

# INSTITUTO TECNOLÓGICO SUPERIOR DE COATZACOALCOS

## INFORME FINAL DE RESIDENCIAS PROFESIONALES

### **CARRERA:**

INGENIERÍA MECATRÓNICA

### **NOMBRE DEL PROYECTO**

SISTEMA DE MONITOREO PARA DETECCIÓN DE FUGAS  
EN UNA TUBERÍA HIDRÁULICA

### **NOMBRE DE LA EMPRESA**

INSTITUTO TECNOLÓGICO SUPERIOR DE  
COATZACOALCOS

### **ASESOR INTERNO**

DR. OMAR HERNÁNDEZ GONZÁLEZ

### **NOMBRE DEL ALUMNO:**

GEOVANI VELÁZQUEZ VEGA

### **NO. DE CONTROL:**

14081004

### **FECHA DE ENTREGA**

DICIEMBRE 2018

## ÍNDICE GENERAL

INTRODUCCIÓN.....	10
1 Capítulo I: Protocolo de Investigación .....	11
1.1 Planteamiento del problema .....	12
1.1.1 Antecedentes .....	12
1.1.2 Enunciado del problema. ....	12
1.1.3 Preguntas de investigación.....	14
1.1.4 Justificación.....	15
1.2 Objetivos .....	17
1.2.1 General.....	17
1.2.2 Específicos .....	17
1.3 Marco conceptual.....	18
1.4 Diseño Metodológico y Técnicas De Investigación .....	21
1.5 Calendario De Trabajo.....	22
2 Capítulo II: Marco Teórico.....	23
2.1 Sistema de monitoreo .....	24
2.1.1 Adquisición de datos .....	26
2.1.1.1 Dispositivo DAQ .....	27
2.1.1.2 Acondicionamiento de Señales .....	27
2.1.1.3 Convertidor Analógico Digital (ADC).....	28
2.1.1.4 Bus de la PC.....	28
2.1.2 Protocolos de comunicación. ....	28
2.1.2.1 Protocolo HART.....	29
2.1.2.2 Comunicación Serial.....	30

2.1.2.2.1	Velocidad de transmisión. ....	30
2.1.2.2.2	Bits de datos: .....	31
2.1.2.2.3	Bits de parada .....	31
2.1.2.2.4	Paridad .....	32
2.1.3	Diseño de interfaces .....	32
2.1.3.1	Labview.....	33
2.1.3.2	Interfaz gráfica del operador: .....	33
2.1.3.3	Modelo del usuario .....	34
2.1.3.4	Modelo del diseñador. ....	35
2.2	Sistema de monitoreo de fugas hidráulicas.....	36
2.2.1	Fugas hidráulicas .....	36
2.2.2	Tipos de fugas.....	36
2.2.3	Métodos de detección de fugas. ....	37
2.2.3.1	Sistema acústico directo .....	37
2.2.3.2	Sistema acústico por correlación .....	38
2.2.3.3	Sistema de sonda de rastreo .....	39
2.2.3.4	Sistema de fotografía termográfica .....	40
2.2.3.5	Sistema del gas trazador.....	41
3	Capítulo III: Análisis y diseño .....	43
3.1	Diseño de adquisición de datos para sensores de flujo y presión .....	44
3.1.1	Arquitectura de Arduino MEGA.....	44
3.1.2	Métodos de conexión entre Arduino y computadora .....	45
3.1.2.1	Conexión inalámbrica por modulo wifi.....	45
3.1.2.2	Conexión inalámbrica de Arduino por Bluetooth .....	47

3.1.3	Sensor de flujo. ....	50
3.1.3.1	Medición de flujo.....	52
3.1.3.2	Algoritmo de medición de flujo y programación en Arduino. ....	54
3.2	Diseño de una interfaz para el monitoreo y adquisición de datos. ....	55
3.2.1	HMI del sistema de monitoreo. ....	56
3.2.2	Comunicación entre Labview y Arduino. ....	57
3.2.2.1	Linx. ....	58
3.2.2.2	VISA Puerto serial. ....	60
3.3	Diseño de un prototipo para la validación de la detección de fugas. .....	62
3.3.1	Diseño de la planta piloto.....	62
3.3.2	Diseño de construcción en Solidworks .....	64
3.3	Diseño de un variador de frecuencia AC para una bomba hidráulica.....	71
3.3.1	TRIAC.....	71
3.3.2	Control por fase.....	72
3.3.3	Diagrama esquemático. ....	73
3.3.4	Diseño de PCB.....	75
4	Capítulo IV: Resultados .....	77
4.1	Simulación del modelo matemático de la tubería con una fuga....	78
4.2	Monitoreo de flujo.....	80
4.3	Control de bomba por medio del variador. ....	82
4.4	Detección de la presencia de la fuga.....	83

5.	CONCLUSIONES .....	85
6.	RECOMENDACIONES.....	86
7.	BIBLIOGRAFÍA .....	87
8.	ANEXOS .....	89

## ÍNDICE DE TABLAS Y FIGURAS.

Figura 2.1 Monitoreo y control de instrumentación industrial .....	24
Figura 2.2 Pantalla de un SCADA en Kimberly Clark Costa Rica .....	26
Figura 2.3 Adquisición de datos.....	27
Figura 2.4 modelo de diseño .....	36
Figura 2.5 Operación para la detección de la fuga por medio del sistema acústico directo. ....	38
Figura 2.6 Operación de la detección de la fuga por medio del sistema acústico de correlación. ....	39
Figura 2.7. Sondas de rastreo: a) Inalámbrica. b) Alámbrica .....	40
Figura 2.8 Operación de la detección de la fuga por medio del sistema de sonda de rastreo .....	40
Figura 2.9 Operación de la detección de la fuga por medio del sistema de termografía .....	41
Figura 2.10 Sistema del gas trazador. a) Operación de la detección de la fuga. b) Equipo de inspección.....	42
Figura 3.1 . Arduino MEGA .....	44
Figura 3.2 Conexión del módulo wifi a Arduino .....	46
Figura 3.3 pines de conexión del módulo wifi.....	47
Figura 3.4 conexión de módulo bluetooth al puerto serie de Arduino. ....	49
Figura 3.5 Puerto serie del módulo bluetooth conectado a Arduino. ....	50
Figura 3.6 Conexión entre sensor YF-S201 y Arduino .....	51
Figura 3.7 Monitor serial para el cálculo de flujo .....	53
Figura 3.8 Caudalímetro Modelo YF- S201 Fuente: Sea .....	54
Figura 3.9 HMI del sistema de monitoreo .....	56
Figura 3.10 Panel de control del HMI del sistema de monitoreo.....	57
Figura 3.11 visualización gráfica de flujos y simulación de tubería.....	58
Figura 3.12 a) Bloque Digital Read de LINX, b) bloque cruce por cero .....	59
Figura 3.13 Panel frontal de interfaz para el monitoreo de flujos en LINX	59

Figura 3.14 Diagrama de bloques de la programación en Labview mediante LINX .....	59
Figura 3.15 Extraccion de numeros de una cadena de caracteres .....	60
Figura 3.16 obtención de datos del array. ....	61
Figura 3.17 Extraccion de numeros adecuado a interfaz de monitoreo ....	61
Figura 3.18 Esquema del prototipo de tuberías Fuente: Delgado, 2014...	63
Figura 3.19 DTI propuesto de sistema de tuberías en Torres, 2014.....	63
Figura 3.20 Sistema de tuberías con 2 líneas y 1 línea de reincorporación. ....	64
Figura 3.21. sistema de tuberías de 6 líneas con 3 de reincorporación....	65
Figura 3.22 Prototipo experimental en Tecnológico de Tuxtla Gutierrez. .	66
Figura 3.23 Prototipo de tuberías en un plano vertical .....	66
Figura 3.24 Vista isométrica completa de la planta .....	67
Figura 3.25 vista de sección lateral .....	68
Figura 3.26 Vista de detalle de conexion a tanque. ....	68
Figura 3.27 Conexion de bomba al sistema de tuberías .....	69
Figura 3.28 Vista completa del recorrido en forma de serpentina.....	69
Figura 3.29 instalación de transmisores; caudal y presión.....	70
Figura 3.30 Válvula que emula una fuga. ....	70
Figura 3.31 Estructura de un TRIAC.....	72
Figura 3.32 Disparo con cruce por cero para un variador de frecuencia ..	73
Figura 3.33 Diagrama esquemático del controlador básico de potencia con una red de snubber; La red de snubber se puede remover si la carga es resistiva .....	73
Figura 3.34 diseño de referrencia de un detector de cruce por cero. ....	74
Figura 3.35 Circuito del variador de frecuencia para bomba de 1/2HP para el control de flujo. ....	75
Figura 3.36 Diseño de placa impresa. ....	76

Figura 3.37 Placa del variador de frecuencia con componentes instalados.	76
Figura 4.1 Condiciones iniciales para la simulacion del modelo matematico de una fuga	78
Figura 4.2 Parametros para el modelo matematico de la tuberia.	78
Figura 4.3 Flujos de entrada y salida simulados con condiciones iniciales.	79
Figura 4.4 Presion simulada en entre los extremos de la tuberia	79
Figura 4.5 Monitoreo de flujo en tiempo real	80
Figura 4.6 HMI del prototipo en tiempo real.	81
Figura 4.7 Prototipo final para pruebas experimentales, con una fuga	81
Figura 4.8 Medicion de flujo con 75% de velocidad de la bomba.	82
Figura 4.9 Medicion de flujo con 56% de velocidad de la bomba	82
Figura 4.10 Control deslizante para la variacion de velocidad en bomba.	83
Figura 4.11 Monitoreo de flujos en tiempo real ante la presencia de una fuga	83
Figura 4.12 Deteccion de la presencia de fuga mediante diferencias de flujos.	84
Figura 4.13 Flujo de entrada y salida en tiempo real y simulado. con fuga ocasionada en el instante 9 seg.	84
Figura 4.14 Visualización de alarma en HMI de la presencia de fuga.	84
Tabla 2.1 tipos de fugas hidráulicas	37
Tabla 3.1 Características del Arduino Mega	45
Tabla 3.2 Modelos de sensores efecto hall	50
Tabla 3.4 Medición de flujo en relación a la frecuencia por sensor yf-s201	54
Tabla 3.5 Parametros del diseño en 3D	67
Tabla 3.6 Componentes para variador de frecuencia.	76



## ÍNDICE DE ANEXOS.

Anexo 1 Programación en Arduino IDE para la obtención de flujos procedentes de los sensores YS-F201.....	89
Anexo 2 Programación en Arduino para un variador de frecuencia monofásico.....	91

## INTRODUCCIÓN.

En este proyecto se realizará la monitorización en tiempo real de variables existentes en un sistema de tuberías como presión y flujo que permitan la detección de una fuga mediante una interfaz en LabView, para ello se implementara un prototipo piloto donde se obtendrán dichas variables

El proyecto consta de varias etapas para su realización, para cada etapa se realizará un análisis para evaluar las deficiencias de cada paso del proceso y de este modo realizar mejoras.

Por lo consiguiente en el capítulo 2 se expondrá la investigación un sistema conceptos que permitan abordar el problema que tienen los sistemas de monitoreo y detección de fugas, es decir, comprenderemos el desarrollo de la teoría en la que se fundamentara el proyecto, con base al planteamiento del problema que se realizara.

Posteriormente en el capítulo 3 se hablará del análisis y diseño, que es la fase de mayor importancia en todo proyecto a implementar, pues en él se especificara las diferentes etapas del proceso y como se planean realizar.

Finalmente se dará a conocer los resultados obtenidos de la investigación del proyecto, así como dar la solución de las preguntas de investigación, dando como evidencia todo el análisis y la información recatada del proyecto.

.

# 1 . Capítulo I: Protocolo de Investigación

## 1.1 Planteamiento del problema

### 1.1.1 Antecedentes

En los últimos años el monitoreo del transporte de fluido en las tuberías ha tenido un importante desarrollo tecnológico, la comunidad hidráulica se ha enfocado en el desarrollo de métodos eficaces para la detección de fallas en tuberías. Las fallas más comunes y estudiadas son: fugas, bloqueos, burbujas de aire, tomas clandestinas.

los principales enfoques para la detección de fugas son: métodos externos, que son los que utilizan dispositivos y/o equipo de alta tecnología, incluso personal técnico especializado para detectar una fuga, y los Métodos Internos, que están constituidos por un sistema de monitoreo en tiempo real y algoritmos basados en el modelo matemático que describe la dinámica de un fluido dentro de una tubería.

A partir de un Sistema de Control y Adquisición de Datos (SCADA) que operan en tiempo real, donde se registran presiones y flujos en puntos limitados de las redes de ductos, se han implementado localizadores automáticos de averías utilizando sensores físicos y virtuales

### 1.1.2 Enunciado del problema.

Uno de los grandes problemas que se presentan en el transporte de fluido es la pérdida de fluidos en las tuberías, debido a fugas ocasionadas por deterioro de la misma y en algunos casos por la extracción ilegal de fluidos, principalmente combustibles. Por esta razón, el problema de detección de fugas en tuberías ha cobrado gran importancia en el ámbito de la investigación alrededor del mundo. Por lo que el monitoreo automático de ductos es un reto para la ingeniería debido a las consecuencias catastróficas que pueden desencadenar los ductos en mal estado, como son la pérdida de vidas humanas y años ambientales con alto impacto ecológico. En general, el objetivo de un sistema automatizado de

supervisión en tiempo real consiste en localizar, tan rápido como sea posible, la presencia de fallas con un mínimo de instrumentación y costo.

El transporte de fluidos por ductos es una función fundamental en procesos cotidianos, debido a la complejidad de las redes de distribución que se diseñan para el transporte de éstos y la variedad de eventos externos que se pueden presentar ya sea por envejecimiento, eventos naturales, extracción ilícita en el caso de hidrocarburos o mantenimientos precarios, es requisito contar con sistemas de monitoreo que den respuestas automáticas de manera rápida a dichos eventos para mejorar la seguridad, reducir costos en hardware y lograr una operación más eficiente de las redes.

### 1.1.3 Preguntas de investigación

- a) ¿Implementar un sistema de monitoreo reducirá fugas hidráulicas?
- b) ¿Qué tipo de eventos no deseados (fallas o daños) se busca diagnosticar en una tubería?
- c) ¿Por qué es importante tener sistemas de monitoreo y detección de fugas en tuberías?
- d) ¿Cómo funcionan los métodos empleados para la detección de una fuga?
- e) ¿Qué instrumental es necesario para el monitoreo de variables en una tubería?

#### 1.1.4 Justificación

Los proyectos de investigación en el Instituto Tecnológico Superior de Coatzacoalcos son significativo de la calidad educativa; la realización de este proyecto dentro de la institución tendrá un gran impacto en el fortalecimiento de las líneas de investigación de la institución, con lo que se impulsara la investigación científica, ya que sirve de base para desarrollar proyectos que plantean la soluciones a problemas propios de la región, logrando que se fomente el desarrollo de residencias profesionales, tesis de licenciatura y artículos científicos-tecnológicos, de esta manera la institución logra prestigio y fortalecimiento en mira de subir los indicadores académicos. Así también se contribuye el desarrollo tecnológico de la región formando mejores profesionales, además será de utilidad para conjuntar el conocimiento teórico-práctico de los alumnos que desarrollen proyectos similares.

Este trabajo de investigación tiene el gran beneficio de ser un tema interesante en el campo del desarrollo de la tecnología de control automático, así también la problemática nacional del robo y fugas en los sistemas de tuberías hidráulicas tanto de combustibles y agua potable; actualmente, entre 30 y 50 por ciento de agua potable que se distribuye en las ciudades de México se pierde por fugas en las redes hidráulicas. El problema es recurrente en casi todas las ciudades del territorio nacional, y las causas son prácticamente las mismas: ductos viejos y deteriorados, la presión variable del agua y el suministro intermitente. Además, la mayor parte de la infraestructura del sistema hídrico ya cumplió su vida útil, por lo que hay grandes pérdidas por fugas y esto se agrava por la mala gestión. Por otro lado, en cada población, no es la misma situación, ya que el suministro es distinto en cada zona del país y existen otros factores como

son la cartografía y distancias de las fuentes de suministros principales, entre otros factores.

Otro gran problema se encuentra en PEMEX, ya que, de acuerdo con el informe anual sobre la situación operativa, programática y financiera de Pemex, en 2013 incrementaron 75 por ciento las tomas clandestinas al pasar de mil 744 detectadas en 2012, a tres mil 52; asimismo, creció en 13.4 por ciento el volumen de combustible sustraído ilícitamente, al pasar de 8.2 millones de barriles en 2012 a 9.3 millones de barriles. Además, este problema es preponderante no solo por el costo del volumen sustraído, sino por los riesgos que las tomas clandestinas conllevan, dado que pueden causar graves daños a las personas involucradas en el robo, a las comunidades, a las instalaciones petroleras y al medio ambiente. Se han propuesto algunas alternativas de solución, recientemente se ha propuesto sistemas no intrusivo combinados, que ponen en función diferentes principios físicos de detección y localización, se han incorporado al SCADA de Pemex



## 1.2 Objetivos

### 1.2.1 General

Diseñar e implementar un sistema de monitoreo para la detección de fugas en tuberías hidráulicas donde dicho sistema de detección se basará en un monitoreo en tiempo real de las variables del sistema.

### 1.2.2 Específicos

- Desarrollar un modelado matemático adecuado que represente la tubería hidráulica en distintas secciones.
- Desarrollar un sistema de monitoreo para el sistema de detección de fugas.
- Desarrollar un prototipo del sistema de tuberías por secciones.
- Implementar y evaluar el sistema de detección de fugas propuesto en un prototipo en tiempo real.
- Analizar la recolección de datos del sistema de monitoreo con el fin de determinar la presencia de posibles fugas en el sistema de tuberías.
- Diseñar una plataforma para la recolección de datos basada en una tarjeta de adquisición de datos.

### 1.3 Marco conceptual

Para el desarrollo de este proyecto es necesario comprender ciertos conceptos que se describen a continuación.

a) Fuga: salida o escape de un líquido o de un gas por una abertura producida accidentalmente en el recipiente que los contiene o en el conducto por el que circulan.

b) Flujo hidráulico: Se define como el volumen de líquido que fluye por una unidad de tiempo.

c) Sistema: es una combinación de componentes que actúan juntos y realizan un objetivo determinado. Un sistema no está necesariamente limitado a los sistemas físicos. El concepto de sistema se puede aplicar a fenómenos abstractos y dinámicos, como los que se encuentran en la economía.

d) Detección: localización de alguna cosa que no puede observarse directamente mediante aparatos o métodos físicos o químicos:

e) Monitoreo: es la acción y efecto de monitorear, se utiliza para nombrar a la supervisión o el control realizado a través de un monitor.

f) Instrumentación industrial: es el grupo de equipamientos y dispositivos que sirven a los ingenieros, para medir, convertir y registrar variables de un proceso y luego, trasmitirlas, evaluarlas y controlarlas con tales fines.

g) Labview: acrónimo de Laboratory Virtual Instrument Engineering Workbench, es una plataforma y entorno de desarrollo para diseñar

sistemas, con un lenguaje de programación visual gráfico. Recomendado para sistemas hardware y software de pruebas, control y diseño, simulado o real y embebido, pues acelera la productividad. El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico.

h) Interfaz: es lo que permite que la interacción entre persona y ordenador ocurra, es decir, la interfaz permite que la persona pueda controlar efectivamente las acciones de la máquina, que la persona reciba respuestas de la máquina que le permitan saber si la interacción es correcta y cómo seguir actuando.

i) HMI: Una interfaz de usuario asistida por ordenador, actualmente una interfaz de uso, también conocida como interfaz hombre-máquina (IHM), forma parte del programa informático que se comunica con el usuario. En ISO 9241-110, el término interfaz de usuario se define como "todas las partes de un sistema interactivo (software o hardware) que proporcionan la información y el control necesarios para que el usuario lleve a cabo una tarea con el sistema interactivo".

j) SCADA: acrónimo de Supervisory Control And Data Acquisition es un concepto que se emplea para realizar un software para ordenadores que permite controlar y supervisar procesos industriales a distancia. Facilita retroalimentación en tiempo real con los dispositivos de campo, y controla el proceso automáticamente.

k) Sensor: es un dispositivo capaz de variar una propiedad ante magnitudes físicas o químicas, y transformarlas con un transductor en variables eléctricas.

l) Adquisición de datos: consiste en la toma de muestras del mundo real (sistema analógico) para generar datos que puedan ser manipulados por un ordenador u otros dispositivos electrónicos (sistema digital). Consiste en tomar un conjunto de señales físicas, convertirlas en tensiones eléctricas y digitalizarlas de manera que se puedan ser procesadas por una computadora.

