

UNIVERSIDAD CATOLICA DE SANTA MARIA

**ESCUELA PROFESIONAL DE INGENIERIA MECANICA, MECANICA-ELECTRICA
Y MECATRONICA**



**DISEÑO E IMPLEMENTACION DE UN MODULO DIDACTICO DE UN SISTEMA DE
CONTROL DE TIPO DISTRIBUIDO CON UN TERMINAL DE DIALOGO HMI COMO
MAESTRO Y DOS CONTROLADORES LOGICOS COMO ESCLAVOS
COMUNICADOS A TRAVES DE UNA RED ETHERNET PARA EL LABORATORIO
DE CONTROL Y AUTOMATIZACION**

Tesis presentada por los Bachilleres:

Flores Arana, Jesús Eduardo

Flores Juárez, Patricio José

Para optar el título profesional de Ingeniero Mecatrónico

AREQUIPA-PERU

2016

AGRADECIMIENTOS

Agradezco a Dios, a Jesús y Giovanna mis padres que siempre me apoyaron desde el inicio de mi carrera hasta ahora, cuando este proyecto ya no es una ilusión. A mi hermano Sergio por decirme siempre que sí se puede ante las complicaciones y amanecidas. Finalmente agradezco a mi enamorada Karen que nunca dejó de apoyarme, me ayudó y dio palabras de aliento y amor en los momentos más difíciles de la construcción de nuestro proyecto.

Jesús Eduardo Flores Arana



Agradezco a Dios, a José Luis y Patricia mis padres que siempre estuvieron conmigo durante toda mi carrera académica educándome y apoyándome. También agradezco a todos mis profesores y Hermanos de las Escuelas Cristianas que fueron una pieza fundamental en mi formación. Finalmente agradezco a mi enamorada María Alejandra por todos esos momentos que fue una fuente de inspiración y amor a mis proyectos.

Patricio José Flores Juárez

RESUMEN

Nuestro trabajo de investigación trata sobre el diseño y la implementación de un módulo didáctico de un sistema de control de tipo distribuido que contará con un terminal de diálogo HMI como maestro y dos controladores lógico programables como esclavos. Cada esclavo controlará un sistema diferente y estos se controlaran directamente desde nuestro maestro HMI sin necesidad de cualquier tipo de manipulación de los controladores esclavos.

Para desarrollar este módulo hicimos uso de una terminal táctil Magelis nueva para la cual usaremos como software de programación el Vijeo Designer que es su software por defecto. Los esclavos serán controladores lógicos Modicon M221 para los cuales usaremos el software de programación gratuito SoMachine Basic. Para la demostración implementamos dos módulos que representaron un control de nivel y un sistema de control domótico simple de una vivienda respectivamente y cada uno será asignado a un esclavo.

Finalmente, nuestra investigación pretendió demostrar que nuestro maestro HMI puede modificar las variables de nuestros sistemas mediante una comunicación Ethernet con los controladores lógicos sin perder eficiencia.

ABSTRACT

Our thesis describes the implementation of a didactic module of a distributed control system which will count with a HMI dialogue terminal as a master and two programmable logic controllers as slaves. Each slave may control a particular system and these may be controlled directly from our master HMI without the necessity of any kind of manipulation of the slave controllers.

To develop this module we will make use of a new Magelis touch screen terminal for which we used Vijeo Designer as its programming software that is its default software. The slaves will be Modicon M221 logic controllers for which we will use SoMachine Basic as their programming software. For the demonstration we implemented two modules that represented a level control and a simple domotic control system of a house respectively and each will be assigned to a slave controller.

Finally, our investigation pretends demonstrate that our HMI master can modify our system's variables through an Ethernet communication with the logic controllers without losing efficiency.

INTRODUCCION

Hoy en día no se puede hablar de comenzar una automatización en viviendas, en empresas privadas y/o públicas, en industrias o en escuelas, porque ya estamos en ella. Pero lo que si podemos decir es que tenemos que seguir avanzando en ésta, porque no podemos quedarnos en que un proceso sea llevado acabo y nosotros tengamos que estar asistiéndolo manualmente, sino tenemos que poder controlar no solo uno sino varios procesos y no manualmente sino poder controlar de manera remota los procesos, manipulando no solo una variable sino muchas al mismo tiempo y cada vez más rápida y eficientemente.

Y se quiere demostrar que aunque existen prejuicios aun sobre este tipo de tecnología, hay que olvidarnos de ciertos miedos hacia la automatización de procesos y darnos cuenta que es la manera más eficiente de controlar desde un proceso pequeño, luego una serie de procesos, una producción masiva, el control de una vivienda o finalmente el control una planta industrial completa.

Recordar que este proyecto tiene fines académicos mas no deja de ser una automatización que pueda ser aplicada en una planta o vivienda, pero en ésta ocasión es para utilizarse como elemento de estudio en próximos años para nuestros compañeros.

El presente proyecto inicia con el Planteamiento Metodológico, donde se describirá y justificará el problema con comparaciones actuales del mercado, que es donde debemos de probar que podemos competir contra grandes marcas donde nuestros estudios quedan comprobados, luego seguimos con el Planteamiento Teórico que brinda los conceptos básicos acerca de redes Ethernet, Controladores lógicos, Terminales táctiles, señales digitales y

analógicas, Domótica, sensores, actuadores, comunicación industrial y finalmente materiales y extras utilizados en este proyecto.

Para garantizar que éste proyecto es confiable y viable, se desarrolló el módulo donde realizamos todas las pruebas necesarias; Una evaluación económica, pruebas de comunicación e incluimos manuales de funcionamiento para el módulo, que sirvan como guías de prácticas para nuestros compañeros en años próximos en el laboratorio de nuestra universidad.



INDICE GENERAL

RESUMEN.....	3
ABSTRACT.....	4
INTRODUCCION.....	5
CAPITULO I: PLANTEAMIENTO METODOLOGICO	
1.1 TITULO.....	21
1.2 PLANTEAMIENTO DEL PROBLEMA.....	21
1.3 JUSTIFICACION.....	23
1.4 HIPOTESIS.....	24
1.5 OBJETIVOS.....	24
1.5.1 OBJETIVO GENERAL.....	24
1.5.2 OBJETIVO ESPECIFICOS.....	25
1.6 ALCANCES.....	25
1.7 ANTECEDENTES.....	26
1.7.1 ANTECEDENTES INTERNACIONALES.....	26
1.7.2 ANTECEDENTES LOCALES.....	27
1.8 DIAGRAMA DE GANTT.....	29
CAPITULO II: PLANTEAMIENTO TEORICO	
2.1 REDES DE COMPUTADORAS.....	30
2.1.1 PAN (Red de administración personal).....	30
2.1.2 LAN (Red de área local).....	30
2.1.3 CAN (Red de área de campus).....	30

2.1.4 WAN (Red de área extensa).....	30
2.1.5 MAN (Red de área metropolitana).....	31
2.1.5.1 DQDB (Dual-Queue Dual-Bus).....	31
2.1.6 Ethernet.....	32
2.1.6.1 Dirección IP.....	33
2.1.6.2 Máscara de subred.....	34
2.2 PLC.....	34
2.2.1 Programación.....	37
2.2.1.1 Lista de instrucciones (IL).....	38
2.2.1.2 Plano de contactos (LD).....	39
2.2.1.3 Diagrama secuencial o Grafcet (SFC).....	40
2.2.2 Señales analógicas y discretas.....	41
2.2.3 SoMachine Basic.....	42
2.3 HMI.....	42
2.3.1 Interfaz de hardware.....	43
2.3.2 Interfaz de software.....	43
2.3.3 Interfaz de software-hardware.....	43
2.4 Sensores y Actuadores.....	44
2.4.1 Sensores.....	44
2.4.1.1 Linealidad y Correlación lineal.....	45
2.4.1.2 Rango de medida.....	46
2.4.1.3 Precisión.....	46
2.4.1.4 Offset o Desviación de cero.....	46

2.4.1.5 Resolución.....	46	
2.4.1.6 Rapidez de respuesta.....	46	
2.4.1.7 Sensibilidad.....	46	
2.4.1.8 Repetitividad.....	46	
2.4.2 Actuadores.....	46	
2.4.2.1 Hidráulicos.....	47	
2.4.2.2 Neumáticos.....	47	
2.4.2.3 Eléctricos.....	48	
2.4.2.4 Mecánicos.....	48	
2.4.2.5 Válvula Solenoide.....	48	
2.4.2.6 Servomotor.....	48	
2.4.2.7 Bomba Hidráulica.....	49	
2.5 Sistema de control.....	50	
2.5.1 Sistema de control de lazo abierto.....	50	
2.5.2 Sistema de control de lazo cerrado.....	51	
2.5.3 Sistema de control distribuido.....	51	
CAPITULO III: DISEÑO Y CONSTRUCCION DE LA SOLUCION		
3.1 Diseño del módulo.....	52	
3.2 Construcción del módulo.....	69	
CAPITULO IV: PROGRAMACION Y COMUNICACION DE LA RED		
4.1 Programación de los PLC.....	113	
4.2 Programación del terminal HMI.....	121	
CAPITULO V: PRUEBAS Y RESULTADOS FINALES.....		138

CONCLUSIONES

RECOMENDACIONES

BIBLIOGRAFIA

ANEXOS



INDICE DE TABLAS

Tabla 1.1 Diagrama de Gantt.....	29
Tabla 2.1 Comparativa de cables Ethernet.....	33



INDICE DE FIGURAS

CAPITULO I: PLANTEAMIENTO METODOLOGICO

CAPITULO II: PLANTEAMIENTO TEORICO

Fig. 2.1 Tipos de redes.....	32
Fig. 2.2 Gráfica tentativa del PLC.....	36
Fig. 2.3 PLC y Código de Ladder.....	38
Fig. 2.4 Tipos de Contactos en Ladder.....	39
Fig. 2.5 Tipos de Bobinas en Ladder.....	40
Fig. 2.6 Señales Analógicas y Digitales.....	42
Fig. 2.7 Grafico estimado de una pantalla de un terminal HMI.....	43
Fig. 2.8 Tipos de Sensores.....	44
Fig. 2.9 Tipos de Actuadores.....	47
Fig. 2.10 Niveles de funcionamiento de un sistema de control distribuido típico.....	51

CAPITULO III: DISEÑO Y CONSTRUCCION DE LA SOLUCION

Fig. 3.1 Elección de los PLC's y el HMI.....	52
Fig. 3.2 Resumen de modulo principal.....	53
Fig. 3.3 Resumen de submodulo de nivel de tanque.....	53
Fig. 3.4 Resumen de submodulo de control de vivienda.....	54
Fig. 3.5 Circuito para luz automática de noche con el TRIAC BT136.....	55
Fig. 3.6 Alternativa para el circuito de detección de humo con el 555.....	57
Fig. 3.7 Circuitos para hacer un control remoto con leds infrarrojo.....	58

Fig. 3.8 Circuito para encender un ventilador con el LM324.....	59
Fig. 3.9 Circuito para detección de movimiento con el 555.....	60
Fig. 3.10 Circuito para el control de giro de un servomotor con el 555.....	61
Fig. 3.11 Cálculo de los valores de resistencias para el giro del servomotor.....	62
Fig. 3.12 Cálculo de otra alternativa de valores de resistencias para el giro del servomotor.....	62
Fig. 3.13 Armado del circuito del servomotor.....	64
Fig. 3.14 Corrección de errores en el armado de los circuitos.....	64
Fig. 3.15 Quemado de la placa del circuito del sensor de movimiento.....	65
Fig. 3.16 Placas listas de los sensores a utilizar.....	65
Fig. 3.17 Placa del circuito de sensor de movimiento.....	66
Fig. 3.18 Placa del circuito del sensor de humo.....	66
Fig. 3.19 Placa del circuito del sensor de nivel de tanque.....	67
Fig. 3.20 Placa del circuito del control del servomotor.....	67
Fig. 3.21 LDR para el circuito de iluminación automática.....	68
Fig. 3.22 Interruptor para el circuito de control total de iluminación.....	68
Fig. 3.23 Diferentes espesores de Acrílico.....	69
Fig. 3.24 Espesor de plancha elegida: 4 mm.....	69
Fig. 3.25 Primer boceto de los planos de la vivienda.....	70
Fig. 3.26 Inicio del armado de los primeros prototipos de la vivienda.....	70
Fig. 3.27 Boceto terminado en papel de la vivienda.....	71
Fig. 3.28 Plano del boceto elegido para la vivienda.....	71
Fig. 3.29 Equivalencias en escala de las medidas de la vivienda.....	72
Fig. 3.30 Distribución de las piezas en la plancha a comprar.....	72

Fig. 3.31 Planos de los tanques a utilizar.....	73
Fig. 3.32 Botella de cloroformo para unir las planchas cortadas de acrílico.....	73
Fig. 3.33 Secado de los tanques de agua ya unidos.....	74
Fig. 3.34 Disposición de los tanques de agua.....	74
Fig. 3.35 Pegado de las paredes del submodulo de vivienda.....	75
Fig. 3.36 Posición final del submodulo de vivienda.....	75
Fig. 3.37 Unimos el piso a la vivienda.....	75
Fig. 3.38 Posicionamiento de las puertas de entrada y garaje.....	76
Fig. 3.39 Puerta de entrada con sus bisagras.....	76
Fig. 3.40 Posicionamiento del techo de la vivienda.....	77
Fig. 3.41 Elección del perfil en U de aluminio.....	77
Fig. 3.42 Elección y posicionamiento de los perfiles en H de plástico.....	77
Fig. 3.43 Colocación de los perfiles en U y en H para el techo de la vivienda.....	78
Fig. 3.44 Sensores en sus respectivos contenedores de Acrílico.....	78
Fig. 3.45 Sistema de calefacción colocado en su posición final.....	79
Fig. 3.46 Sensor de movimiento colocado en su posición final.....	79
Fig. 3.47 Sensor de humo y ventilador disipador de humo colocado en su posición final.....	80
Fig. 3.48 LDR colocado en su posición final.....	80
Fig. 3.49 Sistema de soporte para el eje de la puerta de garaje.....	81
Fig. 3.50 Puerta de garaje montada en su posición final.....	81
Fig. 3.51 Puerta de entrada con sus accesorios.....	81
Fig. 3.52 Ventana de la cocina con sus accesorios.....	82
Fig. 3.53 Ventana de la habitación principal con sus accesorios.....	82

Fig. 3.54 Construcción parcial del submodulo de vivienda.....	83
Fig. 3.55 Elección de las canaletas para el paso de los cables.....	83
Fig. 3.56 Ácido para el quemado de las placas y la placa donde se quemarán los circuitos.....	84
Fig. 3.57 Preparación para el planchado de las placas.....	84
Fig. 3.58 Resultado luego de haber pasado por el ácido las placas.	85
Fig. 3.59 Resultado final de las placas.	85
Fig. 3.60 Nombres impresos en la placa.	86
Fig. 3.61 Elección de la fuente de 24 v.....	86
Fig. 3.62 Elección de la fuente de 12 v.....	87
Fig. 3.63 Cable y terminal de las fuentes.....	87
Fig. 3.64 Distintos tipos de relay.....	88
Fig. 3.65 Distribución de las entradas y salidas con sus respectivos relay.....	88
Fig. 3.66 Posicionamiento de todos los cables conectados a las fuentes en sus posiciones referenciales dentro del sub-modulo.....	89
Fig. 3.67 Pruebas con las fuentes encendidas.....	89
Fig. 3.68 Cableado y puesta de canaletas.....	89
Fig. 3.69 Elección del sensor inductivo industrial.....	90
Fig. 3.70 Armado del soporte para el sensor inductivo.....	90
Fig. 3.71 Posicionamiento del sensor inductivo, del auto y del servomotor dentro del garaje.....	91
Fig. 3.72 Borneras de entradas y salida.	91
Fig. 3.73 Bomba elegida.	91
Fig. 3.74 Datos de placa de la bomba elegida.....	92
Fig. 3.75 Válvula Solenoide.....	92

Fig. 3.76 Válvula Check.....	93
Fig. 3.77 Pulsadores de INICIO y STOP.	93
Fig. 3.78 Válvulas manuales y accesorios.	94
Fig. 3.79 Armado de la tubería de succión.	94
Fig. 3.80 Armado de la tubería de descarga.	94
Fig. 3.81 Armado de la tubería que conectará los dos tanques con la válvula solenoide.....	95
Fig. 3.82 Elección de nuestros PLC's y HMI.....	95
Fig. 3.83 Datos del HMI.....	96
Fig. 3.84 Datos del PLC.....	97
Fig. 3.85 Vista de los PLC's y el HMI.....	99
Fig. 3.86 Elección de los fusibles.	100
Fig. 3.87 Diseño del circuito de luces de alerta de nivel de tanque.....	100
Fig. 3.88 Diseño de los relay del módulo de nivel.....	101
Fig. 3.89 Diseño de los relay de los circuitos alimentados por 12v.....	102
Fig. 3.90 Diseño de los relay de los circuitos alimentados por 24v.....	103
Fig. 3.91 Relays foto de las 3 placas terminadas.....	103
Fig. 3.92 Armado de las borneras del módulo de nivel.....	104
Fig. 3.93 Foto delantera del módulo de nivel.....	104
Fig. 3.94 Todas las placas juntas antes del montaje.....	105
Fig. 3.95 Switch elegido para la conexión de la red interna.....	105
Fig. 3.96 Vista exterior del módulo central de control.....	106
Fig. 3.97 Vista interior del módulo central de control.....	106
Fig. 3.98 Tabla para perforar las entradas en el módulo central.....	107

