sistema operativo integrado y la capacidad de comunicarse con Internet o con las “cosas vecinas”. Uno de los elementos claves de estos sistemas genéricos es que tienden un puente entre las diversas “cosas”, y estas deben tener la capacidad de conectarse con otras, de lo contrario carecen de funcionalidad. El poder real de la IO aparece cuando estos objetos se conectan a un servicio de manera directa o a través de otras “cosas”. En estos sistemas, el servicio desempeña el papel de un administrador invisible que les proporciona capacidades que van desde una simple recopilación y supervisión de datos hasta sus complejos análisis.

Este servidor procede automáticamente sobre nuestros datos y se comunica utilizando servicios de terceros como Twilio o Twitter. El elemento central es un “Canal ThingSpeak”, el que almacena los datos que enviamos a ThingSpeak y está compuesto de los siguientes elementos:

- 8 campos de almacenamiento para datos de cualquier tipo, ya sea de un sensor o de un dispositivo integrado;
- 3 campos de ubicación que pueden ser utilizados para almacenar latitud, longitud y elevación. Estos son muy útiles para rastrear un dispositivo en movimiento.
- 1 campo de estado (un mensaje corto para describir los datos almacenados en el canal).

Para poder utilizarlo necesitamos registrarnos y crear un canal. Una vez que completamos este paso, podemos enviar los datos, permitir que ThingSpeak los procese y también recuperarlos. Esta opción es la que mejor se ajusta a lo requerido.

Al concluir esta etapa, y teniendo en cuenta que se podía acceder de manera simple desde cualquier ordenador a través del explorador de internet, se pensó en crear una aplicación que permita una visualización eficaz, rápida y segura de los datos a través de un celular. Al tener en cuenta que un alto porcentaje de móviles utilizan el sistema Android, trabajamos en una aplicación bajo esa plataforma que mejore la visualización de los datos. Para esta tarea utilizamos Virtuino, una app en su versión gratuita para Android que si bien presenta limitaciones, permite desarrollar proyectos y trabajar vía WiFi, Bluetooth, Ethernet o SMS. Esta App crea una interfaz gráfica personalizada que controla LEDs, interruptores, reguladores, etc., y trabaja con Arduino y ESP8266 entre otras opciones. También puede levantar los datos enviados al servidor [thingspeak.com](http://thingspeak.com) y mostrar gráficos de forma muy intuitiva y elegante [7].

## B. Desarrollo del software a medida

### I. Diseño del modo y la forma de la visualización de los datos



*Figura 27 Logo APP de Diseño*

Para la tarea de edición de la app Android, o sea, el armado de las pestañas, los títulos, las escalas, las gráficas, colores y fuentes, entre otros, recurrimos a la app Virtuíno de edición. Conjuntamente empleamos el CHANNEL ID y la API KEY de solo lectura, esenciales para poder acceder a los datos ubicados en nuestro servidor web.

Hoy por hoy, muchas aplicaciones hacen uso de APIs para comunicarse internamente, otras incluso tienen porciones de sus APIs accesibles al público. API es el acrónimo para Application Programming Interface, en su traducción al español Interfaz de Programación de Aplicaciones. Esta proporciona un medio para acceder a una funcionalidad de determinada aplicación, que usualmente se encuentra dentro de la creación, lectura, actualización y destrucción de contenido -CRUD- aunque esto puede variar. Comúnmente las APIs públicas solamente permiten lectura, es decir, tipos de peticiones GET. Hicimos uso del protocolo HTTP ideado por Tim Berners Lee en el año 1989. Este protocolo ha tenido varias interpretaciones a las que se le han ido anexando distintas funciones. La versión más actual de HTTP es HTTP/2, creada originalmente por Google y soportada por la mayoría de navegadores actuales. HTTP define varios métodos, por ejemplo en su versión 1.0 existían GET, POST y HEAD, pero más tarde en la versión 1.1 fueron implementados los métodos OPTIONS, PUT, DELETE, TRACE y CONNECT. Si bien la mayor parte de la sintaxis como métodos, headers, códigos de estado y URIs se mantienen igual, la manera en la cual la información es encapsulada y transportada entre cliente y servidor ha cambiado [8].

Para nuestro proyecto fueron empleadas las Peticiones GET.