## 1.4 Diseño Metodológico y Técnicas De Investigación

Para realizar el proyecto sistema de monitoreo y detección de fugas en una tubería hidráulica es necesario comprender la base y funcionamiento de sus elementos, así como de los equipos y componentes que intervienen el desarrollo de este mismo. Para ello se llevará a cabo una investigación documental y una investigación experimental con técnicas de investigación aplicadas de los trabajos presentados, enfocados al diseño e implementación práctica de las estrategias de detección de fallas aplicados para la detección de fugas en tuberías hidráulicas.

Se realizará la simulación del modelo matemático implementando un software de simulación, con la finalidad de aprender el comportamiento del sistema ante diversas situaciones o de evaluar estrategias para el funcionamiento del mismo.

Para el diseño y desarrollo del sistema se construirá un prototipo de tuberías a baja escala para pruebas del sistema de monitoreo, este prototipo constará de una sección o segmento de tubería conectada a un tanque de agua a la entrada y una bomba a la salida que retroalimente al tanque, de esta forma el prototipo simulará una tubería en constante flujo, también se le equipará de sensores de presión y flujo en cada uno de los extremos de una determinada sección

Para realizar la monitorización y comprobación de los datos recabados se implementará el software Labview para realizar la comunicación entre la interfaz y la tarjeta de adquisición de datos.

1.5 Calendario De Trabajo

INSTITUTO TECNOLÓGICO SUPERIOR DE COATZACOALCOS  
JEFATURA DE SERVICIO SOCIAL Y RESIDENCIAS PROFESIONALES

Alumno:

Geovani Velazquez Vega

No. De Control

14081004

Correo Electrónico

yoba.vega2401@gmail.com

Teléfono

9211987403

Nombre del Proyecto

Sistema de monitoreo para detección de fugas en una tubería hidráulica

Empresa

Instituto Tecnológico Superior de Coatzacoalcos.

Asesor Externo

M.C Alicia Enriqueta Pérez Yebra

Asesor Interno

Dr. Omar Hernández González

Periodo de Realización:

Agosto – Diciembre del 2018

ACTIVIDAD		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Revisión bibliográfica	P																
	R																
Modelado del caso de estudio	P																
	R																
Simulación del modelo matemático	P																
	R																
Validación del modelo matemático	P																
	R																
Construcción del prototipo	P																
	R																
Refaccionamiento	P																
	R																
Diseño de interfaz para el monitoreo del sistema	P																
	R																
Implementación practica del sistema de monitoreo de detección de fugas.	P																
	R																
Diseño de estrategias de detección de fugas	P																
	R																
Análisis del sistema de detección de fugas	P																
	R																

Fecha de Asesoría	12/08/18	08/08/18	15/08/18	22/08/18	29/09/18	07/09/18	14/09/18	11/10/18	18/10/18	25/10/18	02/11/18	09/11/18	16/11/18	23/11/18	30/11/18	06/12/18
Firma del Docente																
Firma del Alumno																

CA

Entrega de Reportes	Firma del Jefe División			
ITESCO-AC-PO-004-05	25			
		Semana 5 (07-11- de Septiembre)	Semana 9 (18-22 de octubre)	Semana 13 (16 – 20 de Noviembre)
		Reporte I	Reporte II	Reporte III
				Fecha Entrega de Acta de Calificaciones
				15 de Diciembre, 2018

# 2 Capítulo II: Marco Teórico

## 2.1 Sistema de monitoreo

Los sistemas de monitoreo son indispensables en las industrias, donde se pueden implementar con el objetivo de optimizar los recursos disponibles, así como un apropiado uso del agua en los sistemas hidráulicos de abastecimiento, centros de consumo o descargas, donde es necesario medir y conservar constantes algunas magnitudes, como la presión de una línea de abastecimiento, el nivel de una cisterna o el gasto instantáneo en una estación de rebombeo.

Normalmente los operadores llevan a cabo un control manual de las variables, utilizando instrumentos como manómetros Figura 2.1, válvulas manuales etc., control que antes era suficiente; la gradual escasez de agua y los altos costos de energía para la potabilización de la misma, exigen la automatización y el monitoreo de ciertas variables, sin la intervención humana para evitar el desperdicio y la sobreexplotación del recurso.

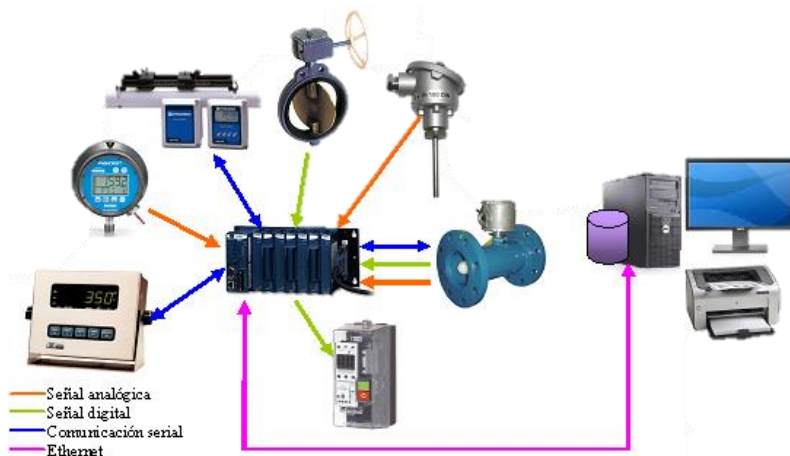


Figura 2.1 Monitoreo y control de instrumentación industrial

El monitoreo generalmente significa ser consciente del estado de un sistema, para observar una situación de cambios que se pueda producir con el tiempo, o sea que no interviene ninguna cuestión manual o automatizada sobre el control de las variables, es únicamente la visualización de los cambios en la instrumentación según las condiciones del sistema; este monitoreo puede realizarse con el personal y sus herramientas en mano, o



bien con la intervención automática de los equipos de instrumentación integrados con dispositivos de transmisión a través de redes locales o remotas, que envíen información a un monitor o a un panel de administración de las variables.

Por otra parte existen los sistemas de monitoreo y control como es el caso de un SCADA (Control Supervisorio y adquisición de datos) que es un sistema de monitoreo y control basado en computadores, que permite supervisar y controlar a distancia una instalación de cualquier tipo; pueden establecer un intercambio de datos entre diferentes componentes que lo integran para llevar a cabo la automatización del monitoreo y supervisión de procesos, permiten entre otras funciones el monitoreo de datos en tiempo real, la supervisión y gestión de alarmas, la conversión de unidades, el almacenamiento histórico, el control de procesos, la generación de reportes y la administración del uso del sistema.

Hoy en día es fácil hallar un sistema SCADA realizando labores de control automático en cualquiera de sus niveles, aunque su labor principal sea de supervisión y control por parte del operador

El siguiente nivel en sistemas de monitoreo SCADA, es la integración de los sistemas GPRS y tecnologías de redes que utilizan protocolos TCP/IP para el envío y transmisión de las variables medidas, con la instrumentación de un proceso definido a un sistema de monitoreo vía WEB, donde los sistemas de supervisión y monitoreo que se ejecutaban normalmente de manera local ahora estén disponibles para consulta en cualquier lugar con acceso a internet a través de un web server (SHUISA, 2018) .

Por ejemplo, los sistemas de monitoreo implementando un SCADA como lo menciona (López, 2015) Kimberly Clark Costa Rica, en su planta de Belén, es una empresa dedicada a la elaboración de productos de papel utilizando como base papel de reciclaje. Los procesos de fabricación de papel son complejos y delicados y por esa razón la firma decidió automatizar

algunos de los principales, utilizando productos de la línea de Foxboro. Al momento de este trabajo se cuenta con automatización en una planta recicladora (incluyendo el proceso de blanqueo), dos máquinas de papel y la planta de tratamiento de aguas.

En Kimberly Clark se ha llegado con ellos hasta el nivel de uso del SCADA, conocido para Foxboro como DCS; este sistema es una interfaz (Figura 2.2) que le permite al operador no solo observar distintas variables sino hacerlas interactuar con el proceso introduciendo ajustes desde el computador en su cuarto de control. El sistema DCS ha permitido a los operadores tener un mayor control de su operación y llevar gráficos de tendencias, los cuales son actualizados constantemente, permitiendo la toma de datos y decisiones oportunas.

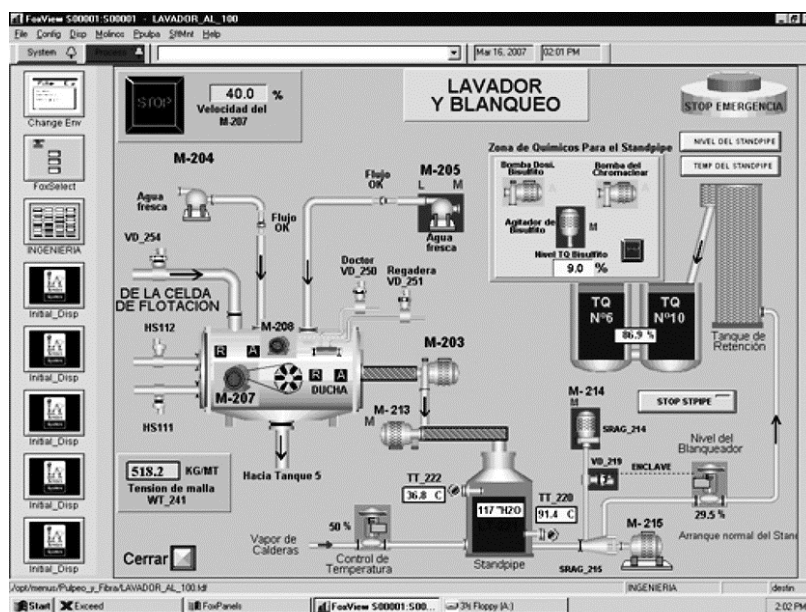


Figura 2.2 Pantalla de un SCADA en Kimberly Clark Costa Rica

### 2.1.1 Adquisición de datos

La adquisición de datos (DAQ) es el proceso de medir con una PC un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión o sonido. Un sistema DAQ consiste de sensores, hardware de medidas DAQ y una PC con software programable. Comparados con los sistemas de

medidas tradicionales, los sistemas DAQ basados en PC aprovechan la potencia del procesamiento, la productividad, la visualización y las habilidades de conectividad de las PCs estándares en la industria proporcionando una solución de medidas más potente, flexible y rentable. (National instruments, 2018).



Figura 2.3 Adquisición de datos

### 2.1.1.1 Dispositivo DAQ

El hardware DAQ actúa como la interfaz entre una PC y señales del mundo exterior. Funciona principalmente como un dispositivo que digitaliza señales analógicas entrantes para que una PC pueda interpretarlas. Los tres componentes clave de un dispositivo DAQ usado para medir una señal son el circuito de acondicionamiento de señales, convertidor analógico-digital (ADC) y un bus de PC. Varios dispositivos DAQ incluyen otras funciones para automatizar sistemas de medidas y procesos. Por ejemplo, los convertidores digitales-analógicos (DACs) envían señales analógicas, las líneas de E/S digital reciben y envían señales digitales y los contadores/temporizadores cuentan y generan pulsos digitales.

### 2.1.1.2 Acondicionamiento de Señales

Las señales de los sensores o del mundo exterior pueden ser ruidosas o demasiado peligrosas para medirse directamente. El circuito de acondicionamiento de señales manipula una señal de tal forma que es

apropiado para entrada a un ADC. Este circuito puede incluir amplificación, atenuación, filtrado y aislamiento. Algunos dispositivos DAQ incluyen acondicionamiento de señales integrado diseñado para medir tipos específicos de sensores.

### **2.1.1.3 Convertidor Analógico Digital (ADC)**

Las señales analógicas de los sensores deben ser convertidas en digitales antes de ser manipuladas por el equipo digital como una PC. Un ADC es un chip que proporciona una representación digital de una señal analógica en un instante de tiempo. En la práctica, las señales analógicas varían continuamente con el tiempo y un ADC realiza "muestras" periódicas de la señal a una razón predefinida. Estas muestras son transferidas a una PC a través de un bus, donde la señal original es reconstruida desde las muestras en software.

### **2.1.1.4 Bus de la PC**

Los dispositivos DAQ se conectan a una PC a través de una ranura o puerto. El bus de la PC sirve como la interfaz de comunicación entre el dispositivo DAQ y la PC para pasar instrucciones y datos medidos. Los dispositivos DAQ se ofrecen en los buses de PC más comunes, incluyendo USB, PCI, PCI Express y Ethernet. Recientemente, los dispositivos DAQ han llegado a estar disponibles para 802.11 Wi-Fi para comunicación inalámbrica. Hay varios tipos de buses y cada uno de ellos ofrece diferentes ventajas para diferentes tipos de aplicaciones.

## **2.1.2 Protocolos de comunicación.**

En informática y telecomunicación, un protocolo de comunicaciones o Protocolo de Comunicación es un sistema de reglas que permiten que dos o más entidades de un sistema de comunicación se comuniquen entre ellas para transmitir información por medio de cualquier tipo de variación de una magnitud física. Se trata de las reglas o el estándar que define la sintaxis,

semántica y sincronización de la comunicación, así como también los posibles métodos de recuperación de errores. Los protocolos pueden ser implementados por hardware, por software, o por una combinación de ambos (Rodríguez-Aragón, s/f)

### **2.1.2.1 Protocolo HART**

Creado a fines de la década de los 80, el protocolo HART (Highway Addressable Remote Transducer) es uno de los estándares líderes en la comunicación con instrumentación inteligente de campo, y continúa aumentando su popularidad y reconocimiento en la industria como un modelo global para este tipo de tareas, existiendo en la actualidad innumerables instalaciones de HART en el mundo., Es muy sencillo de usar y provee una muy efectiva comunicación digital de dos vías. Asimismo, este protocolo es usado simultáneamente con la señal análoga de 4-20 mA, utilizada por los instrumentos tradicionales, provee una excepcional solución de comunicación compatible con gran parte de la base de instrumentos instalados y en uso actualmente, esta característica de compatibilidad asegura que el cableado existente y la estrategia de control actual continuará en el futuro, otra de sus características es la comunicación tipo Maestro-Esclavo.

El protocolo de comunicación HART está basado en el sistema de comunicación telefónica estándar BELL 202 y opera usando el principio del Cambio Codificado de Frecuencia (FSK). La señal digital está construida de dos frecuencias principales, 1200 Hz y 2200 Hz, representando los Bits 1 y 0 (cero), respectivamente. Las ondas seno de estas dos frecuencias están superim-puestas en la señal de corriente continua análoga de 4-20 mA DC. Con esto, cables de comunicación con señales análogas transportan a la vez comunicación análoga y digital

### 2.1.2.2 Comunicación Serial.

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

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

Típicamente, la comunicación serial se utiliza para transmitir datos en formato ASCII. Para realizar la comunicación se utilizan 3 líneas de transmisión: (1) Tierra (o referencia), (2) Transmitir, (3) Recibir. Debido a que la transmisión es asincrónica, es posible enviar datos por una línea mientras se reciben datos por otra. Existen otras líneas disponibles para realizar handshaking, o intercambio de pulsos de sincronización, pero no son requeridas. Las características más importantes de la comunicación serial son la velocidad de transmisión, los bits de datos, los bits de parada, y la paridad. Para que dos puertos se puedan comunicar, es necesario que las características sean iguales.

#### 2.1.2.2.1 Velocidad de transmisión.



Indica el número de bits por segundo que se transfieren, y se mide en baudios, por ejemplo, 300 baudios representan 300 bits por segundo. Cuando se hace referencia a los ciclos de reloj se está hablando de la velocidad de transmisión. Por ejemplo, si el protocolo hace una llamada a 4800 ciclos de reloj, entonces el reloj está corriendo a 4800 Hz, lo que significa que el puerto serial está muestreando las líneas de transmisión a 4800 Hz. Las velocidades de transmisión más comunes para las líneas telefónicas son de 14400, 28800, y 33600. Es posible tener velocidades más altas, pero se reduciría la distancia máxima posible entre los dispositivos. Las altas velocidades se utilizan cuando los dispositivos se encuentran uno junto al otro, como es el caso de dispositivos GPIB.

#### **2.1.2.2.2 Bits de datos:**

Se refiere a la cantidad de bits en la transmisión. Cuando la computadora envía un paquete de información, el tamaño de ese paquete no necesariamente será de 8 bits. Las cantidades más comunes de bits por paquete son 5, 7 y 8 bits. El número de bits que se envía depende en el tipo de información que se transfiere. Por ejemplo, el ASCII estándar tiene un rango de 0 a 127, es decir, utiliza 7 bits; para ASCII extendido es de 0 a 255, lo que utiliza 8 bits. Si el tipo de datos que se está transfiriendo es texto simple (ASCII estándar), entonces es suficiente con utilizar 7 bits por paquete para la comunicación. Un paquete se refiere a una transferencia de byte, incluyendo los bits de inicio/parada, bits de datos, y paridad. Debido a que el número actual de bits depende en el protocolo que se seleccione, el término paquete se usará para referirse a todos los casos.

#### **2.1.2.2.3 Bits de parada**

Usado para indicar el fin de la comunicación de un solo paquete. Los valores típicos son 1, 1.5 o 2 bits. Debido a la manera como se transfiere la información a través de las líneas de comunicación y que cada dispositivo

tiene su propio reloj, es posible que los dos dispositivos no estén sincronizados. Por lo tanto, los bits de parada no sólo indican el fin de la transmisión sino además dan un margen de tolerancia para esa diferencia de los relojes. Mientras más bits de parada se usen, mayor será la tolerancia a la sincronía de los relojes, sin embargo, la transmisión será más lenta.

#### **2.1.2.2.4 Paridad**

Es una forma sencilla de verificar si hay errores en la transmisión serial. Existen cuatro tipos de paridad: par, impar, marcada y espaciada. La opción de no usar paridad alguna también está disponible. Para paridad par e impar, el puerto serial fijará el bit de paridad (el último bit después de los bits de datos) a un valor para asegurarse que la transmisión tenga un número par o impar de bits en estado alto lógico. Por ejemplo, si la información a transmitir es 011 y la paridad es par, el bit de paridad sería 0 para mantener el número de bits en estado alto lógico como par. Si la paridad seleccionada fuera impar, entonces el bit de paridad sería 1, para tener 3 bits en estado alto lógico. La paridad marcada y espaciada en realidad no verifican el estado de los bits de datos; simplemente fija el bit de paridad en estado lógico alto para la marcada, y en estado lógico bajo para la espaciada. Esto permite al dispositivo receptor conocer de antemano el estado de un bit, lo que serviría para determinar si hay ruido que esté afectando de manera negativa la transmisión de los datos, o si los relojes de los dispositivos no están sincronizados.

#### **2.1.3 Diseño de interfaces**

La Interfaz de Usuario, en adelante IU, de un programa es un conjunto de elementos hardware y software de una computadora que presentan información al usuario y le permiten interactuar con la información y con la computadora. También se puede considerar parte de la IU la



documentación (manuales, ayuda, referencia, tutoriales) que acompaña al hardware y al software. (Gómez, 2000)

### 2.1.3.1 Labview

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) es un ambiente para el desarrollo de aplicaciones de propósito general que permite controlar dispositivos y realizar análisis de datos. También utiliza un lenguaje de programación gráfica llamado lenguaje G, el cual a través de diagramas de bloques brinda una solución gráfica a un problema de programación. (National Instruments", s/f)

Este software permite visualizar resultados mediante interfaces de usuario de "clic-arrastre" y visualizadores de datos integrados. Este software garantiza la compatibilidad con otras herramientas ya que puede interactuar o reutilizar bibliotecas de otro software y lenguajes de fuente abierta como Arduino. (National Instruments, s/f)

LabVIEW cuenta con dos interfaces: panel frontal y diagrama de bloques. Estas cuentan con paletas que contienen los objetos necesarios para implementar y desarrollar tareas.

### 2.1.3.2 Interfaz gráfica del operador:

Proporciona al operador las funciones de control y supervisión de la planta. El proceso que se supervisará se representa mediante sinópticos gráficos almacenados en el ordenador y generados desde el editor incorporado en el SCADA o importados desde otra aplicación de uso general etc. durante la configuración del paquete. Los sinópticos están formados por un fondo fijo y varias zonas activas que cambian dinámicamente de formas y colores, según los valores leídos en la planta o en respuesta a las acciones del operador. Deben tenerse en cuenta las siguientes consideraciones a la hora de diseñar las pantallas:

- Las pantallas deben tener apariencia consistente, con zonas diferenciadas para mostrar la planta (sinópticos), las botoneras y entradas de mando (control) y las salidas de mensajes del sistema (estados, alarmas).
- La representación del proceso se realizará preferentemente mediante sinópticos que se desarrollan de izquierda a derecha.
- La información presentada aparecerá sobre el elemento gráfico que la genera o soporta, y las señales de control estarán agrupadas por funciones.
- La clasificación por colores ayuda a la comprensión rápida de la información.
- Los colores deben usarse de forma consistente en toda la aplicación: si rojo significa peligro o alarma y verde indica normalidad, estos serán sus significados en cualquier parte de la aplicación.