Fig. 3.99 Módulo central ya perforado.....	107
Fig. 3.100 Inserción de los terminales.....	108
Fig. 3.101 Vista posterior de la puerta del módulo.....	108
Fig. 3.102 Puerta del módulo terminada con el terminal HMI colocado.....	109
Fig. 3.103 Vista posterior de la puerta terminada.....	109
Fig. 3.104 Posicionamiento interno de los elementos necesarios.....	110
Fig. 3.105 Cableado interno con los elementos ya fijos.....	110
Fig. 3.106 Cableado de los PLC con los terminales circulares.....	111
Fig. 3.107 Montaje completo de nuestro módulo.....	112
 CAPITULO IV: PROGRAMACION Y COMUNICACION DE LA RED	
Fig. 4.1 Interfaz de SoMachine Basic con el PLC ya elegido.....	113
Fig. 4.2 Dirección IP asignada para Ethernet del primer PLC.....	114
Fig. 4.3 Programa en Ladder para el módulo de nivel de tanque (1).....	115
Fig. 4.4 Programa en Ladder para el módulo de nivel de tanque (2).....	116
Fig. 4.5 Dirección IP asignada para Ethernet del segunda PLC.....	116
Fig. 4.6 Programa en Ladder para el módulo de domótica (1).....	118
Fig. 4.7 Programa en Ladder para el módulo de domótica (2).....	118
Fig. 4.8 Programa en Ladder para el módulo de domótica (3).....	119
Fig. 4.9 Programa en Ladder para el módulo de domótica (4).....	119
Fig. 4.10 Terminal RJ-45.....	120
Fig. 4.11 PLC ya cargado con el programa detectando entradas y salidas.....	120
Fig. 4.12 Inicialización de la pantalla de Vijeo Designer.....	121

Fig. 4.13 Valores IP asignados para el terminal HMI.....	121
Fig. 4.14 Respuesta del terminal para la inicialización.....	122
Fig. 4.15 Interfaz de Vijeo Designer para el diseño del SCADA.....	122
Fig. 4.16 Asignando la IP del terminal para la descarga de datos.....	123
Fig. 4.17 Panel de entrada.....	123
Fig. 4.18 Panel principal.....	124
Fig. 4.19 Panel de control domótico.....	124
Fig. 4.20 Panel de control del sensor de movimiento.....	125
Fig. 4.21 Panel de control del sensor de humo.....	125
Fig. 4.22 Panel de control del sistema de calefacción.....	126
Fig. 4.23 Panel de control del sistema de alumbrado.....	126
Fig. 4.24 Panel de control de la puerta de garaje.....	127
Fig. 4.25 Panel de control de nivel de tanque.....	127
Fig. 4.26 Panel de control de llenado y vaciado de la simulación de un tanque de agua.....	128
Fig. 4.27 Ventana de acciones.....	128
Fig. 4.28 Ventana de variables internas y externas.....	134
Fig. 4.29 Ventana para la configuración de la comunicación con el primer PLC.....	134
Fig. 4.30 Ventana para la configuración de la comunicación con el segundo PLC.....	135
Fig. 4.31 Descarga del Vijeo Designer al terminal HMI Magelis.....	135
Fig. 4.32 Pantalla de encendido del terminal HMI Magelis.....	136
Fig. 4.33 Pruebas de comunicación en la red.....	136
[Fig. 4.34 Programa ya cargado en el terminal HMI Magelis.....	137

CAPITULO V: PRUEBAS Y RESULTADOS FINALES

Fig. 5.1 HMI conectado a través del cable Ethernet.....	138
Fig. 5.2 PLC's conectados a través del cable Ethernet.....	139
Fig. 5.3 Error de comunicación.....	140
Fig. 5.4 Luces de Entrada apagadas.....	141
Fig. 5.5 Activando las luces de entrada en el HMI.....	141
Fig. 5.6 Respuesta del PLC a la activación de Luces de Entrada.....	142
Fig. 5.7 Luces de Entrada encendidas.....	143
Fig. 5.8 Luces Internas apagadas.....	143
Fig. 5.9 Luces Internas encendidas.....	144
Fig. 5.10 Ventilador de Humo apagado.....	144
Fig. 5.11 Respuesta del PLC a la activación del sensor.....	145
Fig. 5.12 Respuesta del HMI a la activación del sensor.....	146
Fig. 5.13 Ventilador de Humo encendido.....	146
Fig. 5.14 Focos de calefactor apagados.....	147
Fig. 5.15 Respuesta del PLC a los focos de calefactor encendidos.....	148
Fig. 5.16 Respuesta del HMI al sensor LM35 con temperatura bajas.....	148
Fig. 5.17 Focos de calefactor encendidos.....	149
Fig. 5.18 Ventilador de calefactor apagado.....	149
Fig. 5.19 Respuesta del PLC a la activación del ventilador de calefactor.....	150
Fig. 5.20 Respuesta del HMI al sensor LM35 con temperaturas altas.....	150
Fig. 5.21 Ventilador de calefacción encendido.....	151
Fig. 5.22 Respuesta del HMI a alarma de intruso apagada.....	151

Fig. 5.23 Respuesta del PLC a alarma de intruso encendida.....	152
Fig. 5.24 Respuesta del HMI a alarma de intruso encendida.....	152
Fig. 5.25 Respuesta del HMI a la activación de la bomba.....	153
Fig. 5.26 Luego del llenado del tanque, la respuesta del HMI a la activación de la válvula solenoide.....	154
Fig. 5.27 Simulación en el HMI del drenado del tanque.....	154
Fig. 5.28 Presentación final lista.....	155



CAPITULO I

PLANTEAMIENTO METODOLOGICO

1.1 Titulo

Diseño e Implementación de un módulo didáctico de un sistema de control de tipo distribuido con un terminal de dialogo HMI como maestro y dos controladores lógicos como esclavos comunicados a través de una red Ethernet para el laboratorio de Control y Automatización.

1.2 Planteamiento del problema

Desde los inicios de la humanidad, el hombre ha buscado herramientas que pueda utilizar para facilitar la realización de diversas actividades cotidianas en sus vidas. Estas herramientas fueron creadas a mano a partir de la naturaleza por el hombre. A medida que el hombre fue evolucionando, sus actividades cotidianas también y así mismo eran necesarias aún más herramientas diferentes cada vez con un mayor grado de complejidad en su construcción hasta el punto en que se llegó a necesitar de una herramienta para la creación de otra.

Con el paso de los años se fueron desarrollando diferentes herramientas que permitían al hombre facilitar sus tareas con el fin de que el trabajo sea mayor para la herramienta y menor para el hombre hasta llegar al punto en que la herramienta trabaje sin necesidad de alguien que la opere. Pero este desarrollo tenía un límite y este fue la energía, para realizar cualquier trabajo es necesaria energía que era proporcionada por el trabajo del hombre directamente.

Luego se descubrió la energía almacenada en los combustibles que podía transferirse en forma de calor a ciertas herramientas para que estas funcionen, por ejemplo las locomotoras y los

automóviles, pero estas seguían siendo manipuladas por alguien. Cuando se descubrió la energía eléctrica se abrió una puerta a muchas innovaciones en el campo de las herramientas y la automatización que continúan hasta el día de hoy.

En la actualidad se puede observar a herramientas creando herramientas que luego crearan otras herramientas, pero siempre es necesario que el hombre supervise y controle estos procesos y por lo tanto a cada uno de los componentes. Para esta tarea de supervisión también se han creado herramientas que en este caso las conocemos como sistemas de control.

Los sistemas de control se encargan de administrar, dirigir, ordenar o regular el comportamiento de otros sistemas y el hombre es el encargado de supervisar que el control sea bueno, pero en una industria actual la cantidad de sistemas que necesitan ser controlados es bastante grande, por eso el hombre creó lo que hoy conocemos como sistema de control distribuido.

Este sistema se aplica a grandes industrias y está formado por dos clases de controladores, los maestros y los esclavos, esto quiere decir que tendremos sistemas de control que controlen otros sistemas de control para así poder juntar varios procesos en uno solo y en vez de necesitar 5 supervisores para 5 procesos, solo necesitaremos de 1 que supervise al sistema controlado por el maestro.

Para esto es necesaria una comunicación eficiente que en nuestro caso será una comunicación Ethernet que es un estándar de redes de área local o LAN para computadores que se tomó como base para la redacción del estándar internacional IEEE 802.3.

1.3 Justificación

El control de procesos está actualmente cobrando un gran apoyo, siendo utilizado en las nuevas formas de automatización mediante el control por PLC's con programación sencilla, lógica difusa, redes neuronales o controlador PID, pero lo resaltante de esto es que se está implementando en todo tipo de industrias, viviendas o empresas. A pesar de un primer rechazo por desconocimiento teórico, cada vez es mayor la aceptación del público y el anhelo de tener un sistema completamente automatizado sea en casa, empresa o una industria.

Nuestra investigación plantea que para lograr que una automatización sea completa es necesario más que poder controlar un proceso sin estar personal y manualmente modificando parámetros, poder controlar este subproceso a través de un controlador mayor llamado maestro, que nos dé resultados de lo que necesitamos de varios procesos separados e independientes uno de otro que pueden o no estar en el mismo recinto, pero que sin necesidad de estar presentes podamos no solo obtener los resultados sino también poder variar y manipular variables para modificar dichos resultados.

La investigación también se plantea para ser materia de estudio y aplicación para que la formación profesional impartida en la universidad sea también práctica, en áreas de estudio que solo cuentan con parte teórica.

El valor agregado de este proyecto se basa en el desarrollo de un módulo físico que pueda demostrar todos los puntos anteriormente mencionados, de la ejecución de órdenes a través de un terminal HMI sin intervenir directamente sobre el subproceso, implementado con costos y materiales accesibles, y dejando guías de laboratorio para su uso en el laboratorio de nuestra escuela profesional en adelante.

Como innovación extra, en nuestro módulo de pruebas tenemos dos tipos de sensores: industriales, con estándares internacionales, y otros que no los son y que pueden ser implementados fácilmente pero que no necesariamente cumplen los estándares más comunes (como la tensión de la señal de salida). Nuestra investigación puesta en práctica al momento del diseño de nuestros sensores, determina que estamos dejando a nuestra casa de estudios una tecnología propia que puede ser modificada, ampliada o simplemente utilizada para alguna práctica en algún curso. Nótese que no son sensores industriales sino son sensores que simulan los diferentes procesos con los que trabajamos en nuestro módulo. Pero no dejan de ser elementos que nos sirvan para cualquier aplicación fuera de este proyecto. Al tener ambos tipos de sensores en un mismo sistema podemos comprobar la versatilidad que tienen los PLC's escogidos, puesto que no discriminan el ser componentes de la misma empresa, de empresas distintas o tecnología nuestra.

1.4 Hipótesis

Dado que un controlador "A" pueda controlar las variables "M, N y P" de un proceso y otro controlador "B" pueda controlar las variables "X, Y y Z" de otro proceso, es posible diseñar e implementar un controlador "C" que controle "A y B" y por ende también a las variables "M, N, P, X, Y y Z" de manera remota.

1.5 Objetivos

1.5.1 Objetivo General

Diseñar e Implementar un módulo didáctico de un sistema de control de tipo distribuido con un terminal de dialogo HMI Magelis como maestro y dos controladores lógicos Modicon M221 como esclavos comunicados a través de una red Ethernet.

1.5.2 Objetivos Específicos

- 1.5.2.1 Diseñar e implementar una interfaz gráfica humano-máquina para nuestra terminal de dialogo (maestro).
- 1.5.2.2 Realizar una comunicación Ethernet eficiente mediante la cual sea posible enviar y recibir información para armar resultados y modificar variables en los procesos independientes.
- 1.5.2.3 Diseñar y construir un módulo donde este implementado en forma física nuestro sistema de control maestro-esclavo y a su vez dos módulos de prueba donde estarán montadas las maquetas de nuestros procesos.
- 1.5.2.4 Diseñar y realizar las guías de prácticas para comprender el uso y todas las funciones de nuestro proyecto: “Diseño e Implementación de un módulo didáctico de un sistema de control de tipo distribuido con un terminal de dialogo HMI como maestro y dos controladores lógicos como esclavos comunicados a través de una red Ethernet para el laboratorio de Control y Automatización” para ayudar con el aprendizaje de este tipo de sistema.

1.6 Alcances

El presente proyecto está orientado a facilitar el control de dos plantas con un acceso remoto a ellas, modificando variables sin necesidad de estar presente en la misma planta, gracias a la red Ethernet. Tiene la finalidad de explicar y demostrar de forma física la conexión esclavo-maestro estudiada en clases teóricas, no optando por una conexión inalámbrica pero si dejándola como recomendación para una posible próxima ampliación.

Servirá para demostrar que lo estudiado en nuestros años como estudiantes de nuestra casa de estudios Universidad Católica de Santa María, es utilizado para la creación de redes a

nivel industrial que pueden ser aplicados en la vida profesional como una opción para el control de alguna empresa o de alguna vivienda.

Dejaremos para nuestros compañeros un módulo físico implementado con lo que este proyecto específica, que servirá para ser utilizado en la explicación y practica de redes a nivel industrial o doméstico, con ejemplos reales del mundo en que vivimos.

1.7 Antecedentes

1.7.1 Antecedentes Internacionales

1.7.1.1 *"Diseño e implementación de un sistema de control de dispositivos Maestro - Esclavo basados en la Red Industrial AS-i para el Laboratorio de Automatización Industrial"* (2015).

Tesista: Byron José Orellana Proaño

Universidad Politécnica Salesiana de Ecuador

Es un proyecto que enseña las diversas formas de redes que existen y la forma de conectarlas, utilizaron el programa TIA PORTAL V.13 para la configuración de red y direccionamiento de los esclavos bajo las normas en AS-i. Realizaron además un manual de prácticas para capacitar a los estudiantes próximos a venir sobre la red AS-i y la interacción con Profinet porque el PLC y el HMI se comunican a través de Profinet.

1.7.1.2 *"Diseño e implementación de un sistema de control distribuido didáctico para el laboratorio de control de procesos"* (2006).

Tesista: Henry Bolívar Andagana Junta

Edison Fernando Churo Guayasamín

Escuela Politécnica Nacional de Ecuador

El objetivo del proyecto fue enlazar 4 módulos, los cuales controlan flujo, presión y temperatura conformando de ésta manera un sistema de control distribuido. Se usó el protocolo de comunicación Modbus y fueron monitoreados y controlados mediante

computadoras que forman parte de una red local. EL software SCADA utilizado fue Lookout, que permite control, monitoreo y la elaboración de los HMI's.

1.7.2 Antecedentes Locales

1.7.2.1 *"Diseño e Implementación de un sistema de control industrial, para monitoreo de temperatura, empleando un controlador lógico programable"* (2009)

Tesista: Yuddy Carla Ayala Campana

Carlos Enrique Chávez Quispe

Universidad Católica de Santa María

Es un proyecto en el que se diseña e implementa un sistema de control de temperatura para un proceso industrial de enfriamiento de agua simulado utilizando un PLC Twido y una Interfaz Hombre Máquina (HMI) Magelis XBT R. La comunicación entre el terminal y el PLC es mediante Modbus según el principio esclavo – maestro, pero siendo la función de este terminal netamente de monitoreo. También se hizo uso de sensores muy difundidos en la mayoría de procesos industriales, la termocupla tipo J y la Pt100, así como una electroválvula (Danfoss) como actuador.

1.7.2.2 *"Diseño de RTNet/IP protocolo para Aplicaciones en tiempo real de sistemas de control distribuido"* (2007).

Tesista: Carlos Enrique Reinoso Espinoza

Universidad Católica de Santa María

El propósito de su investigación fue demostrar que el protocolo presentado fue capaz de establecer un intercambio de datos en tiempo real en su tecnología de red no necesariamente diseñada para este comportamiento como la norma IEE 802.3 que define variables de acceso en medios no determinísticos.

1.7.2.3 *"Diseño e implementación de un módulo de caudal controlado por PLC y supervisión de proceso mediante SCADA"*(2012).

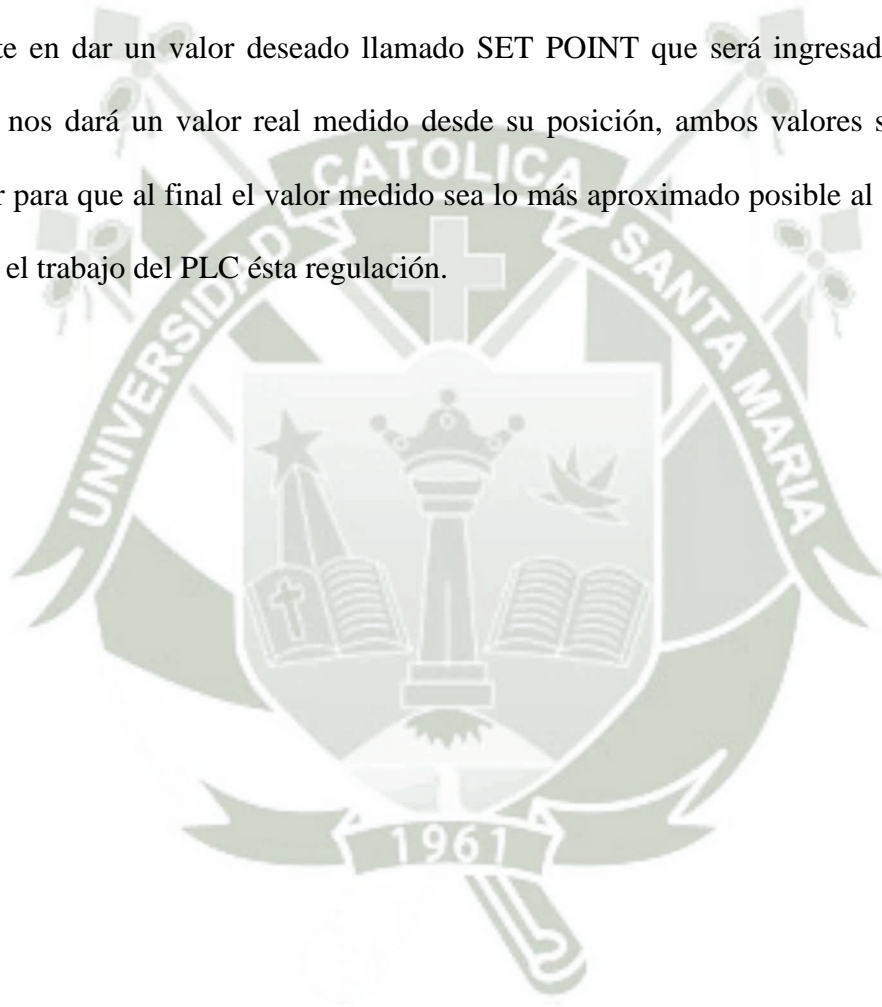
Tesista: Edgar Chaca Cuti

Fernando David Siles Nates

Universidad Católica de Santa María

Es un módulo didáctico para el centro de automatización de la universidad, para que los alumnos realicen prácticas de control y procesos de automatización.