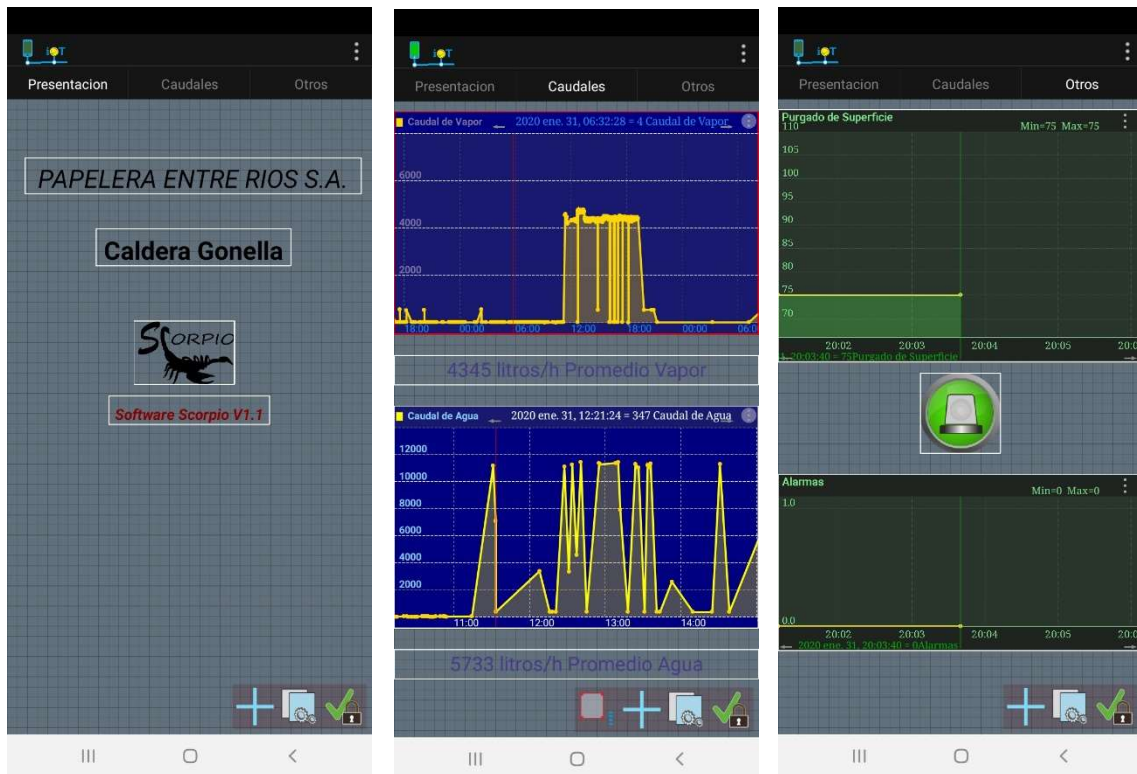


Figura 28 Edición de Pantallas de visualización Android

Una vez que finaliza la edición se guarda el proyecto en un archivo con extensión .vrt



Figura 29 Logo APP de Visualización

Finalmente con la app Virtuino Viewer cargamos el archivo creado y ya queda listo para usar. Esto nos es útil para que el usuario no tenga acceso a realizar ninguna modificación crítica sobre la aplicación ya que no da la opción de hacerlo. Podrá visualizar datos actuales y en vivo, o cargar datos de otro lapso de tiempo en el que se guardaron los mismos. También se puede observar todos los puntos de la gráfica o promediar por lapsos de tiempo, y de esta forma, no sobrecargar el grafico.

Proyecto cargado en Virtuino Viewer:

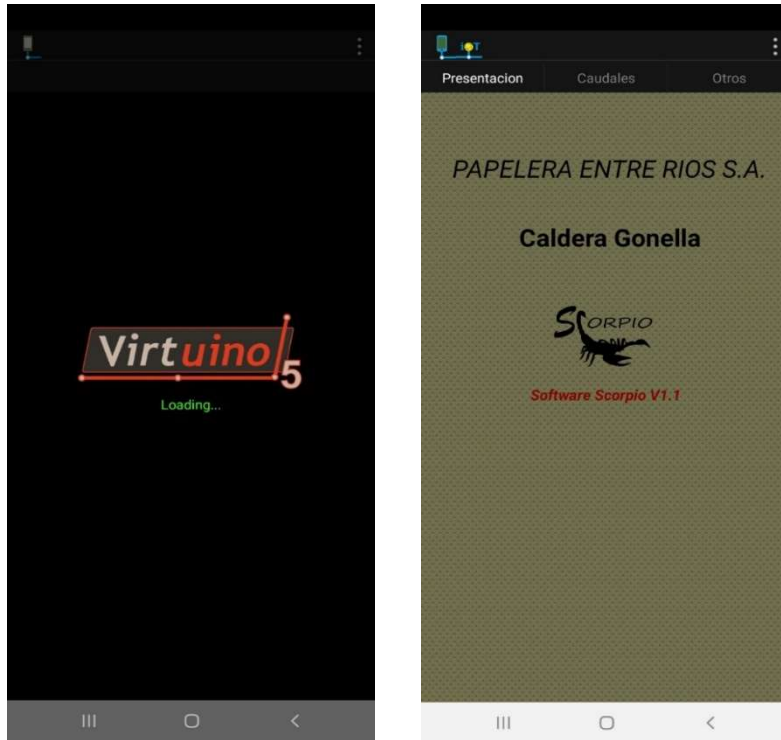


Figura 30 Pantalla Inicial

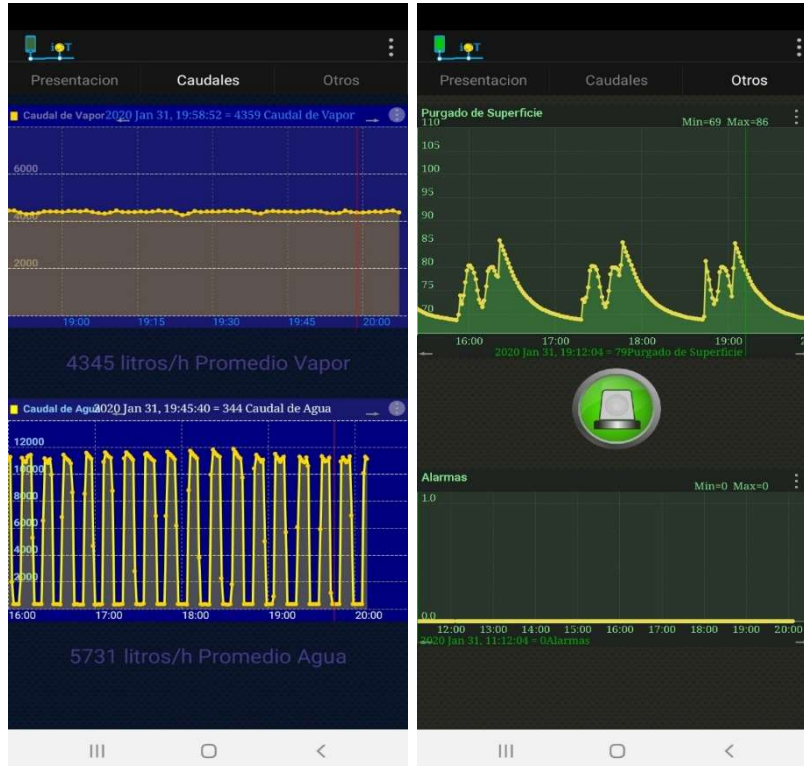


Figura 31 Pantallas Caudales y Otros

## C. Análisis de la aplicación obtenida

La respuesta de la aplicación realizada tuvo muy buena aceptación, los datos se refrescan de forma fluida.

El contraste de las gráficas temporales ayuda mucho para la lectura de los valores. La interacción que se puede realizar a través de la pantalla táctil tanto para realizar un zoom (que es el efecto de acercamiento o alejamiento de la gráfica) o para arrastrar temporalmente la gráfica y visualizar diferentes rangos de tiempos, es lo que le da un dinamismo y una capacidad de plasmar los datos de forma amena para el usuario.

En general la aplicación es muy amigable, confiable, predecible y simple en su utilización.

## 2.2 Diseño Completo

### 2.2.1 Diseño final

En el diseño del Hardware como así también, el Software, se tuvieron en cuenta los factores que podrían influir directa o indirectamente sobre su desempeño, tanto en su ubicación, como la temperatura, polución, protección mecánica, fiabilidad, seguridad, eficiencia, factores ambientales, organizativos, de trabajo, entre otros.

El Hardware es simple y se comportó de manera fiable. Se realizaron leves correcciones de las lecturas para su trazado. El caudal de vapor se contrastó con el Graficador circular Honeywell y el controlador UDC 3300, dos elementos de medición instalados desde hace tiempo utilizados como indicadores del comportamiento de la caldera.

También se redefinieron los valores enviados al servidor WEB. La APP Android se desarrolló perfectamente con la cantidad de observadores que se conectaron (tres), cada uno utilizando un dispositivo móvil desde donde ingresaron con la llave electrónica correspondiente.

### 2.2.2 Prestaciones

La placa adquisidora de datos se desempeñó de forma fiable y no se mostró perjudicada por el ámbito industrial.

La transmisión de datos por RS485 es muy buena, hay muy poca pérdida de paquetes de datos (aproximadamente un 3,2%).

El algoritmo desarrollado tiene la particularidad que puede ser fácilmente exportado a distintas plataformas de software (Windows, iOS o Linux) manteniendo iguales parámetros y características de funcionamiento.

El software está en una Pc de escritorio fija, y el mismo correrá las 24hs del día. El paquete de archivos que se le provee a la empresa posee lo indispensable para su funcionamiento en un sistema operativo Windows 10. Solo se deberá tener en cuenta que el programa debe

ejecutarse en una sola PC, de lo contrario, habrá inconvenientes con el servidor web que reciba los datos de los softwares en funcionamiento.

La APP Android puede extenderse a varios usuarios, solo hay que crearlos desde la página del servidor WEB y luego ingresar la llave electrónica en cada móvil.

ThingView es la aplicación de Thingspeak para plataforma Android, para ver los datos cargados al servidor aunque su visualización es muy rústica.

Things Speak es la aplicación de Thingspeak para plataforma iOS, donde se pueden ver los datos en el servidor aunque su visualización es muy básica.

Desde la página web [www.thingspeak.com](http://www.thingspeak.com) a través del explorador de internet, sea cual fuese el dispositivo que lo corra, se podrá acceder a los mismos datos, solo se deberá tener una clave única e ingresar con el cuidado de no modificar parámetros, porque pueden estar siendo utilizados por los usuarios espectadores.

