



ESPE
ESCUELA POLITÉCNICA DEL EJÉRCITO
CAMINO A LA EXCELENCIA

DEPARTAMENTO DE ELÈCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELCTROMECAÁNICA

TEMA: DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO MÓDULO DE CONEXIÓN ENTRE UN SIMULADOR VIRTUAL DE PROCESOS INDUSTRIALES Y UN PROGRAMADOR LÓGICO CONTROLABLE EN EL LABORATORIO DE PLC DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN LATACUNGA.

AUTOR: ADRIÁN FERNANDO GÓMEZ ESPÍN

DIRECTOR: ING. WASHINGTON FREIRE



ANTECEDENTES

OBJETIVOS

**SISTEMAS
HÍBRIDOS**

**DESCRIPCIÓN
DEL PROTOTIPO**

**OBJETIVOS DE
LA PROPUESTA**

**DESARROLLO
DEL PROTOTIPO**

**ANÁLISIS DE
RESULTADOS**

CONCLUSIONES

RECOMENDACIONES

**E
S
T
R
U
C
T
U
R
A**



La cantidad de equipos no satisface con la carga horaria que tiene el laboratorio, debido a la demanda de prácticas que los estudiantes buscan realizar ya sea por abordar o solucionar problemas de procesos industriales para la adquisición de conocimientos o resolución de algún problema.



GENERAL

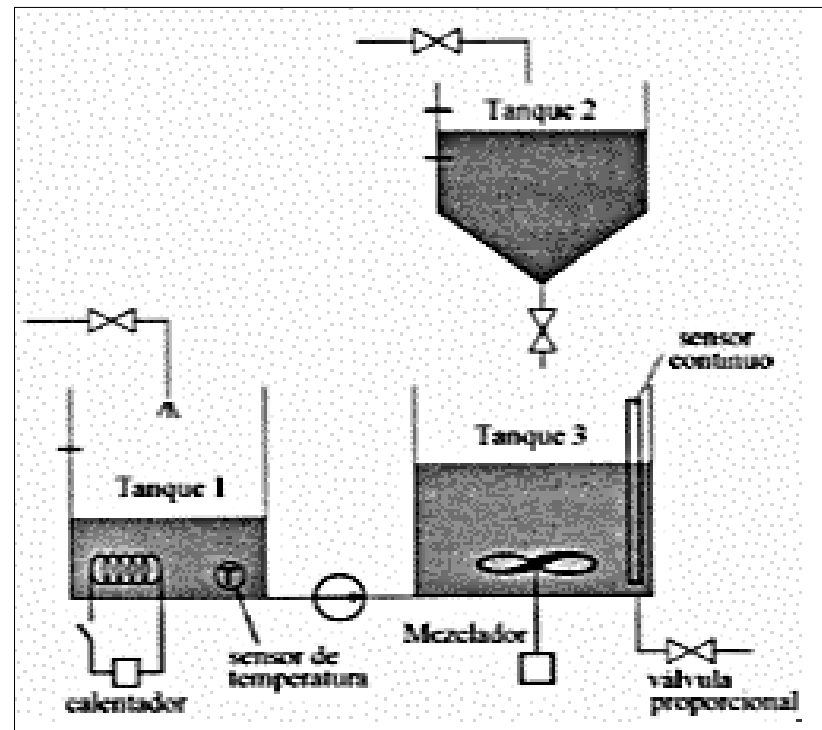
- Diseñar e Implementar un prototipo de módulo de conexión entre un simulador de procesos industriales y un Programador Lógico Programable en el laboratorio de PLC de la Universidad de las Fuerzas Armadas Espe Extensión Latacunga.

ESPECÍFICOS

- Determinar procesos industriales que permitan programar PLC's.
- Simular y modelar procesos industriales de control de empresas regionales.
- Acondicionar, procesar y programar señales de entradas y salidas ya sean análogas/digitales de los diferentes procesos industriales de control.
- Implementar un sistema de conexión para enlazar PLC's con la simulación de procesos industriales

La interacción entre software y hardware es conocida como sistema híbrido

Un buen modelo debe poseer al menos tres características: aproximarse cualitativa y cuantitativamente al comportamiento observado del sistema real; ser simple, de modo que se pueda realizar análisis, simulación y diseño en tiempos breves; y adecuarse al propósito para el que será empleado.



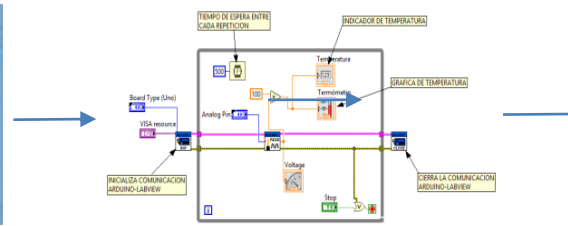
VISUALIZACIÓN DEL CONTROL EN LABVIEW



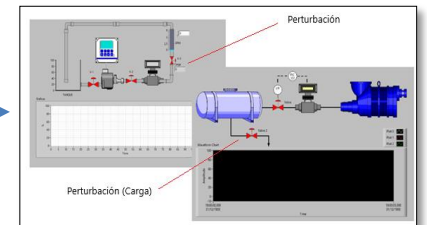
Software LabVIEW



Módulos DSC

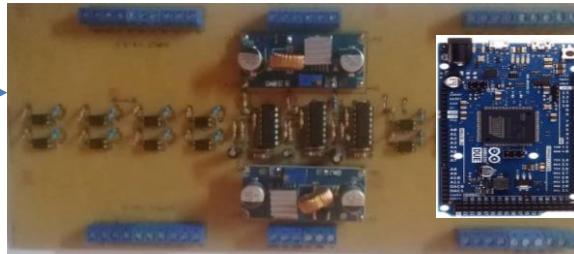
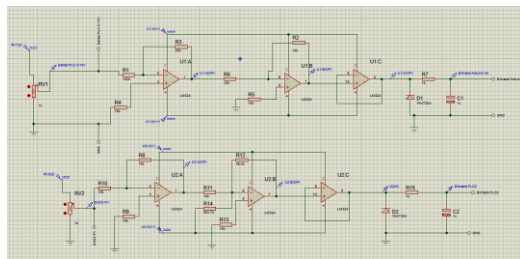