Se hizo de un sensor de caudal industrial. El funcionamiento en sí del procesos total consiste en dar un valor deseado llamado SET POINT que será ingresado al PLC y el sensor nos dará un valor real medido desde su posición, ambos valores se tendrán que regular para que al final el valor medido sea lo más aproximado posible al valor deseado, siendo el trabajo del PLC ésta regulación.



1.8 Diagrama de Gantt

TIEMPO ACTIVIDAD	Febrero		Marzo				Abril				Mayo			
	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Plan de Tesis	X	X	X											
Presentación del plan				X										
Diseño de un prototipo en papel					X									
Adquisición de Dispositivo						X								
Distribución sensores						X								
Programación de los controladores						X	X							
Primeras pruebas físicas							X							
Armado del módulo en físico							X	X						
Ensamble completo del módulo y controladores								X						
Armado de planta de nivel								X	X					
Armado de planta de domótica								X	X					
Ensamble completo total										X	X			
Pruebas finales											X			
Armado de las guías de prácticas											X	X		
Presentación borrador												X		
Levantamiento observaciones												X	X	
Sustentación Tesis													X	

Tabla 1.1 Diagrama de Gantt

CAPITULO II

PLANTEAMIENTO TEORICO

2.1 Redes de computadoras

Existen 4 tipos principales de redes, las redes PAN, CAN, LAN Y WAN.

2.1.1 PAN (red de administración personal)

Son redes pequeñas, que están formadas por no más de 8 equipos.

2.1.2 LAN (red de área local)

Es la más conocida, es una red pequeña dentro de un área de una oficina, un edificio, etc. Son redes muy rápidas con una velocidad entre 10 y 100 Mbps y normalmente se utiliza un cable sencillo (coaxial o UTP) al que están conectadas todas las computadoras.

Se caracteriza por que sus canales son propios de una entidad o empresa, los enlaces y conexiones son de alta velocidad, las estaciones de trabajo están cerca unas de otras, para el trabajo de oficina es más eficiente y productiva una red de este tipo porque agiliza la administración. Su organización permite compartir datos, imágenes, variables, etc. Y tiene una tasa de error mucho menor que redes de mayor tamaño como las redes WAN.

2.1.3 CAN (red de área de campus)

Es la reunión de varias LAN que están en diferentes lugares geográficamente unidos en un mismo campus bajo la propiedad de una misma entidad, como universidades, empresas de trabajo, etc.

2.1.4 WAN (red de área extensa)

Son redes que interconectan puntos en diferentes lugares del mundo, países y continentes. Así como este tipo de red cubre una mayor distancia que los demás tipos de red, también su velocidad de transmisión es menor que las demás.

Está formada por una gran cantidad de computadoras interconectadas (hosts), por medio de subredes más pequeñas para poder ejecutar programas, aplicaciones, enviar o recibir resultados, etc.

Las redes LAN generalmente se conectan a las redes WAN, para poder obtener mejores y más variados servicios, como conectarse a Internet. Estas redes son más complejas puesto que deben poder enrutar la información que se requiera pero de todas las redes conectadas a ésta.

2.1.5 MAN (red de área metropolitana)

Son las redes que tan solo pueden estar en una determinada área geográfica, máxima de 4 Km. Son redes con dos buses unidireccionales, eso quiere decir que solo tiene una dirección sea de envío o recibo de datos, es muy similar a las redes LAN, pero con la diferencia que usan DQDB para la resolución de conflictos.

DQDB contiene 2 buses unidireccionales, que enlaza a todas las estaciones, cada bus tiene una cabecera y un fin. Cuando llega el momento de transmitir, una computadora que quiera enviar información, si está ubicada a la izquierda usa el bus superior, de estar a la derecha usa el bus inferior.

2.1.5.1 DQDB (Dual-Queue Dual-Bus)

En castellano significa Bus Dual de Cola Distribuida, es un sistema de conexión para una red multi-acceso, utilizando un bus dual y teniendo la información de manera organizada en una cola distribuida.

Cuenta con dos buses y una estación generadora de tramas o paquetes de datos al inicio y al final de ambos buses, una conectada al Bus superior A y una conectada al Bus inferior B. Y en cualquier momento del recorrido pueden recibir y enviar datos de cualquier equipo que esté conectada a esta.



TIPOS DE REDES

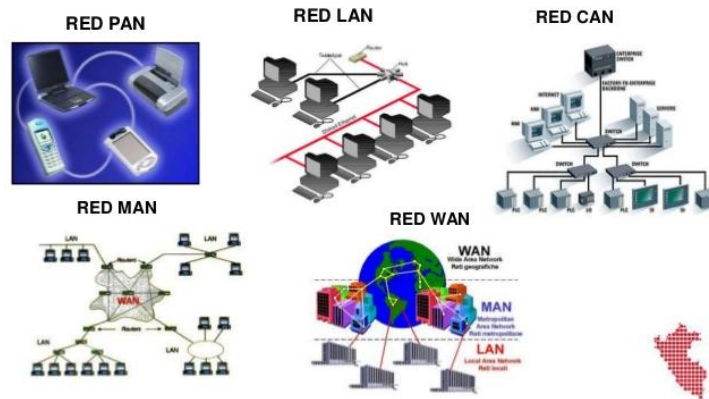


Fig. 2.1 Tipos de redes

Fuente: <http://image.slidesharecdn.com/conectividaddelaredlanenlasaipycrtdelasii-150614004211-lva1-app6892/95/conectividad-de-la-red-lan-4-638.jpg?cb=1434245612>

2.1.6 ETHERNET

Ethernet es una familia de las tecnologías de redes computacionales comúnmente usadas en redes de área local (LAN). Se introdujo al mercado en 1980 y fue estandarizada por primera vez en 1983 como el estándar IEEE 802.3 y ha sido mejorada para soportar un mayor rango de bits y de distancias desde entonces. Con el tiempo, Ethernet ha reemplazado en su mayoría a sus competencias en tecnologías LAN cableadas.

El Ethernet 10BASE5 original usa cable coaxial pero las últimas variantes Ethernet usan par trenzado o fibra óptica como enlaces en conjunto con hubs o switches. Con el curso de la historia, las tasas de transferencia de datos de Ethernet se han incrementado desde 2.94 megabits por segundo (Mbps) hasta 100 gigabits por segundo (Gbps) y se espera que llegue a 400 Gbps en 2017.

Su principio de funcionamiento es que todos los equipos están conectados a una misma línea de transmisión y la comunicación se lleva a cabo por medio de un protocolo llamado “Carrier Sense Multiple Acces with Collision Detect” (CSMA/CD). Este es un protocolo de acceso múltiple que monitorea la portadora: detección de colisiones y detección de portadora.

Con esto, cualquier equipo está autorizado a transmitir en la línea cuando quiera y sin prioridad alguna entre ellos. Esto se da de manera simple:

- Se verifica que la línea de comunicación no esté en uso antes de realizarse la transmisión.
- Si la línea es usada por dos equipos al mismo tiempo, se produce una colisión de datos.
- Se interrumpe cualquier comunicación durante un tiempo cualquiera, después una vez que el primero ha excedido el periodo de tiempo, puede volver a transmitir.

Por estos motivos se eligió la red Ethernet en nuestro módulo

Abreviatura	Nombre	Cable	Conector	Velocidad	Puertos
10Base2	Ethernet delgado (Thin Ethernet)	Cable coaxial (50 Ohms) de diámetro delgado	BNC	10 Mb/s	185 m
10Base5	Ethernet grueso (Thick Ethernet)	Cable coaxial de diámetro ancho (10,16 mm)	BNC	10Mb/s	500 m
10Base-T	Ethernet estándar	Par trenzado (categoría 3)	RJ-45	10 Mb/s	100 m
100Base-TX	Ethernet veloz (Fast Ethernet)	Doble par trenzado (categoría 5)	RJ-45	100 Mb/s	100 m
100Base-FX	Ethernet veloz (Fast Ethernet)	Fibra óptica multimodo (tipo 62,5/125)		100 Mb/s	2 km
1000Base-T	Ethernet Gigabit	Doble par trenzado (categoría 5)	RJ-45	1000 Mb/s	100 m
1000Base-LX	Ethernet Gigabit	Fibra óptica monomodo o multimodo		1000 Mb/s	550 m
1000Base-SX	Ethernet Gigabit	Fibra óptica multimodo		1000 Mbit/s	550 m
10GBase-SR	Ethernet de 10 Gigabits	Fibra óptica multimodo		10 Gbit/s	500 m
10GBase-LX4	Ethernet de 10 Gigabits	Fibra óptica multimodo		10 Gbit/s	500 m

Tabla 2.1 Comparativa de cables Ethernet

Fuente: <http://es.ccm.net/contents/672-ethernet>

2.1.6.1 IP

Una dirección IP es un número que se le da a un dispositivo dentro de una red para poder identificarlo y comunicarse con él. Este número es lógico y jerárquico y se rige mediante el Protocolo de internet (IP) que corresponde al nivel de red del modelo TCP/IP.

Este número puede ser estático, como el de muchas páginas web, o puede ser dinámico y por lo tanto se modificaría según la conveniencia de cada usuario en sus redes privadas.

No debe confundirse este número con el número MAC, este último es un número único global de las tarjetas de red de diferentes dispositivos, mientras que el número IP funciona solo dentro de una red específica, por lo tanto puede haber varios números IP dentro de diferentes redes, pero solo un número MAC en cualquier red.

2.1.6.2 Mascara de Subred

Es una máscara local de bits que especifica qué bits de la dirección IP especifican una red IP determinada o un host dentro de una subred. Se utiliza para “enmascarar” una porción de una dirección IP de modo que el TCP/IP pueda determinar si cualquier dirección IP está en una red local o remota. Cada equipo configurado con el TCP/IP debe tener una máscara de subred definida.

Un valor que permite que una red sea subdividida y proporciona asignaciones de direcciones más complejas. El formato de la máscara de subred es nnn.nnn.nnn.nnn, por ejemplo, 255.255.255.0.

2.2 PLC

Un PLC o controlador lógico programable es una computadora digital usada para la automatización de procesos electromecánicos típicos como el control de maquinaria en una línea

de producción industrial, juegos mecánicos en parques de diversiones, etc. Los PLC son usados por muchas maquinas en muchas industrias. Los PLC's son diseñados con múltiples arreglos de entradas y salidas digitales y analógicas, rangos de temperatura extendidos, inmunidad al ruido eléctrico y resistencia a la vibración e impacto. Usualmente los programas para el control de las operaciones de equipos están guardados en memorias no volátiles o en memorias con batería de reserva.

Antes de los PLC, el control, secuencia y los enclavamientos de seguridad para la fabricación de automóviles estaban principalmente constituidos por relés, temporizadores de levas, cadenas de rodillos y controladores de lazo cerrado dedicados. La cantidad de estos dispositivos podían llegar a ser cientos o miles, por lo tanto, para actualizar estos sistemas a los últimos modelos requeridos se necesitaba emplear mucho tiempo y dinero debido a que los electricistas tenían que volver a cablear todos los relés individualmente para cambiar sus características operacionales.

Las computadoras digitales, siendo dispositivos programables para propósitos generales, fueron aplicadas con prontitud a los procesos de control de las industrias. Las primeras computadoras requerían a programadores especialistas y un control estricto del entorno operativo para la temperatura, limpieza y calidad de energía. Usar una computadora de propósitos generales requería protegerla de las condiciones de la planta. Una computadora industrial de control tendría bastantes atributos: toleraría el ambiente de trabajo, soportaría entradas y salidas discretas (bits) de una manera fácilmente ampliable, no requeriría años de capacitación para ser usada y permitiría que su operación sea monitoreada. El tiempo de respuesta de cualquier sistema de computadora debe ser lo suficientemente rápido para que el control sea útil, la velocidad requerida varía de acuerdo a la naturaleza del proceso.

En 1968 GM Hydra-Matic (la división de transmisión automática de General Motors) publicó una solicitud de propuestas para un reemplazo electrónico para los sistemas de relés conectados por cables basados en un documento técnico escrito por el ingeniero Edward R. Clark. La propuesta ganadora provino de Bedford Associates desde Bedford, Massachusetts. El primer PLC, designado el 084 porque fue el proyecto ochenta y cuatro de Bedford Associates, fue el resultado. Bedford Associates empezó una nueva compañía dedicada al desarrollo, fabricación, venta y mantenimiento de este nuevo producto: Modicon, iniciales para Modular Digital Controller. Una de las personas que trabajó en el proyecto fue Dick Morley, quien es considerado el “padre” del PLC. La marca Modicon fue vendida en 1977 a Gould Electronics, después fue adquirida por una compañía alemana AEG y luego por la compañía Francesa Schneider Electric, la dueña actual.



Fig. 2.2 Gráfica tentativa del PLC

Fuente: <http://www2.schneider-electric.com/images/pictures/customers/satisfy-our-customers/m340-unitypro.jpg>

2.2.1 Programación

Los primeros PLC's, a mediados de los 90's, fueron programados usando paneles de programación propios o terminales de programación de propósito especial, que muchas veces tenían teclas de función dedicadas representando los elementos de los programas del PLC. Algunas terminales propias de programación mostraban los elementos de los programas del PLC como símbolos gráficos, pero eran comunes las representaciones simples con caracteres ASCII de contactos, bobinas y cables. Los programas eran guardados en carretes de cinta magnética o cinta de cassette. Los mecanismos para impresión y documentación eran mínimos debido a la falta de capacidad de memoria. Los PLC's más antiguos usaban una memoria de núcleo magnético no volátil.

Recientemente, los PLC's son programados usando software aplicativo en computadoras personales, que ahora representan la lógica en gráficos en vez de símbolos ASCII. La computadora es conectada al PLC mediante Ethernet, RS-232, RS-485 o RS-422. El software de programación permite entrar y editar la lógica estilo Ladder. Generalmente el software provee funciones para la depuración del sistema y localización de problemas del software del PLC, por ejemplo, resaltando porciones de la lógica para mostrar el estado actual durante la operación o vía simulación. El software cargara y descargara el programa del PLC, para fines de restaurar y tener copias de respaldo (backup).



Fig. 2.3 PLC y Código de Ladder

Fuente: <http://www.logismarket.es/ip/gdo-electric-programacion-plc-y-hmi-programacion-de-plc-y-hmi-800497-FGR.jpg>

2.2.1.1 Lista de Instrucciones (IL)

Una lista de instrucciones consta de una serie de instrucciones. Cada instrucción empieza en una nueva línea de programación, y contiene un operador y según sea la forma de operación, uno o más operandos separados por una coma.

En una instrucción se puede encontrar una marca de identificador seguida de un doble punto.

Se puede añadir un comentario como último elemento de una línea de instrucción.

También se pueden insertar líneas vacías entre instrucciones sin ninguna dificultad.

Ejemplo de programa en lista de instrucciones:

LD 17

ST int (*Comentario*)

GE 5

JMPC next

LD idword

EQ instruct.sdword

STN test

NEXT;

2.2.1.2 Plano de Contactos (LD)

El plano de contactos es un lenguaje de programación gráfica, se basa en el principio aplicado a los esquemas eléctricos.

-Contactos:

El plano de contactos consta de una serie de redes. Cada red consta en el lado izquierdo a su vez de una red de contactos (que van dentro de dos líneas paralelas), que de izquierda a derecha ofrecen estados “conectado” o “desconectado”.

Estos estados corresponden a los estaos lógicos VERDADERO y FALSO. A cada contacto corresponde una variable booleana (Por ejemplo A, B, C y Salida). Cuando esta variable es verdadera, entonces el estado continúa de izquierda a derecha sobre la línea de conexión, de lo contrario se mantiene la conexión derecha en estado “desconectado”.

Los contactos pueden estar paralelos o enseriados e inclusive una combinación serie – paralelo o paralelo – serie tal como en los circuitos eléctricos.