### 2.2.3 Fotos del hardware y capturas de pantalla de software



*Figura 32 Gabinete*



*Figura 33 Tablero final*

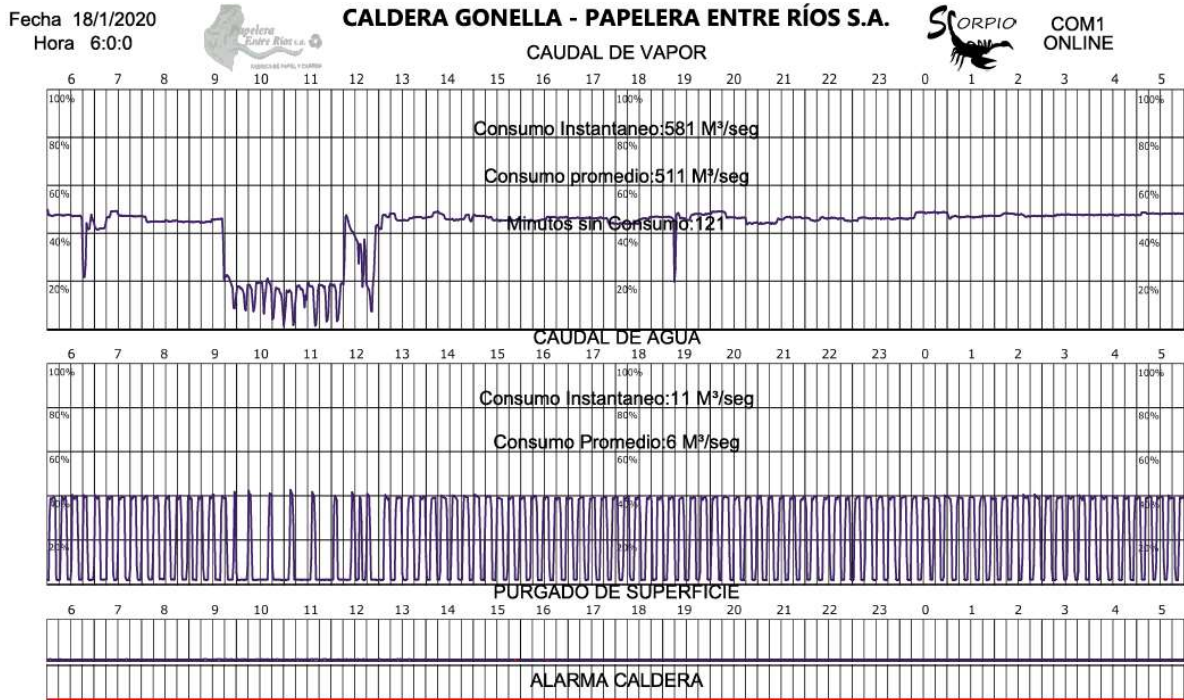


Figura 34 Almacenamiento en archivo PDF



Figura 35 Pestañas Caudales y Otros APP



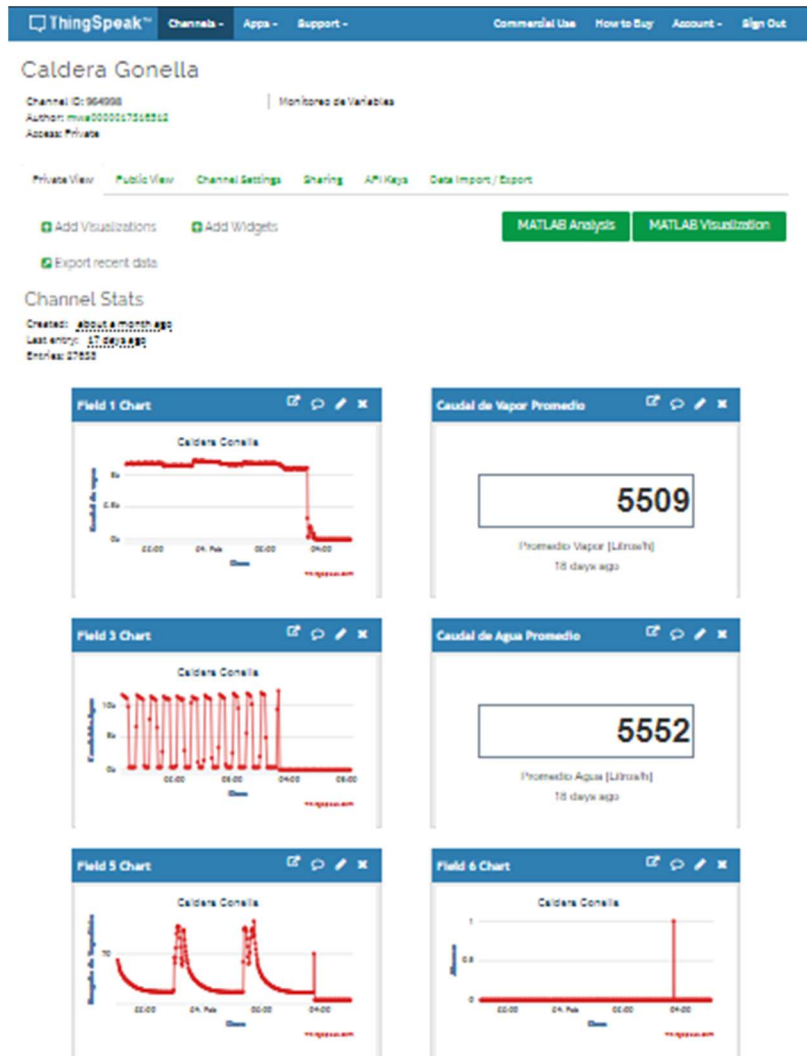


Figura 36 Desde explorador de internet página thingspeak

## Capítulo 3: Resultados

En términos generales los resultados obtenidos fueron muy buenos, tanto con el software ubicado en la pc de escritorio, como con la aplicación Android que trabaja con el servidor web. Toda la información queda debidamente almacenada tanto en la pc como en el servidor.

No se sabe con precisión cual es el límite de almacenamiento de datos en el servidor, pero hasta la fecha todos los datos enviados siguen almacenados y se los pueden observar rápidamente.

El software en la Pc es estable y consume pocos recursos, por lo que no ralentiza el sistema operativo.

La aplicación Android es muy buena, los datos se refrescan de forma fluida.

El contraste de las gráficas temporales ayuda mucho en la lectura de los valores. La interacción que se puede realizar a través de la pantalla táctil tanto para realizar un zoom (que es el efecto de acercamiento o alejamiento de la gráfica) o para arrastrar temporalmente la gráfica y visualizar diferentes rangos de tiempos, es lo que le da un dinamismo y una capacidad de plasmar los datos de forma amena al usuario.

Se ha notado que raras veces la aplicación produce algunos puntos en sus graficas que se solapan a los verdaderos, lo cual no es problema porque la forma ideal de corregirlos es limpiar la gráfica y cargar nuevamente los datos.

La aplicación móvil es lo pujante y colorido del proyecto, lo que termina dándole un desenlace formidable, con rigidez y siendo amigable, confiable, predecible y simple en su utilización.

Esta nueva herramienta ya ha dado frutos.

Al poder observar el funcionamiento del sensor de purgado de superficie se advirtió que el controlador no tenía un normal funcionamiento.