Códigos VI – Diagramas de bloque



HMI de la simulación del proceso

ENLACE DEL PROTOTIPO – PROTOCOLO DE COMUNICACIÓN



CONTROL DEL PROCESO

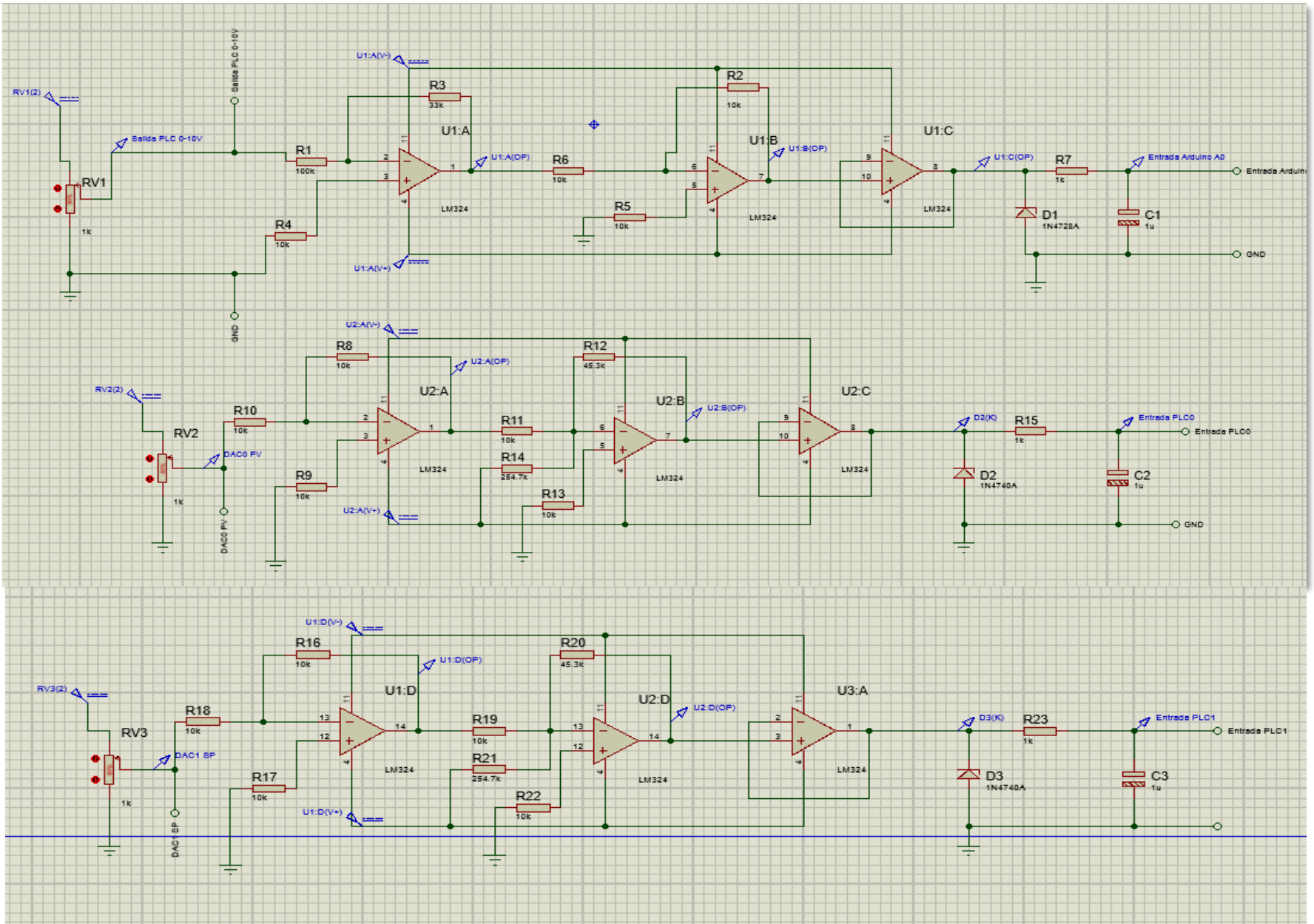
Objetivo General

- Diseñar e Implementar un prototipo módulo de conexión por medio de la estructuración de un circuito electrónico que permita manipular variables de ingeniería de una tarjeta electrónica programable y de un software de simulación de procesos industriales mediante modelación VI con diagramas de bloques.

Objetivos Específicos

- Vincular el software de simulación LabVIEW con la plataforma de código abierto Arduino para la manipulación de variables análogas y digitales.
- Realizar el acondicionamiento de las señales de voltaje entre PLC y tarjeta electrónica Arduino que permitan la conexión de dispositivos y aplicaciones.
- Desarrollar el HMI respectivo para cada uno de los procesos industriales en el software LabVIEW (Caudal, Temperatura, Presión, Nivel).
- Realizar un menú didáctico en el Software LabVIEW, que permita a los estudiantes interactuar de una forma intuitiva con la simulación de los procesos industriales.