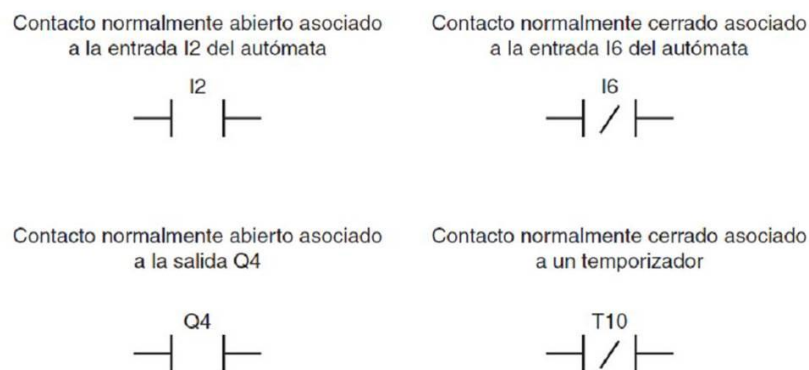


Fig. 2.4 Tipos de Contactos en Ladder

Fuente: <https://automatismoindustrial.com/d-9-4-lenguaje-de-contactos/>

-Bobinas:

En el lado derecho de una red se encuentran las denominadas bobinas. Estas se pueden conectar solamente de forma horizontal. Una bobina da el valor de la conexión de izquierda a derecha y lo copia en la variable booleana correspondiente. Los contactos y las bobinas también pueden ser negados.

Cuando una bobina esta negada, entonces se copia el valor negado de la variable booleana correspondiente Cuando el contacto esta negado entonces solo conecta cuando la variable correspondiente es falsa.



Fig. 2.5 Tipos de Bobinas en Ladder

Fuente: <https://automatismoindustrial.com/d-9-4-lenguaje-de-contactos/>

2.2.1.3 Diagrama secuencial o Grafcet (SFC)

El lenguaje secuencial o cuadro de funciones secuenciales (Secuencial Function Chart) es un lenguaje gráfico orientado, este posibilita el desarrollo secuencial de diferentes acciones en función del tiempo o condiciones específicas.

-Pasos, Acciones:

Los bloques utilizados en el lenguaje secuencial consisten de una serie de pasos unidos entre sí. Cada paso tiene un programa que asigna una acción específica.

-Transición:

Entre cada paso se encuentra las transiciones que determinan las condiciones del programa. Cada condición está asignada a una variable lógica.

2.2.2 Señales analógicas y discretas

Las señales discretas se comportan como interruptores binarios, produciendo simplemente una señal ON/OFF (1/0, Verdadero/Falso). Pulsadores, sensores fotoeléctricos y finales de carrera son ejemplos de dispositivos que envían señales discretas.

Las señales discretas puede ser enviadas usando corriente o voltaje, donde un rango específico es designado como ON otro como OFF. Por ejemplo, un PLC puede usar 24 V DC E/S, con valores sobre los 22V DC representando ON, valores debajo de 2 V DC representando OFF y valores intermedios indefinidos. Inicialmente, los PLC solo tenían E/S discretas.

Las señales analógicas son como controles de volumen, con un rango de valores entre cero y escala llena. Estos son típicamente interpretados como valores enteros (contables) por el PLC, con varios rangos de precisión dependiendo del dispositivo y el número de bits disponibles para almacenar los datos. Como los PLC típicamente usan procesadores binarios de 16 bits, los valores enteros están limitados entre -32 768 y +32 767.

Presión, temperatura, flujo y peso son muchas veces representados por señales analógicas. Las señales analógicas pueden usar voltaje o corriente con magnitudes proporcionales al valor de la señal de proceso. Por ejemplo, una entrada analógica entre 0 a 10 V o 4-20 mA puede convertirse en un valor entero entre 0 y 32767.

Las entradas de corriente son menos sensibles al ruido eléctrico comparadas con las de voltaje.

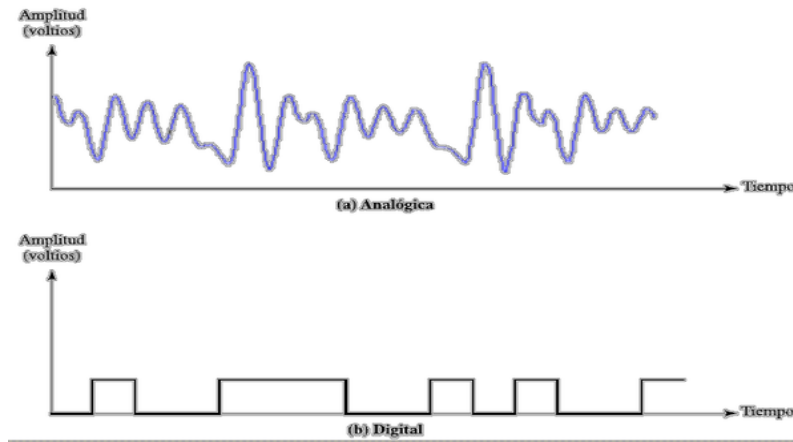


Fig. 2.6 Señales Analógicas y Digitales

Fuente: http://upload.wikimedia.org/wikipedia/commons/thumb/f/fe/Se%C3%B1ales_analogica-digital.png/600px-Se%C3%B1ales_analogica-digital.png

2.2.3 SoMachine Basic

So Machine Basic es un software de Schneider Electric que nos permite programar en los lenguajes de Lista de Instrucciones (IL), Diagrama de contactos (LD) y Grafset (SFC). Este es un software de licencia libre y solo es compatible con la gama de PLC's Modicon M221. Tiene la opción de configurarse en 8 idiomas incluidos el inglés y el español. Este software se puede descargar de la página de Schneider Electric.

2.3 HMI

HMI o interfaz humano-máquina es la solución hecha por los humanos para poder comunicarse con las máquinas.

Entonces es el medio por el cual una persona puede resolver, recibir y enviar datos, de una máquina, equipo, dispositivo u ordenador, y que comprende todas las maneras en las cuales podría haber contacto físico entre el usuario y el equipo.



Fig. 2.7 Grafico estimado de una pantalla de un terminal HMI

Fuente: <http://www.aispro.com/Media/Default/images/content/Stainless%20Steel%20HMI%20Panel%20PC%20Control%20and%20Monitoring%20System.jpg>

Su proceso de control, recepción y envío de información, se divide en 3 sub-procesos:

2.3.1 Interfaz de Hardware

Es el nivel en donde se encuentran los dispositivos utilizados para procesar, entregar e ingresar datos desde el exterior.

2.3.2 Interfaz de Software

Es el nivel donde se entrega la información de los procesos de control, generalmente es lo que se observa en la pantalla.

2.3.3 Interfaz de Software-Hardware

Es el nivel que realiza la comunicación entre la máquina y la persona, hace que la máquina entienda lo que se pidió hacer y al humano entender lo que las señales analógicas y digitales representan en el proceso de forma gráfica y/o legible.

2.4 Sensores y Actuadores

2.4.1 Sensores

Un sensor es un dispositivo que toma una magnitud del exterior y la convierte en otra magnitud, generalmente es eléctrica, para que podamos comprender lo que esta señal exterior representa numéricamente en nuestro proceso.



Fig. 2.8 Tipos de Sensores

Fuente: <http://www.lostipos.com/wp-content/uploads/Tipos-de-sensores.jpg>

Normalmente los sensores son fabricados por componentes pasivos, son llamados así puesto que varían su magnitud en función de alguna variable externa. Y una minoría de componentes activos, llamados así porque pueden modificar las magnitudes que ingresan en ellos sin necesidad de que alguien los maniobre o manipule.

Algunos tipos de sensores son:

- Sensor de posición lineal.
- Sensor de posición angular.
- Sensor de desplazamiento.

- Sensor de deformación por fuerza.
- Sensor de deformación por par de torsión.
- Sensor de velocidad lineal.
- Sensor de velocidad angular.
- Sensor de aceleración.
- Sensor de presión.
- Sensor de caudal.
- Sensor de temperatura.
- Sensor de presencia.
- Sensores táctiles.
- Sensor de visión artificial.
- Sensor de proximidad.
- Sensor acústico.
- Sensor de acidez.
- Sensor de gases y humo.
- Sensor de luz.
- Sensor de captura de movimiento.

Tienen las siguientes características:

2.4.1.1 Linealidad y Correlación lineal

Se puede entender fácil con la relación de a mayor magnitud física externa, mayor magnitud de respuesta del sensor y para los sensores todo o nada, si es sí es sí, si es no es no, siempre si es sí será sí y si es no será no.

2.4.1.2 Rango de Medida

Es el dominio en la magnitud de medición que se aplica sobre el sensor.

2.4.1.3 Precisión

Es el error de una medición en un lapso de tiempo.

2.4.1.4 Offset o Desviación de cero

Es el valor que nos arroja cuando la variable de entrada es nula.

2.4.1.5 Resolución

Es la variación más pequeña que puede detectar el sensor de la señal de entrada.

2.4.1.6 Rapidez de Respuesta

Es la velocidad en la cual el sensor analiza, recibe y envía la señal de salida desde el momento que se propone la señal de entrada

2.4.1.7 Sensibilidad

Es el mínimo valor que se requiere en la entrada para que reconozca y pueda enviar un valor de salida

2.4.1.8 Repetitividad

Es el error que da al repetir muchas veces la misma medición.

2.4.2 Actuadores

Un actuador es un tipo de motor que es responsable de mover o controlar un mecanismo o un sistema.



Fig. 2.9 Tipos de Actuadores

Fuente: <http://ahorroenergi.files.wordpress.com/2012/05/actuadores-neumaticos.jpg>

Es operado por alguna fuente de energía, típicamente corriente eléctrica, presión hidráulica o presión neumática y convierte esta energía en movimiento. Un actuador es el mecanismo que controla las acciones de un sistema sobre un ambiente. El sistema de control puede ser simple (un arreglo mecánico o sistema electrónico), basado en software (por ejemplo un controlador de impresora, un sistema de control robótico), un humano, o cualquier otra entrada.

2.4.2.1 Hidráulicos

Estos actuadores consisten en cilindros o motores de fluido que usan la energía hidráulica para lograr la operación mecánica. Debido a que es casi imposible comprimir los líquidos, los actuadores hidráulicos puede lograr grandes fuerzas.

2.4.2.2 Neumáticos

Estos actuadores convierten la energía formada por aire comprimido a altas presiones en movimientos lineales o rotacionales. Estos permites grandes fuerzas producidas en base a pequeños cambios en la presión del aire.

2.4.2.3 Eléctricos

Estos actuadores convierten la energía eléctrica en movimiento. Esta es una de las formas más limpias y de fácil acceso porque no involucra aceite.

2.4.2.4 Mecánicos

Estos actuadores convierten movimiento rotacional en movimiento lineal o viceversa. Pueden incluir en su diseño engranajes, poleas, rieles, cadenas u otros dispositivos.

2.4.2.5 Válvula Solenoide

Es una válvula de gas o fluidos que se abre al recibir una señal de voltaje específica en un electroimán también conocido como solenoide. Al pasar una corriente eléctrica a través de la bobina, se genera un campo electromagnético de cierta intensidad en el interior. Un émbolo fabricado de metal ferroso es atraído por la fuerza magnética hacia el centro de la bobina, lo que proporciona el movimiento necesario para accionar la válvula

Las señales más comunes para activar estas válvulas son 5 VDC, 12 VDC o 24 VDC

2.4.2.6 Servomotor

Un servomotor es un motor eléctrico con la diferencia de que al servomotor se le puede controlar fácilmente la posición sin necesidad de un sensor como un encoder o un resolver que son frecuentemente usados en los motores eléctricos.

La desventaja del servomotor es que tiene un rango limitado de movimiento, que suele ser 180° o 360°.

Su posición es controlada por una modulación de ancho de pulsos o también conocida como PWM.

Estos motores suelen consumir bastante corriente y tienen mayor fuerza comparados con los motores comunes y corrientes.

2.4.2.7 Bomba Hidráulica

La bomba hidráulica es una máquina que tiene como función proporcionar movimiento a un fluido líquido o una mezcla de líquidos y sólidos pero que cumplan ciertas características de fluidos.

Suelen convertir energía eléctrica en energía cinética mediante un motor eléctrico y luego se transforma la energía cinética en energía mecánica o presión para mover un fluido de diferentes formas según sea el tipo de bomba. Pero también existen bombas que no usan energía eléctrica, sino otros tipos que mencionaremos más adelante.

Las bombas se clasifican según su principio de funcionamiento en:

- Bombas de desplazamiento positivo: en las que el principio de funcionamiento está basado en la hidrostática. Y estas a su vez se clasifican en:
 - Bombas de émbolo alternativo
 - Bombas volumétricas rotativas
- Bomba roto dinámica: en las que el principio de funcionamiento está basado en el intercambio de cantidad de movimiento entre la máquina y el fluido, aplicando la hidrodinámica. Y estas a su vez se clasifican en:
 - Radiales o centrífugas
 - Axiales
 - Diagonales o helicentrífugas

Y según el tipo de accionamiento en:

- Electrobombas
- Bombas neumáticas
- Bombas de accionamiento hidráulico

2.5 Sistemas de control

Un sistema de control es un conjunto de diversos dispositivos como sensores, actuadores, controladores, etc. que se dedican a dirigir, regular, administrar y ordenar el modo de funcionamiento de otros dispositivos o sistemas.

El término en sí de “sistema de control” se refiere a sistemas manuales en los cuales un operador de, por ejemplo, abrir o cerrar una válvula, encender o apagar una máquina teniendo en cuenta también que se realice en un entorno seguro.

Un control automático secuencial es aquel que tiene sensores y actuadores regidos por una secuencia específica que no necesariamente tiene que tener un dispositivo específico, por ejemplo los sistemas neumáticos o electro neumáticos. Pero es común encontrar secuencias dadas por controladores lógicos programables.

Como sistemas de control existen los sistemas de lazos abiertos y lazos cerrados.

2.5.1 Sistema de control de lazo abierto

Los sistemas de control de lazo abierto son aquellos en los que la variable de salida no influye en la variable que se controla, o sea no se tiene una referencia, por lo tanto cada valor de la entrada tendrá su propio valor de salida.

En este tipo de sistemas es muy importante la calibración de los controladores y tratar de evitar las perturbaciones, porque estas afectan considerablemente la variable de salida.

Un ejemplo de un sistema de control de lazo cerrado sería un semáforo común y corriente, la variante de entrada es el tiempo que será asignada a cada luz, el tiempo siempre será el mismo sin importar el tráfico de las calles.

2.5.2 Sistema de control de lazo cerrado

Los sistemas de control de lazo cerrado son aquellos en los que la variable de salida si tiene un efecto directo sobre la variable que se controla, o sea se tiene una referencia o una retroalimentación. Este sistema está mejor preparado en contra de las perturbaciones, no es tan necesaria la calibración del controlador, ya que la retroalimentación permite acomodar la variable que se controla. Un ejemplo de un sistema de control de lazo abierto sería el sistema de un refrigerador, el cual se le exige una temperatura específica que sería la variable de salida y cuando nosotros ingresamos un alimento caliente, se detecta y el refrigerador empieza a funcionar con tal de volver a la temperatura exigida por el usuario.

2.5.3 Sistema de control distribuido

Un sistema de control distribuido es un sistema de control en el cual todos los elementos de control están distribuidos por toda la planta a diferencia de los sistemas no distribuidos que tienen un solo elemento central de control. En el sistema de control distribuido también existe una jerarquía de controladores conectados por distintos niveles de redes de comunicación para efectuar comandos y monitoreo. Este sistema consta de muchos elementos, varios de ellos controladores, sensores y actuadores, pero a diferencia de otros sistemas también consta de elementos de comunicación.

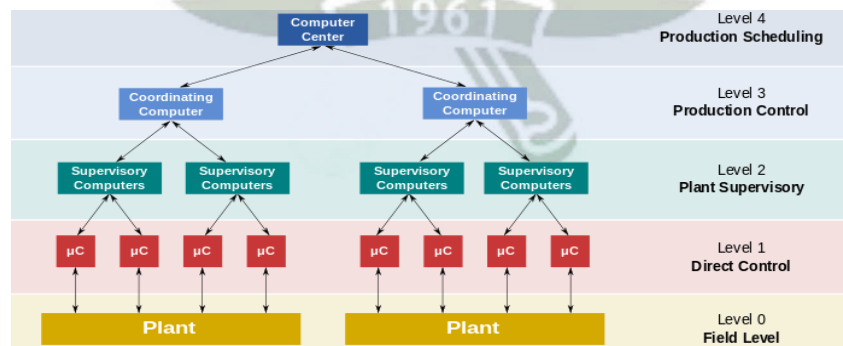


Fig. 2.10 Niveles de funcionamiento de un sistema de control distribuido típico

Fuente: <http://wikipedia.org>

CAPITULO III

DISEÑO Y CONSTRUCCIÓN DE LA SOLUCIÓN

3.1 Diseño del modulo

Hicimos la primera posible elección de los controladores lógicos que usaremos, luego que hicimos una revisión de nuestras posibilidades en distintas marcas, decidimos elegir Schneider- Electric que teníamos buenas referencias, luego se explicará el criterio de elección, ésta marca cumplía con todas las características que necesitábamos, y ésta fue la primera decisión:

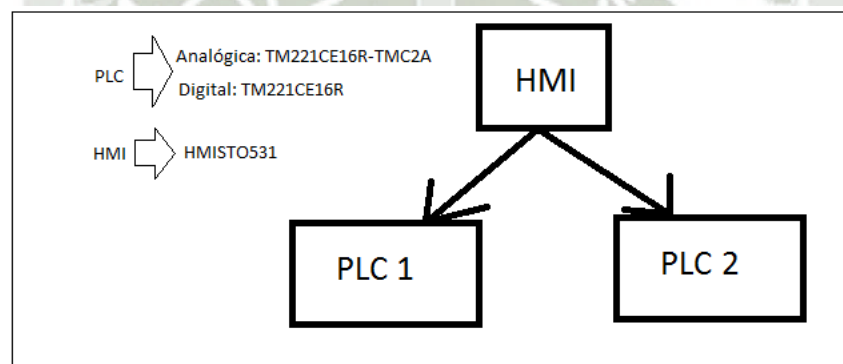


Fig. 3.1 Elección de los PLC's y el HMI

Fuente: Tesis: Diseño e Implementación de un módulo didáctico de un sistema de control distribuido con un terminal de dialogo HMI como maestro y dos controladores lógicos como esclavos comunicados a través de una red Ethernet para el laboratorio de control y automatización – UCSM de Arequipa en el año 2016.