Al abrirse la purga el lapso correspondiente (tiempo de purga seteado + lapso de eliminación para la nivelación de solidos suspendidos), la válvula se cierra y debería transcurrir un tiempo establecido (seteado a traves de jumpers), el cual no era respetado.

El controlador BC1100 cuenta con varias horas de uso y a su vez ha sido reparado en otra oportunidad. Al ser una falla que se puede enmendar se comenzó a trabajar de inmediato. Para absorber el mal funcionamiento se colocó un temporizador que asegura el tiempo de mantención de cierre de la purga.

En largos lapsos de tiempo se observó el comportamiento del controlador y al ver que ocasionalmente perdía señal o era inestable, se optó por desarmar el sensor ubicado en la caldera y limpiar el sarro en su interior.

Luego de la intervención sobre el sensor las lecturas fueron satisfactorias.

## Capítulo 4: Análisis de Costos

ITEM	CANTIDAD	P. UNITARIO (US\$)	PRECIO (US\$)
Placa Arduino UNO	1	30	30
Caja plástica Roker 71x71x50mm	2	6	12
Fuente Switching Regin Power 5VDC 8a 50w	1	58	58
Gabinete metálico grado de IP65 placa interfaz	1	67	67
Fuente 5VDC 2A tipo cargador	1	29	29
Cable adaptador USB a RS232 (Db9) con cable Mallado Manhattan	1	36	36
MCP6004	2	4,2	8,4
PC817C	6	0,72	4,32
Materiales para placas de conversión serie RS-232 a RS485	1	12	12
Par trenzado UTP CAT6 GLC (100 metros)	100	1,43	143
Resistencias ¼ [W]	50	0,2	10
Diodo zener 5.1V	10	0,17	1,7
Placa de fibra 10x10	1	5,25	5,25
Bornera para placa PCB	10	0,7	7
Capacitores cerámicos	5	0,05	0,25
Termomagnetica unipolar 6A	1	8	8
Gabinete acrílico para Arduino Uno	1	7,2	7,2
Horas dedicadas	750	15	11250
<b>TOTAL</b>			<b>11689,12</b>

Cotización dólar 1US\$=\$70,87 peso Argentino

7 de Julio 2020

*Tabla 1 Costos*

El análisis de costo se elaboró teniendo en cuenta la investigación, el diseño, el desarrollo y proceso de armado que se necesitó para cada dispositivo.