DISEÑO DEL CIRCUITO ELECTRÓNICO



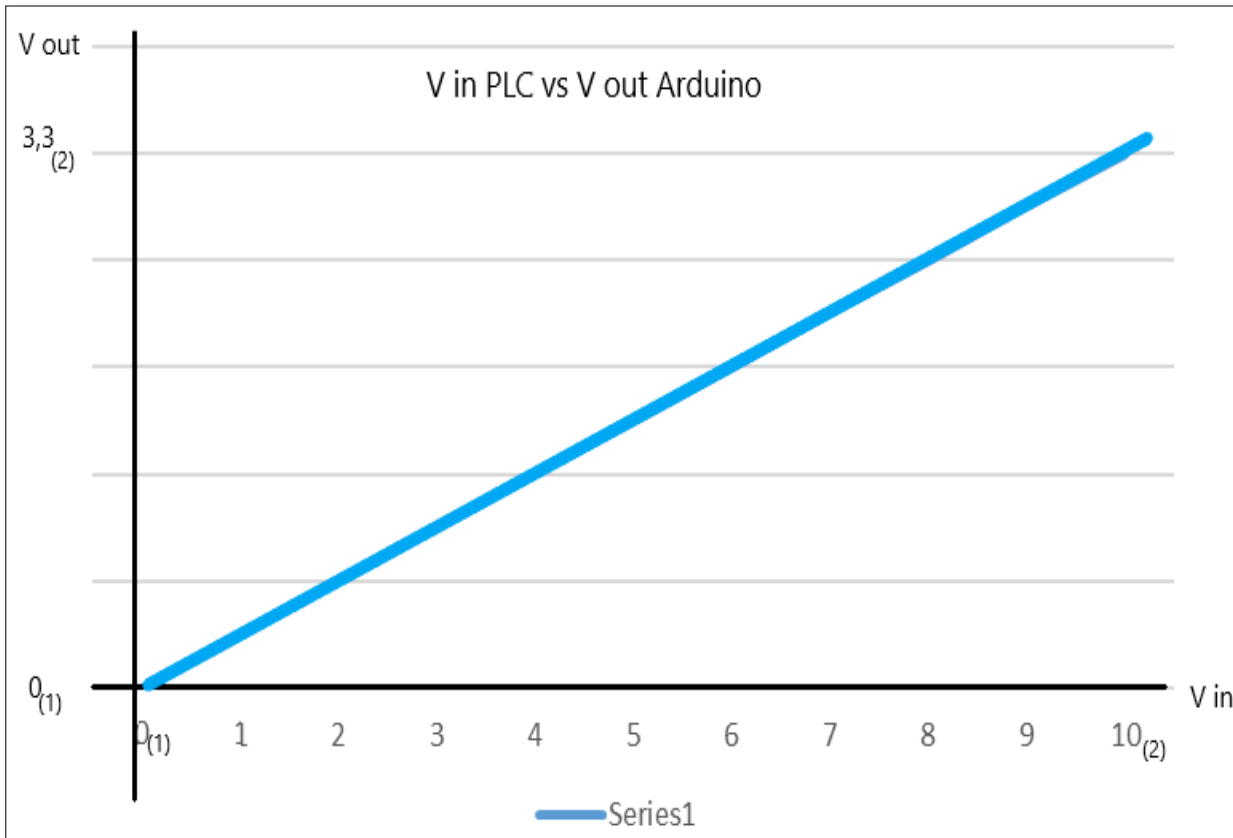
ACONDICIONAMIENTO PLC - ARDUINO

$$V_{out} = m * V_{in} + b$$

$$m = \frac{V_{out(2)} - V_{out(1)}}{V_{in(2)} - V_{in(1)}}$$

$$m = \frac{3,3 - 0}{10 - 0}$$

$$m = 0,33$$

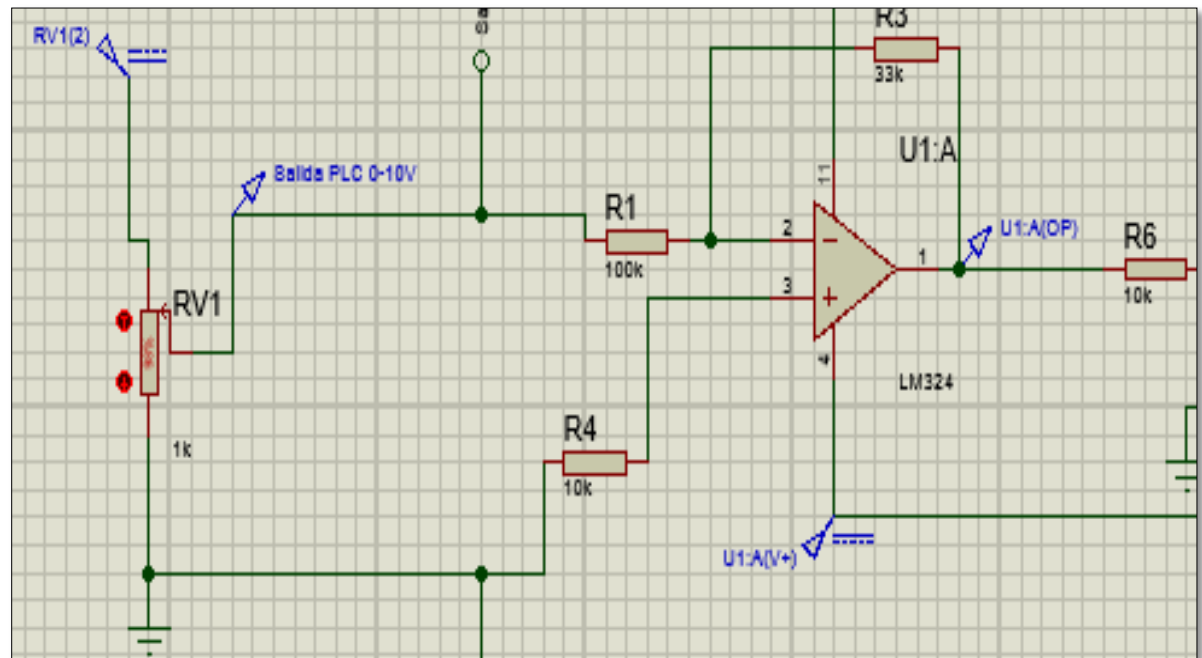


$$m = \frac{R_3}{R_1}$$

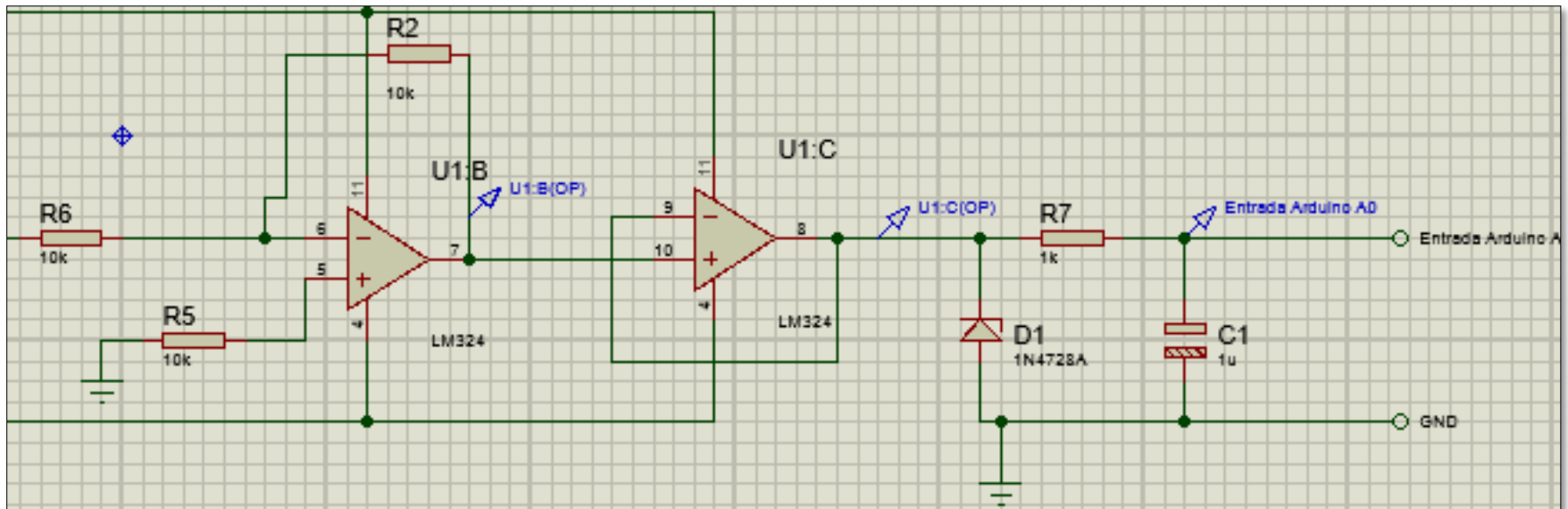
$$R_3 = m * R_1$$

$$R_3 = (0,33)(10 K\Omega)$$

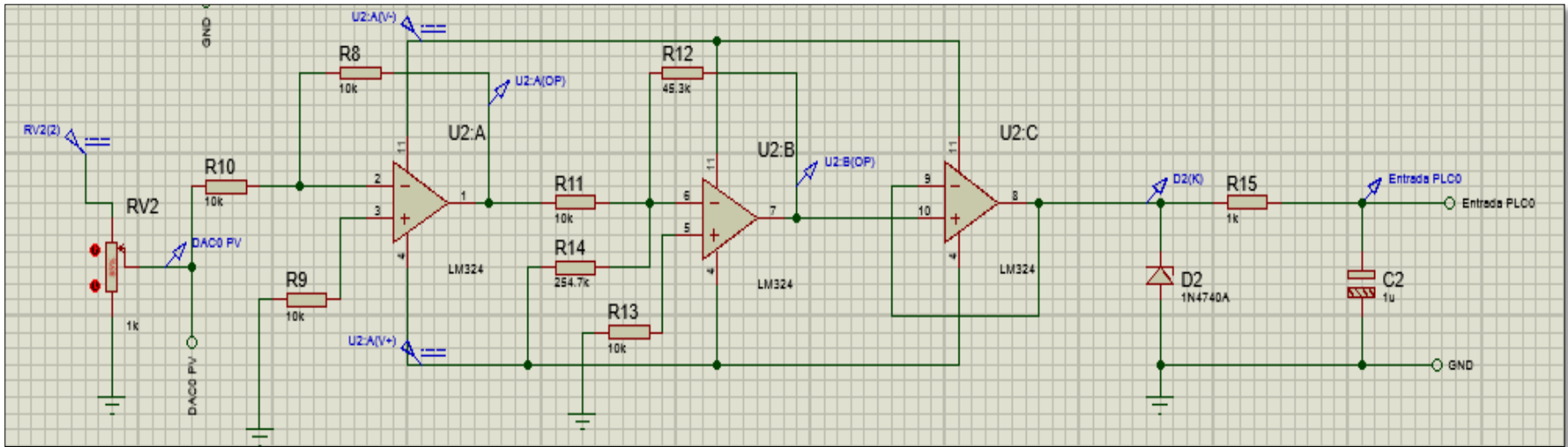
$$R_3 = 33 K\Omega$$



CIRCUITO DE PROTECCIÓN Y SEGUIDOR DE VOLTAJE



ACONDICIONAMIENTO ARDUINO - PLC



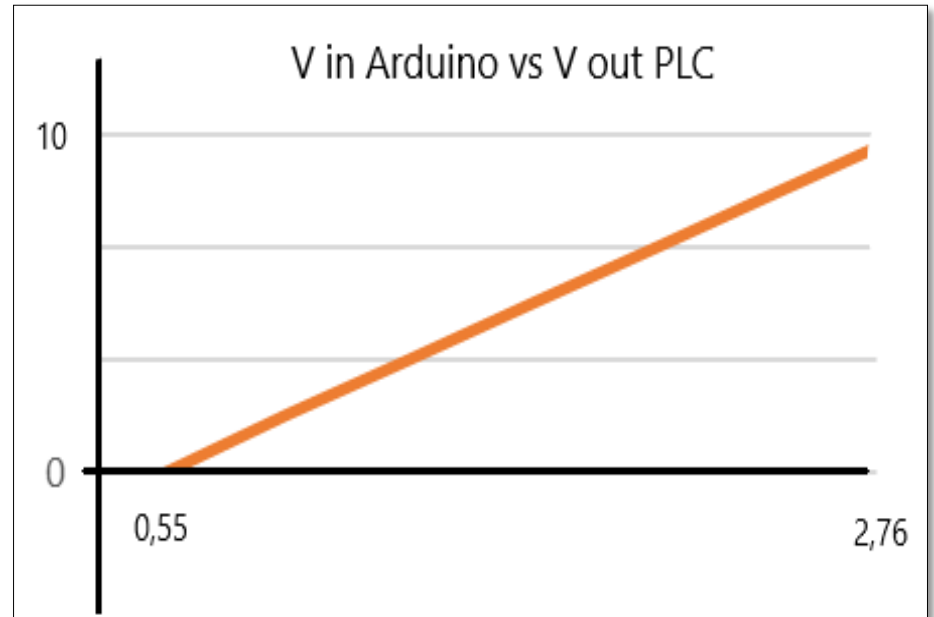
$$m_1 = \frac{10 - 0}{2,76 - 0,55}$$

$$m_1 = 4,53$$

$R_{11} = 10 \text{ K}\Omega$ se obtiene:

$$R_{12} = 4,53 * 10 \text{ K}\Omega$$

$$R_{12} = 45,3 \text{ K}\Omega$$



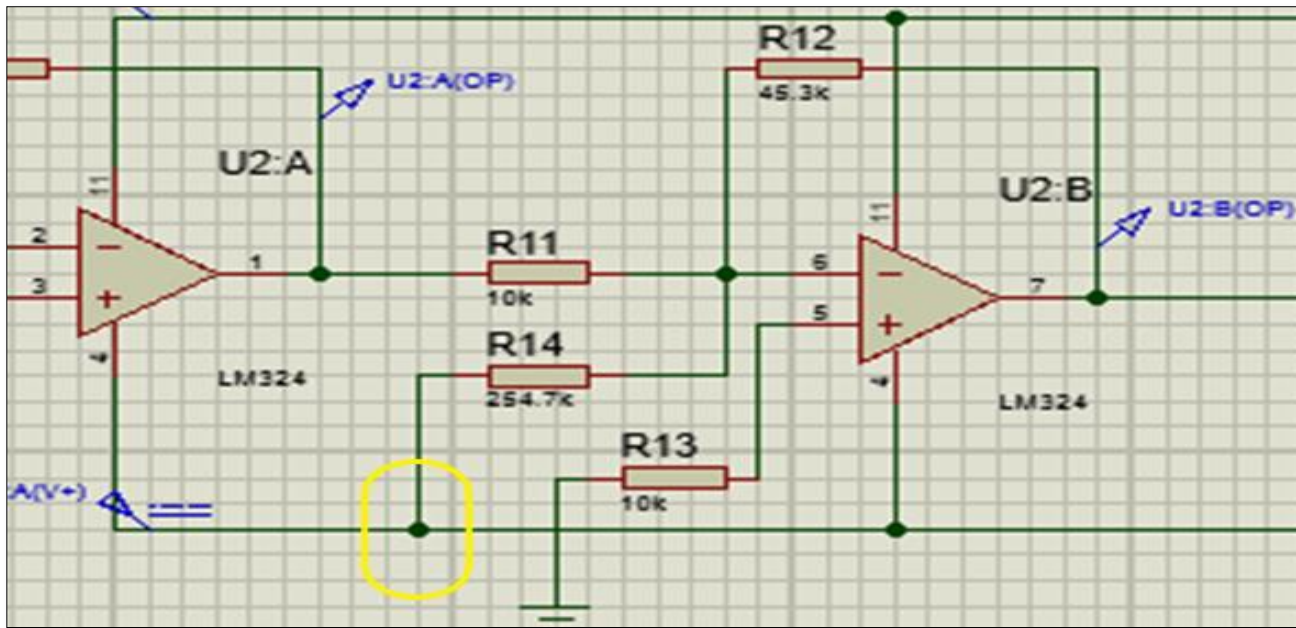
Min: 0,55 V

Max: 2,76 V

$$V_{out} = m * V_{in} + b$$

$$0 = 4.53(0.55) + b$$

$$b = -2,49$$



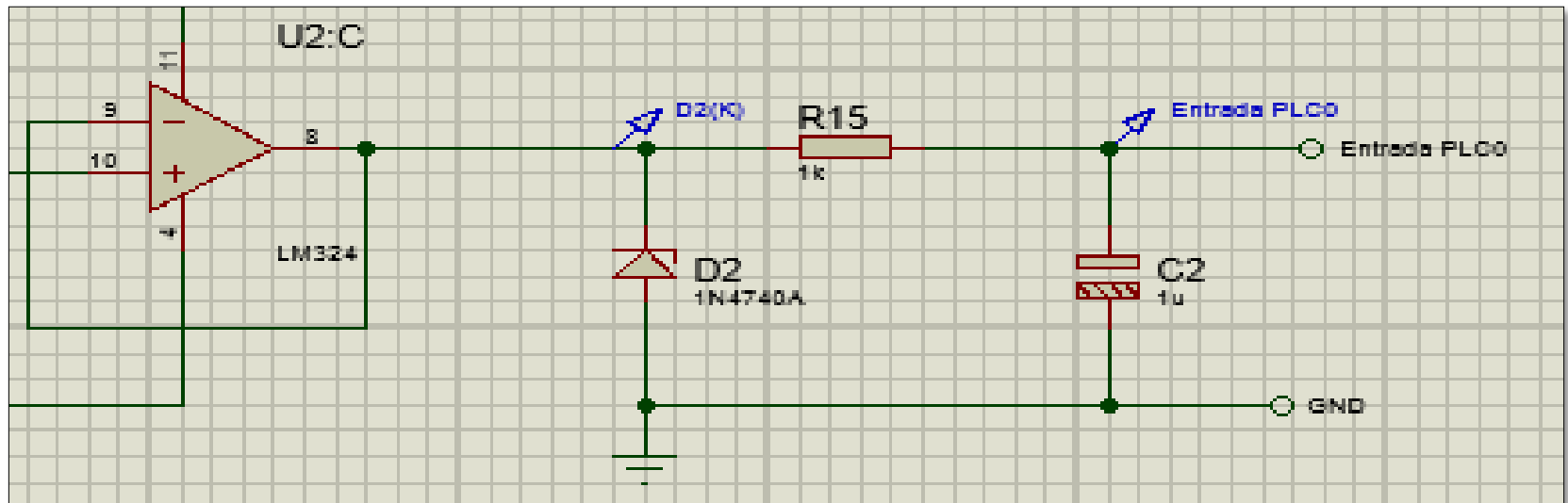
$$b = \frac{R_{12}}{R_{14}} * V_{CC}$$

$$R_{14} = \frac{R_{12}}{b} * V_{CC}$$

$$V_{CC} = 14 V$$

$$R_{14} = 254,7 K\Omega$$

CIRCUITO DE PROTECCIÓN Y SEGUIDOR DE VOLTAJE



PROTOCOLO DE COMUNICACIÓN OPC ENTRE LABVIEW Y ARDUINO



The screenshot shows the 'KEPServerEX - Runtime (Demo Expires 01:05:02)' application window. A red callout box points to a button labeled 'Click to add a channel.' with the text 'Añadir un nuevo canal' below it. A 'New Channel - Identification' dialog box is open, showing a text input field for 'Channel name:' containing the text 'MOD'. A red circle highlights this input field. The dialog box also contains instructions: 'A channel name can be from 1 to 256 characters in length. Names can not contain periods, double quotations or start with an underscore.' At the bottom of the dialog are buttons for '< Back', 'Next >', 'Cancel', and 'Help'. Below the dialog is a log table with columns 'Date', 'Time', and 'Source'.

Date	Time	Source
1/6/2019	3:06:51 PM	KEPServerEX\Runtime
1/6/2019	3:07:48 PM	KEPServerEX\Runtime
1/6/2019	3:11:44 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	Modbus RTU Serial
1/6/2019	3:12:40 PM	KEPServerEX\Runtime
1/6/2019	3:13:29 PM	OPC .NET Server
1/6/2019	3:24:12 PM	KEPServerEX\Runtime

KEPServerEX

File Edit View Tools Runtime Help

Click to add a channel.

KraussMaffei MC4 Ethernet
 Lufkin Modbus Serial
 Memory Based
 Mettler Toledo Serial
 Micro DCI
 Mitsubishi CNC Ethernet
 Mitsubishi Ethernet
 Mitsubishi FX
 Mitsubishi FX Net
 Mitsubishi Serial
 Modbus ASCII Serial
 Modbus Plus
Modbus RTU Serial
 Modbus Slave RTU Serial
 Modbus TCP/IP Ethernet
 MITConnect Client
 ODBC Client
 Omni Flow Computer
 Omron FINS Ethernet
 Omron FINS Serial
 Omron Host Link
 Omron NJ Ethernet
 Omron Process Suite
 Omron Toolbus
 OPC DA Client
 OPC UA Client
 OPC XML-DA Client
 Optimization OptiLogic
 Opto 22 Ethernet

Enable diagnostics

< Back Next > Cancel Help

Date	Time	Source
1/6/2019	3:06:51 PM	KEPServerEX\Runtime
1/6/2019	3:07:48 PM	KEPServerEX\Runtime
1/6/2019	3:11:44 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	Modbus RTU Serial
1/6/2019	3:12:40 PM	KEPServerEX\Runtime
1/6/2019	3:13:29 PM	OPC .NET Server
1/6/2019	3:34:12 PM	KEPServerEX\Runtime

KEPServerEX - Runtime (Demo Expires 01:05:02)

File Edit View Tools Runtime Help

Click to add a channel.

New Channel - Communications

Connection type
 Physical medium COM Port

Serial Port Settings
 COM ID 121
 Baud rate 9600
 Data bits 8
 Parity None
 Stop bits 1
 Flow control None

Flow control
 Select the flow control type required by the target device.

< Back Next > Cancel Help

Date	Time	Source
1/6/2019	3:06:51 PM	KEPServerEX\Runtime
1/6/2019	3:07:48 PM	KEPServerEX\Runtime
1/6/2019	3:11:44 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	Modbus RTU Serial
1/6/2019	3:12:40 PM	KEPServerEX\Runtime
1/6/2019	3:13:29 PM	OPC .NET Server



Nuevo Dispositivo - Arduino

MODBUS

Click to add a device.

Device ... / Model ID Description

Click to add a device.

New Device - Name

A device name can be from 1 to 256 characters in length.

Names can not contain periods, double quotations or start with an underscore.

Device name:

Device 1

< Back Next > Cancel Help

Date	Time	Source
1/6/2019	3:06:51 PM	KEPServerEX\Runtime
1/6/2019	3:07:48 PM	KEPServerEX\Runtime
1/6/2019	3:11:44 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	Modbus RTU Serial
1/6/2019	3:12:40 PM	KEPServerEX\Runtime
1/6/2019	3:13:29 PM	OPC.NET Server

New Device - ID

The device you are defining may be multidropped as part of a network of devices. In order to communicate with the device, it must be assigned a unique ID.

Your documentation for the device may refer to this as a "Network ID" or "Network Address."

Device ID:

Decimal

< Back Next > Cancel Help

KEPServerEX - Runtime (Demo Expires 01:03:14)

File Edit View Tools Runtime Help

MODBUS / Device1

Tag Name / Address Data Type Scan Rate Scaling Description

Click to add a static tag. Tags are not required, but are browsable by OPC clients.