Previendo dificultades en la observación del color, debe añadirse alguna forma de redundancia, sobre todo en los mensajes de alarma y atención: textos adicionales, símbolos gráficos dinámicos, intermitencias u otros. La redundancia como un componente de seguridad SCADA consiste en lograr respaldos de información, duplicar (cuando un elemento asume la función de otro), contar con centros de control separados geográficamente que proporcionen redundancia y, por tanto, protección contra los ataques humanos y desastres naturales. Este elemento permite seguir operando, aunque el sistema primario esté desactivado, funcionando incluso de manera remota (Krutz, 2006).

### 2.1.3.3 Modelo del usuario

El usuario tiene su visión personal del sistema, y espera que éste se comporte de una cierta forma. Se puede conocer el modelo del usuario estudiándolo, ya sea realizando test de usabilidad, entrevistas, o a través

de una realimentación. Una interfaz debe facilitar el proceso de crear un modelo mental efectivo.

Para ello son de gran utilidad las metáforas, que asocian un dominio nuevo a uno ya conocido por el usuario. Un ejemplo típico es la metáfora del escritorio, común a la mayoría de las interfaces gráficas actuales.

#### **2.1.3.4 Modelo del diseñador.**

El diseñador mezcla las necesidades, ideas, deseos del usuario y los materiales de que dispone el programador para diseñar un producto de software. Es un intermediario entre ambos.

El modelo del diseñador describe los objetos que utiliza el usuario, su presentación al mismo y las técnicas de interacción para su manipulación. Consta de tres partes: presentación, interacción y relaciones entre los objetos (Figura 2.4).

La presentación es lo que primero capta la atención del usuario, pero más tarde pasa a un segundo plano, y adquiere más importancia la interacción con el producto para poder satisfacer sus expectativas. La presentación no es lo más relevante y un abuso en la misma (por ejemplo, en el color) puede ser contraproducente, distrayendo al usuario.

La segunda parte del modelo define las técnicas de interacción del usuario, a través de diversos dispositivos.

La tercera es la más importante, y es donde el diseñador determina la metáfora adecuada que encaja con el modelo mental del usuario. El modelo debe comenzar por esta parte e ir hacia arriba. Una vez definida la metáfora y los objetos del interfaz, los aspectos visuales saldrán de una manera lógica y fácil

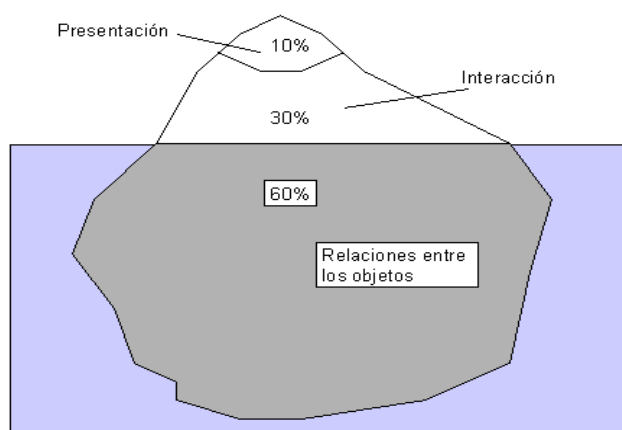


Figura 2.4 modelo de diseño

## 2.2 Sistema de monitoreo de fugas hidráulicas

### 2.2.1 Fugas hidráulicas

Una fuga es una salida de agua no controlada en cualquiera de los componentes del sistema de distribución de agua potable; con mayor frecuencia ocurren en uniones de tuberías, codos, roturas de conductos y válvulas.

En los sistemas de conducción de agua a presión es común que se presenten fugas de este líquido. Las fugas pueden deberse al agrietamiento transversal, aplastamiento o agrietamiento longitudinal de las tuberías, la corrosión, el mal junteo de los tubos o la falla de las válvulas que pueden incrementar las fugas en una red.

### 2.2.2 Tipos de fugas

Las fugas pueden clasificarse de acuerdo a si son visibles u ocultas. Las fugas visibles brotan a la superficie y humedecen el terreno y son identificadas y/o reportadas por algún observador. Las fugas ocultas generalmente se filtran y no se sabe su localización, dichas fugas son pequeñas al inicio y crecen con el tiempo. Algunas de las fugas ocurren en el cuerpo de la tubería y otras en los accesorios de la misma. Además, las

fugas se pueden clasificar dependiendo de su forma en: fisuras u orificios; permitiendo que a su vez las rajaduras se dividan en: longitudinales, transversales y combinadas; tal como se observa en la Tabla 2.1 [O'Day, 1982].

Fugas	Rajadura	Longitudinal	
		Transversal	
		Combinada	
	Orificio	Orificio	

Tabla 2.1 tipos de fugas hidráulicas

## 2.2.3 Métodos de detección de fugas.

Existen diversos sistemas de detección de fugas en tuberías, dichos sistemas cuentan con múltiples ventajas y desventajas sobre otros sistemas, por lo que, al haber una amplia gama de necesidades también existe una amplia gama de características propias de cada sistema. (Rosas, 2017)

### 2.2.3.1 Sistema acústico directo

El sistema acústico directo de detección de fugas es uno de los sistemas más utilizados actualmente, debido a su fácil implementación tanto en fugas domésticas como industriales. Este sistema consiste en la inspección personal de un operador, el cual cuenta con un equipo que le permite escuchar el sonido emitido por el fluido que escapa de la tubería que lo transporta y así ubicar su posición [Hunaidi, 2000].

Dicho equipo consiste en un par de hidrófonos que en un principio captan la señal acústica y la transmiten a un sistema de amplificación (46 dB), el cual finalmente la transmite a los audífonos que porta el operador. El

sistema de amplificación también cuenta con un regulador de volumen para proporcionar protección a la audición del operador, el cual puede observar mediante una pantalla las variaciones en los decibeles que están en función de la posición del hidrófono con respecto de la fuga; a lo largo de la tubería Figura 2.5 (Sewering, 2012).

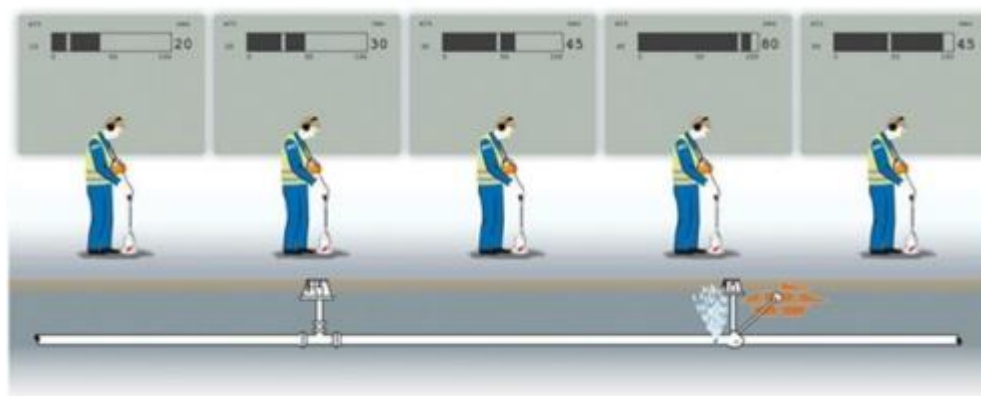


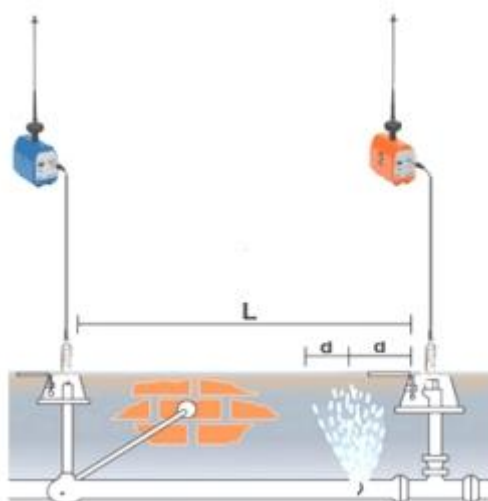
Figura 2.5 Operación para la detección de la fuga por medio del sistema acústico directo.

Este método tiene la ventaja de que solo requiere un operador para el equipo, pero aun así tiene la desventaja de que se requiere que el operador cuente con experiencia en el manejo del equipo, bajos niveles de perturbación en la superficie donde se está censando el sonido de la fuga y que además ésta no ocurra en una tubería de plástico (Hunaidi O., 2000), debido a que se han presentado complicaciones en la localización de la fuga, ya que el plástico tiende a amortiguar la emisión acústica de la fuga.

### 2.2.3.2 Sistema acústico por correlación

El sistema acústico de detección de fugas por correlación es un sistema implementado en varias partes del mundo, debido a que es posible recuperar la inversión inicial en poco tiempo determinado. Este sistema consiste en la sujeción de un hidrófono o acelerómetro (dependiendo de la exactitud deseada) a dos terminales de una tubería entre las cuales se espera identificar la fuga (Fujitecom, 2010). Los hidrófonos captan la señal

acústica y vía inalámbrica la envían a un sistema de correlación (Fujitecom, 2010). Dicho sistema llevará a cabo un proceso de correlación basado en los tiempos y distancias que tarda en recorrer la tubería dicha señal acústica captada por cada hidrófono (SSLD, 2009) (Figura 2.6).



*Figura 2.6 Operación de la detección de la fuga por medio del sistema acústico de correlación.*

De la Figura 2.6 se puede apreciar cómo la correlación es posible debido a que existe una diferencia en las distancias que hay entre la fuga y ambas terminales, provocando así un retardo entre ambas señales, al detectar este retardo y conociendo la velocidad de propagación del sonido “c” dentro de la tubería, se determina la posición de la fuga con respecto a ambas distancias. Una vez detectada la fuga, la pantalla del sistema de correlación muestra la ubicación de la fuga con respecto de ambas terminales, así como el retraso de tiempo que hubo en la señal.

### 2.2.3.3 Sistema de sonda de rastreo

El sistema de sonda de rastreo ha sido ampliamente aceptado por la comunidad industrial, debido a que es aplicado a sistemas donde se requiere inspeccionar grandes distancias. La sonda insertada puede ser alámbrica o inalámbrica este sistema consiste en insertar la sonda de rastreo en una toma y al finalizar su recorrido, se procede a retirar la sonda



por otra toma. La sonda viaja registrando (Figura 2.8) una señal acústica conforme se desplaza a lo largo de la tubería, para que al extraerla se proceda a analizar los datos registrados. Cuando la sonda es inalámbrica viaja a través de la tubería arrastrada por el flujo y al final de su recorrido se extrae por medio de una red (PT, 2013) (Figura 2.7a-b).



Figura 2.7. Sondas de rastreo: a) Inalámbrica. b) Alámbrica

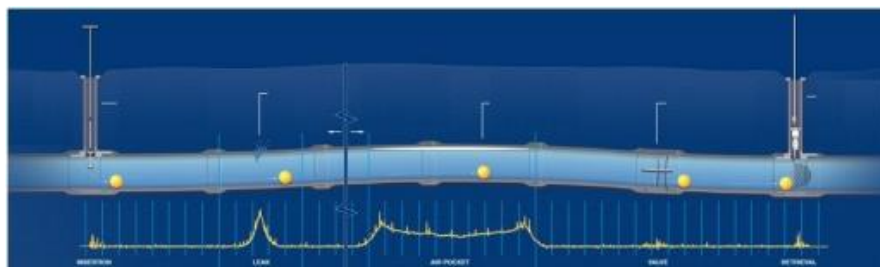


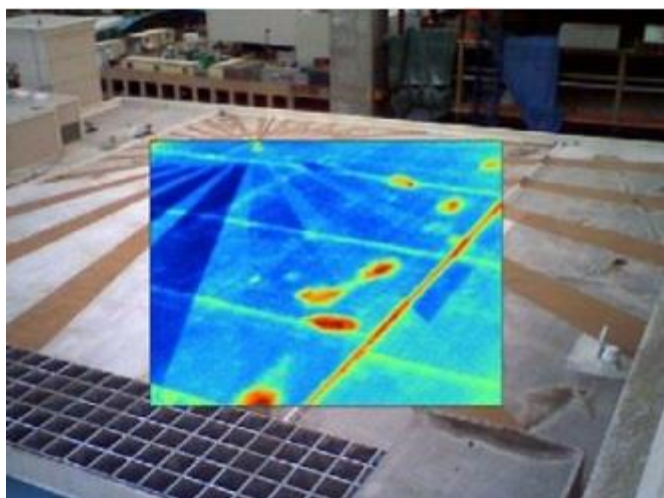
Figura 2.8 Operación de la detección de la fuga por medio del sistema de sonda de rastreo

#### 2.2.3.4 Sistema de fotografía termográfica

El sistema de fotografía termográfica para detección de fugas es uno de los sistemas más eficaces en cuanto a rapidez de implementación y sobre todo es común ver su aplicación al tratar de inspeccionar las fachadas de los edificios para identificar tanto fugas de agua en intramuros, como fugas en el aire acondicionado en ventanas, techos y demás lugares poco accesibles. Este sistema que consiste en la detección de fugas por medio de termografía. La radiación infrarroja emitida por un objeto es detectada y medida por medio de un radiómetro infrarrojo, el cual contienen un sensor que convierte la radiación emitida por los objetos en señales eléctricas que



a su vez son expresadas en una escala de colores, haciendo posible así la detección de la fuga (Figura 2.9) (Fluke, 2013).



*Figura 2.9 Operación de la detección de la fuga por medio del sistema de termografía*

Este sistema cuenta con la gran ventaja de que puede localizar más de una fuga al mismo tiempo, así como también no se requiere de gran experiencia para operar la cámara termográfica y que ésta a su vez, permite tomar fotografías de forma continua cada 2 s; así como también el hecho de que puede operar en un rango de temperaturas  $-20^{\circ}\text{C}$  a  $600^{\circ}\text{C}$  del objeto fotografiado con una precisión del 2% (Fluke, 2013).

La desventaja que presenta este sistema radica principalmente en que no puede cuantificar la fuga y que, además, el operador debe de tener nociones de donde se localiza la fuga, de lo contrario tendría que fotografiar la red de tuberías por completo.

#### **2.2.3.5 Sistema del gas trazador.**

El sistema del gas trazador ha sido ampliamente aceptado por la industria, debido a su versatilidad en cuando a la detección de fugas en redes y ha sido ampliamente aceptada en la industria que maneja flujo de gas por su gran sensibilidad. Este sistema consiste en la detección de fugas por medio de la inspección personalizada de un operador, el cual cuenta con un equipo

de medición capaz de censar el gas trazador que previamente es inyectado a la tubería que se desee inspeccionar y que escapa por la fuga.

En el caso de que las tuberías se encuentren enterradas se le puede añadir un colorante al gas trazador a manera de tener una ayuda visual al momento de la inspección (Sewering,2013) (Figura 2.8).

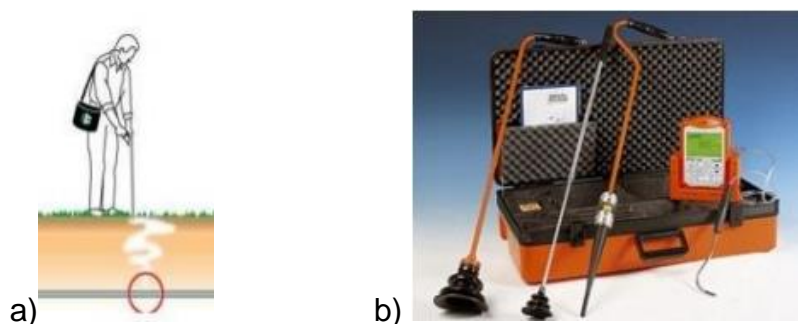


Figura 2.10 Sistema del gas trazador. a) Operación de la detección de la fuga. b) Equipo de inspección.

# 3 . Capitulo III: Análisis y diseño

### 3.1 Diseño de adquisición de datos para sensores de flujo y presión

#### 3.1.1 Arquitectura de Arduino MEGA

El Arduino MEGA es una de las múltiples placas de Arduino, es un microcontrolador basado en ATmega 2560. Cuenta con 54 puertos o terminales de conexión que pueden ser codificadas como entradas o salidas.

Se optó por implementar Arduino mega para la adquisición de datos procedentes de los sensores debido a que tiene una mayoría de terminales dedicadas a interrupciones por hardware una ventaja ante las demás placas así mismo, un mayor número de terminales de puerto serial y la facilidad de conexión con otros dispositivos periféricos.

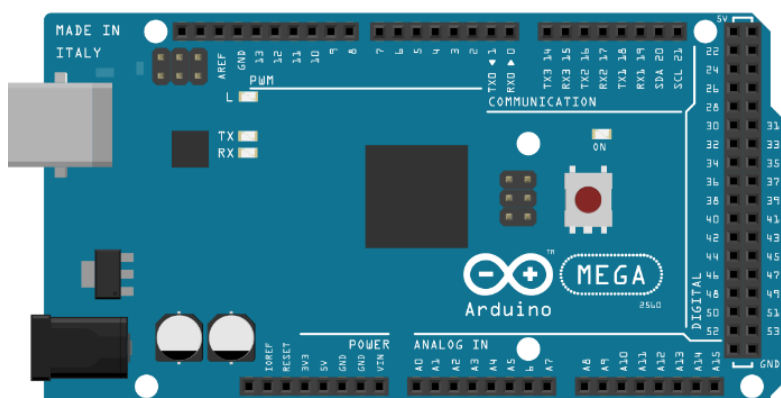


Figura 3.1 . Arduino MEGA

Dentro de las características principales del Arduino MEGA se mencionan los siguientes datos técnicos Tabla 3.1:

Arduino MEGA	
Microcontrolador	ATmega 2560
Voltaje de operación	5V
Voltaje de entrada(recomendado)	7-12V
Voltaje de entrada(límites)	6-20V

Pines digitales I/O	54 (15 con salida PWM)
Pines digitales	16
Pines de interrupciones	5
Corriente DC por pin I/O	40 mA
Corriente DC para el pin 3.3V	50 mA
Memoria Flash	128 KB de los cuales 4 KB usados para bootloader
SRAM	8 KB
EEPROM	4 KB
Velocidad del reloj	16 MHz

*Tabla 3.1 Características del Arduino Mega*

### 3.1.2 Métodos de conexión entre Arduino y computadora

Existen diversos métodos para establecer comunicación entre Arduino y la PC, lo más habitual y común es conectarlo mediante cable USB por el puerto serial, dado que la comunicación casi siempre es por serial, y Arduino cuenta con pines del puerto serie es fácil conectar dispositivos que cuentan con pines de comunicación serial.

#### 3.1.2.1 Conexión inalámbrica por modulo wifi.

Al igual que la comunicación por cable USB, la comunicación con el módulo se realiza mediante puerto serie. A diferencia de la alimentación del módulo, que en ningún caso debe exceder los 3.6V, existe un amplio debate sobre si los pines RX y TX (Figura 3.3) son tolerantes a 5V (Es decir, si pueden conectarse directamente a un sistema a un 5V).

En principio, conectar RX y TX a un sistema de 5V no parece dañar la placa de forma inmediata. De hecho, encontramos indicios en la documentación que apuntan a que podrían ser tolerantes a 5V, sin quedar totalmente claro. ara la conexión con Arduino podemos emplear el puerto serie normal, pero esto implica que tendremos que desconectar el ESP01 cuando queramos

programar Arduino, y no podremos usar el puerto serie para la comunicación con el PC.

Por este motivo frecuentemente se emplea un puerto serie por software. No obstante, tener en cuenta que esto supone una carga de procesamiento importante para Arduino.

Por tanto, la conexión vista desde Arduino sería la siguiente como se muestra en la Figura 3.2.