Autores: Flores Arana, Jesús Eduardo y Flores Juárez, Patricio José.

Luego de esto, decidimos empezar a armar nuestro modulo con los submodulos que decidimos, el submodulo de nivel de tanque y el submodulo de control de vivienda. Resumimos entonces las posibilidades que teníamos en mente en un listado de papel.

-Modulo Central:

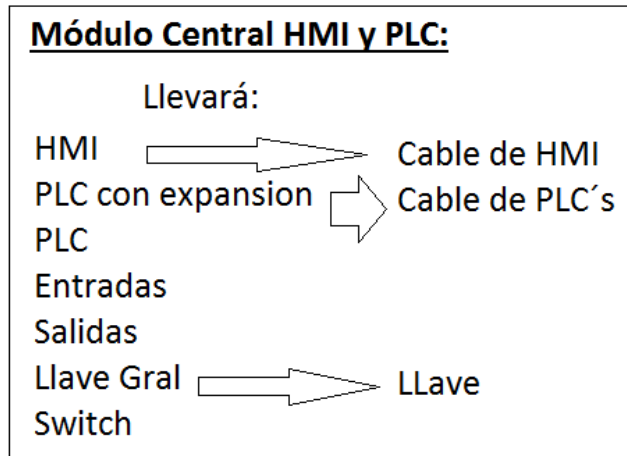


Fig. 3.2 Resumen de modulo principal

Fuente: Propia

-Submodulo de Control de un Tanque de Agua:

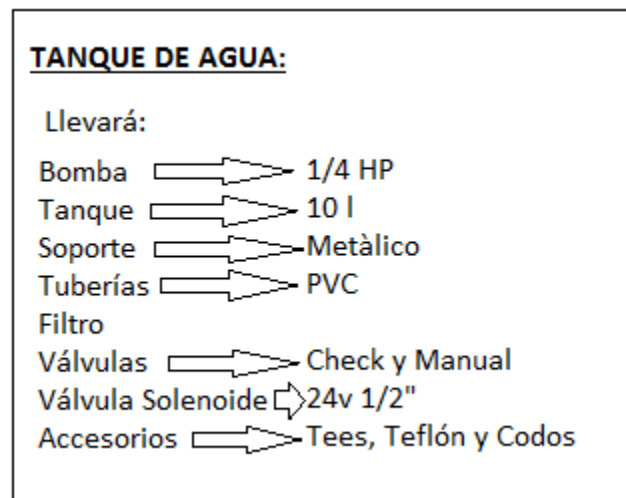


Fig. 3.3 Resumen de sub módulo de nivel de tanque

Fuente: Propia

-Submodulo de Control de una vivienda:

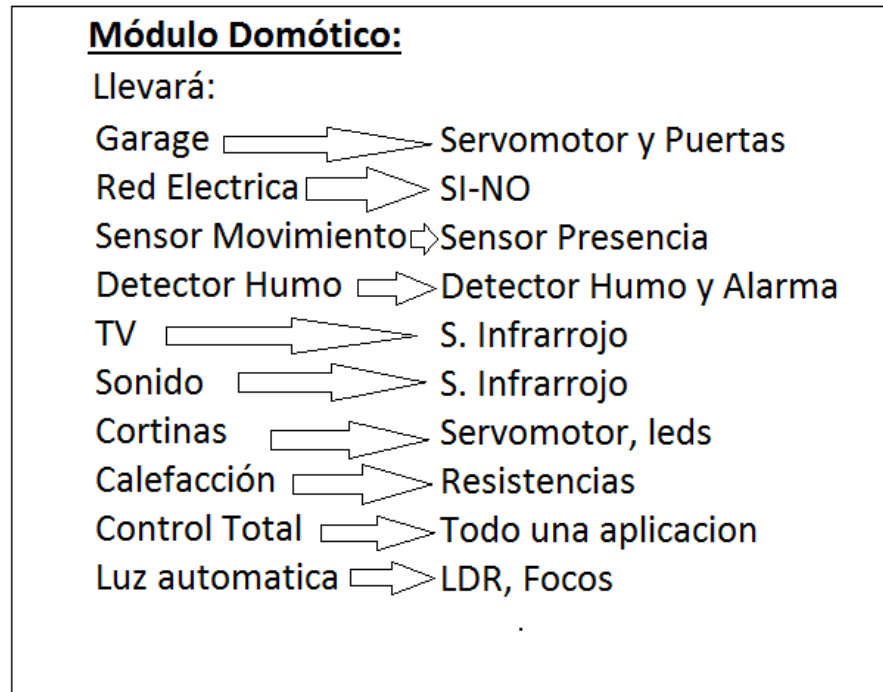


Fig. 3.4 Resumen de sub módulo de control de vivienda

Fuente: Propia

Elegimos aquí lo que controlaríamos y lo que no controlaríamos, en base a nuestras posibilidades, como se vio en la imagen estudiamos las mejores posibilidades para el control de vivienda y comenzamos por allí, decidimos que controlaríamos lo siguiente:

- Sensor de movimiento.
- Sensor de humo.
- Sistema de calefaccion.
- Apertura de puerta de garage.
- Iluminacion automatica.
- Control total de sistema electrico.

Pero cotizamos los sensores e instrumentos necesarios para esto y decidimos hacerlos nosotros mismos para saber exactamente que es lo que vamos a controlar, y así fue como realizamos los sensores necesarios para la aplicación requerida en cada espacio de nuestro submodulo de vivienda.

Circuitos en primer boceto de:

-Luz de Noche:

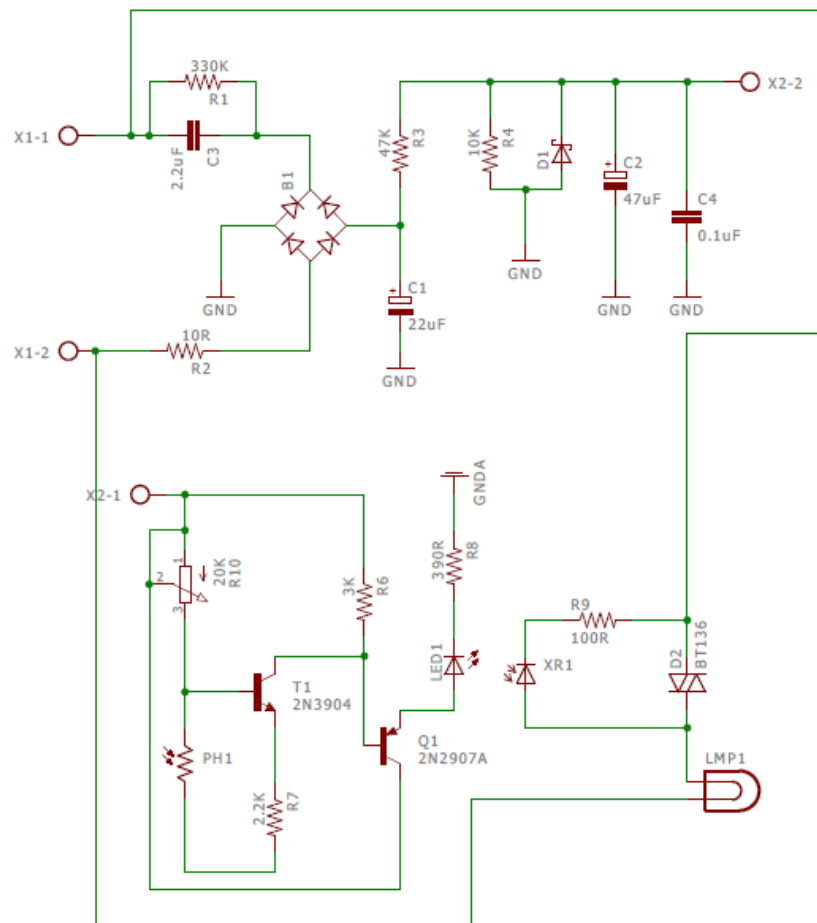


Fig. 3.5 Circuito para luz automática de noche con el TRIAC BT136

Fuente: Propia

Este circuito activa una luz en base a la variación del LDR, a menor luminosidad, menor luz emite el foco o led, pero está integrado ya con un circuito de potencia. En este circuito

nuestros transistores deben entrar en estado de saturación para encender la lámpara LMP1, para que esto suceda nuestras corrientes de base deben ser:

$$I_B > \frac{I_{Csat}}{\beta}$$

Dónde:

I_B = Corriente de base

I_{Csat} = Corriente de Colector de Saturación

β = Beta del transistor

Para poder obtener esta corriente de base es necesario calcular las resistencias que vamos a colocar en nuestro circuito, esto lo hacemos despejando la corriente de colector de saturación

$$I_{Csat} = \frac{V_{cc}}{R_C + R_E}$$

Dónde:

V_{cc} = Voltaje de alimentación

R_C = Resistencia de colector

R_E = Resistencia de emisor

Reemplazando los valores con los datos de los transistores reales pudimos hallar que:

$$R_C = 3k$$

$$R_E = 2.2k$$

Estos valores fueron acomodados a valores comerciales de resistencias.

-DETECTOR DE HUMO:

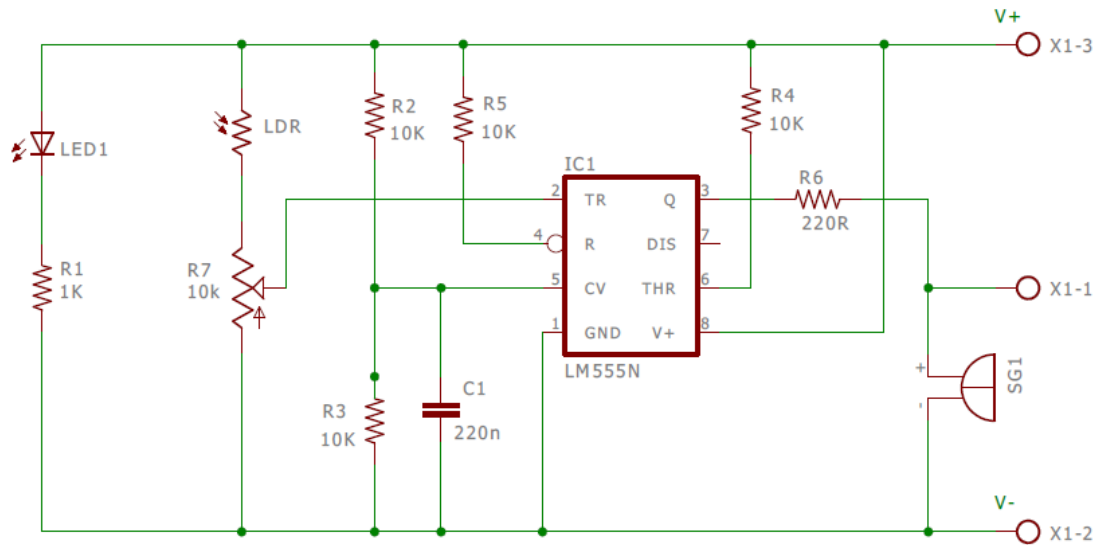


Fig. 3.6 Alternativa para el circuito de detección de humo con el 555

Fuente: Propia

Estas 3 variedades de circuito son en si el mismo, pero con unas pequeñas variaciones para activar un relay, una bocina o una lámpara. Pero lo que usa es un circuito básico de barrera pero con un LDR, que lo que hace es detectar la luz de un led que le apunta directamente, cuando este haz de luz se ve modificado por humo o cualquier cosa que impida que se vea con claridad la luz, activa una señal que activa un ventilador puesto estratégicamente para que disipe el humo hasta que la luz se vea de nuevo con claridad.

Para este circuito decidimos usar un NE555 como comparador con memoria, por esto es que como entrada tenemos al pin 2 (Disparo) y al pin 5 (Control de voltaje).

En estado de reposo, mientras reciba luz el sensor de luz LDR, su resistencia será baja, por lo tanto estará activado el Reset del flip flop interno del NE555. Pero al momento de cortar la luz, la resistencia del LDR baja, lo que permite que el voltaje de entrada por el pin 2 (Disparo)

sea mayor y active el Set del flip flop interno del NE555 y así pueda enviar por la salida una señal que según la hoja de datos del circuito integrado es aproximadamente:

$$OUT = \frac{1}{3}V_{cc}$$

Y permanecerá así hasta que la luz vuelva a llegar, suba la resistencia y se envíe la señal por el pin 5 que resetea la memoria flip flop.

Entonces colocamos todas resistencias genérica de 10k para bajar la corriente y evitar daños en nuestro circuito integrado.

A la salida se colocó una resistencia de 220 ohm para bajar menos la corriente debido a que el zumbador requiere cierta alimentación.

-CONTROL REMOTO:

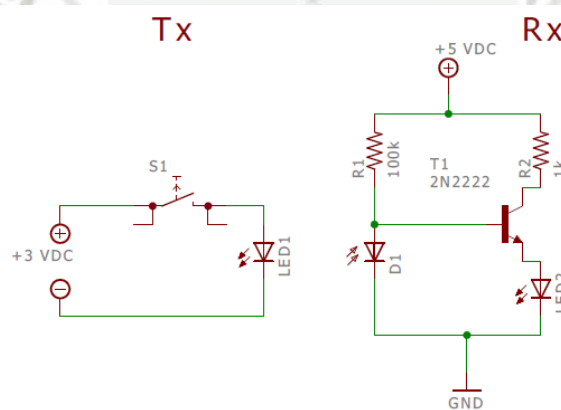


Fig. 3.7 Circuitos para hacer un control remoto con leds infrarrojo

Fuente: Propia

Este circuito sirve para recibir un bit desde el control por decirlo así, al pulsar un botón, cierra circuito y recibimos una señal en el circuito de respuesta a través de los leds infrarrojos.

-ENCENDER VENTILADOR:

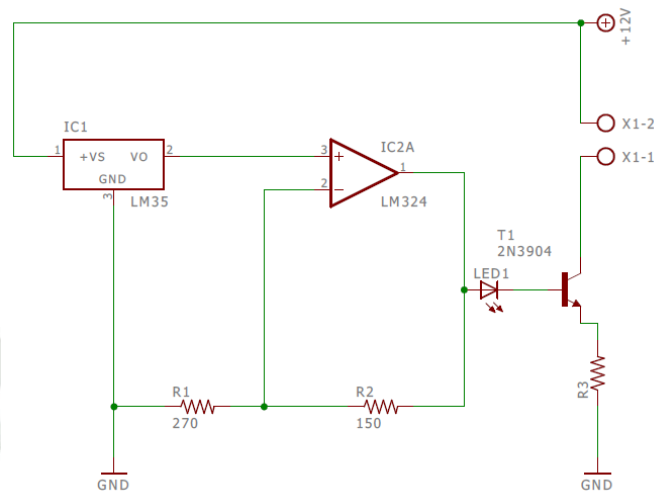


Fig. 3.8 Circuito para encender un ventilador con el LM324

Fuente: Propia

Con este circuito pudimos encender el ventilador sin necesidad de un pulsador, un relay o un contacto. Amplificando valores del LM35 a través del LM324, y luego a través del transistor consiguiendo el valor de voltaje necesario para encender el ventilador y ponerlo en marcha.

En este circuito, los componentes principales son el sensor LM35 que a medida que aumenta su temperatura, también aumenta su voltaje de salida. Y este voltaje luego es comparado con un LM324 al cual le colocamos resistencias solo para bajar la corriente debido al led, pero que no es necesario realizar un gran cálculo.

Cuando la temperatura llega a cierto nivel, el voltaje es suficiente para permitir que conduzca el comparado y este luego enciende un led y llega corriente al Transistor T1, que entra en estado de saturación y permite el paso de señal a través de él.

-DETECTOR DE MOVIMIENTO:

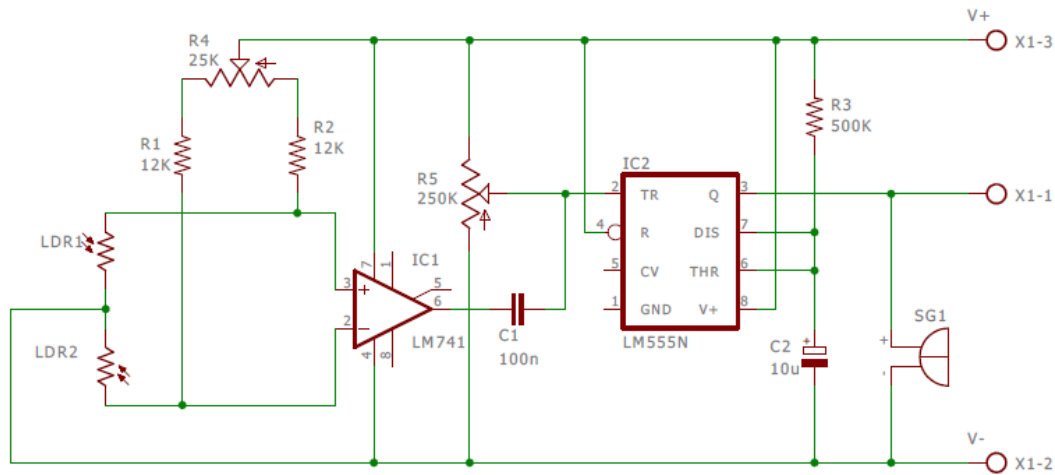


Fig. 3.9 Circuito para detección de movimiento con el 555

Fuente: Propia

Entendemos que el principio que utilizamos es la de comparar a través del LM741 la luminosidad que recibe cada ldr y cuando es diferente entre ellas se activa una alarma, puesto que gracias a esta comparación sabemos que cuando la cantidad luminosa en cada ldr es igual, nada se está moviendo. En el momento en que algo se mueve produce cambios en las sombras lo cual activa el NE555 emitiendo un voltaje que activará una alarma y nos avisara cuando éste esté activo.