Suponiendo que el producto fuera el resultado de una producción en serie para un mercado competitivo, estimando solo los costos de manufactura, es decir, los materiales directos, la mano de obra directa y los costos indirectos de fabricación (Fijos: Son los que se mantienen de manera independiente al volumen de producción, como el mantenimiento de los edificios, la depreciación y el arrendamiento. Variables: Estos sí cambian según el volumen de producción, como la adquisición de material indirecto y la contratación de mano de obra para tareas relacionadas con la producción), se podría amortizar los importes llegando a un valor estimado de US\$2500.

Cabe destacar que la materia prima se cotiza en dólares y depende su valor de los impuestos que se le apliquen a la importación de los elementos electrónicos, por esta razón se ofrece un valor de venta aproximado a los costos comerciales a Julio de 2020.

Luego se percibiría un arancel por el servicio de almacenamiento y uso del servidor web. Actualmente la licencia estándar costa USD 650/anual con un máximo de canales permitidos igual a 250.

Se estima que una cuota semestral sería de USD 70.

## Capítulo 5: Discusión y Conclusión.

### 5.1 Posibles mejoras

#### 5.1.1 Hardware

Una posibilidad para perfeccionar podría darse al agregar una comunicación bilateral y poder realizar acciones sobre el objeto desde el celular o la pc de escritorio.

Además, desde la propia empresa, plantearon la idea de tener una pantalla que muestre lo mismo que la pc de escritorio, al lado del tablero general de la Caldera. Para realizarlo no se necesitaría mucho más que una pc que a través de uno de sus puertos USB reciba la trama directamente desde el Arduino sin ninguna conversión de por medio.

#### 5.1.2 Software

El software de la PC es sencillo, brinda la posibilidad de agregar opciones que se lleven a cabo sobre el objeto a través de actuadores, válvulas, etc. pero teniendo en cuenta que puede haber retardos grandes y de distintas duraciones.

El servidor Web es libre, aunque al utilizar el servicio con fines de lucro tiene un costo, para lo que sería ideal tener un servidor propio.

Algo similar ocurre con la APP Android la cual tiene sus limitaciones, entonces poder realizar una aplicación propia sería lo conveniente, y a la medida del proyecto que queramos desarrollar. Esporádicamente se producen errores de datos cruzados en gráficas temporales, pero esto es propio de la aplicación y no hay forma de evitarlos. También raras veces carga los valores de los puntos de los chars y no los gráfica. Para enmendar esto solo se limpian las gráficas y se vuelven a cargar los puntos.

Para nuevos proyectos se podrían incorporar desde una mayor utilización de sensores, hasta el acondicionamiento de otras señales, y cálculos sobre ellas para ser monitoreadas desde la misma pantalla.

### 5.2 Conclusión

La elección del tema para el proyecto surgió de la idea de unir conocimientos en programación de Arduino y la programación Java y así crear un programa basado en software libre.

Otro aspecto por lo que se consideró avanzar en este proyecto, fue determinado por la inexistencia comercial de un sistema que se adapte a este propósito, y que a su vez utilice herramientas o una tecnología de bajo coste no pensada para un ambiente industrial. El

agregado sustancial de poder acceder a los datos desde cualquier dispositivo y en cualquier parte del mundo es lo que termina dándole el propósito que buscaba.

Los datos siempre han sido de vital importancia para medir, controlar y tomar decisiones en las empresas; su veracidad, relevancia, oportunidad y presentación son las características que los convierten en información útil para el usuario, o en su defecto lo opuesto.

El acceso a información útil ha sido un reto que al día de hoy muchas organizaciones no han podido resolver y siguen operando y tomando decisiones con información limitada, imprecisa y desactualizada; desafortunadamente han aprendido a vivir con ello, sin embargo, las exigencias y tendencias actuales en los negocios exigen procesos ágiles y resilientes soportados por información útil y en tiempo real. Las empresas ya no podrán sobrevivir en los mercados actuales operando y tomando decisiones con información limitada, imprecisa y/o desactualizada; las empresas líderes están transformándose hacia una nueva era digital con acceso a grandes volúmenes de datos, aquellas que logren operar y tomar decisiones con información útil y en tiempo real tendrán mejores resultados y una ventaja competitiva en su mercado.

Este proyecto es solo el comienzo...

En el apartado de costos se debe tener en cuenta que la implementación fue específica para un determinado instrumento o conjunto de instrumentos, por ello no se puede contrastar directamente con otro producto similar del mercado ya que en este no encontramos existencia alguna.

Este proyecto se encaró con la premisa de que debería ser ampliamente adaptable, por lo que las mejoras o nuevos desarrollos que se podrían incorporar van desde la utilización de más sensores, al acondicionamiento y cálculo de otras variables para ser monitoreadas desde cualquier dispositivo con acceso a Internet.