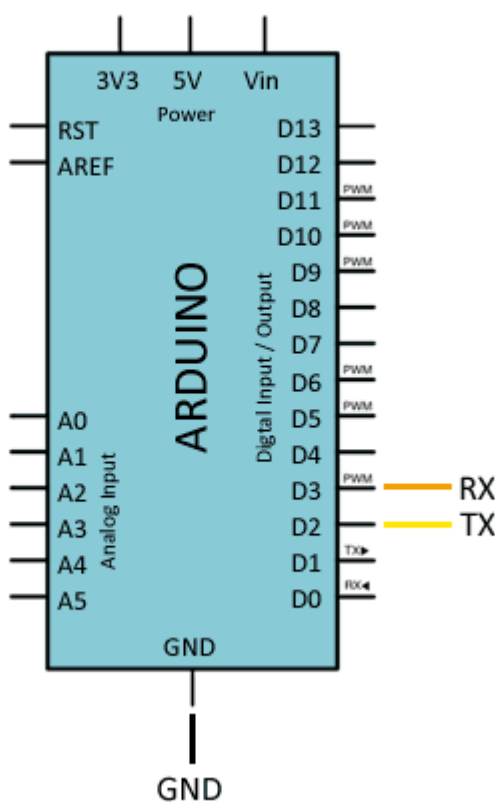
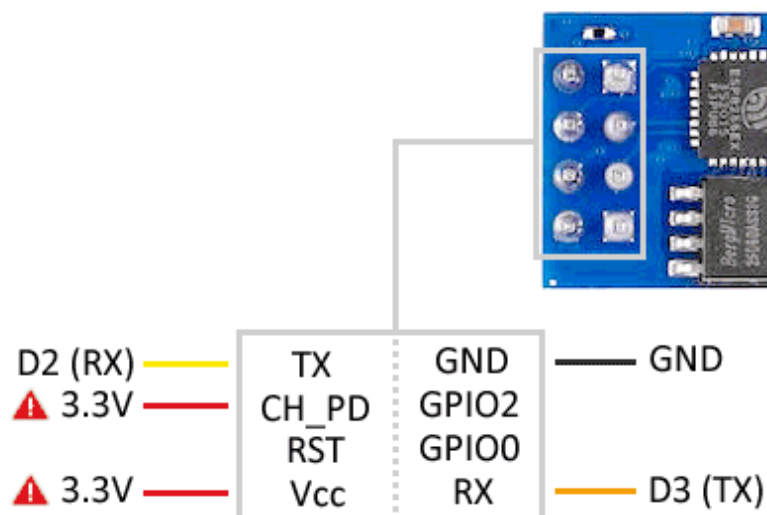


Figura 3.2 Conexión del módulo wifi a Arduino

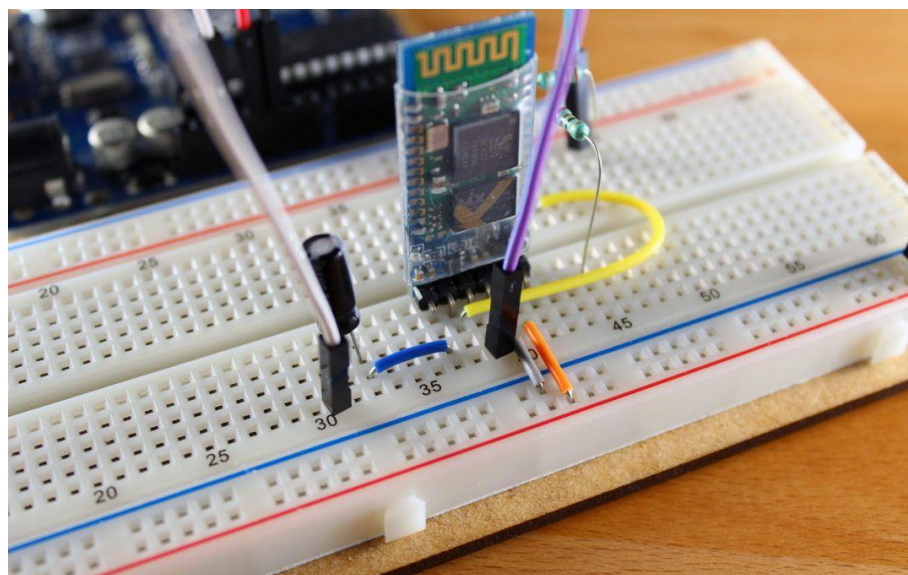
En cuanto a las velocidades, el ESP01 puede configurarse a 9600, 19200, 38400, 74880, 115200, 230400, 460800 y 921600. Por defecto suelen venir, según fabricante, a 9600 o 115200.



Vcc 3.3V. Al usar alimentación externa SIEMPRE poner GND común.

Figura 3.3 pines de conexión del módulo wifi

### 3.1.2.2 Conexión inalámbrica de Arduino por Bluetooth



Configurando el módulo HC-05 para la programación inalámbrica de Arduino por Bluetooth



Antes de que el HC-05 funcione como un “programador de Arduino”, debemos configurar el módulo para que se comporte como lo requerimos. Debemos asegurarnos que el módulo tenga los siguientes parámetros ajustados:

En el caso de Arduino UNO con Optiboot: UART configurada a 115200 baudios, 8 bits, 1 bit de parada, sin paridad

En el caso de Arduino Duemilanove / Decimilia: UART configurada a 56700 baudios, 8 bits, 1 bit de parada, sin paridad

Configurar la polaridad del pin de state para la polaridad adecuada

Los comandos a enviar al HC-05 para configurarlo son los siguientes:

- 1 AT+ORGL
- 2 AT+ROLE=0
- 3 AT+POLAR=1,0
- 4 AT+UART=115200,0,0
- 5 AT+INIT

Conexión del Bluetooth HC-05 para la programación inalámbrica de Arduino por Bluetooth

Conectaremos los pines TX y RX del bluetooth a los pines RX y TX del Arduino respectivamente (van cruzados). El pin status del bluetooth pasará a funcionar como reset y debemos conectarlo al pin “reset” del Arduino a través de un capacitor cerámico de 100 nF. Las resistencias que se muestran en el circuito son para reducir el voltaje del pin TX (salida de 5 volts) de Arduino a algo cercano a 3.3 Volts, que es el nivel lógico que admite el módulo bluetooth. Las conexiones van como se muestra en la siguiente Figura 3.4:



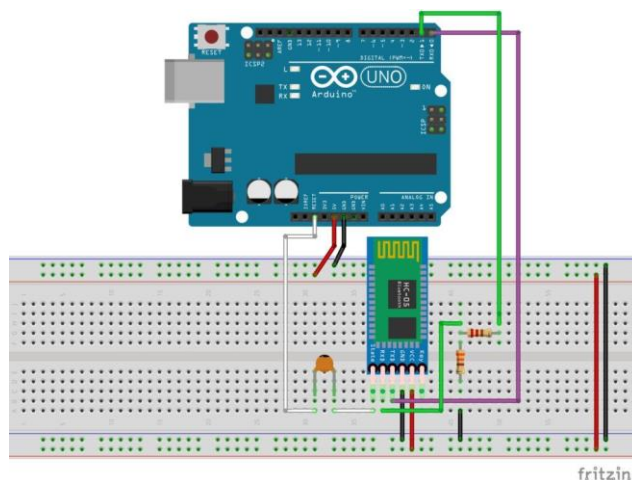


Figura 3.4 conexión de modulo bluetooth al puerto serie de Arduino.

En este momento ya tenemos el hardware listo para intentar realizar la programación de Arduino por Bluetooth. Por lo que podemos abrir nuestro IDE y realizar el proceso de forma habitual, veremos que es muy similar a lo que haríamos para cargar un sketch mediante un cable USB.

Abrimos el IDE de Arduino como es habitual y seleccionamos el nuevo puerto serie () que se creó cuando emparejamos el bluetooth con la PC.

La carga del programa debería transcurrir de forma normal, hasta que se comience a ejecutar el código cargado en el Arduino. Si hay errores puede deberse a que conectamos alguna de las señales incorrectamente o que no configuramos correctamente la polaridad del pin “state”.

La ventaja de tener una interfaz en Labview por puerto serial es que permite la conexión entre cualquier dispositivo sin tener que cargar un firmware que comunique a ambos.

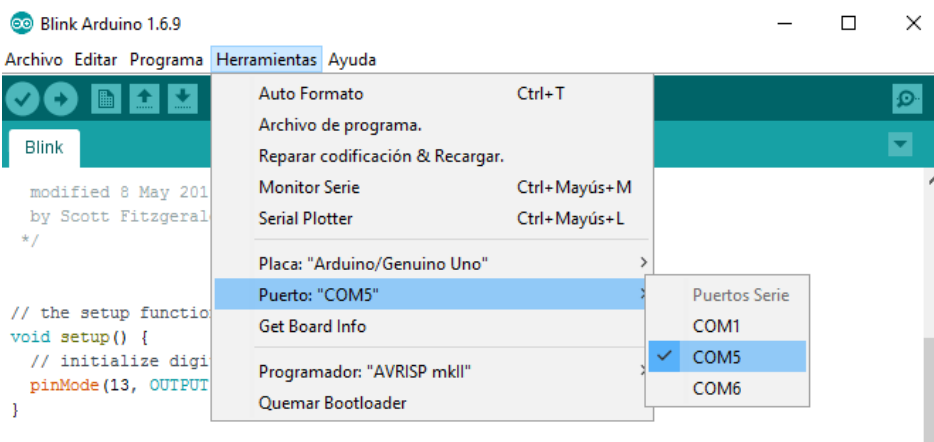


Figura 3.5 Puerto serie del módulo bluetooth conectado a Arduino.

### 3.1.3 Sensor de flujo.

Los sensores de flujo YF-S201, FS300A, FS400A ver tabla son muy usados en varias aplicaciones, puesto que el flujo o caudal es parámetro necesario en varios procesos, a nivel doméstico lo podemos usar para medir el consumo de agua.




Sensor de flujo	Modelo	Conexión	Caudal
	YF-S201	½"	1-30 L/min
	FS300A	¾"	1-60 L/min
	FS400A	1"	1-60 L/min

Tabla 3.2 Modelos de sensores efecto hall

Fuente: *naylamps mechatronics. 2018*

El sensor internamente tiene un rotor cuyas paletas tiene un imán, la cámara en donde se encuentra el rotor es totalmente aislado evitando fugas de

agua, externamente a la cámara tiene un sensor de efecto hall que detecta el campo magnético del imán de las paletas y con esto el movimiento del rotor, el sensor de efecto hall envía los pulsos por uno de los cables del sensor, los pulsos deberán ser convertidos posteriormente a flujo pero esto ya es tarea del Arduino o controlador que se desee usar.

Es importante mencionar que el medidor de flujo, detallado en la Tabla 3.2 y la Figura 3.2, es la parte más importante de la tubería. El sensor se encuentra en la línea de los tubos y. El sensor es un dispositivo rentable pero no es un medidor de flujo de precisión. Sin embargo, es adecuado Para las tareas requeridas en este experimento. (Adafruit Flow Meter, 2014)

La conexión entre arduino y el sensor de flujo lo describen en (naylamp mechatronics, 2018) ver figura 3.2

Como se puede observar el cable amarillo va conectado directamente al pin 2 del Arduino, utilizamos este pin porque en el programa vamos a usar la interrupción externa. Arduino Uno solo tiene interrupciones externas en los pines 2 y 3.

Podemos usar la misma conexión si estamos trabajando con un Arduino Nano, Mini, Mega o Micro, puesto que todas estas placas tienen interrupción externa en el pin 2.

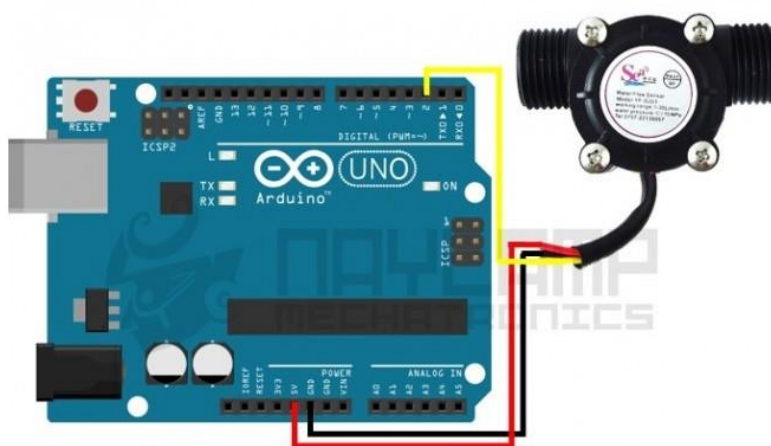


Figura 3.6 Conexión entre sensor YF-S201 y Arduino

Datos técnicos de sensor de flujo YF-S201	
Tensión de trabajo	5 to 18VDC
Máximo consumo de corriente	15mA @ 5V
Caudal de trabajo	1 to 30 Litros/Minuto
Rango de temperatura de	-25 to 80°C
Rango de humedad de trabajo	35%-80% RH
Presión máxima del agua	2.0 Mpa
Ciclo de trabajo de salida	50% +-10%
Características del pulso de	Frecuencia (Hz) = 7.5 * Flujo (L/min)
Pulsos por litro	450
Diámetro de tubería permitida	1/2"
Tamaño	2.5" x 1.4" x 1.4"

### 3.1.3.1 Medición de flujo.

La medición de flujo según (Naylamp mechatronics, s/f) es necesario saber cómo dato el factor de conversión de frecuencia a caudal, dicho dato nos lo da el fabricante.

Teniendo el factor de conversión, el programa se resume a calcular la frecuencia de los pulsos del sensor, para posteriormente con el factor de conversión escalar de frecuencia a caudal.

A continuación, se muestra el programa para calcular la frecuencia de los pulsos y el caudal de agua que realiza Naylamp Mechatronics, 2016):

```
void loop ()
{
    float frecuencia=ObtenerFrecuencia(); //obtenemos la Frecuencia de los pulsos
    en Hz
    float caudal_L_m=frecuencia/factor_conversion; //calculamos el caudal en L/m
    float caudal_L_h=caudal_L_m*60; //calculamos el caudal en L/h

    //-----Enviamos por el puerto serie-----
    Serial.print ("FrecuenciaPulsos: ");
```

```
Serial.print (frecuencia,0);
Serial.print ("Hz\tCaudal: ");
Serial.print (caudal_L_m,3);
Serial.print (" L/m\t");
Serial.print (caudal_L_h,3);
Serial.println ("L/h");
}
```

El resultado se muestra en el monitor serial de Arduino IDE Figura 3.7

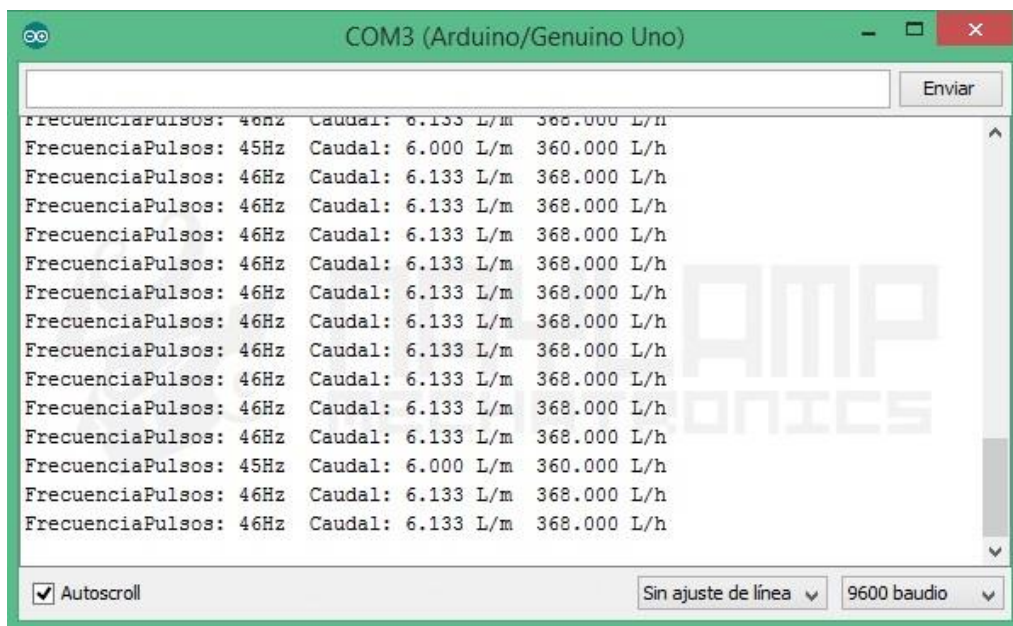


Figura 3.7 Monitor serial para el cálculo de flujo

Fuente: Naylamp Mechatronics, 2016

La medición de flujo para este trabajo se hizo mediante sensores de efecto hall YF-S201(figura 3.2) ya descritos anteriormente, estos sensores según sus hojas de especificaciones se calcula el flujo en función de la frecuencia (tabla 3.2) en un rango de 2 a 30 L/min.



Figura 3.8 Caudalímetro Modelo YF- S201 Fuente: Sea

.Flow Range:100L/H-/1800H-L/H

Flow (L/H)	Frezq.(HZ)	Erro range
120L/H	16	±10
240L/H	32.5	
360L/H	49.3	
480L/H	65.5	
600L/H	82	
720L/H	90.2	

Tabla 3.3 Medición de flujo en relación a la frecuencia por sensor yf-s201

### 3.1.3.2 Algoritmo de medición de flujo y programación en Arduino.

En base a lo citado anteriormente y mediante los datos proporcionados por las especificaciones del sensor se puede calcular el flujo en litros/minutos de la siguiente forma.

- Se calcula el número de pulsos que el sensor envía en un determinado tiempo.
- Se obtiene la frecuencia a partir de la siguiente ecuación 3.1



- (3.1)  $f = \frac{\text{numero de pulsos}}{\text{tiempo}}$

- Mediante la siguiente ecuación 3.2 se obtiene el flujo en l/min

$$(3.2) q = \frac{f}{\text{Factor de conversion}}$$

Donde el factor de conversión es una constante obtenida de la tabla 3.2.

Dado a que los sensores envían pulsos cuando existe paso de líquido en ellos la programación (ver anexo 3.1) se realizó en Arduino haciendo uso de interrupciones y funciones de conteo por hardware para detectar el mínimo pulso y evitar errores en la medición, posteriormente se calcula el flujo mediante las ecuaciones 3.1 y 3.2 y se envía por el puerto serial a Labview para su procesamiento y monitoreo.

### 3.2 Diseño de una interfaz para el monitoreo y adquisición de datos

Gracias a la gran variedad de funciones, con las que cuenta LabVIEW, nos fue posible desarrollar un sistema de monitoreo y control en tiempo real, implementando la adquisición de señales analógicas y digitales, así como la comunicación serial; además de realizar cálculos, manejo de base de datos, generación de reportes utilizando hojas de cálculo y despliegues gráficos de la información.

En este trabajo se diseñó una interfaz gráfica en la que se lograra monitorear los flujos y presiones del sistema en una gráfica así también se realizó el diseño del HMI para el control de la bomba y visualización de variables.

### 3.2.1 HMI del sistema de monitoreo.

Para la presentación de los datos del sistema de tuberías se realizó un HMI (figura 3.3) en Labview, en ella se puede observar los flujos y presiones, el control del encendido y apagado de la bomba, así como un control para la variación del flujo mediante la bomba, también cuenta con indicadores alarma de presencia de fugas el cual el sistema de monitoreo utiliza para iniciar el algoritmo de localización de fugas.

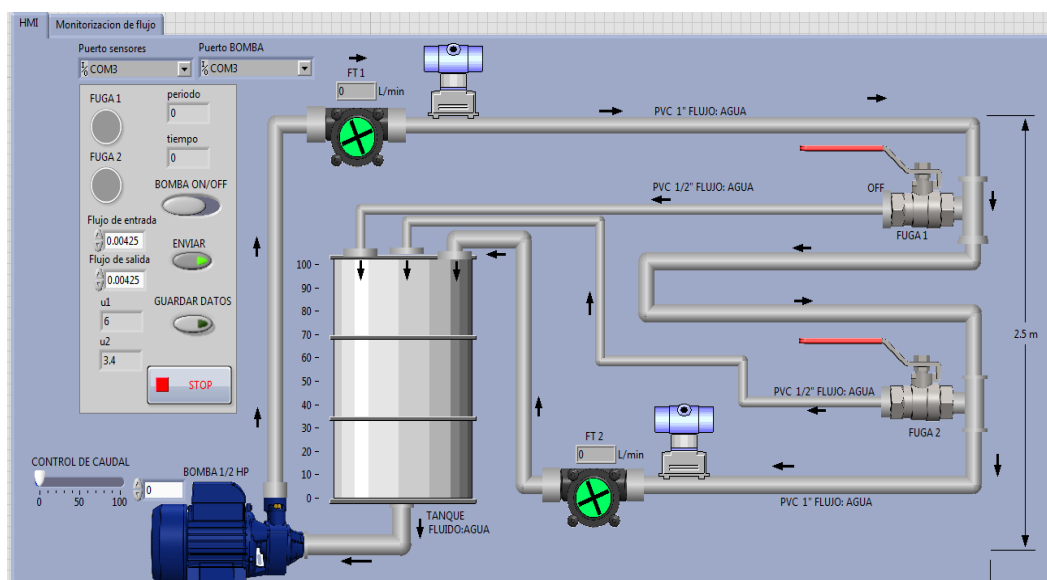


Figura 3.9 HMI del sistema de monitoreo

Fuente: Elaboración propia.