Para este circuito se utilizó un NE555 en funcionamiento monoestable y un comparador que nos permita reconocer cuando hay un cambio en la medida de los sensores de luz LDR.

El circuito del NE555 monoestable es gobernado por la siguiente ecuación:

$$T = 1.1 * R1 * C1$$

Dónde:

T = Periodo de la señal de salida

$R1$ = Resistencia 1 en los pines 6 y 7 con alimentación

C1 = Capacitor 1 en los pines 6 y 7 con tierra

Entonces decidimos empezar a probar con distintos componente comerciales con tal de obtener una señal con un periodo aceptable y finalmente optamos por colocar una resistencia de 500k y un capacitor de 10uF por lo que el periodo nos da:

$$T = 1.1 * 500 * 10^3 * 10^{-6}$$

$$T = 5.5 \text{ segundos}$$

Las resistencias R1, R2 y el potenciómetro R4 nos permiten cambiar la sensibilidad del detector y utilizamos resistencias no tan altas debido a la máxima resistencia que pueden tener los sensores LDR, el potenciómetro R5 nos permite cambiar la corriente en la entrada, utilizamos una de 250k por seguridad.

El LM741 funciona como comparador, al cambiar el estado de los LDR, deja de enviar señal, esto lo estabiliza el capacitor de 100nF y así el pin 2 pasa a nivel bajo lo que dispara el pulso en el pin 3 con un periodo de 5.5 segundos.

-CONTROL DEL SERVOMOTOR:

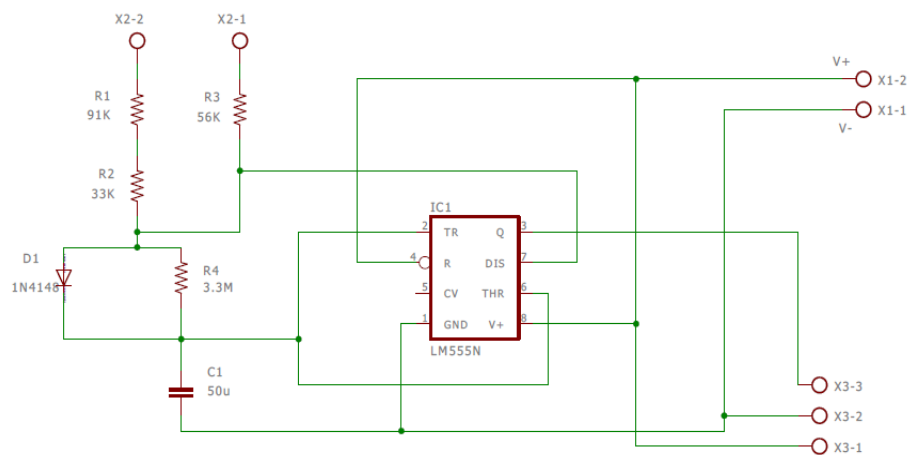


Fig. 3.10 Circuito para el control de giro de un servomotor con el 555

Fuente: Propia

Con este simple y sencillo circuito, logramos controlar el giro del servomotor mediante el giro de un potenciómetro conectado al NE555, simulando la apertura y cierre de una puerta de un garaje. El siguiente reto fue poder controlar exactamente los 90 grados de diferencia que debe haber entre cerrado y abierto, para lo cual probamos múltiples soluciones y adoptamos una de ellas, y cambiamos el potenciómetro por dos resistencias una de apertura y otra de cierre de la puerta.

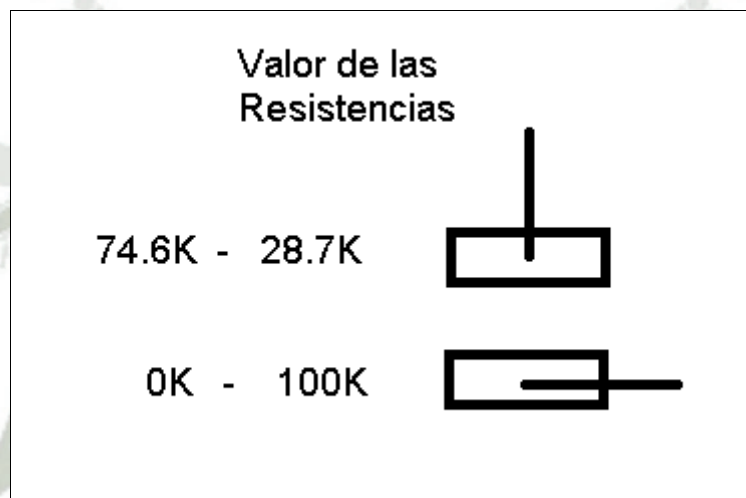


Fig. 3.11 Cálculo de los valores de resistencias para el giro del servomotor

Fuente: Propia

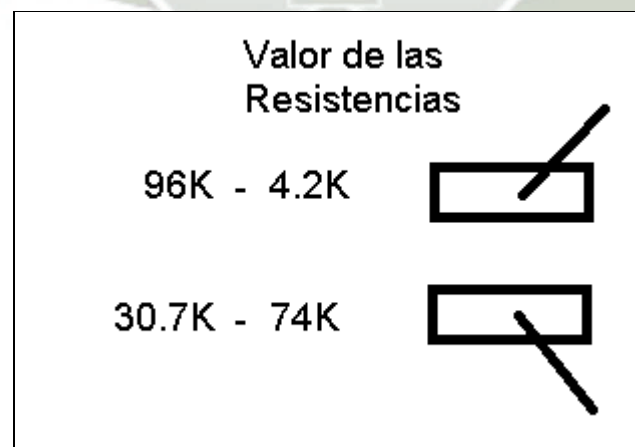


Fig. 3.12 Cálculo de otra alternativa de valores de resistencias para el giro del servomotor

Fuente: Propia

En el caso un servomotor, lo que controla su posición es la modulación de anchos de pulsos, que con un NE555 se puede realizar utilizando el modo de funcionamiento astable cuya ecuación es la siguiente:

$$f = \frac{1.44}{C * (Ra + 2Rb)}$$

Dónde:

f = Frecuencia de la señal de salida

C = Capacitor entre pin 6 y tierra

Ra = Primera resistencia entre alimentación y pin 7

Rb = Segunda resistencia entre pines 7 y 6

Para nuestro caso, necesitamos que el servomotor se encuentre solo en 2 posiciones. 0 y 90° aproximadamente. Para esto decidimos utilizar un capacitor de 50uF y una resistencia Rb de 3.3M, luego en base a eso obtendríamos las otras resistencias utilizando un potenciómetro y midiendo la resistencia en la posición deseada.

La primera posición nos salió 56k que utilizando la fórmula nos sale una frecuencia:

$$f = \frac{1.44}{50 * 10^{-6} * (56000 + 2 * 3.3 * 10^6)} = 0.004327 \text{ Hz}$$

Y en la segunda posición medimos aproximadamente una resistencia de 124k y nos da una frecuencia de:

$$f = \frac{1.44}{50 * 10^{-6} * (124000 + 2 * 3.3 * 10^6)} = 0.004283 \text{ Hz}$$

Entonces ya mediante la programación activaremos o bien la primera frecuencia o bien la segunda y así podremos intercalar ambas posiciones del servomotor.

El diodo colocado nos sirve para evitar contracorrientes.

Y empezamos el armado:

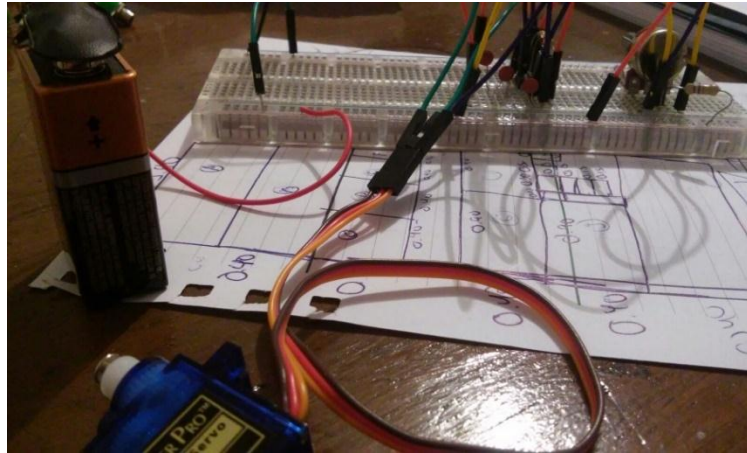


Fig. 3.13 Armado del circuito del servomotor

Fuente: Propia

Con el armado de los circuitos hechos en papel, empezamos con el testeo de voltajes y corrientes, corrección de errores, aumento o reducción del valor de resistencias y elección de accesorios.



Fig. 3.14 Corrección de errores en el armado de los circuitos

Fuente: Propia

Y luego el quemado provisional de las placas:

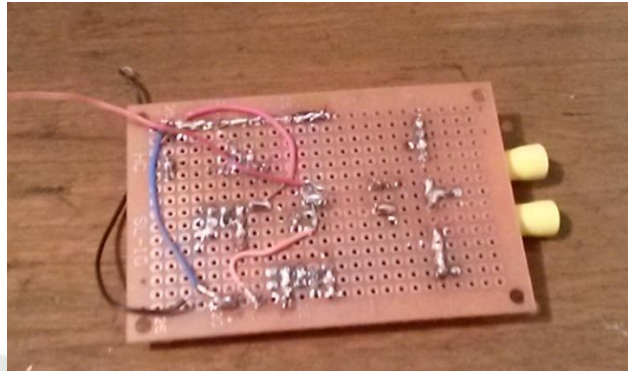


Fig. 3.15 Quemado de la placa del circuito del sensor de movimiento

Fuente: Propia

Y resultando las 3 de la siguiente manera:

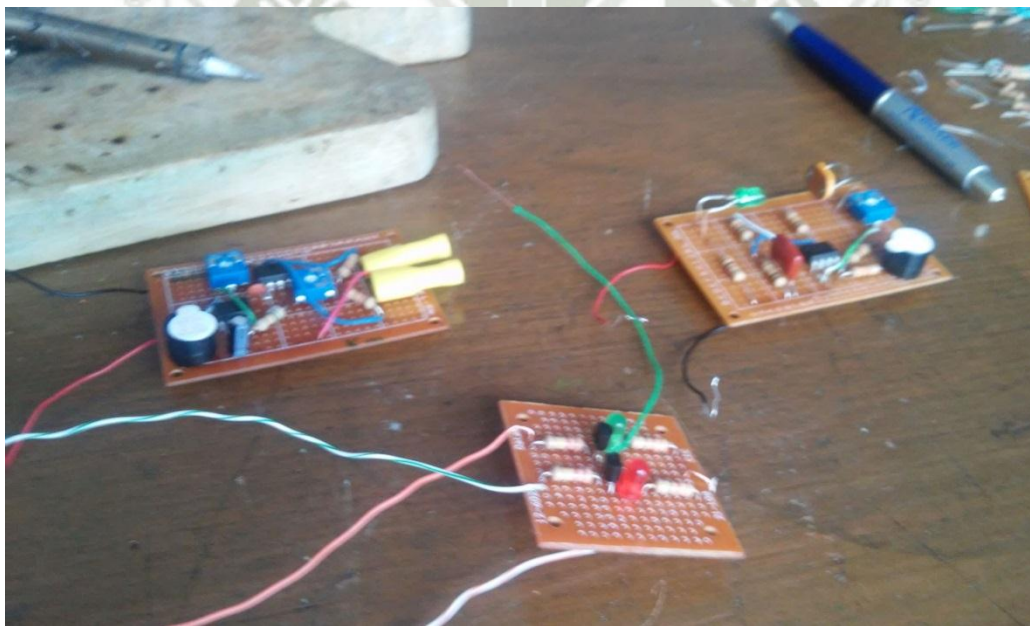


Fig. 3.16 Placas listas de los sensores a utilizar

Fuente: Propia

Analizando cada uno:

-SENSOR DE MOVIMIENTO:

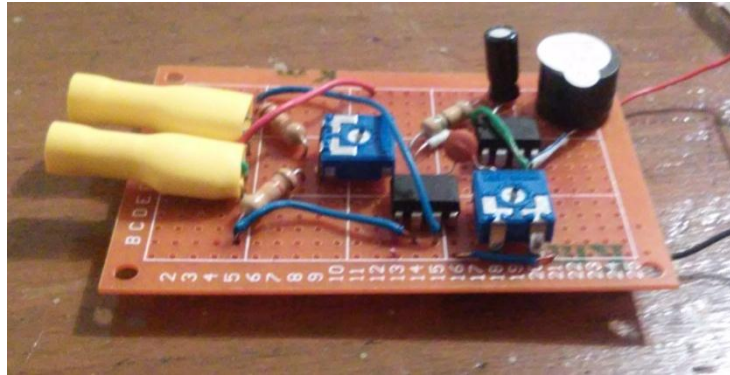


Fig. 3.17 Placa del circuito de sensor de movimiento

Fuente: Propia

Con la característica que lo que analiza es la diferencia de sombras gracias al comparador así detecta algún movimiento y suena una alarma.

-SENSOR DE HUMO:

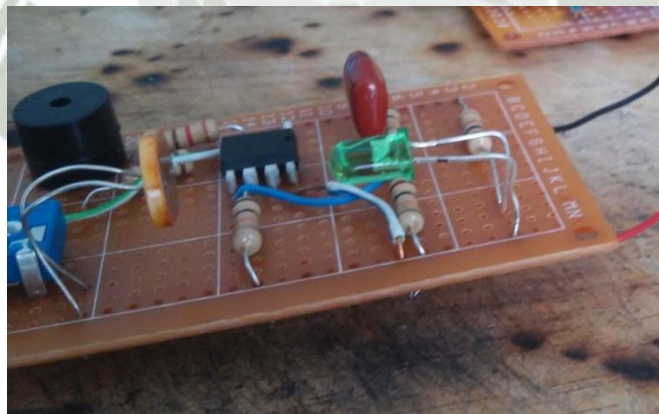


Fig. 3.18 Placa del circuito del sensor de humo

Fuente: Propia

Funciona como un sensor de barrera. Cuando la recepción del LDR de la luz que emite el led es interrumpida, suena una alarma para que luego se pueda activar un ventilador que ventile el humo que interrumpe la visión de la luz.

-SENSOR DE NIVEL:

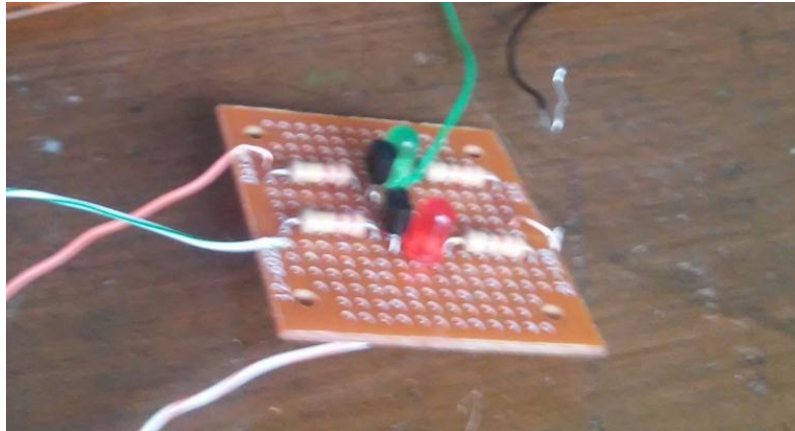


Fig. 3.19 Placa del circuito del sensor de nivel de tanque

Fuente: Propia

Tiene dos led de aviso, el de color verde es cuando el nivel del tanque es el elegido, y el led de color rojo que avisa cuando está casi en el limite de nivel maximo del tanque que rebalsaria al seguir asi.

-CONTROLADOR DE SERVOMOTOR:

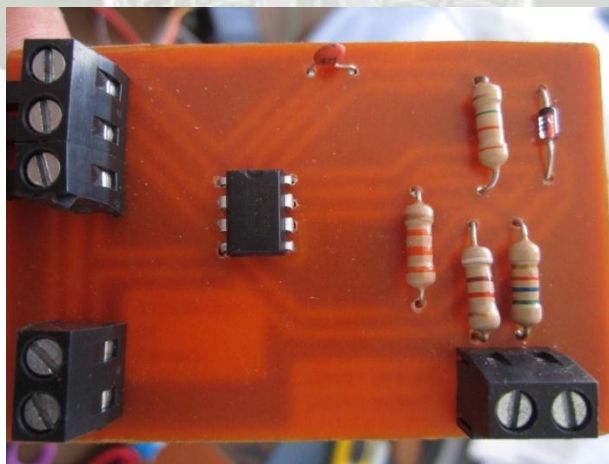


Fig. 3.20 Placa del circuito del control del servomotor

Fuente: Propia

Para éste circuito utilizaremos el NE555 para controlar la apertura o el cerrado de una puerta de garaje. Tenemos dos borneras de apertura y cerrado con un arreglo de resistencias analizados más arriba en la Fig. 3.11 y Fig. 3.12.