## Capítulo 6: Literatura Citada.

[1] Checksum Fletchers. URL: <https://www.drdoobs.com/database/fletchers-checksum/184408761>

[2] Datasheet SN75176A Differential Bus Transceiver

[3] Norma TIA/EIA RS232C

[4] Ingeniería en Microcontroladores. Norma la TIA/EIA-485. URL: <http://www.electronica60norte.com/mwfls/pdf/rs-485.pdf>

[5] Cable de comunicación Manhattan. URL: [https://s3.amazonaws.com/assets.mhint/downloads/42706/205146\\_datasheet\\_spanish.pdf](https://s3.amazonaws.com/assets.mhint/downloads/42706/205146_datasheet_spanish.pdf)

[6] Adobe Acrobat. URL: <https://acrobat.adobe.com/la/es/acrobat/about-adobe-pdf.html>

[7] IoT analytics platform service. URL: <https://www.Thingspeak.com>

[8] Que es una API. URL: <https://thescienceofcode.azurewebsites.net/Articles/Show/596fd1d471bec27e2c2e0896>

[9] Microsoft. Internet de las cosas (IoT). URL: <https://www.microsoft.com/es-es/internet-of-things/>.

[10] IoT Agenda. What is Internet of Everything (IoE). URL: <http://internetofthingsagenda.techtarget.com/definition/Internet-of-Everything-IoE>.

[11] TECNOLOGIA fácil. Domotica domestica: Casas Inteligentes, 2019. URL: <http://tecnologia-facil.com/que-es/domotica-domestica-casas-inteligentes/>.

[12] Asociación Española de Domotica e Inmotica CEDOM. Que es Domotica. URL: <http://www.cedom.es/sobre-domotica/que-es-domotica>.

[13] Arduino. Arduino -Software, 2019. URL: <https://www.arduino.cc/en/main/software#>.

[14] Arduino. Getting Started - First step with Arduino, 2019. URL: <http://www.arduino.org/learning/getting-started>.

[15] Fritzing. Getting Started. URL: <http://fritzing.org/learning/>.

[16] URL: <https://aprendiendoarduino.wordpress.com/2017/03/29/que-es-iot/>

[17] Foros de Arduino. URL: <http://forum.arduino.cc>

[18] “RMS calculation for energy meter applications using ADE7756”, Notas de aplicación Analog Devices AN-578

[19] “A power meter reference design base don ADE7756”, Notas de aplicación Analog Devices AN-564

[20] Processing -Software, 2019. URL:  
<https://processing.org/>.

## Apéndice A – Hojas de Datos

En este anexo se adjuntan Datasheet de los componentes más importantes utilizados en el proyecto.



# MCP6001/2/4

## 1 MHz Bandwidth Low Power Op Amp

### Features

- Available in 8C-70-5 and 8OT-23-5 packages
- 1 MHz Gain Bandwidth Product (typ.)
- Rail-to-Rail Input/Output
- Supply Voltage: 1.8V to 5.5V
- Supply Current:  $I_Q = 100 \mu A$  (typ.)
- 90° Phase Margin (typ.)
- Temperature Range:
  - Industrial: -40°C to +85°C
  - Extended: -40°C to +125°C
- Available in Single, Dual and Quad Packages

### Applications

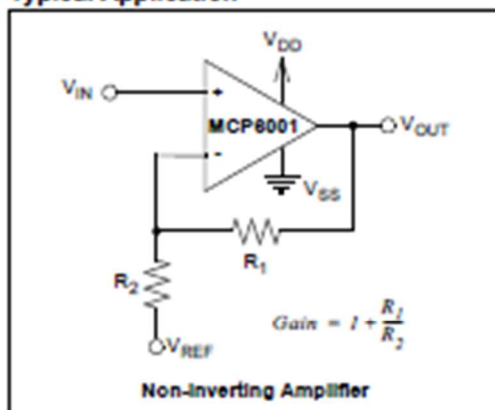
- Automotive
- Portable Equipment
- Photodiode Pre-amps
- Analog Filters
- Notebooks and PDAs
- Battery-Powered Systems

### Available Tools

Spice Macro Models (at [www.microchip.com](http://www.microchip.com))

FilterLab® Software (at [www.microchip.com](http://www.microchip.com))

### Typical Application

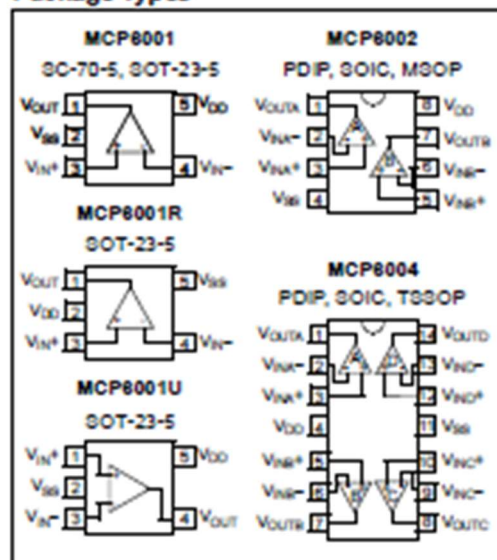


### Description

The Microchip Technology Inc. MCP6001/2/4 family of operational amplifiers (op amps) is specifically designed for general-purpose applications. This family has a 1 MHz gain bandwidth product and 90° phase margin (typ.). It also maintains 45° phase margin (typ.) with 500 pF capacitive load. This family operates from a single supply voltage as low as 1.8V, while drawing 100  $\mu A$  (typ.) quiescent current. Additionally, the MCP6001/2/4 supports rail-to-rail input and output swing, with a common mode input voltage range of  $V_{DD} + 300 mV$  to  $V_{SS} - 300 mV$ . This family of operational amplifiers is designed with Microchip's advanced CMOS process.