El hmi se diseñó con la posibilidad de registrar y guardar los datos, a un archivo Excel, el operador puede decidir si quiere guarda los datos o no mediante un botón.

En el panel de control (figura 3.5) se muestran datos como el tiempo, parámetros de simulación de la tubería, un botón para él envío de datos a Arduino y un botón para detener el programa.

También se incluye controles para los parámetros de la simulación del modelo matemático que representa la tubería, tales como flujos de entrada y presiones.



En el capítulo IV se describe con más detalle lo anterior y se muestran los resultados obtenidos.

En la figura Figura 3.9 se puede apreciar que el HMI cuenta con dos puertos serial, uno está dedicado a los sensores y otro al variador de frecuencia para la bomba.

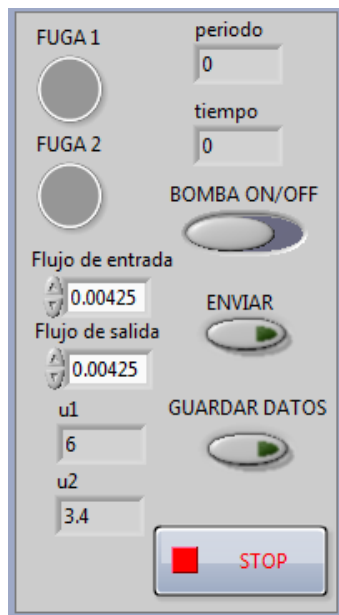


Figura 3.10 Panel de control del HMI del sistema de monitoreo.

Fuente: elaboración propia

La interfaz también incorpora una pestaña en la cual se visualiza gráficamente el flujo de entrada y salida en tiempo real (figura 3.4).

### 3.2.2 Comunicación entre Labview y Arduino.

La comunicación entre Labview y arduino se hizo mediante dos formas, implementando el toolkit de Maker Hub LINX y VISA o comunicación serial. En los siguientes puntos se describe cada método empleado, haciendo énfasis en la comunicación hecha a partir de VISA o serial, ya que según los resultados muestra un mínimo de error en comparación con LINX.

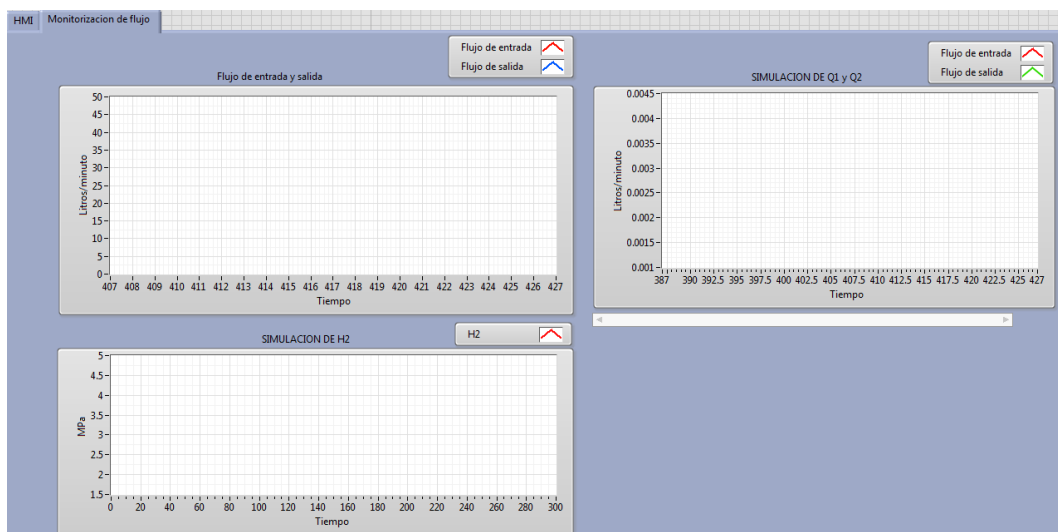


Figura 3.11 visualización grafica de flujos y simulación de tubería

Fuente: elaboración propia

### 3.2.2.1 Linx.

En la interfaz hecha a partir del complemento LINX para la lectura de cada sensor la programación para el cálculo de pulsos, frecuencia, flujo y volumen se realizó desde Labview (Figura 3.14) haciendo uso de los bloques del complemento, es importante mencionar que LINX carga un firmware al microprocesador de arduino el cual no se puede modificar y depende de los complementos que utilicemos en Labview.

Dado que en LINX carece del uso de interrupciones externas la lectura de los sensores se hace a través del bloque *Digital Read* (Figura 3.12a), el conteo de pulsos mediante el bloque *Crossing* (Figura 3.12b) para la detección del flanco de subida, esta señal a su vez es enviada a un condicional el cual incrementa un contador que guarda el valor de la cantidad de pulsos, el contador es reiniciado a cero cuando se ha alcanzado el tiempo de muestreo, una vez ocurrido esto se calcula la frecuencia y el flujo.

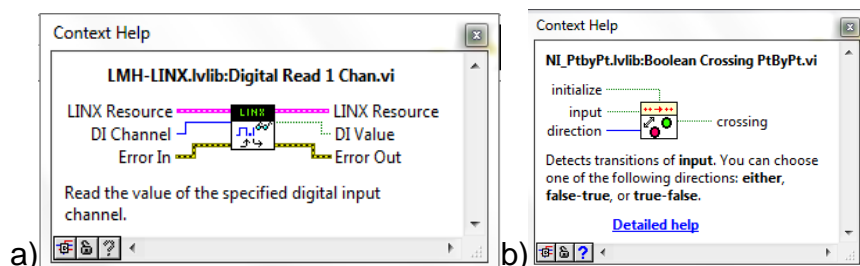


Figura 3.12 a)Bloque Digital Read de LINX, b) bloque cruce por cero

En el panel frontal se muestra una gráfica, indicadores numéricos para los flujos, un control para el rango de la gráfica Figura 3.6

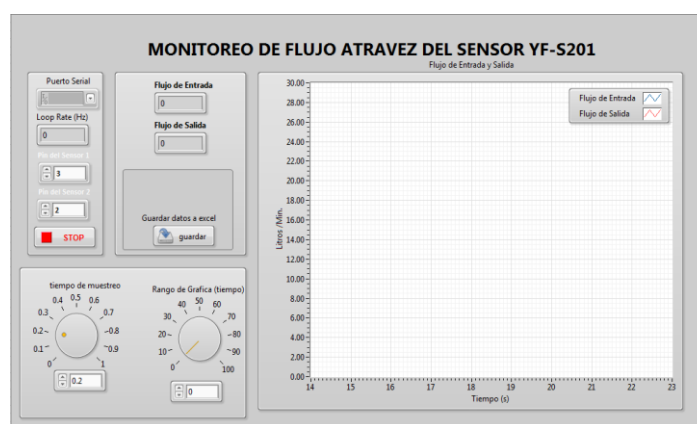


Figura 3.13 Panel frontal de interfaz para el monitoreo de flujos en LINX

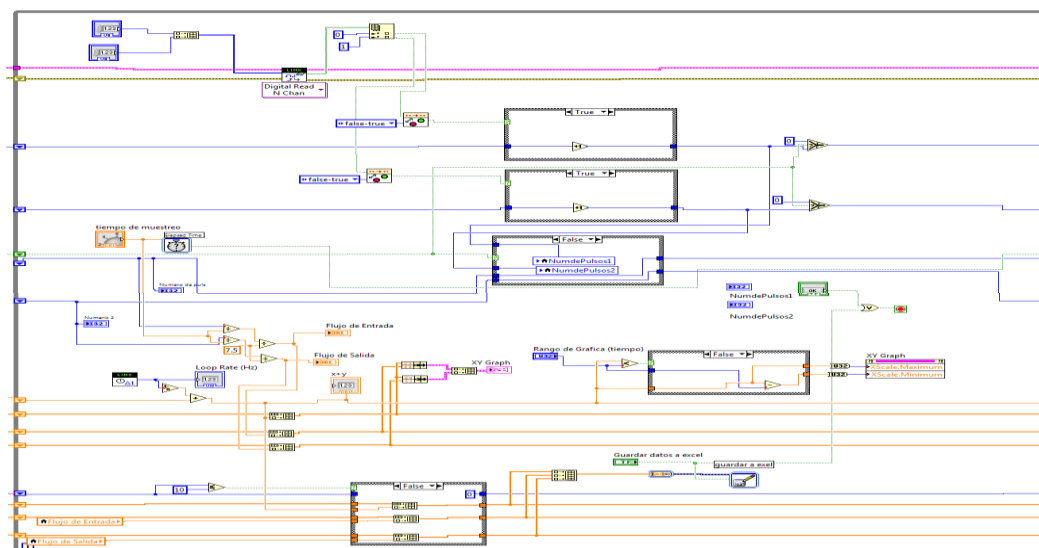


Figura 3.14 Diagrama de bloques de la programación en Labview mediante LINX

### 3.2.2.2 VISA Puerto serial.

El diseño del panel frontal y la visualización de las variables se elaboró en Labview, el cálculo de la frecuencia y la obtención del flujo de cada sensor se hizo mediante un programa en el IDE de Arduino ( Anexo 1 ) dado que se emplean interrupciones por hardware, la primera parte del código se basa en obtener un conteo de pulsos en un determinado tiempo, la segunda parte es hacer el cálculo para la frecuencia, flujo y volumen, por último, se envía por el puerto serie dichas variables en cadena de texto.

Labview recibe los datos de Arduino en forma de caracteres para lo cual gran parte del programa en Labview es el convertir dichos datos a valores numéricos y poder ser representados.

Es por eso que haremos énfasis a la parte más importante del programa la cual describiré más adelante.

La comunicación entre Labview y arduino se hace de forma serial, tanto como arduino y Labview envían datos en cadena de caracteres, el cual en ambos casos se tiene que hacer una traducción o conversión a valores numéricos.

Para dicha comunicación se hizo uso de programación de referencia que incluye (Figura 3.15) Labview. Se puede apreciar que dicho programa ignora caracteres que no sean números.

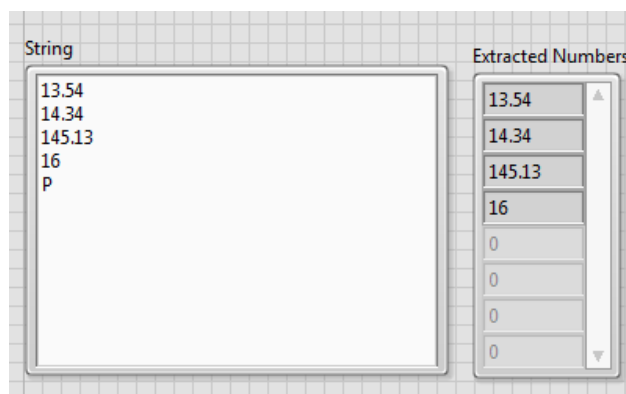


Figura 3.15 Extracción de números de una cadena de caracteres

Fuente: National instruments

La implementación en la interfaz del programa anterior se hizo haciendo un buffer de datos procedentes de arduino, para luego una vez reunidos todos los datos hasta el carácter final, introducirlos al programa que extrae números, es importante mencionar que los datos numéricos se guardan en un array para lo cual después es necesario utilizar un index array. El cual nos devuelve el valor de cada elemento del array Figura 3.16.

En la Figura 3.17 se muestra el uso del programa mencionado anteriormente dentro de la interfaz del sistema de monitoreo.

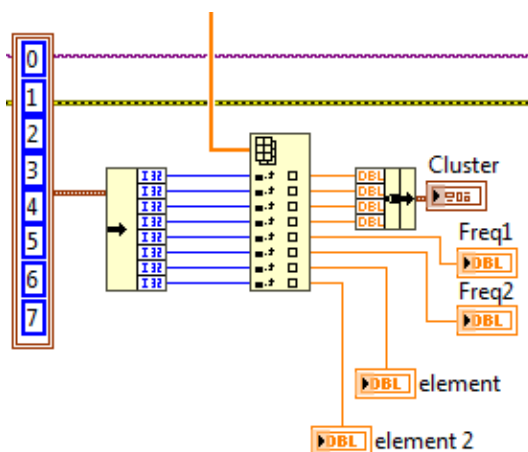


Figura 3.16 obtención de datos del array.

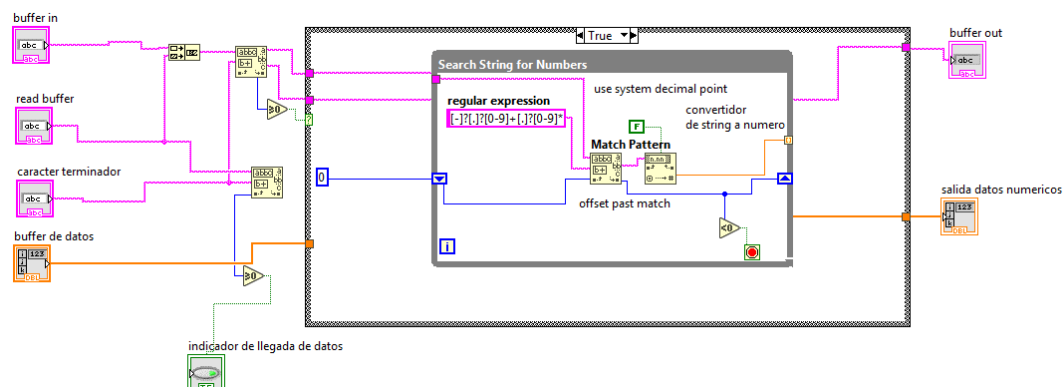


Figura 3.17 Extracción de números adecuado a interfaz de monitoreo

### 3.3 Diseño de un prototipo para la validación de la detección de fugas.

El prototipo consiste en un sistema de tuberías para el monitoreo y detección de fugas el cual tiene la finalidad de hacer circular agua empleando una bomba y un tanque para la reincorporación del agua, dicho prototipo simulara una tubería en constante flujo, los transmisores de flujo y presión colocados en el inicio y fin de la tubería tienen como objetivo medir si existen diferencias en el flujo de entrada y salida, así como las presiones en ambos extremos, de esta forma se podrá determinar alguna posible fuga, para lo cual el prototipo se equipara de varias válvulas que emularan una fuga en un determinado punto de la tubería, para evitar que el agua de las fugas se desperdicie se emplearan tuberías que reincorporen el agua al tanque.

Para la variación del flujo se diseñó un variador de frecuencia incorporado al dispositivo de bombeo el cual fue una bomba de 1/2 HP monofásica 127V. Para la construcción del sistema de circulación de agua se emplearán tubos de PVC hidráulico de 1" y para el sistema de reincorporación de agua de las fugas se empleó PVC hidráulico de 1/2".

#### 3.3.1 Diseño de la planta piloto.

Para el presente trabajo, se consideraron prototipos de tuberías citados, (Delgado, 2014) propone un prototipo el cual transporta agua y cuyos tubos están fabricados con Polipropileno Copolímero Random (PP-R) bajo la norma de fabricación NMX-E-226/2- 1998-SCFI. El diagrama esquemático del prototipo se muestra en la Fig. 3.5.

Un diseño similar es el que se presenta en (Torres, 2014) Donde la tubería se encuentra en circuito cerrado (figura 3.6), se compone de un tanque de 750 (L) (E1) que alimenta de agua toda la tubería, utilizando una bomba centrífuga (E2) aguas arriba. La tubería cuenta también con tres válvulas (V1, V2, V3) que son utilizadas para emular fugas. Estas válvulas se

encuentran a 20 (m), 41,5 (m) y 63 (m) respectivamente a partir del punto de alimentación.

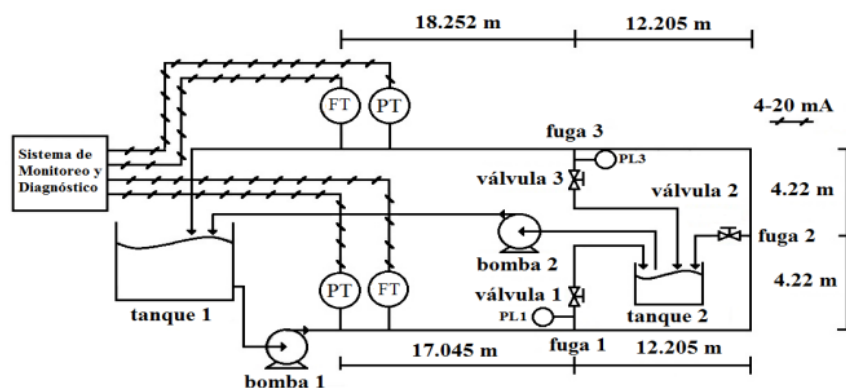


Figura 3.18 Esquema del prototipo de tuberías Fuente: Delgado, 2014

La longitud real de la tubería es de  $L = 68,4$  (m), sin embargo, una longitud equivalente fue estimada considerando la fricción y los accesorios que la componen (codos, conectores, etc.). Si la fricción varía demasiado, la estimación de la longitud equivalente, así como las posiciones equivalentes de las válvulas, pueden estimarse utilizando el observador presentado en Navarro et al. (2011).

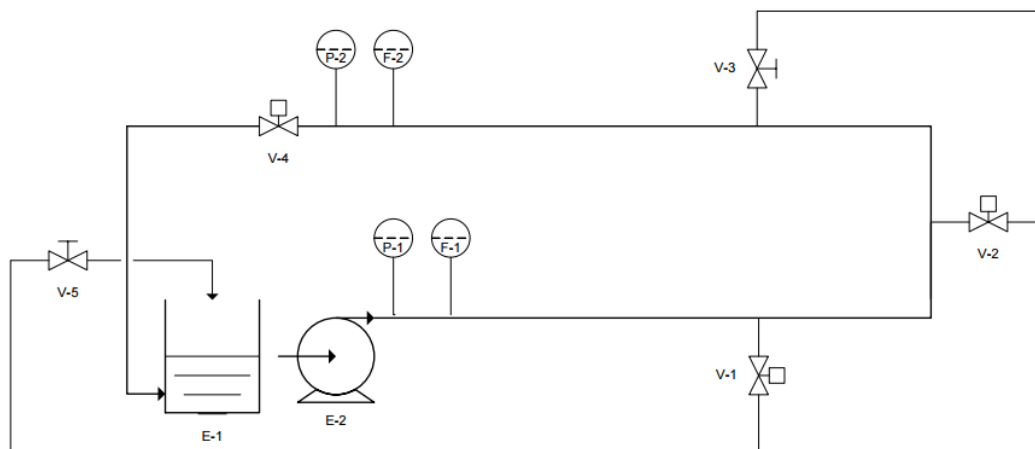


Figura 3.19 DTI propuesto de sistema de tuberías en Torres, 2014

En cuanto a diseño de la planta se contemplaron dos propuestas; un sistema de tuberías con 2 (figura 3.3) y 6 (figura 3.4) líneas, se realizó un diagrama de tuberías e instrumentación del sistema completo para ambos casos en el cual se especifica cada componente.

### 3.3.2 Diseño de construcción en Solidworks

En cuanto a diseño de construcción para el prototipo piloto de validación, se tomó en cuenta trabajos realizados con anterioridad, a continuación, se muestran referencias de diseños.