-ILUMINACION AUTOMATICA:

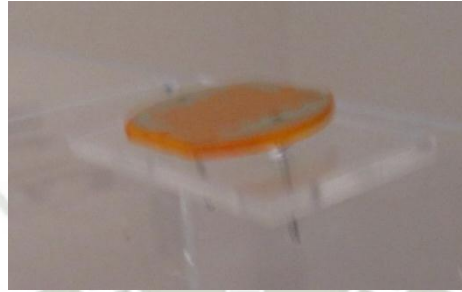


Fig. 3.21 LDR para el circuito de iluminación automática

Fuente: Propia

Utilizaremos también un sensor LDR para variar el voltaje de salidas dependiendo de la iluminación total que tenga la habitación.

-CONTROL TOTAL DE ILUMINACIÓN:



Fig. 3.22 Interruptor para el circuito de control total de iluminación

Fuente: Propia

3.2 Construcción del modulo

Luego de tener los circuitos ya listos decidimos construir el submodulo de vivienda, simulando una casa real, eligiendo los materiales mas adecuados decidimos utilizar Acrílico para las paredes de la vivienda por ser transparente y resistente a la vez, preguntamos y cotizamos los diferentes tipos de Acrílicos:



Fig. 3.23 Diferentes espesores de Acrílico

Fuente: Propia

Y elegimos el espesor de 4 mm por su resistencia y su buen acabado, vimos de distintos espesores mas unos mas delgados otros mas gruesos, pero decidimos por ese porque cumplia con lo que deseabamos.



Fig. 3.24 Espesor de plancha elegida: 4 mm

Fuente: Propia

Luego empezamos a hacer distintos tipos de viviendas para desarrollar nuestro proyecto, formas internas externas, varias formas de distribuir las habitaciones para poder utilizar al maximo el espacio de la vivienda para nuestro cometido.

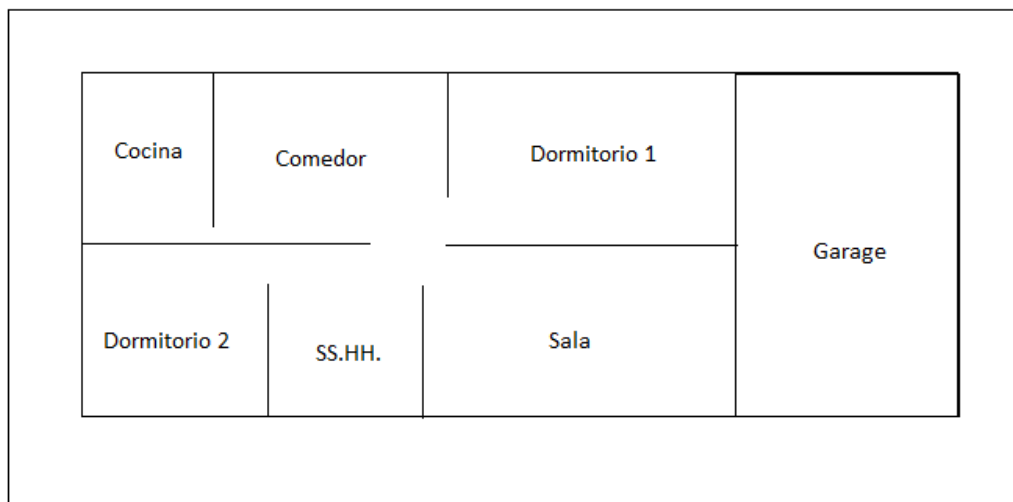


Fig. 3.25 Primer boceto de los planos de la vivienda

Fuente: Propia

Luego de haber decidido la distribución, lo hicimos en maqueta de papel para encontrar la proporción correcta.



Fig. 3.26 Inicio del armado de los primeros prototipos de la vivienda

Fuente: Propia



Fig. 3.27 Boceto terminado en papel de la vivienda

Fuente: Propia

Luego de tener la proporción correcta de la vivienda la pasamos a planos escritos para poder hacer los cortes en las planchas de Acrílico que compramos.

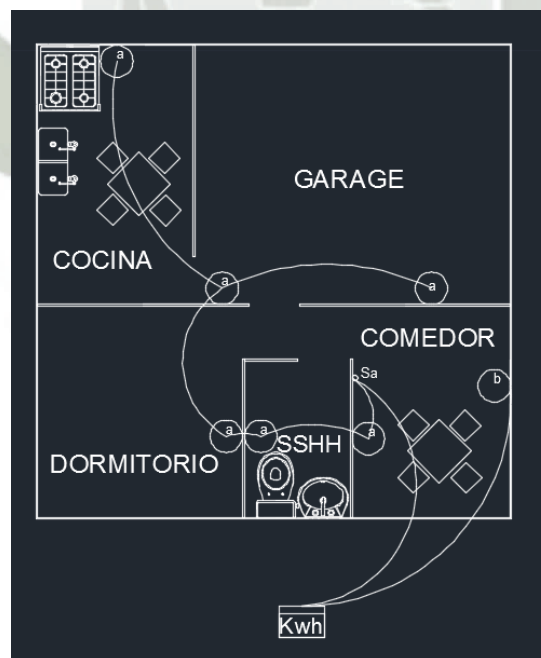


Fig. 3.28 Plano del boceto elegido para la vivienda

Fuente: Propia

Y sus equivalencias en la medida del submodulo a escala.

<u>EQUIVALENCIAS EN ESCALA 7:1</u>	
2 de 7m x 2.1m =>	1m x 0.3m
2 de 0.7m x 2.1m =>	0.1m x 0.3m
4 de 6.3m x 2.1m =>	0.9m x 0.3m
2 de 5.6m x 2.1m =>	0.8m x 0.3m
3 de 2.1m x 2.1m =>	0.3m x 0.3m
1 de 3.5m x 2.1m =>	0.5m x 0.3m
1 de 4.2m x 2.1m =>	0.6m x 0.3m

Fig. 3.29 Equivalencias en escala de las medidas de la vivienda

Fuente: Propia

Y aquí la distribución de las piezas en la plancha de Acrílico, las cuales tiene las dimensiones de 1.20m x 1.80m.

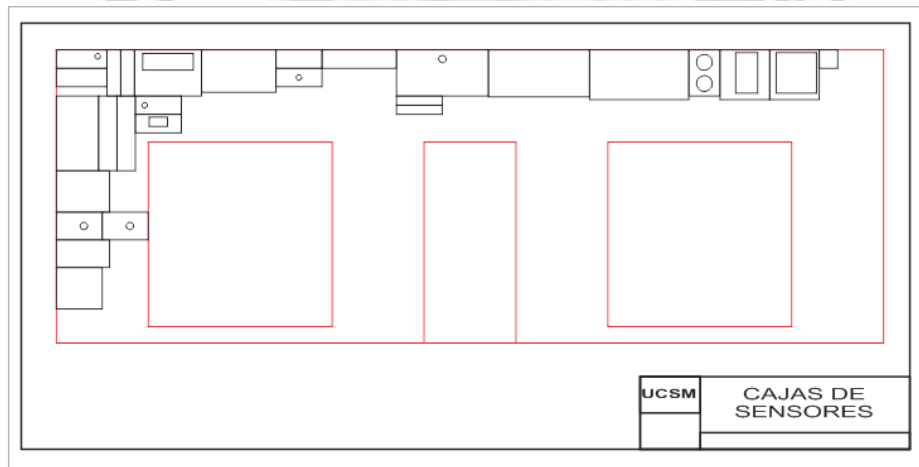


Fig. 3.30 Distribución de las piezas en la plancha a comprar

Fuente: Propia

Y también los planos de los tanques para el submodulo de nivel de tanque:

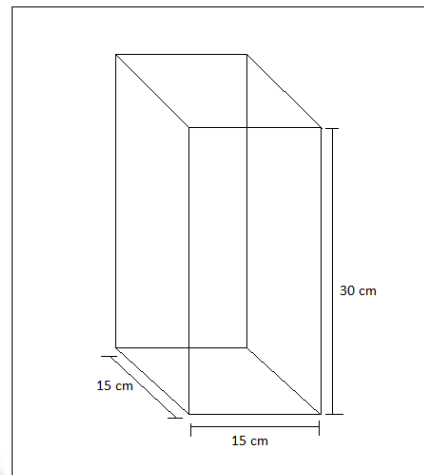


Fig. 3.31 Planos de los tanques a utilizar

Fuente: Propia

Y al tener los planos los llevamos a cortar y empezamos el armado, decidimos utilizar cloroformo para el pegado de las placas.

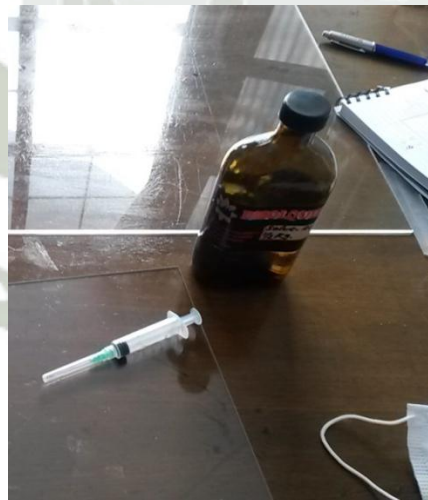


Fig. 3.32 Botella de cloroformo para unir las planchas cortadas de acrílico

Fuente: Propia



Fig. 3.33 Secado de los tanques de agua ya unidos

Fuente: Propia

Y la disposición de los mismos.



Fig. 3.34 Disposición de los tanques de agua

Fuente: Propia

Luego de terminar ésta parte y probar que este completamente sellado, iniciamos con el armado de la casa.



Fig. 3.35 Pegado de las paredes del submodulo de vivienda

Fuente: Propia

Pegado completo de las paredes:



Fig. 3.36 Posición final del submodulo de vivienda

Fuente: Propia

Pegado del piso para mayor estabilidad de la vivienda.



Fig. 3.37 Unimos el piso a la vivienda

Fuente: Propia

Posicionamiento de la puerta de entrada:



Fig. 3.38 Posicionamiento de las puertas de entrada y garaje

Fuente: Propia

Como una puerta se selló con una visagra:



Fig. 3.39 Puerta de entrada con sus bisagras

Fuente: Propia

Luego para el techo no utilizamos Acrílico en vez de ello utilizamos policarbonato por ser un material de excelente acabado y peso mínimo, lo medimos y cortamos para que nos quede así:

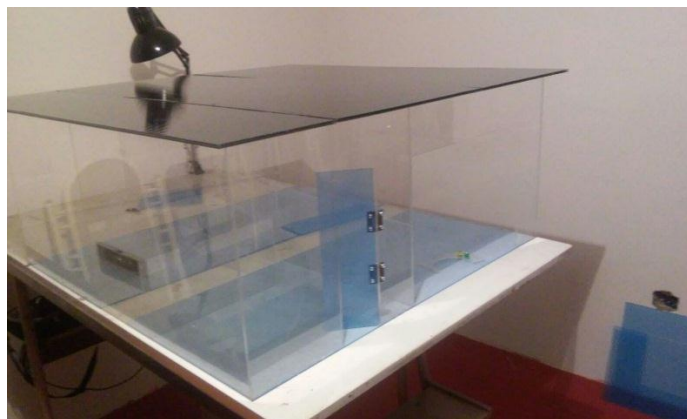


Fig. 3.40 Posicionamiento del techo de la vivienda

Fuente: Propia

Para unir las placas en el techo y para darle un mejor acabado al techo pusimos perfiles H de material Plástico, y para los bordes del techo utilizamos perfiles U de material Aluminio, los cuales medimos y cortamos para que quede exacto y los sellamos con Silicona.

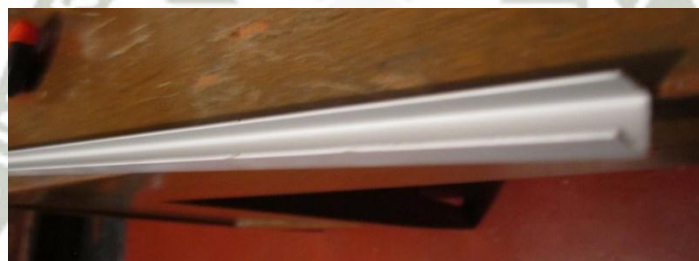


Fig. 3.41 Elección del perfil en U de aluminio

Fuente: Propia

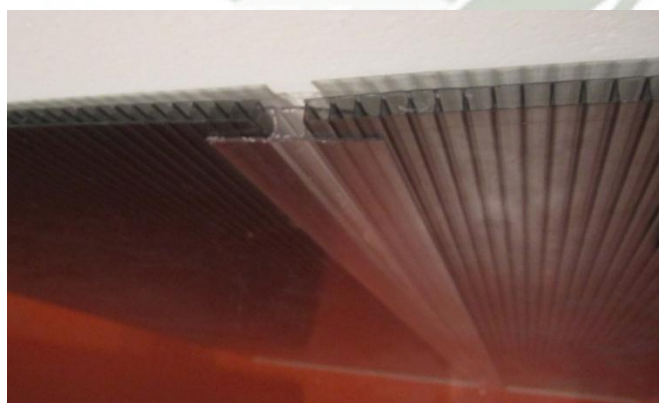


Fig. 3.42 Elección y posicionamiento de los perfiles en H de plástico

Fuente: Propia

Elegimos el perfil en H de plástico puesto que tiene mayor flexibilidad que el de aluminio, y al considerar nuestro techo movable, sabemos que tendrá que soportar bastante movimiento por esa razón se eligió el material de este perfil.



Fig. 3.43 Colocación de los perfiles en U y en H para el techo de la vivienda

Fuente: Propia

Luego hicimos con el mismo material Acrílico, las cajas que soportarían a los sensores en la vivienda y en el tanque, para que al ser de este material se pueda mirar y observar desde afuera las placas de los circuitos y su funcionamiento.

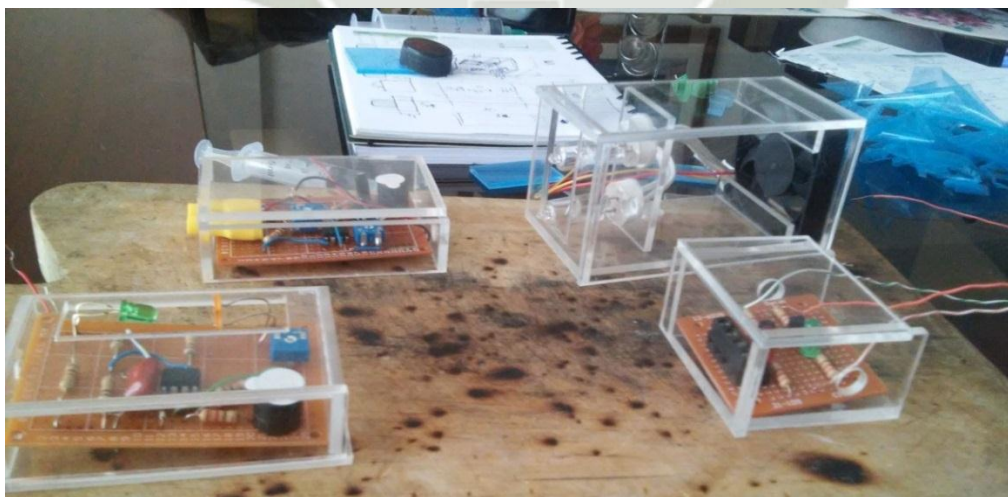


Fig. 3.44 Sensores en sus respectivos contenedores de Acrílico

Fuente: Propia

Como se vio cada caja tiene una tapa para poder acceder a los sensores.
Luego pegamos las cajas de sensores y se fijaron las tapas a la vivienda.