The MCP6001/2/4 family is available in the industrial and extended temperature ranges. It also has a power supply range of 1.8V to 5.5V.

### Package Types

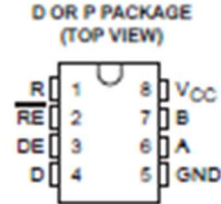




## SN75176A DIFFERENTIAL BUS TRANSCEIVER

SLLS100A – JUNE 1984 – REVISED MAY 1995

- Bidirectional Transceiver
- Meets or Exceeds the Requirements of ANSI Standards EIA/TIA-422-B and ITU Recommendation V.11
- Designed for Multipoint Transmission on Long Bus Lines in Noisy Environments
- 3-State Driver and Receiver Outputs
- Individual Driver and Receiver Enables
- Wide Positive and Negative Input/Output Bus Voltage Ranges
- Driver Output Capability . . . ±60 mA Max
- Thermal-Shutdown Protection
- Driver Positive- and Negative-Current Limiting
- Receiver Input Impedance . . . 12 kΩ Min
- Receiver Input Sensitivity . . . ±200 mV
- Receiver Input Hysteresis . . . 50 mV Typ
- Operates From Single 5-V Supply
- Low Power Requirements



### description

The SN75176A differential bus transceiver is a monolithic integrated circuit designed for bidirectional data communication on multipoint bus-transmission lines. It is designed for balanced transmission lines and meets ANSI Standard EIA/TIA-422-B and ITU Recommendation V.11.

The SN75176A combines a 3-state differential line driver and a differential input line receiver, both of which operate from a single 5-V power supply. The driver and receiver have active-high and active-low enables, respectively, that can be externally connected together to function as a direction control. The driver differential outputs and the receiver differential inputs are connected internally to form differential input/output (I/O) bus ports that are designed to offer minimum loading to the bus whenever the driver is disabled or  $V_{CC} = 0$ . These ports feature wide positive and negative common-mode voltage ranges making the device suitable for party-line applications.

The driver is designed to handle loads up to 60 mA of sink or source current. The driver features positive- and negative-current limiting and thermal shutdown for protection from line fault conditions. Thermal shutdown is designed to occur at a junction temperature of approximately 150°C. The receiver features a minimum input impedance of 12 kΩ, an input sensitivity of ±200 mV, and a typical input hysteresis of 50 mV.

The SN75176A can be used in transmission-line applications employing the SN75172 and SN75174 quadruple differential line drivers and SN75173 and SN75175 quadruple differential line receivers.

The SN75176A is characterized for operation from 0°C to 70°C.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.



POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 1995, Texas Instruments Incorporated

# PC817X Series

## DIP 4pin General Purpose Photocoupler

\*4-channel package type is also available.  
(model No. **PC847X Series**)



### ■ Description

**PC817X Series** contains an IRED optically coupled to a phototransistor.

It is packaged in a 4pin DIP, available in wide-lead spacing option and SMT gullwing lead-form option.

Input-output Isolation voltage(rms) is 5.0kV.

Collector-emitter voltage is 80V(\*) and CTR is 50% to 600% at Input current of 5mA.

### ■ Features

1. 4pin DIP package
2. Double transfer mold package (ideal for Flow Soldering)
3. High collector-emitter voltage ( $V_{CE0}$ :80V(\*)
4. Current transfer ratio (CTR : MIN. 50% at  $I_F=5\text{ mA}$ ,  $V_{CE}=5\text{ V}$ )
5. Several CTR ranks available
6. High Isolation voltage between Input and output ( $V_{iso(rms)}$  : 5.0 kV)

(\*) Up to Date code "P7" (July 2002)  $V_{CE0}$  : 35V.  
From the production Date code "J5" (May 1997) to "P7" (July 2002), however the products were screened by  $BV_{CE0}>70\text{ V}$ .

### ■ Agency approvals/Compliance

1. Recognized by UL1577 (Double protection Isolation), file No. E64380 (as model No. **PC817**)
2. Package resin : UL flammability grade (94V-0)

### ■ Applications

1. I/O Isolation for MCUs (Micro Controller Units)
2. Noise suppression in switching circuits
3. Signal transmission between circuits of different potentials and Impedances

Notice: The content of data sheet is subject to change without prior notice.  
In the absence of confirmation by device specification sheets, SHARP takes no responsibility for any defects that may occur in equipment using any SHARP devices shown in catalogs, data books, etc. Contact SHARP in order to obtain the latest device specification sheets before using any SHARP devices.