Como propone (Santos, Bermudez, López, Puig y Torres 2018) un método para pruebas experimentales para la detección de fugas y localización el cual se llevó a cabo en una planta piloto ubicada en el Laboratorio de

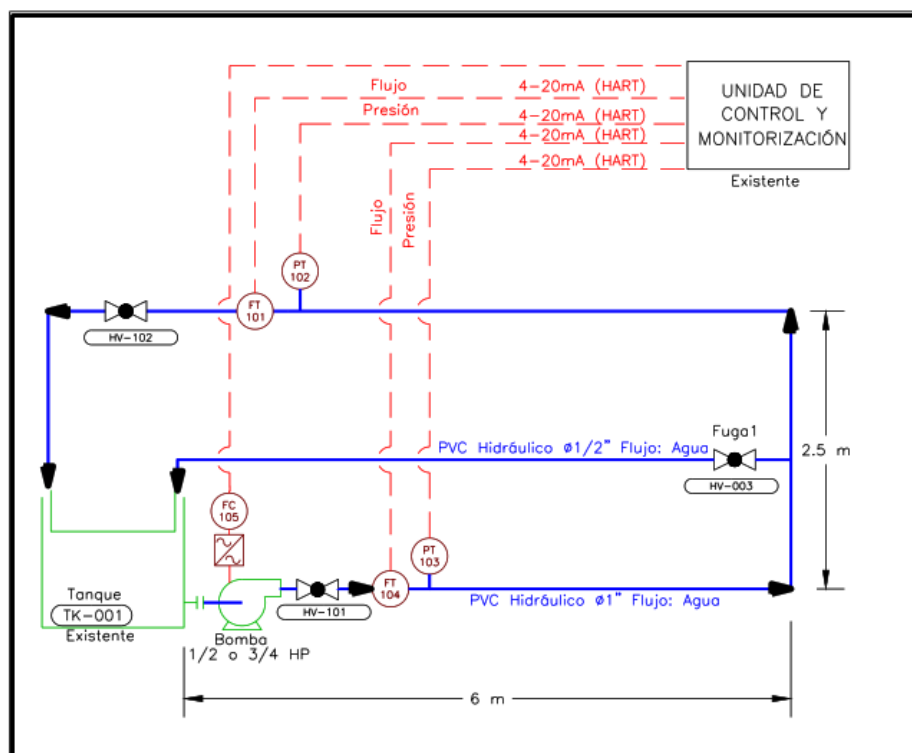


Figura 3.20 Sistema de tuberías con 2 líneas y 1 línea de reincorporación.



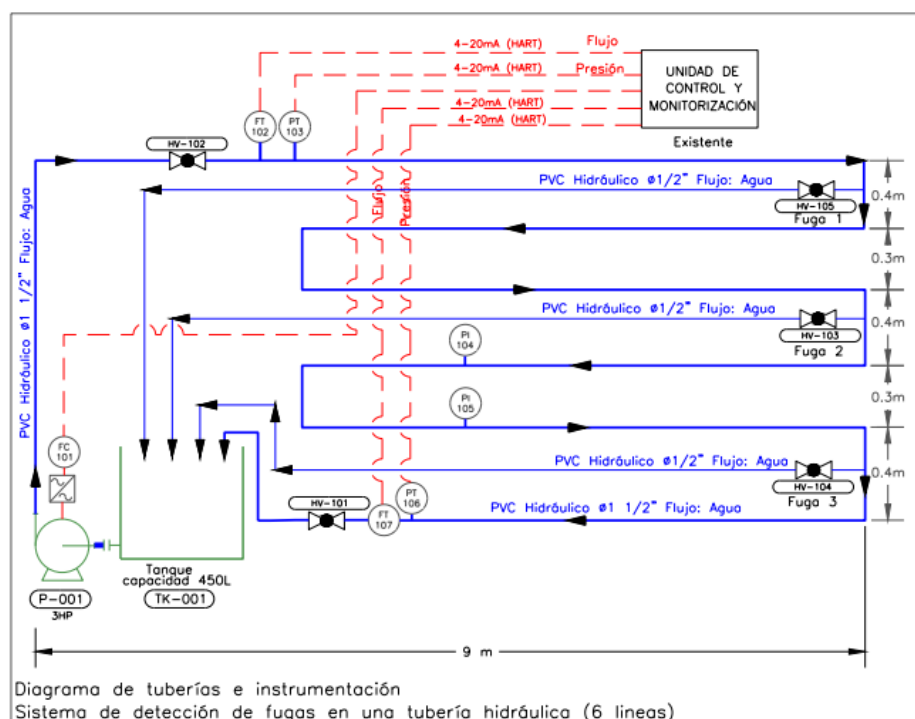


Figura 3.21. sistema de tuberías de 6 líneas con 3 de reincorporación.

hidráulica en el Instituto de Tecnología de Tuxtla Gutiérrez. Esta tubería experimental tiene forma de serpentina en un plano horizontal (Figura. 3.8) con una longitud equivalente de 57.76 m y está hecha de PVC de 2 pulgadas de diámetro, el agua se acciona con una bomba centrífuga cuya potencia es controlada por un variador de frecuencia.

En Torres proponen un prototipo distinto al mencionado; la instalación experimental para detección de fugas es una tubería de acero de cuatro pulgadas de diámetro, cédula 40, en forma de espiral sobre un plano vertical, como lo muestra la figura 4. La entrada del flujo está en la parte inferior y su salida en la parte superior. Se tiene una cisterna de 10 000 litros de agua potable, una bomba hidráulica de 7.5 HP y válvulas de paso en la entrada y en la salida. Los sensores de flujo y presión se ubican en ambos extremos del ducto. Aproximadamente en la parte media de cada tubo horizontal se haya una válvula de 1 in de diámetro para simular fugas.



*Figura 3.22 Prototipo experimental en Tecnológico de Tuxtla Gutierrez.*

*Fuente: (Santos et al., 2018, p.56)*



*Figura 3.23 Prototipo de tuberías en un plano vertical*

Con lo anterior se decidió diseñar un sistema de tuberías en un plano vertical y con la propuesta del DTI de 6 líneas Figura 3.21, la planta se diseñó en solidworks contemplando lo siguiente (:

Parámetro	Valor
Longitud total de tubería	51 m
Diámetro de tubería	2 pulgadas
Tipo de tubería	PVC Sch 40
Altura	2 m
Largo	8 m
Área de instalación	8x9m
1 Tanque	750 L
1 Bomba centrífuga monofásica	3 HP 2 X 1 ½ "
3 Válvulas de fuga	½ "

Tabla 3.4 Parámetros del diseño en 3D

En la Figura 3.24 se muestra una vista isométrica de la planta diseñada para el sistema de monitoreo

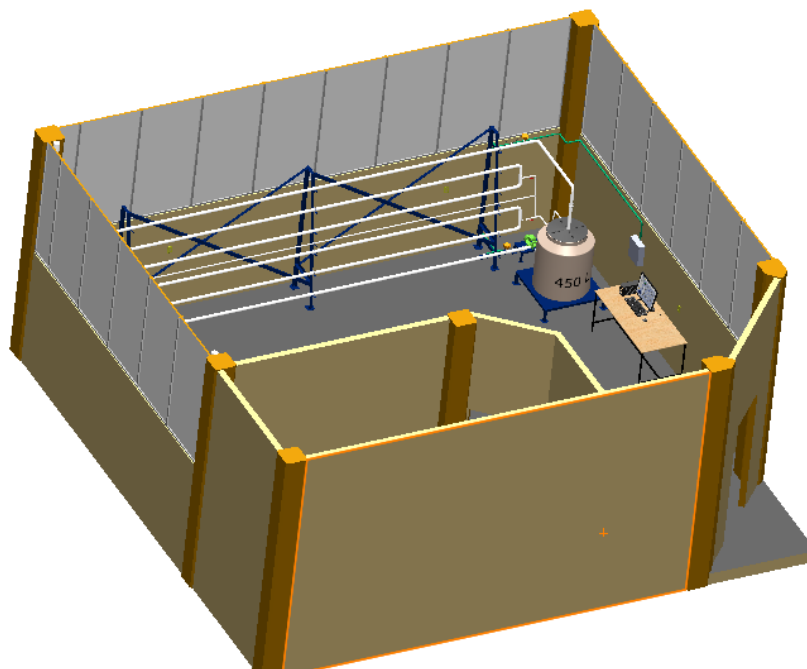
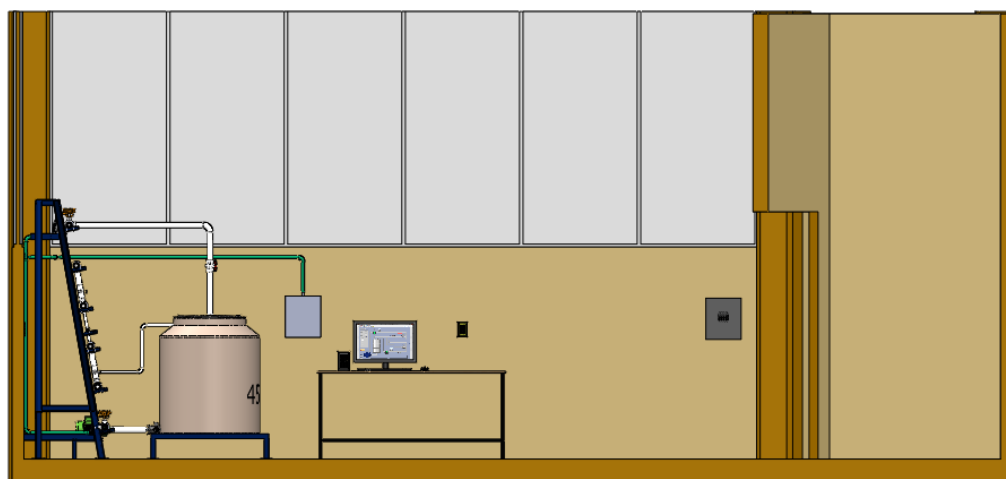


Figura 3.24 Vista isométrica completa de la planta

Fuente: elaboración propia



*Figura 3.25 vista de sección lateral*

*Fuente: elaboración propia*

En la Figura 3.25 se muestra la vista de sección en la cual podemos ver la estación de control con el sistema de monitoreo instalado.

En la Figura 3.26 podemos ver una vista que nos muestra el conexionado de la tubería al tanque, aquí mismo se puede apreciar las estructuras necesarias tanto para la bomba, tanque y tuberías, también se aprecia que el sistema de adquisición de datos está ubicado a un lado del equipo de control.



*Figura 3.26 Vista de detalle de conexión a tanque.*

*Fuente: elaboración propia*

La conexión de la bomba centrífuga la podemos ver en la Figura 3.27

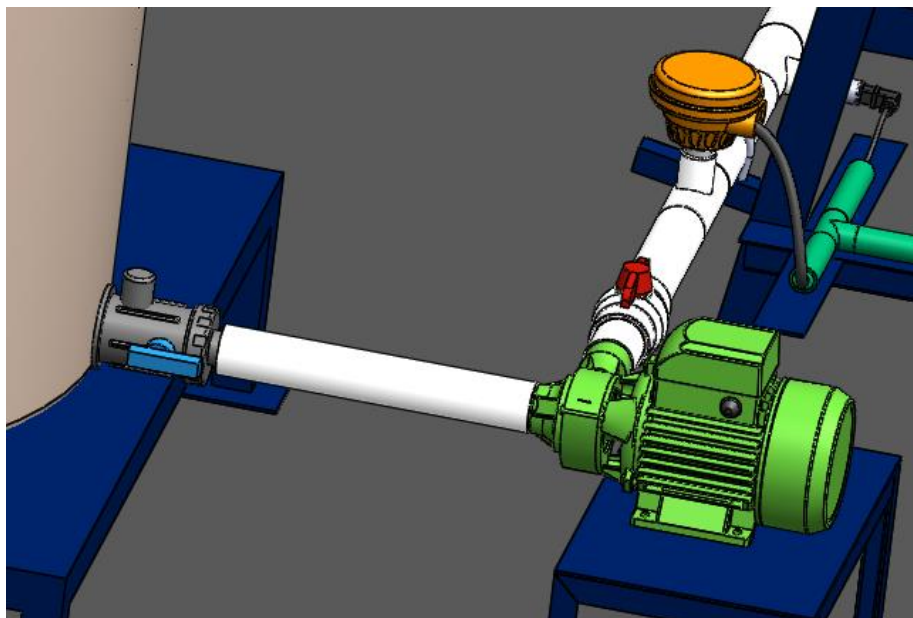


Figura 3.27 Conexión de bomba al sistema de tuberías

Fuente: elaboración propia

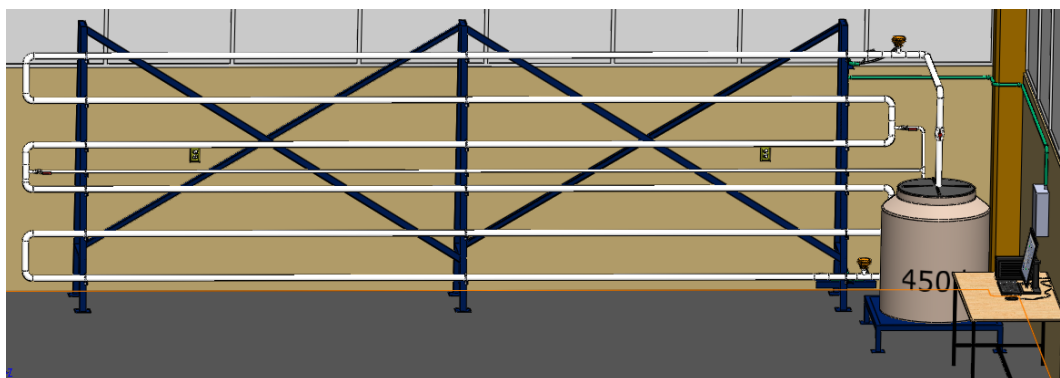


Figura 3.28 Vista completa del recorrido en forma de serpentina.

Fuente: elaboración propia

Se optó por un recorrido como el mostrado en la Figura 3.28 ya que con esto se reduce demasiado espacio del prototipo y se tiene un mejor acceso a las válvulas que emulan una fuga.

A manera de referencia se muestra una vista (Figura 3.29) de detalle de la correcta instalación de los transmisores de caudal y presión.



Las fugas emuladas serán controladas por válvulas colocadas en ubicaciones conocidas, estas válvulas para más detalle se muestran en la Figura 3.30

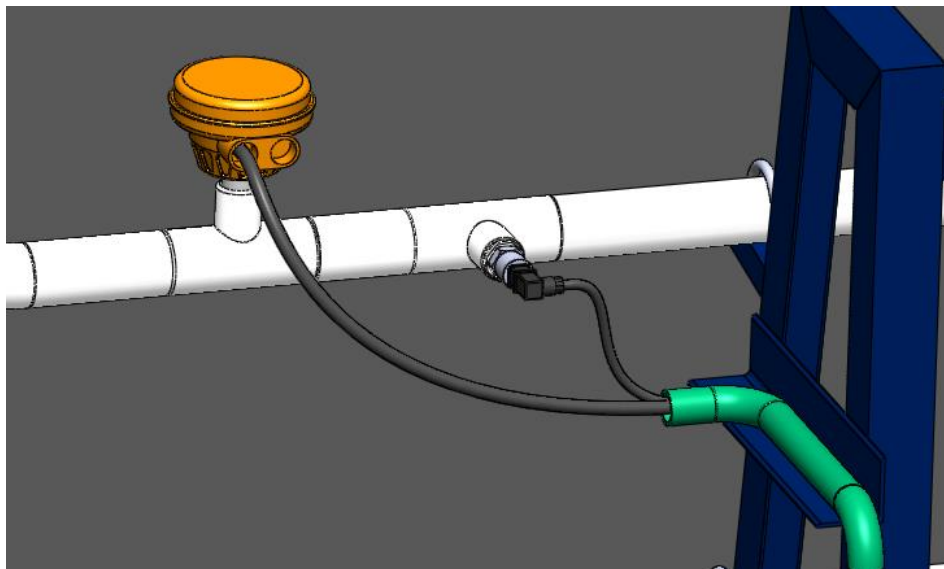


Figura 3.29 instalación de transmisores; caudal y presión.

Fuente: elaboración propia

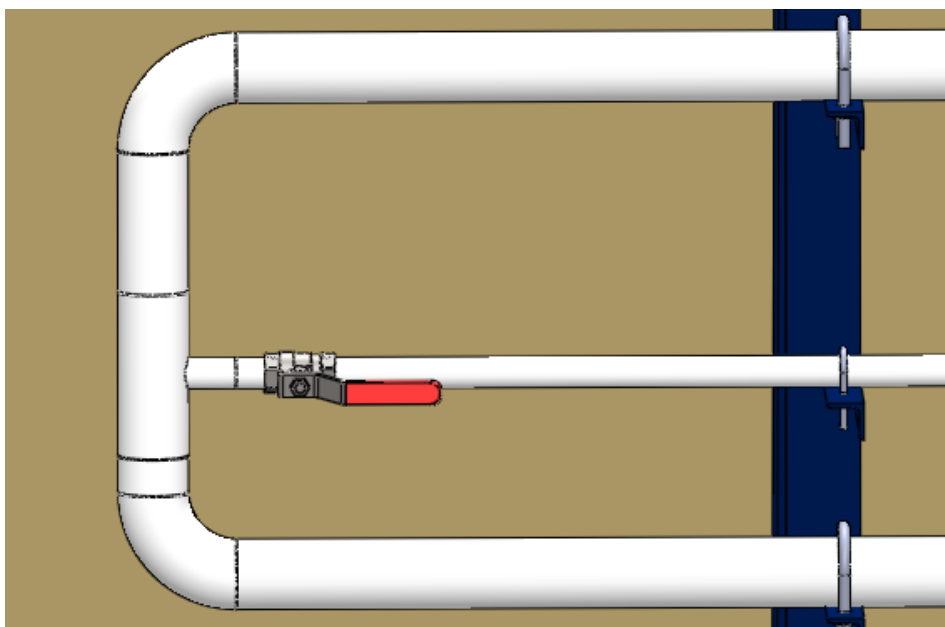


Figura 3.30 Válvula que emula una fuga.

Fuente: elaboración propia

### 3.3 Diseño de un variador de frecuencia AC para una bomba hidráulica.

En los requisitos de las industrias y talleres modernos, un controlador de velocidad ofrece la flexibilidad para mejorar el comportamiento de un motor que los circuitos comunes de encendido/apagado no permiten. En estos escenarios, los tiristores son ampliamente utilizados para controlar la corriente alterna, ya que cuando se activan correctamente su pérdida de potencia es mínima debido a la trayectoria de baja resistencia entre ánodo y cátodo, también los componentes externos son mínimos y menos complejos, lo convierten en el dispositivo perfecto Para una aplicación de bajo coste tal como control de velocidad de un motor universal. (Álvarez, Araque, Merino 2017).

En este trabajo de investigación se describe la implementación de un controlador análogo para variar la velocidad en un motor monofásico universal mediante el ajuste por fase con el uso de un TRIAC como interruptor AC de corriente alterna.

Es por eso que se diseñó un circuito variador de fase para la bomba monofásica de 1/2 HP del prototipo piloto, controlado desde Labview con ayuda de un Arduino extra.

#### 3.3.1 TRIAC.

Un TRIAC es como un DIAC con una terminal compuerta. Un TRIAC puede ser disparado por un pulso de corriente en la compuerta y no requiere voltaje de ruptura para iniciar la conducción, como el DIAC. Básicamente, se puede pensar en un TRIAC simplemente como dos SCR conectados en paralelo y en direcciones opuestas con una terminal común, la compuerta. A diferencia del SCR, el TRIAC puede conducir corriente en una u otra dirección cuando es activado, según la polaridad del voltaje a través de sus terminales A1 y A2. La Figura 3.31 muestra la construcción básica y el símbolo esquemático de un TRIAC. (Álvarez et al., 2017)

En un TRIAC cuando la corriente y voltaje en la compuerta (gate) es igual o supera el nivel de disparo (50 mA y 1.3 V para el BTA41) [4] se enciende y la conexión ánodo-cátodo pasa de alta impedancia a baja resistencia y se mantienen así, incluso después de remover los niveles de disparo. Para un apagado del TRIAC se pueden usar varios métodos como la desconexión de la fuente, cortocircuito entre ánodo – catodo y voltaje negativo en la compuerta.

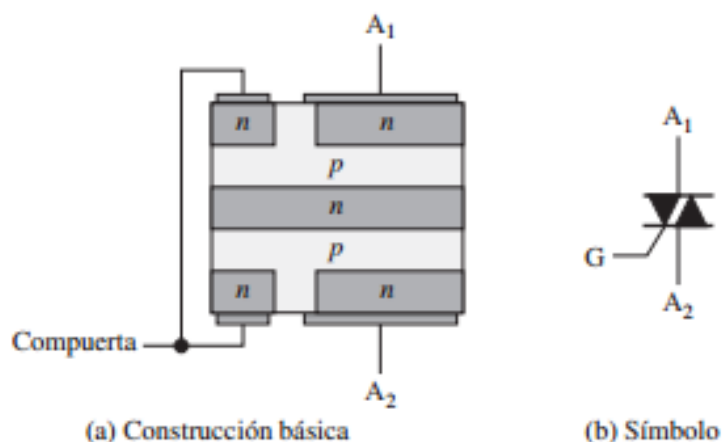


Figura 3.31 Estructura de un TRIAC

Debido a que se implementó una bomba de 1/2HP 6 Amp, se decidió seleccionar el Triac BTA16 para una corriente máxima de 16 Amp.