Date	Time	Source
1/6/2019	3:11:44 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	KEPServerEX\Runtime
1/6/2019	3:12:20 PM	Modbus RTU Serial
1/6/2019	3:12:40 PM	KEPServerEX\Runtime
1/6/2019	3:13:29 PM	OPC .NET Server
1/6/2019	3:34:12 PM	KEPServerEX\Runtime
1/6/2019	3:34:32 PM	Licensing
1/6/2019	4:29:29 PM	KEPServerEX\Configuration

Tag Properties

General Scaling

Identification

Name:

Address:

Description:

Data properties

Data type: Default

Client access: Read/Write

Scan rate: 100 milliseconds

Note: This scan rate is applied for non-OPC clients. It only applies to OPC clients when the device scan rate mode is set to 'Respect tag specified rate'.

OK Cancel Apply Help

Tag's

Tag Properties

General Scaling

Identification

Name: IN

Address: 30001

Description:

Data properties

Data type: Word

Client access: Read Only

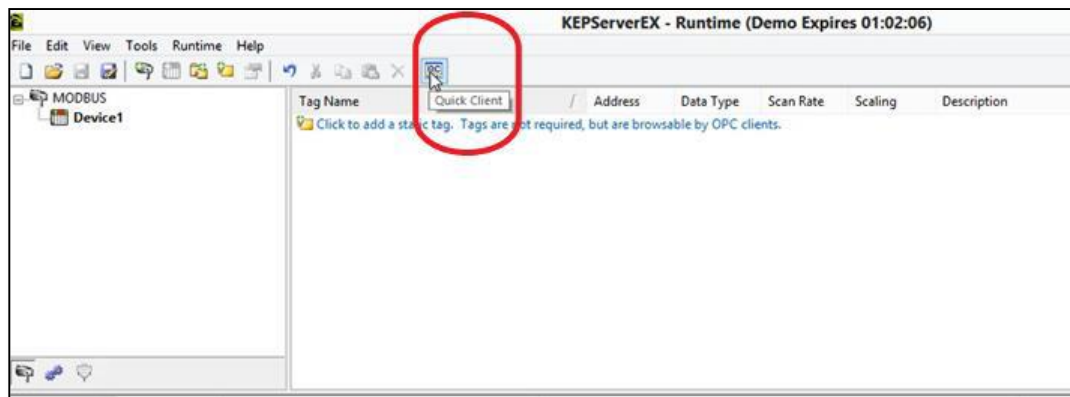
Scan rate: 100 milliseconds

Note: This scan rate is applied for non-OPC clients. It only applies to OPC clients when the device scan rate mode is set to 'Respect tag specified rate'.

OK Cancel Apply Help

Stopping Modbus RTU Serial device driver

VERIFICACIÓN

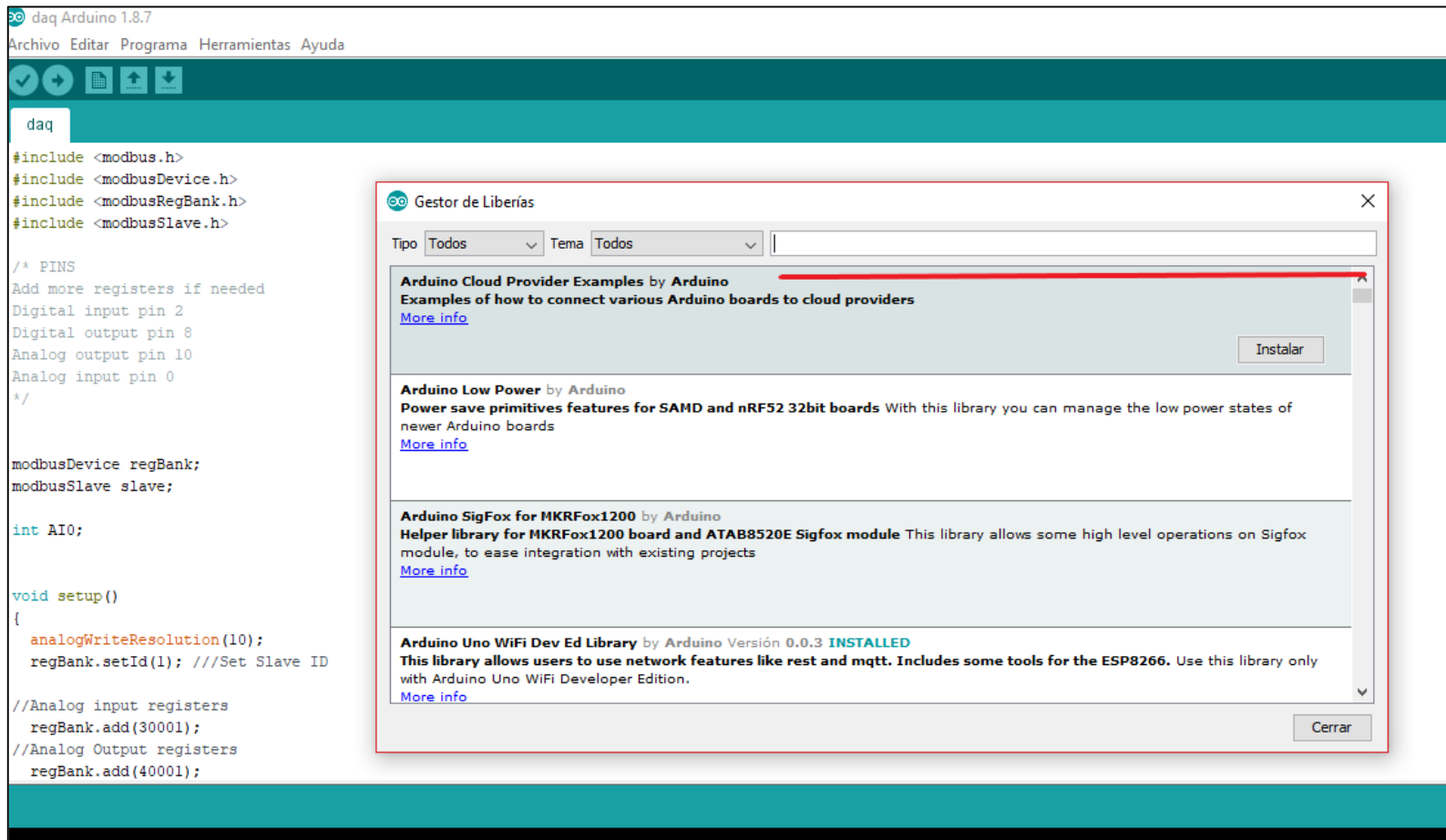


The screenshot shows the KEPServerEX - Runtime interface with the OPC Quick Client window open. The title bar of the OPC Quick Client window reads 'OPC Quick Client - Untitled *'. The menu bar includes File, Edit, View, Tools, and Help. The main area displays a tree view with 'Kepware.KEPServerEX.V5' and its sub-items. Below the tree is a table with columns: Item ID, Data Type, Value, Timestamp, Quality, and Update Count. The table is circled in red. Below the table is an event log with columns: Date, Time, and Event.

Item ID	Data Type	Value	Timestamp	Quality	Update Count
Modbus.Device1.Out_Digital	Boolean	0	16:33:01.985	Good	1
Modbus.Device1.Out_Analog1	Word	475	16:33:01.955	Good	1
Modbus.Device1.Out_Analog	Word	879	16:33:01.953	Good	1
Modbus.Device1.InAnalog	Word	0	16:33:01.919	Good	1
Modbus.Device1.In_Digital	Boolean	0	16:33:01.883	Good	1

Date	Time	Event
1/6/2019	4:33:01 PM	Added group 'Modbus.Device1' to 'Kepware.KEPServerEX.V5'.
1/6/2019	4:33:01 PM	Added 24 items to group 'Modbus.Device1._System'.
1/6/2019	4:33:01 PM	Added 5 items to group 'Modbus.Device1'.