### 3.3.2 Control por fase.

El control por fase es la forma más común de un control de potencia con TRIACs, consiste en disparar el tiristor en determinados momentos de la onda, a diferentes ángulos como se muestra en la Figura 3.32 ; Estas fases de disparo resultan en conducción (parte sombreada) que se ve reflejado en el motor como una variación de potencia y por ende de velocidad. (Alvarez et al., 2017)



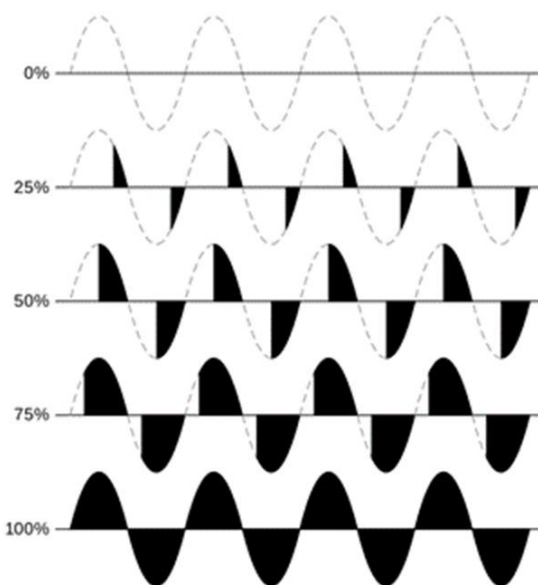


Figura 3.32 Disparo con cruce por cero para un variador de frecuencia

### 3.3.3 Diagrama esquemático.

Un circuito para el control de potencia usando tiristores es mostrado en la Figura 3.33. este circuito ofrece un control de onda completa y posee cuatro componentes entre ellos dos tiristores, el primero es el DIAC el cual dispara el TRIAC acorde al voltaje del DIAC seleccionado, el segundo es el TRIAC encargado de conducir y conectar la carga principal. Además, incluye dos resistencias variables  $R_1$  y  $R_2$  y un capacitor  $C_1$  para modificar el ángulo de disparo.

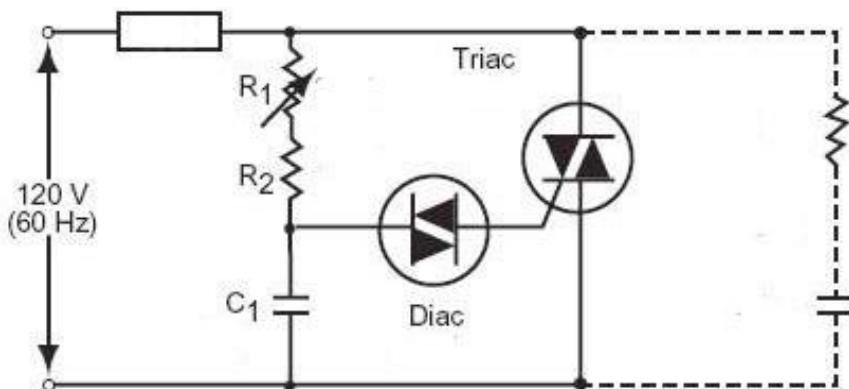


Figura 3.33 Diagrama esquemático del controlador básico de potencia con una red de snubber; La red de snubber se puede remover si la carga es resistiva

El diseño implementado en este proyecto difiere un poco al controlador básico Figura 3.33 en la forma de controlar el disparo, ya que se hace mediante Arduino y controlado desde Labview, fue necesario remplazar el DIAC por un opto TRIAC para la activación de la compuerta del TRIAC BTA16, también se remplazaron los componentes que modifican el ángulo de disparo por una señal digital procedente de Arduino, esta señal es activada con un retraso de tiempo cada vez que la corriente alterna pasa por cero. Para la detección por cruce por cero se hizo uso de un puente de diodos y un opto acoplador para el envío del pulso hacia Arduino como se muestre en Figura 3.34.

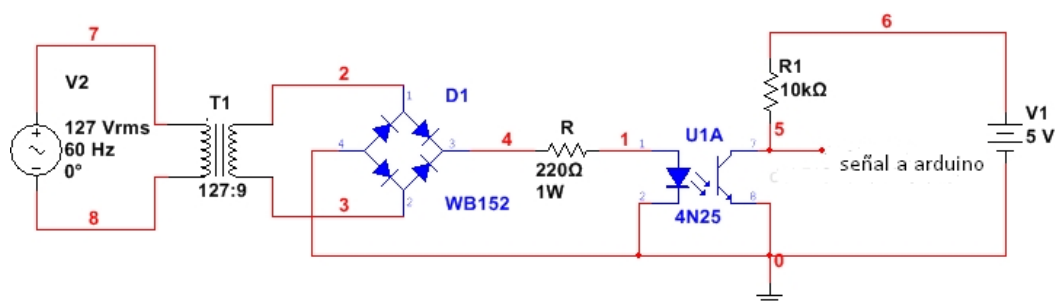


Figura 3.34 diseño de referencia de un detector de cruce por cero.

Ya que la bomba centrífuga es una carga inductiva es necesario el uso de una red de snubber que es un circuito que absorbe energía, suprime las transciendes de voltaje generadas por cargas inductivas en la desconexión eléctrica o mecánica, los cuales son implementadas mayormente en cargas inductivas.

La red de snubber implementada es de tipo RC (Figura 3.33 líneas punteadas), esta se conecta en paralelo al tiristor para prevenir activaciones erróneas y daños potenciales en el mismo.

Finalmente el circuito quedo de la siguiente manera Figura 3.35.

El programa para Arduino se encuentra en el Anexo 2.

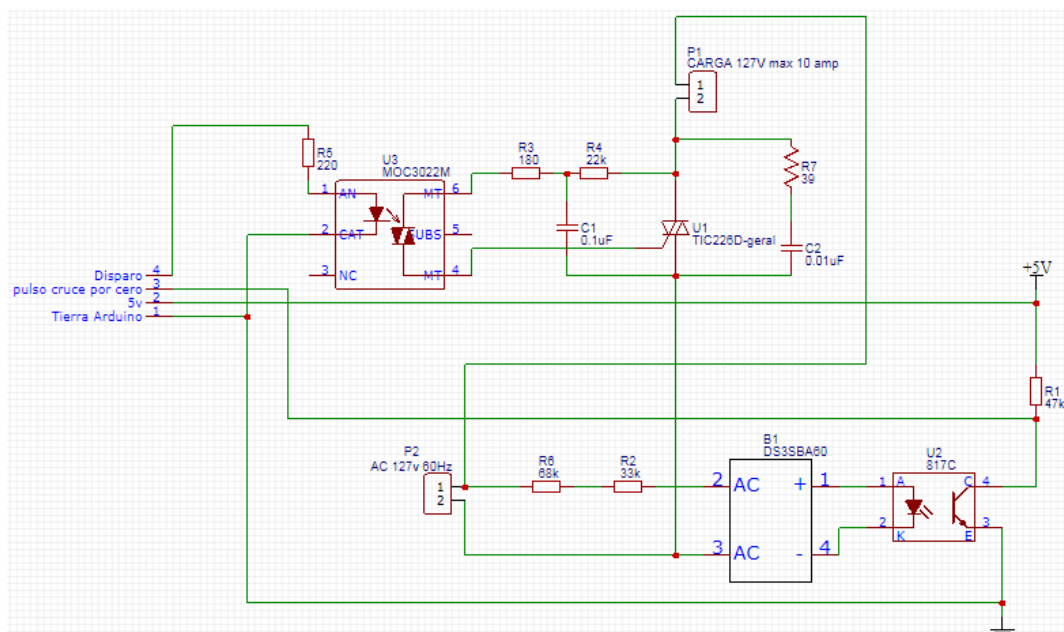


Figura 3.35 Circuito del variador de frecuencia para bomba de 1/2HP para el control de flujo.

Fuente: elaboración propia

### 3.3.4 Diseño de PCB.

La pcb fue diseñada haciendo uso de un software Easy EDA, el grosor de las pistas y la distribución de los componentes fue tomado en cuenta de acuerdo a la corriente que soporta el triac.

Los materiales necesarios para la elaboración de la placa se mencionan a continuación en la Tabla 3.5.

Componente	Valor	Cantidad
TRIAC BTA16	16 A max	1
Puente de diodos D3SBA60	4 A	1
OptoTRIAC MOC3022		1
Terminal de bloques	2 pines 16A	2
Optoacoplador 81C		1
Pin Header hembra	4pines	1
Capacitor cerámico	0.1uf 250v	1
Capacitor cerámico	0.01uf 250v	1



# 4      Capitulo IV: Resultados

## Simulación del modelo matemático de la tubería con una fuga.

Los resultados que se presentan a continuación se obtuvieron de la simulación del modelo matemático de una tubería con una fuga inducida con parámetros establecidos y condiciones iniciales.

El modelo matemático que representa los flujos en una tubería a partir de que se inicia una fuga está dado por (Torres, 2014):

$$\dot{Q}_1 = -\mu Q_1 |Q_1| + \frac{a1}{\Delta z1} (H_{in} - H_2)$$

$$\dot{H}_2 = \frac{a1}{\Delta z1} (Q_1 - Q_2 - Q_f)$$

$$\dot{Q}_2 = -\mu Q_2 |Q_2| + \frac{a1}{L - \Delta z1} (H_2 - H_{out})$$

G	L	b	D	f
9.81	85	407.75	0.0635	0.0226
m/s <sup>2</sup>	m	m/s	M	s/u

Figura 4.1 Condiciones iniciales para la simulación del modelo matemático de una fuga

Símbolo	Parámetro	Unidades
G	Gravedad	m/s <sup>2</sup>
B	Velocidad de onda de onda de presión	m/s
Φ	Diámetro	m
F	Fricción	s/u

Figura 4.2 Parámetros para el modelo matemático de la tubería.

La fuga fue inducida en el instante 3s (Figura 4.3) , se aprecia que el flujo de salida aumenta debido a la fuga, y el flujo de salida disminuye debido al gasto en la fuga ocasionada.

También se observa que la presión entre extremos aumenta debido a la presencia de la fuga como se muestra en la Figura 4.4

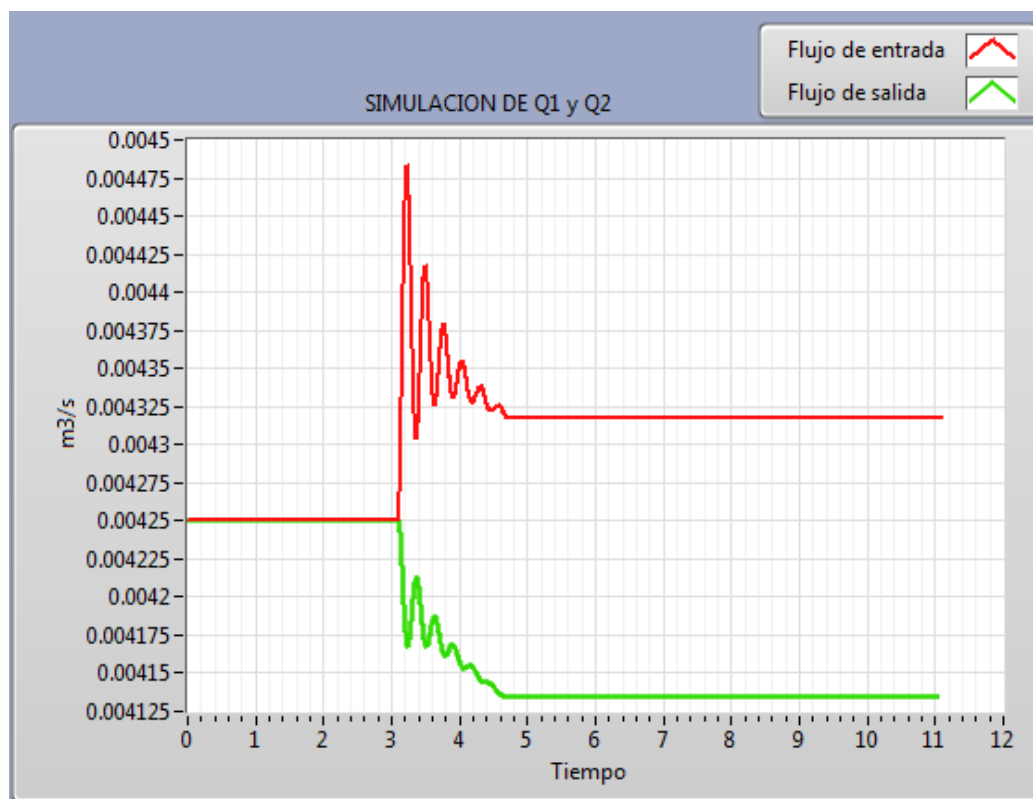


Figura 4.3 Flujos de entrada y salida simulados con condiciones iniciales.

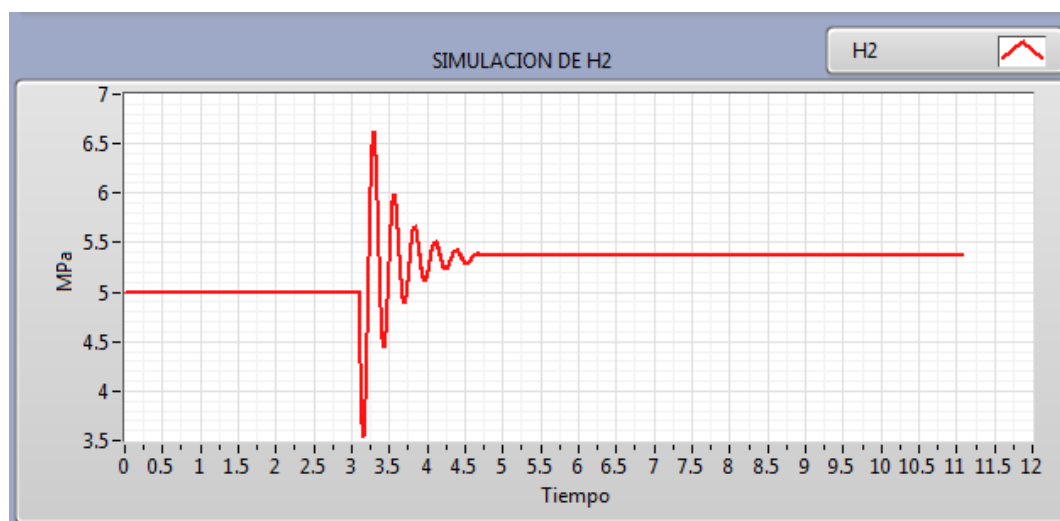


Figura 4.4 Presión simulada en entre los extremos de la tubería

## 4.2 Monitoreo de flujo.

Para la medición y monitoreo de flujo se hizo uso del prototipo de tuberías descrito con anterioridad, con una pequeña modificación (Figura 4.7), la Figura 4.5 muestra la medición de flujos en tiempo real, se observa además que existe una pérdida de caudal entre el flujo de entrada y salida, al parecer se mantiene constante ese gasto, es por eso que para la detección de la fuga se tendrá en cuenta, ya que cuando el sistema compare el gasto con el gasto si fuga podrá determinar si existe o no una posible falla.

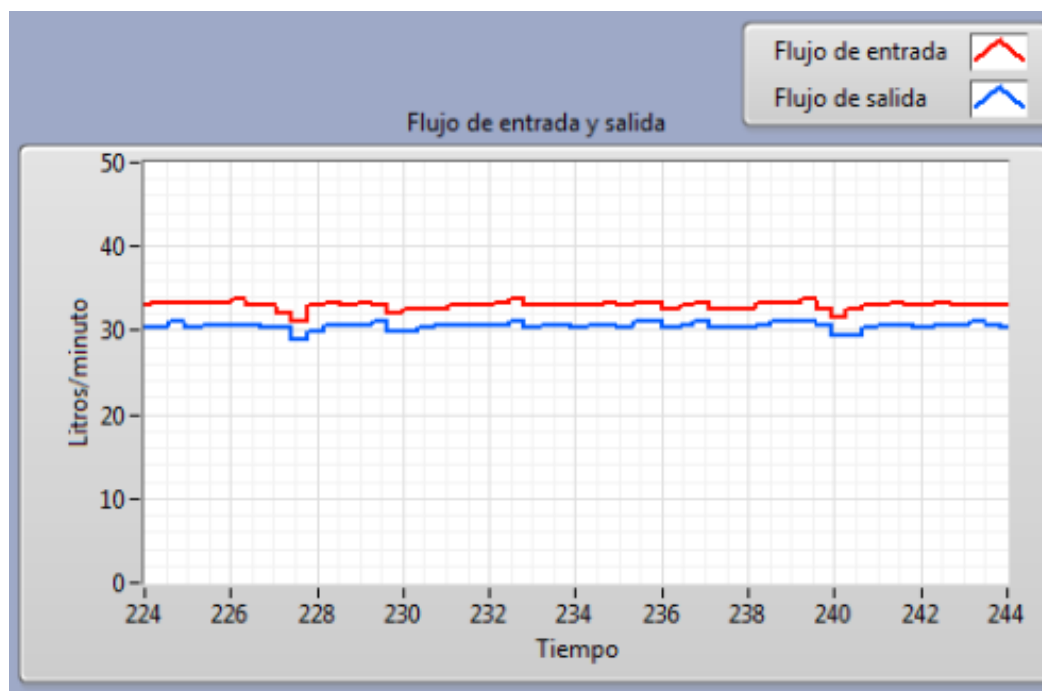


Figura 4.5 Monitoreo de flujo en tiempo real



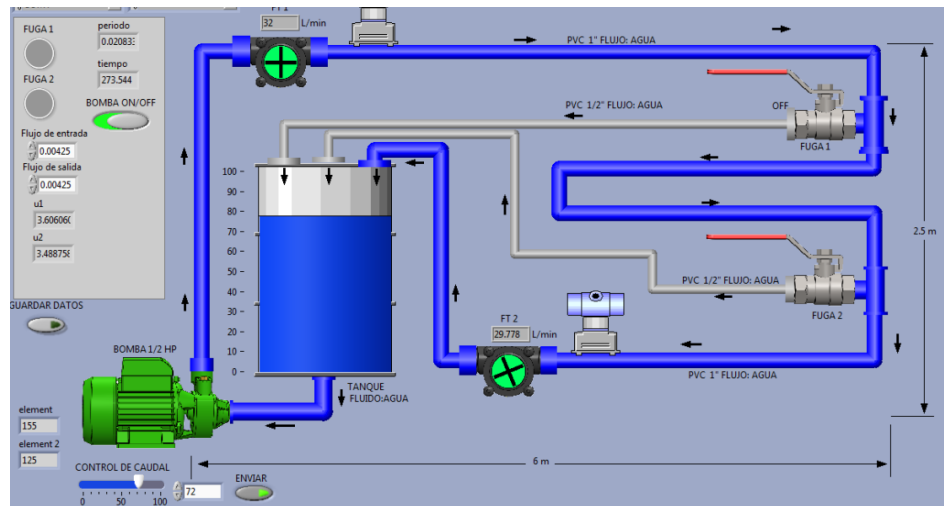


Figura 4.6 HMI del prototipo en tiempo real.



Figura 4.7 Prototipo final para pruebas experimentales, con una fuga

### 4.3 Control de bomba por medio del variador.

En las Figura 4.8 y Figura 4.9 se muestran las variaciones de flujo mediante el control deslizante del variador de velocidad de la bomba Figura 4.10

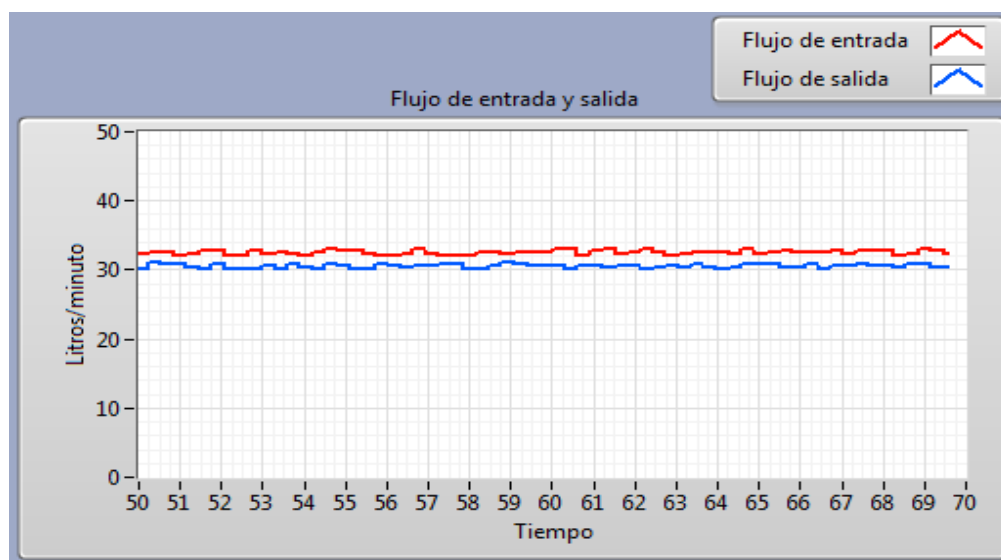


Figura 4.8 Medición de flujo con 75% de velocidad de la bomba.

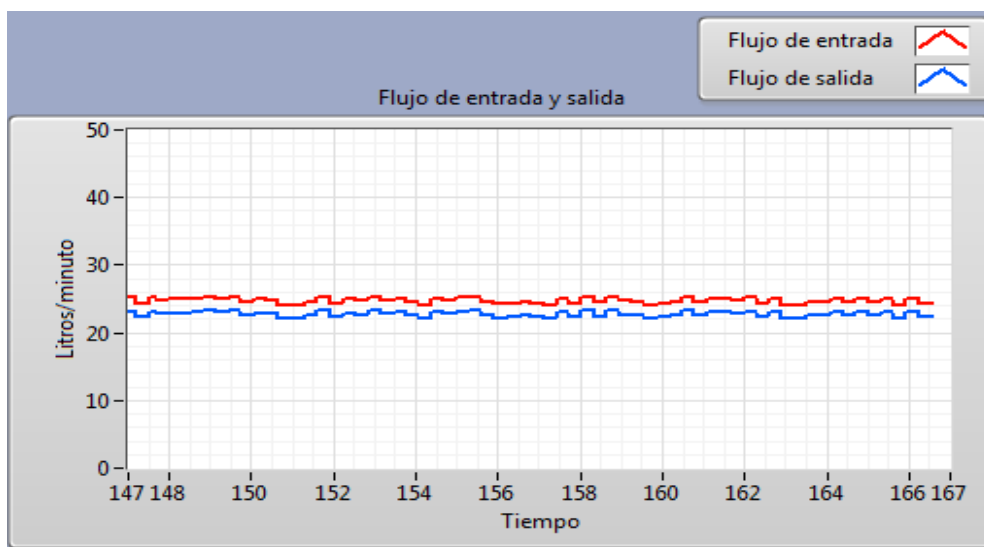


Figura 4.9 Medición de flujo con 56% de velocidad de la bomba

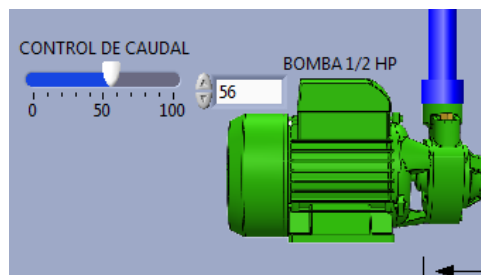


Figura 4.10 Control deslizante para la variación de velocidad en bomba.

#### 4.4 Detección de la presencia de la fuga.

Las pruebas experimentales en el prototipo piloto al emular una fuga mediante las válvulas se observa en la Figura 4.11 y con las condiciones que se muestran a continuación:

Bomba a un 75% de la velocidad

Flujo de salida ~26 L/m

Flujo de entrada ~27.5 L/m

Fuga iniciada en el instante 9seg.

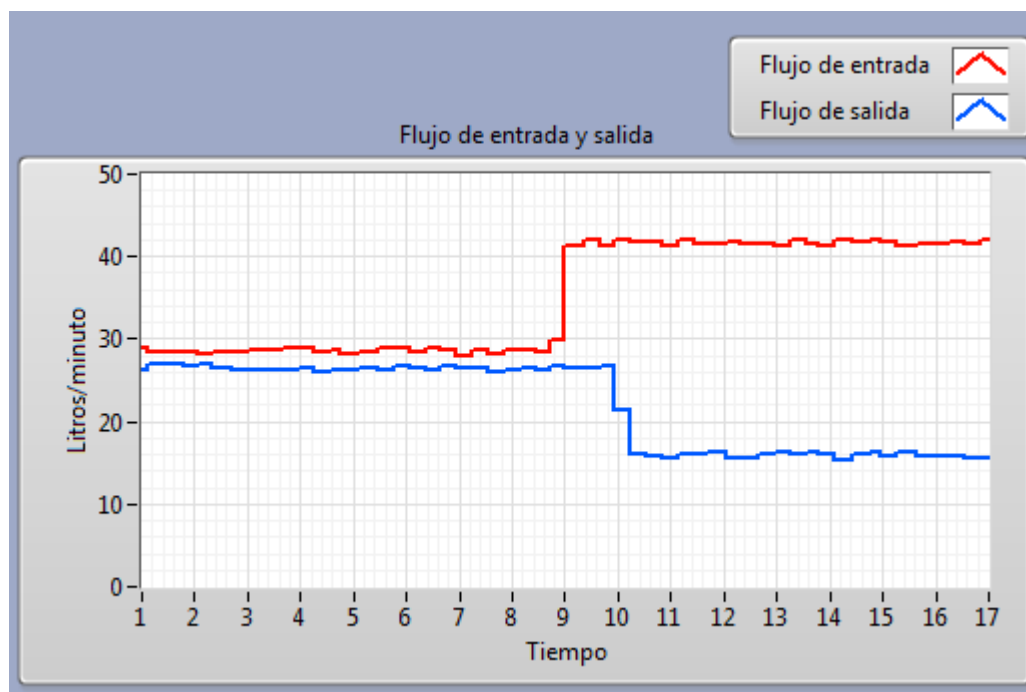


Figura 4.11 Monitoreo de flujos en tiempo real ante la presencia de una fuga

La activación de la alarma en el HMI Figura 4.14 de presencia de fuga se hizo mediante una diferencia de flujos Figura 4.12, si esa diferencia es mayor al gasto sin fuga inducida se activa la alarma de presencia de fuga. Cabe mencionar que esta alarma a su vez activa la fuga inducida en la simulación Figura 4.13 del modelo matemático de la tubería en presencia de una fuga.

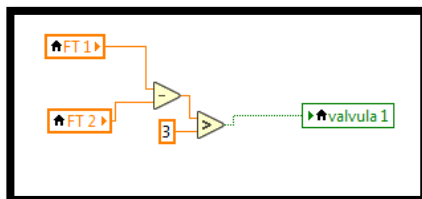


Figura 4.12 Detección de la presencia de fuga mediante diferencias de flujos.

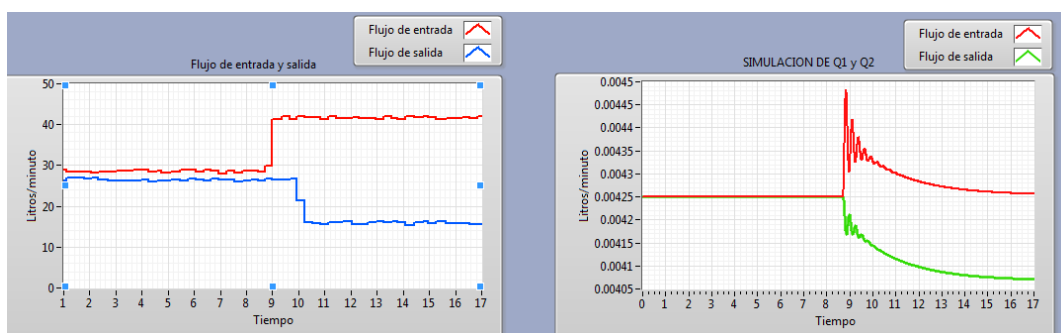


Figura 4.13 Flujo de entrada y salida en tiempo real y simulado. con fuga ocasionada en el instante 9 seg.

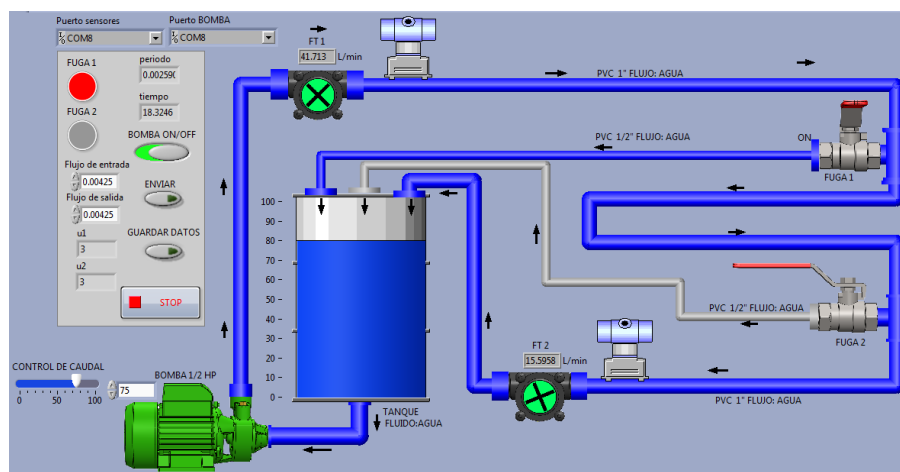


Figura 4.14 Visualización de alarma en HMI de la presencia de fuga.

## 5. CONCLUSIONES

Dentro de los objetivos planteados en este proyecto se han desarrollado varios aspectos; monitorización de flujo en tiempo real implementando sensores de flujo montados en un prototipo, también se logró hacer un diseño en 3d de construcción de una planta piloto para pruebas experimentales.

Se logró la construcción de un prototipo similar al diseñado en este proyecto de investigación en él fue posible implementar el sistema de monitoreo en tiempo real.

Así mismo se generó y desarrollo un software para el monitoreo y control de elementos en el sistema capaz de detectar la existencia de una fuga en una tubería mediante una diferencia de flujos en un tiempo menor a 1 s, para el desarrollo del software fue posible el uso de Labview y la comunicación vía serial a Arduino.

También fue posible realizar una simulación del modelo matemático que representa a una tubería ante una fuga, dicha simulación fue iniciada a partir de la existencia real de la fuga, cabe mencionar que se consideraron condiciones iniciales.

## 6. RECOMENDACIONES.

Debido a la falta de tiempo el proyecto pudo desarrollarse en un 80%, el diseño del prototipo original, no pudo ser implementado, además que el sistema solo detecta una posible fuga ya que el algoritmo de localización de fugas no se implementó.

Se propone la implementación del prototipo diseñado con todos sus elementos que lo componen, además se sugiere que el sistema de tuberías sea totalmente automatizado para pruebas experimentales

Se recomienda ampliar el modelo matemático de la tubería para diversas configuraciones de configuraciones partiendo del prototipo propuesto hasta redes complejas de tuberías con más de una fuga al mismo tiempo.

Finalmente se sugiere implementar un algoritmo de localización de fugas en el sistema en tiempo real mediante Labview.

## 7. BIBLIOGRAFÍA

- ¿Qué es Adquisición de Datos? - National Instruments. (s/f). Recuperado el 4 de septiembre de 2018, de <https://www.ni.com/data-acquisition/what-is/esa/>.
- ¿Qué es LabVIEW? - National Instruments. (s/f). Recuperado el 10 de diciembre de 2018, de <http://www.ni.com/es-mx/shop/labview.html>
- Comunicación Serial: Conceptos Generales - National Instruments. (s/f). Recuperado el 3 de octubre de 2018, de <http://digital.ni.com/public.nsf/allkb/039001258CEF8FB686256E0F005888D1>
- Suhissa. (s/f). Sistemas de monitoreo. Recuperado el 9 de diciembre de 2018, de <https://suhissa.com.mx/sistemas-de-monitoreo/>.
- Fujitecom,(2010),How to work Leak Noise Correlator LC-2500, en línea: <http://www.fujitecom.com/howto/LC2500.html>, fecha de consulta: 28 de Enero de 2013.
- Gómez, L. (2000). Diseño de Interfaces de Usuario Principios, Prototipos y Heurísticas para Evaluación.
- Hunaidi O., (2000), Detecting Leaks in Water-Distribution Pipes, National Research Council of Canada, Ottawa, Canada.



- Krutz, R. (2006). Securing SCADA Systems. Indiana: Wiley Publishing Inc.
- Álvarez Castro, Juan David, Araque de los Rios, Oscar, Merino, Yvonne, Dispositivo electrónico para controlar la frecuencia en un motor monofásico de corriente alterna. Scientia Et Technica [en línea] 2017, 22 (Octubre-Diciembre) : [Fecha de consulta: 12 de diciembre de 2018] Disponible en: <<http://www.redalyc.org/articulo.oa?id=84955649002>> ISSN 0122-1701.
- Rodríguez-Aragón, L. J. (s/f). Tema 4: Internet y Teleinformática Informática Básica, 8.
- O'day, D, (1982), Organizing and analyzing leak and break data for making main replacement decisions, Journal of American Water Works Association, 74 (11), pp.:588–594.
- Sewering, (2012), Receptor AQUAPHON@A100, En Línea:[http://www.sewerin.com/cms/fileadmin/redakteure/Prospekte/pro\\_aquaphon\\_a100\\_es.pdf](http://www.sewerin.com/cms/fileadmin/redakteure/Prospekte/pro_aquaphon_a100_es.pdf), fecha de consulta: 28 de Enero de 2013.



## 8. ANEXOS

### Anexo 1 Programación en Arduino IDE para la obtención de flujos procedentes de los sensores YS-F201.

```
#define factor_conversion1 7.5
#define factor_conversion2 7.5
float TiempoMuestreo=0.3; //en segundos
volatile long NumPulsos1;//variable para la cantidad de pulsos recibidos por el sensor 1
volatile long NumPulsos2;//variable para la cantidad de pulsos recibidos por el sensor 2
int PinSensor1 = 2;    //Sensor1 conectado en el pin 2
int PinSensor2 = 3;    //Sensor2 conectado en el pin 3/float factor_conversion
int pinsensorpresion0 = 0;
int pinsensorpresion1 = 1;
float volumen1=0.0; //volumen del sensor 1
float volumen2=0.0; //volumen del sensor 1
float Fre1=0.0;
float Fre2=0.0;
float TM=0;
Long dt=0; //variación de tiempo por cada bucle
Long t0=0; //millis() del bucle anterior
char Tm=' ';
int u1, u2;
//---Función que se ejecuta en interrupción-----
void ContarPulsos1()
{
    NumPulsos1++; //incrementamos la variable de pulsos
}
//---Función que se ejecuta en interrupción-----
void ContarPulsos2()
{
    NumPulsos2++; //incrementamos la variable de pulsos
}
void setup()
{
    Serial.begin(9600);
    pinMode(PinSensor1, INPUT);
    pinMode(PinSensor2, INPUT);
    attachInterrupt(0,ContarPulsos1,RISING); //(Interrupción 0(Pin2),función,Flanco de
subida)
    attachInterrupt(1,ContarPulsos2,RISING);
    t0=millis();
}
```

```
void loop ()
{
    u1= analogRead(pinsensorpresion0); //sensor de Pression1
    u2= analogRead(pinsensorpresion1); //sensor de Pression1
    float frecuencia1=0.0; //obtenemos la frecuencia de los pulsos del sensor 1 en Hz
    float frecuencia2=0.0; //obtenemos la frecuencia de los pulsos del sensor 2 en Hz
    NumPulsos1 = 0;
    NumPulsos2 = 0; //Ponemos a 0 el número de pulsos
    interrupts(); //Habilitamos las interrupciones
    delay(TiempoMuestreo*1000); //muestra de 200ms
    noInterrupts(); //Deshabilitamos las interrupciones
    frecuencia1=NumPulsos1/TiempoMuestreo; //Hz(pulsos por segundo)
    frecuencia2=NumPulsos2/TiempoMuestreo;
    Fre1=frecuencia1;
    Fre2=frecuencia2;
    float caudal1_L_m=frecuencia1/factor_conversion1; //calculamos el caudal en L/m
    float caudal2_L_m=frecuencia2/factor_conversion2; //calculamos el caudal en L/m
    dt=millis()-t0; //calculamos la variación de tiempo
    t0=millis();

    volumen1=volumen1+(caudal1_L_m/60.0)*(dt/1000.0); // volumen(L)=caudal(L/s)*tiempo(s)
    volumen2=volumen2+(caudal2_L_m/60.0)*(dt/1000.0); // volumen(L)=caudal(L/s)*tiempo(s)
    u1= analogRead(pinsensorpresion0);
    u2= analogRead(pinsensorpresion1);

    //-----Enviamos por el puerto serie-----
    Serial.println (caudal1_L_m,3);
    Serial.println (caudal2_L_m,3);
    Serial.println (volumen1,3);
    Serial.println (volumen2,3);
    Serial.println (Fre1,3);
    Serial.println (Fre2,3);
    Serial.println (u1);
    Serial.println (u2);
    Serial.println ("P");
}
```

## Anexo 2 Programación en Arduino para un variador de frecuencia monofásico.

```
//variador de frecuencia controlado desde el puerto serie mediante
Labview
//Autor: Geovani Velazquez Vega.
#define factor_conversion1 7.5
#define factor_conversion2 7.5
float TiempoMuestreo=0.3; //en segundos
volatile long NumPulsos1;//variable para la cantidad de pulsos
recibidos por el sensor 1
volatile long NumPulsos2;//variable para la cantidad de pulsos
recibidos por el sensosr 2
int PinSensor1 = 2;    //Sensor1 conectado en el pin 2
int PinSensor2 = 3;    //Sensor2 conectado en el pin 3/float
factor_conversion
int pinsensorpresion0 = 0;
int pinsensorpresion1 = 1;
float volumen1=0.0; //volumen del sensor 1
float volumen2=0.0; //volumen del sensor 1
float Fre1=0.0;
float Fre2=0.0;
float TM=0;
long dt=0; //variación de tiempo por cada bucle
long t0=0; //millis() del bucle anterior
char Tm=' ';
int u1, u2;

//---Función que se ejecuta en interrupción-----
void ContarPulsos1()
{
    NumPulsos1++; //incrementamos la variable de pulsos
}
//---Función que se ejecuta en interrupción-----
void ContarPulsos2()
{
    NumPulsos2++;//incrementamos la variable de pulsos
}
void setup()
{
    Serial.begin(9600);
    pinMode(PinSensor1, INPUT);
```

```
pinMode(PinSensor2, INPUT);
attachInterrupt(0, ContarPulsos1, RISING); //(Interrupción
0(Pin2), función, Flanco de subida)
attachInterrupt(1, ContarPulsos2, RISING);
t0=millis();
}

void loop ()
{
    u1= analogRead(pinsensorpresion0); //sensor de Pression1
    u2= analogRead(pinsensorpresion1); //sensor de Pression1
    float frecuencia1=0.0; //obtenemos la frecuencia de los pulsos del
sensor 1 en Hz
    float frecuencia2=0.0; //obtenemos la frecuencia de los pulsos del
sensor 2 en Hz
    NumPulsos1 = 0;
    NumPulsos2 = 0; //Ponemos a 0 el número de pulsos
    interrupts(); //Habilitamos las interrupciones
    delay(TiempoMuestreo*1000); //muestra de 200ms
    noInterrupts(); //Deshabilitamos las interrupciones
    frecuencia1=NumPulsos1/TiempoMuestreo; //Hz(pulsos por segundo)
    frecuencia2=NumPulsos2/TiempoMuestreo;
    Fre1=frecuencia1;
    Fre2=frecuencia2;
    float caudal1_L_m=frecuencia1/factor_conversion1; //calculamos el
caudal en L/m
    float caudal2_L_m=frecuencia2/factor_conversion2; //calculamos el
caudal en L/m
    dt=millis()-t0; //calculamos la variación de tiempo
    t0=millis();

    volumen1=volumen1+(caudal1_L_m/60.0)*(dt/1000.0); //
volumen(L)=caudal(L/s)*tiempo(s)
    volumen2=volumen2+(caudal2_L_m/60.0)*(dt/1000.0); //
volumen(L)=caudal(L/s)*tiempo(s)
    u1= analogRead(pinsensorpresion0);
    u2= analogRead(pinsensorpresion1);

    //-----Enviamos por el puerto serie-----
    Serial.println (caudal1_L_m,3);
    Serial.println (caudal2_L_m,3);
    Serial.println (volumen1,3);
    Serial.println (volumen2,3);
```

```
Serial.println (Fre1,3);  
Serial.println (Fre2,3);  
Serial.println (u1);  
Serial.println (u2);  
Serial.println ("P");  
}
```