PROGRAMACIÓN DE LA TARJETA ARDUINO



The image shows the Arduino IDE interface. The main window displays a code editor with the following code:

```
daq
#include <modbus.h>
#include <modbusDevice.h>
#include <modbusRegBank.h>
#include <modbusSlave.h>

/* PINS
Add more registers if needed
Digital input pin 2
Digital output pin 8
Analog output pin 10
Analog input pin 0
*/

modbusDevice regBank;
modbusSlave slave;

int AI0;

void setup()
{
  analogWriteResolution(10);
  regBank.setId(1); ///Set Slave ID

//Analog input registers
  regBank.add(30001);
//Analog Output registers
  regBank.add(40001);
```

The Library Manager window is open, showing a list of libraries:

- Arduino Cloud Provider Examples** by Arduino
Examples of how to connect various Arduino boards to cloud providers
[More info](#) Instalar
- Arduino Low Power** by Arduino
Power save primitives features for SAMD and nRF52 32bit boards With this library you can manage the low power states of newer Arduino boards
[More info](#)
- Arduino SigFox for MKRFox1200** by Arduino
Helper library for MKRFox1200 board and ATAB8520E Sigfox module This library allows some high level operations on Sigfox module, to ease integration with existing projects
[More info](#)
- Arduino Uno WiFi Dev Ed Library** by Arduino Versión 0.0.3 **INSTALLED**
This library allows users to use network features like rest and mqtt. Includes some tools for the ESP8266. Use this library only with Arduino Uno WiFi Developer Edition.
[More info](#) Cerrar



```
daq Arduino 1.8.7
Archivo Editar Programa Herramientas Ayuda
daq
#include <modbus.h>
#include <modbusDevice.h>
#include <modbusRegBank.h>
#include <modbusSlave.h>

/* PINS
Add more registers if needed
Digital input pin 2
Digital output pin 8
Analog output pin 10
Analog input pin 0
*/

modbusDevice regBank;
modbusSlave slave;

int AI0;

void setup()
{
  analogWriteResolution(10);
  regBank.setId(1); ///Set Slave I

//Analog input registers
  regBank.add(30001);
//Analog Output registers
  regBank.add(40001);
```

```
daq Arduino 1.8.7
Archivo Editar Programa Herramientas Ayuda
daq
  regBank.add(40001);
  regBank.add(40002);

  slave._device = &regBank;
  slave.setBaud(9600);
}
void loop(){
  while(1){

//Analog input  ***READ Twice deliberately

  AI0 = analogRead(0);

  regBank.set(30001, (word) AI0);
  delay(10);

//Analog output

  word AO10 = regBank.get(40001);

  analogWrite(DAC0,AO10);
```



DISEÑO HMI DEL MENÚ Y DE LA SIMULACIÓN DE LOS PROCESOS INDUSTRIALES



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO MÓDULO DE CONEXIÓN ENTRE UN SIMULADOR VIRTUAL

DE PROCESOS INDUSTRIALES Y UN PROGRAMADOR LÓGICO CONTROLABLE

EN EL LABORATORIO DE PLC DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN LATACUNGA.

MENU DE INGRESO

1

Caudal

2

Nivel

3

Presion

4

Temperatura



AUTOR:

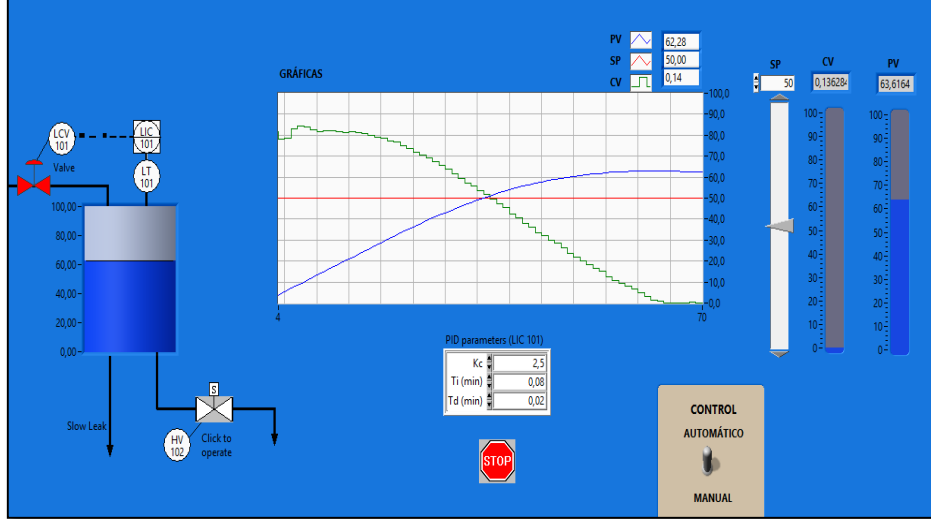
Adrián Fernando Gómez Espín

LATACUNGA - 2019

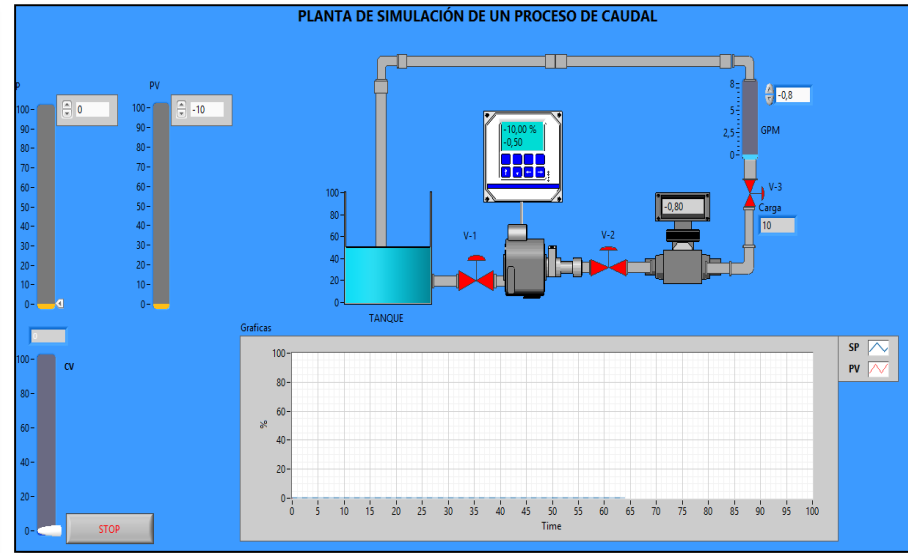


ESPE
ESCUELA POLITÉCNICA DEL EJÉRCITO
CAMINO A LA EXCELENCIA

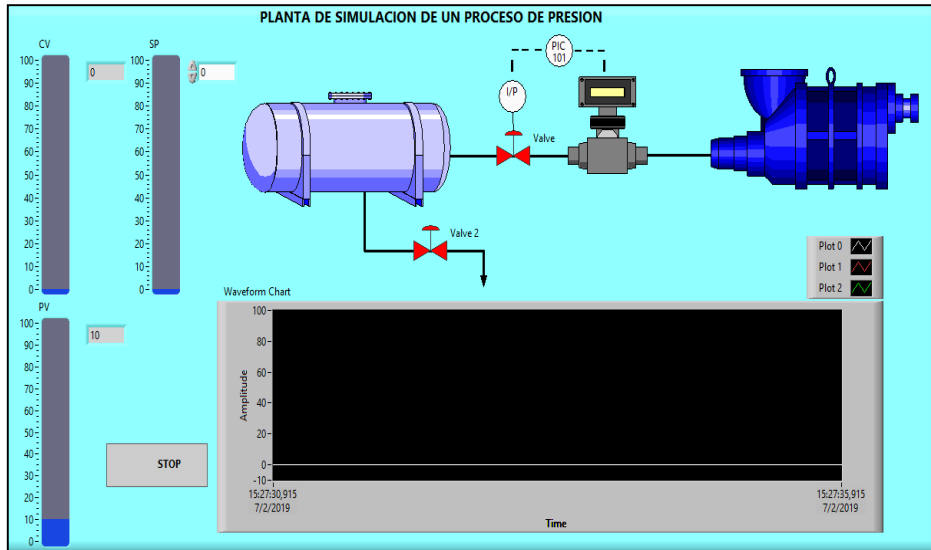
PLANTA DE SIMULACION DE UN PROCESO DE NIVEL



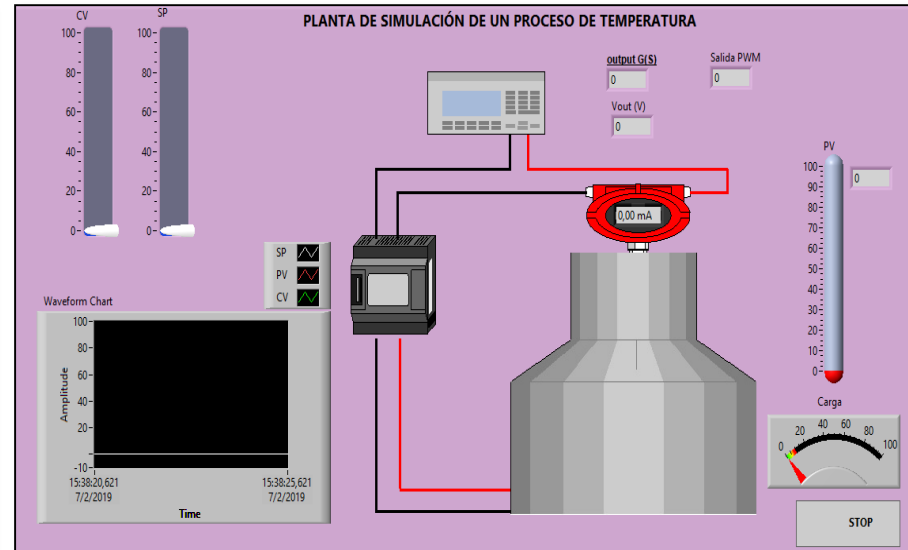
PLANTA DE SIMULACIÓN DE UN PROCESO DE CAUDAL

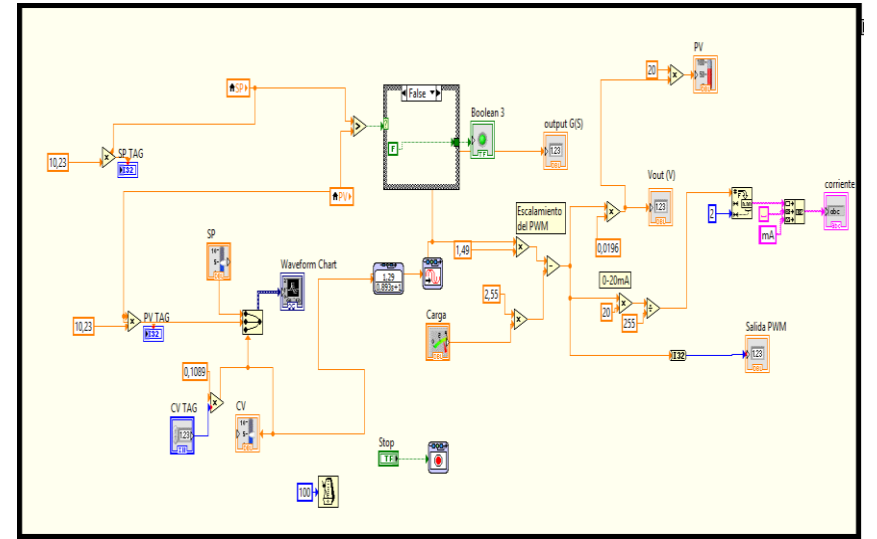
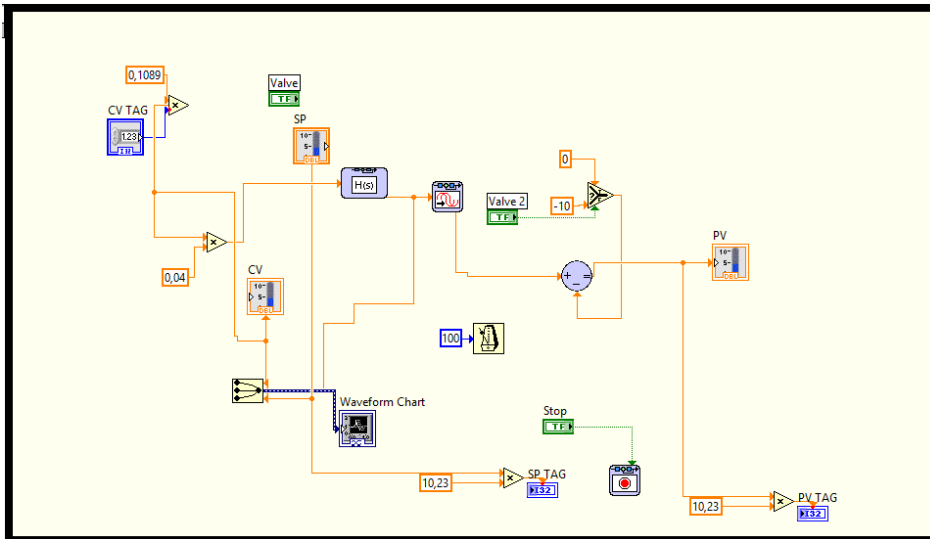
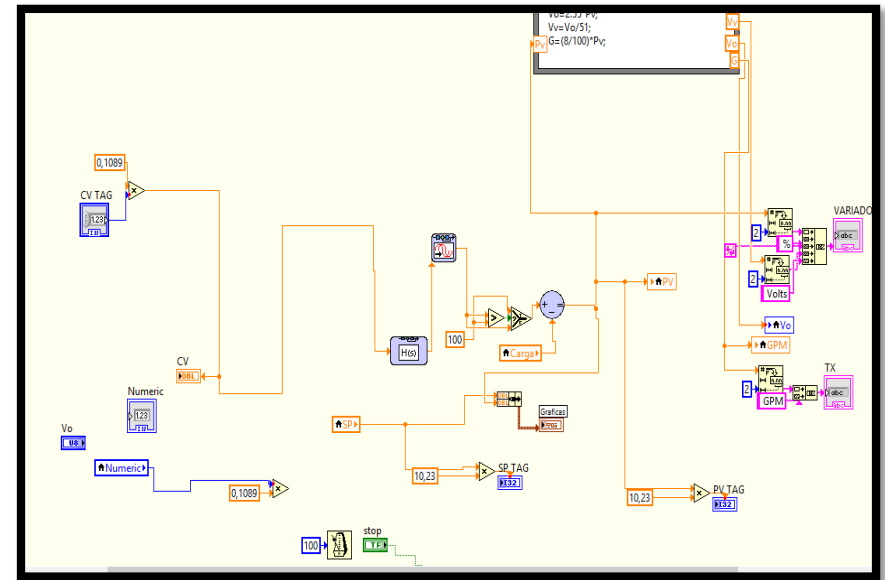
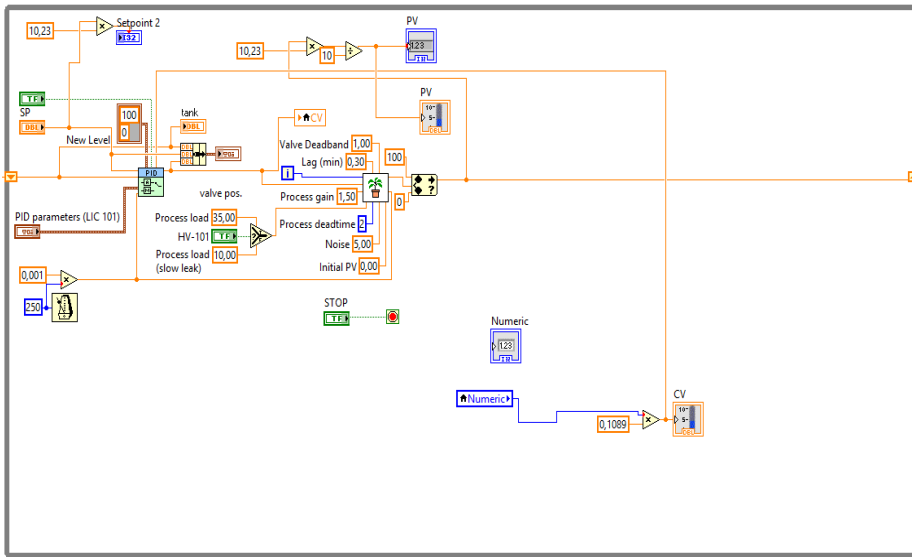


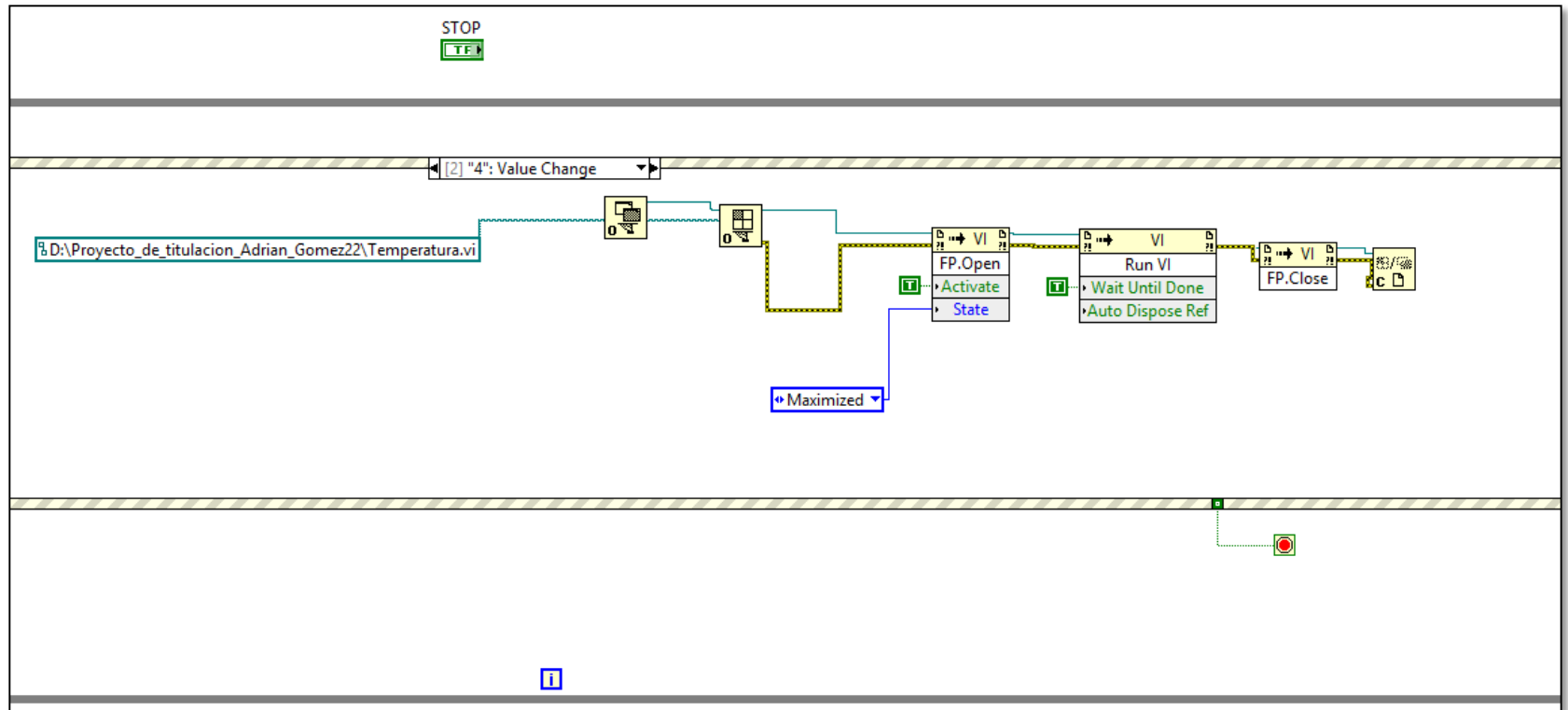
PLANTA DE SIMULACION DE UN PROCESO DE PRESION



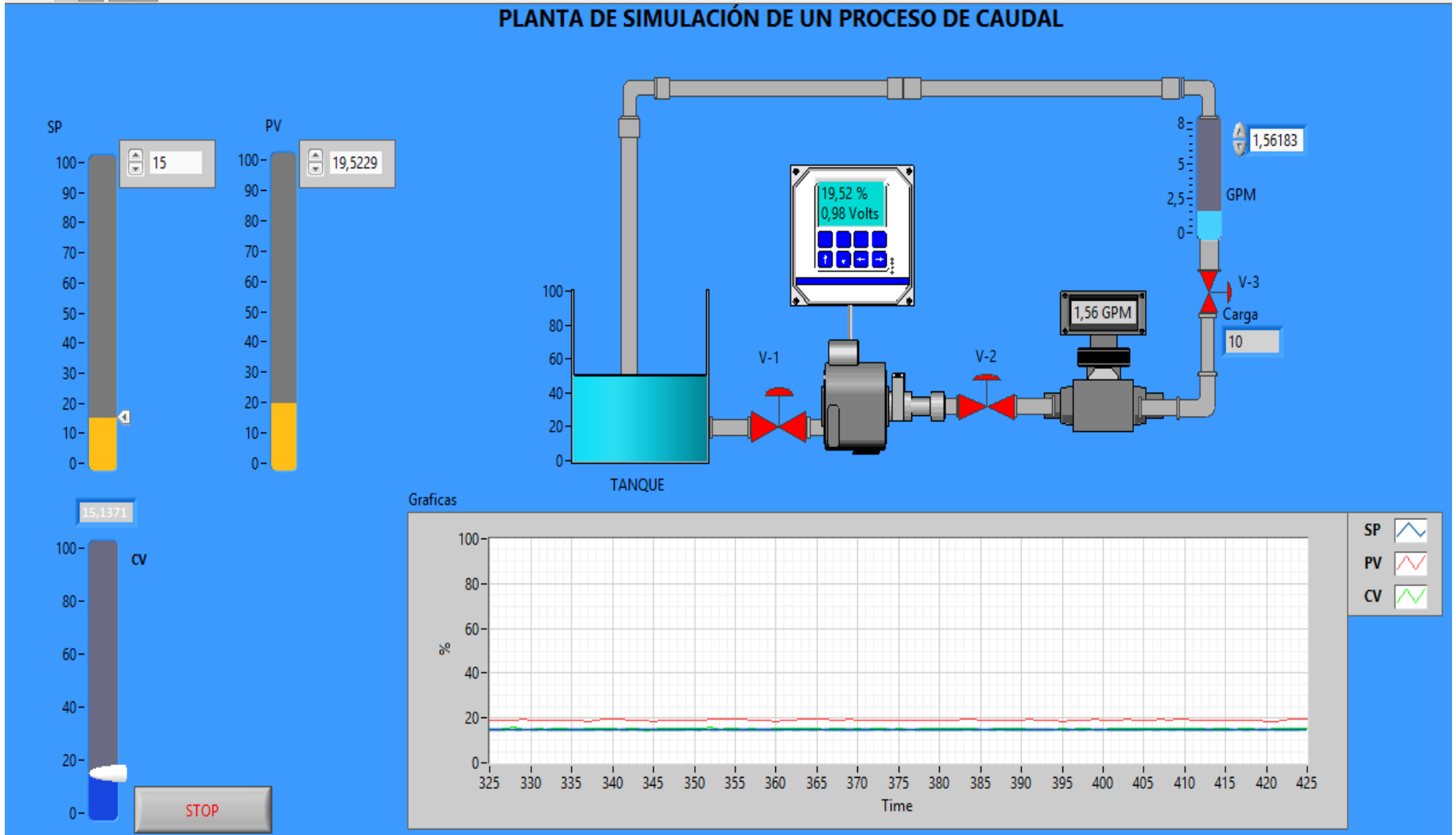
PLANTA DE SIMULACIÓN DE UN PROCESO DE TEMPERATURA







ANALISIS VALORES OBTENIDOS



% SET POINT	CV (VOLTAJE)	PV (VOLTAJE)
0	0	-0.03
15	1.53	1.46
30	3.31	2.91
45	4.55	4.45
60	6.02	5.91
75	7.51	7.45
90	9.11	8.8
100	10	9.71

$$e = \frac{Valor_{medido} - Valor_{real}}{Valor_{real}} * 100\%$$

% SET POINT	ERROR CV
0	0
15	2
30	10
45	1
60	0.3
75	0.1
90	1.2
100	0
	<u>%ERROR = 1.825</u>



- Se diseñó e implementó un prototipo de conexión didáctico a nivel software y hardware en base a circuitos electrónicos y a características de elementos integrantes a plantas de procesos industriales del laboratorio que facilita el aprendizaje de los estudiantes, beneficia el tiempo de ejecución de proyectos y aumentará la capacidad de experimentación en el laboratorio.
- Se desarrollaron Tag's con el protocolo de comunicación adecuado que permite y trabaja con señales de entradas y salidas análogas de 0, 3,3 y 10 V según la programación interna establecida y con señales digitales tipo sourcing para el control del proceso simulado.
- Las mediciones físicas de las variables de Control, Proceso y SetPoint presentaron variaciones de error al ser comparados con los simulados en LabVIEW tanto en la planta física como en el prototipo debido a tolerancias de las resistencias lineales del circuito electrónico, escalamientos y enlace tardío.
- La simulación de cada proceso posee un modelo matemático con una función de transferencia establecida, la modificación del proceso se lo realizó en el HMI mediante la programación de nuevos VI y SubVI que permiten el control de la ventana de simulación con una similitud del 95 por ciento a las plantas físicas que se encuentran en el laboratorio

- Se debe trabajar con señales de entradas análogas al PLC de 0 – 10 V.
- Los tipos de entrada y salida de señales digitales de los PLC's a utilizar en el enlace deben ser módulos de salida discreta tipo relé sourcing.
- Tomar en cuenta las propiedades que deben ser modificadas al implantar un nuevo protocolo de comunicación como: puertos, esclavos, paridad, etc.
- Se recomienda conocer la dirección precisa de las Tag's para que éstas puedan ser leída como el tipo de dato acorde a lo requerido (Char, Byte, Long. Word, ETC), o seleccionar la opción default para que automáticamente se ubique en la dirección y sea leído de forma correcta.
- Se debe realizar una buena configuración del OPC con el simulador y la tarjeta electrónica programable para evitar fallos.
- Es necesario tener el modelo matemático de la planta física, pues optimizará la simulación del proceso cuando sea introducida en la programación VI.

GRACIAS POR SU ATENCIÓN