-SISTEMA DE CALEFACCION:

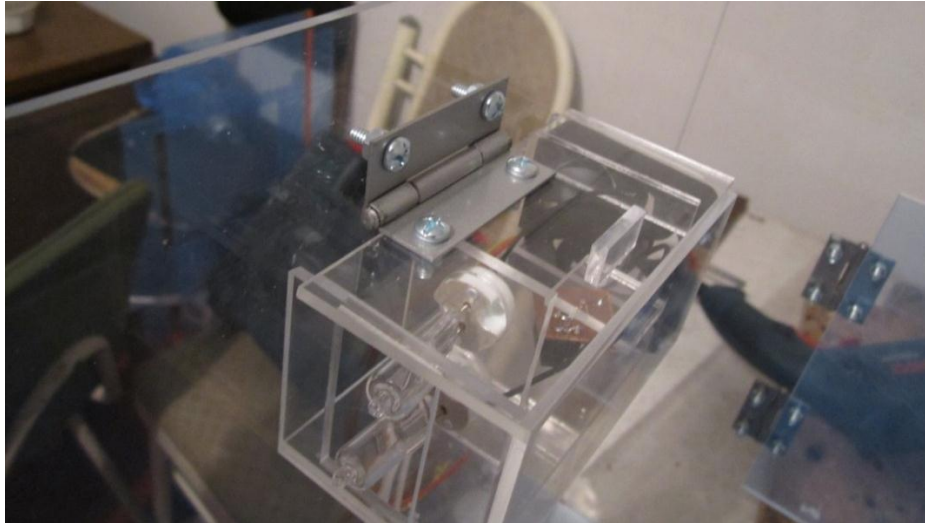


Fig. 3.45 Sistema de calefacción colocado en su posición final

Fuente: Propia

-SENSOR DE MOVIMIENTO:

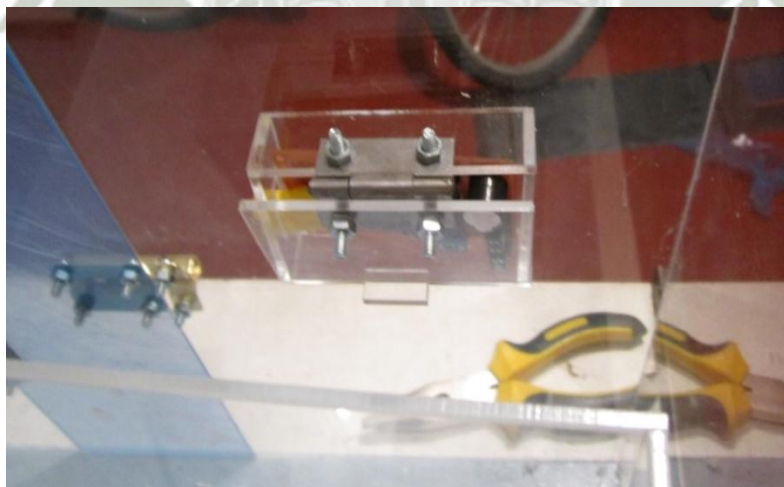


Fig. 3.46 Sensor de movimiento colocado en su posición final

Fuente: Propia

-DETECTOR DE HUMO:

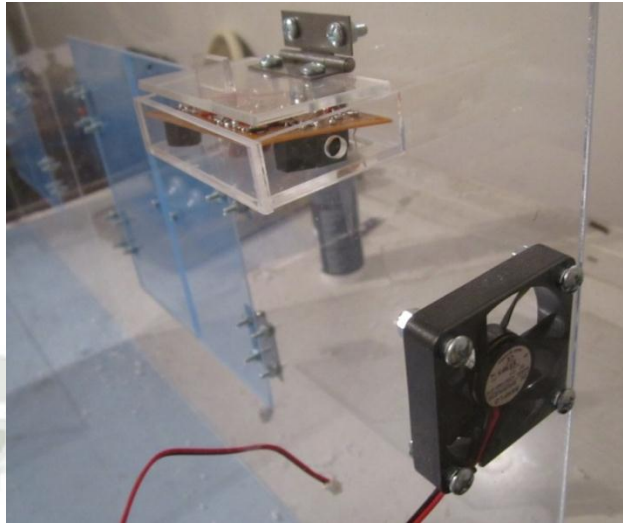


Fig. 3.47 Sensor de humo y ventilador disipador de humo colocados en su posición final

Fuente: Propia

-SENSOR DE LUZ:

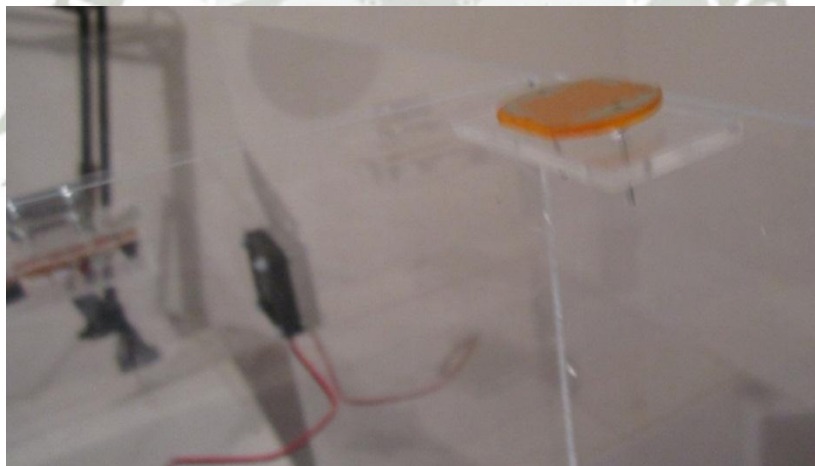


Fig. 3.48 LDR colocado en su posición final

Fuente: Propia

Luego iniciamos el armado de la puerta de garage con el sistema de giro de un eje que ensamblado a la puerta, al girar levante ésta para que se pueda acceder al garage mismo.

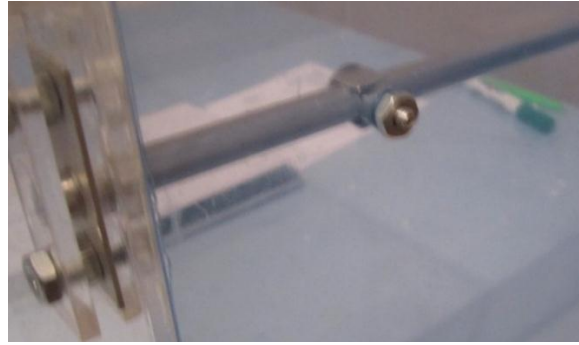


Fig. 3.49 Sistema de soporte para el eje de la puerta de garaje

Fuente: Propia



Fig. 3.50 Puerta de garaje montada en su posición final

Fuente: Propia

Iniciamos el pegado de los accesorios en las ventanas y puertas.



Fig. 3.51 Puerta de entrada con sus accesorios

Fuente: Propia



Fig. 3.52 Ventana de la cocina con sus accesorios

Fuente: Propia



Fig. 3.53 Ventana de la habitación principal con sus accesorios

Fuente: Propia

Con un resultado parcial bastante cercano al pensado desde un inicio.

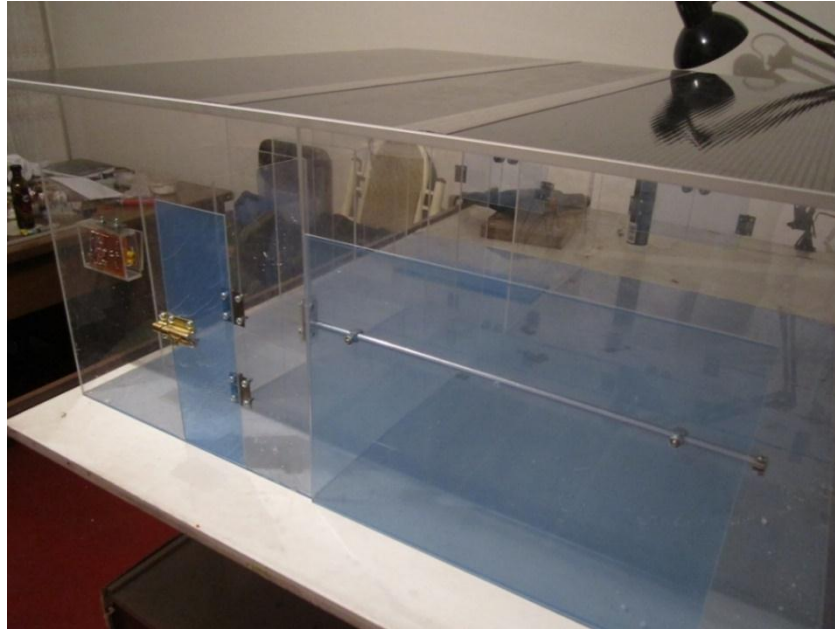


Fig. 3.54 Construcción parcial del submodulo de vivienda

Fuente: Propia

Luego elegimos las canaletas para el paso de los cables dentro del submodulo por lo que elegimos la mas pequeña con el fin de seguir manteniendo la escala de ésta misma.

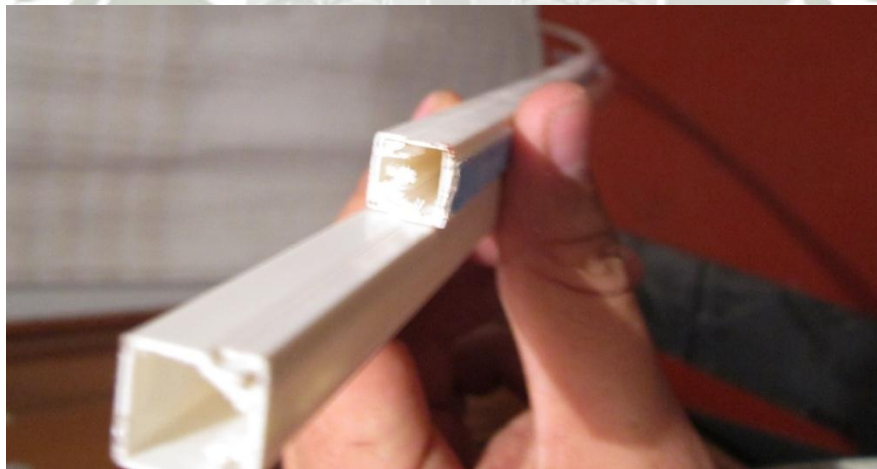


Fig. 3.55 Elección de las canaletas para el paso de los cables

Fuente: Propia



Fig. 3.56 Ácido para el quemado de las placas y la placa donde se quemarán los circuitos.

Fuente: Propia

El ácido es “Cloruro Férrico” y la placa es de baquelita.



Fig. 3.57 Preparación para el planchado de las placas

Fuente: Propia

El primer paso fue envolver con el papel impreso del circuito a la placa de baquelita, siempre observando que exceda un poco de todos los costados para que se pueda pegar con la

cinta luego. El siguiente paso fue planchar por encima con una plancha cualquiera, por unos 10 a 15 minutos en temperatura para algodón, presionando fuerte y moviendo a todas las esquinas.



Fig. 3.58 Resultado luego de haber pasado por el ácido las placas.

Fuente: Propia

Luego de haber planchado se quedará impreso en la plancha las pistas impresas en el papel. La pasamos por agua para enfriar y sacar el papel a medias, debe quedar un poco opaco como en la foto y luego pasar por el ácido, manteniendo ahí por una hora aproximadamente.

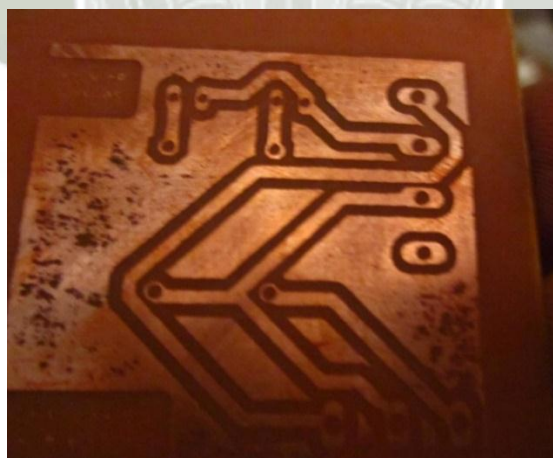


Fig. 3.59 Resultado final de las placas.

Fuente: Propia

Y el resultado fue éste, se pudo reconocer fácilmente todas las pistas, luego con un taladro se perforó y luego la soldamos con estaño y quedaron listas para usar.



Fig. 3.60 Nombres impresos en la placa.

Fuente: Propia

Imprimimos nuestros nombres en las placas como muestra de nuestra autoría y también el nombre del sensor o detector al cual pertenece el circuito.



Fig. 3.61 Elección de la fuente de 24 v

Fuente: Propia

Para la fuente de 24v. Elegimos una fuente que soporte 10 A. Para que no exista luego problemas con la alimentación de los focos, ventiladores o actuadores.



Fig. 3.62 Elección de la fuente de 12 v

Fuente: Propia

Para la fuente de 12v. Elegimos una fuente que soporte 15 A. Para que no tengamos problemas de alimentación a los sensores, los focos, los ventiladores y los actuadores. Ésta fuente tiene la posibilidad de trabajar con 110v o 230v escogimos 230v porque es la señal que recibimos en nuestros terminales en nuestra ciudad.



Fig. 3.63 Cable y terminal de las fuentes

Fuente: Propia

Para protección de las fuentes elegimos un cable de alimentación vulcanizado y un terminal que soporta 15 A. Altas temperaturas y golpes sin dañarse.

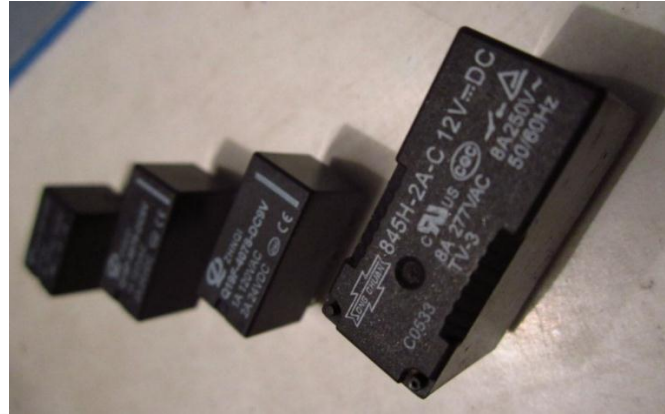


Fig. 3.64 Distintos tipos de relay

Fuente: Propia

Para la parte final del sub modulo, escogimos que todas las entradas, salidas para el PLC serían activados por relay de distintos voltajes dependiendo de las salidas de cada sensor.

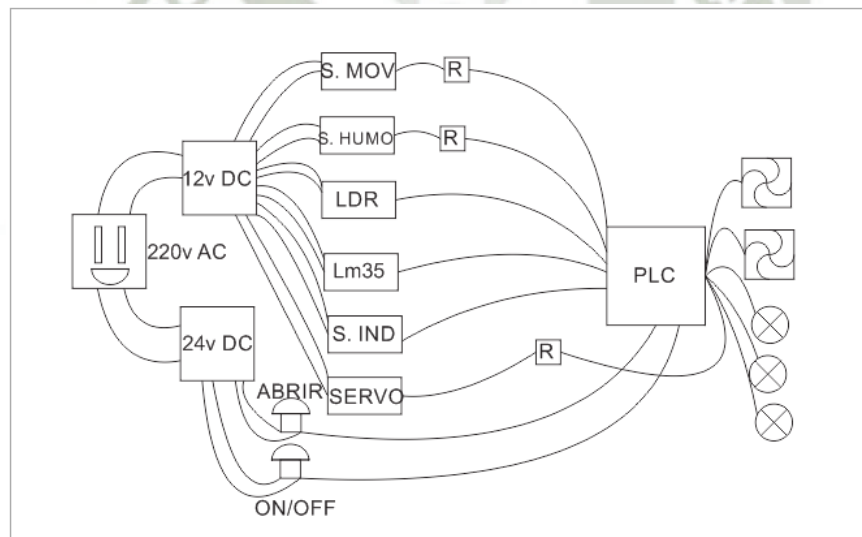


Fig. 3.65 Distribución de las entradas y salidas con sus respectivos relay

Fuente: Propia

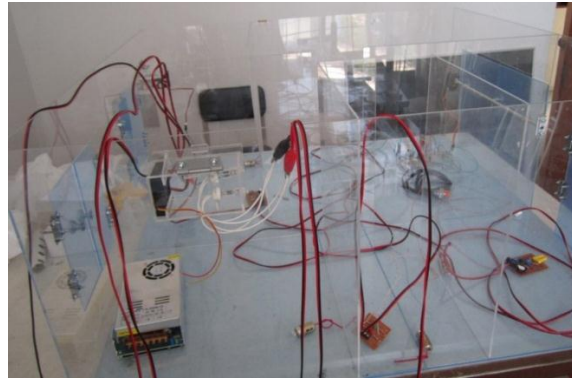


Fig. 3.66 Posicionamiento de todos los cables conectados a las fuentes en sus posiciones referenciales dentro del sub-modulo

Fuente: Propia

Pusimos una posición referencial dentro de la casa solo para poder probar los distintos sensores.



Fig. 3.67 Pruebas con las fuentes encendidas

Fuente: Propia

Luego conectamos a las fuentes y se probaron los circuitos.



Fig. 3.68 Cableado y puesta de canaletas

Fuente: Propia

Con las pruebas finalizadas, iniciamos el cableado de todos los sensores pero ya en sus posiciones finales.



Fig. 3.69 Elección del sensor inductivo industrial

Fuente: Propia

Para completar nuestros sensores compramos un sensor inductivo de uso industrial, para comprobar que no solo nuestro proyecto funciona en sensores caseros sino en industriales y puede ser utilizado para ponerlo en funcionamiento en cualquier tipo de sistema sea real o simulado. Las especificaciones técnicas se pueden encontrar en “ANEXOS”.

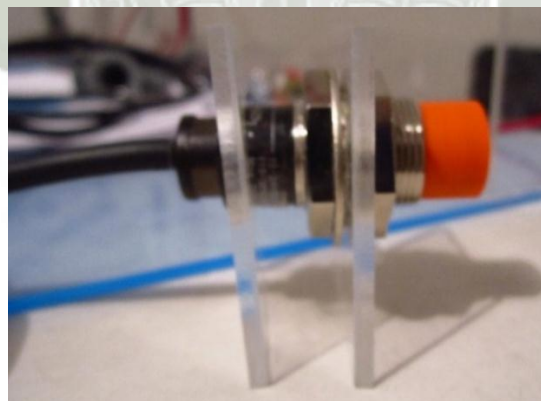


Fig. 3.70 Armado del soporte para el sensor inductivo

Fuente: Propia

Para posicionar el sensor inductivo dentro de la casa se hizo un soporte con acrílico para poder posicionarlo con comodidad.

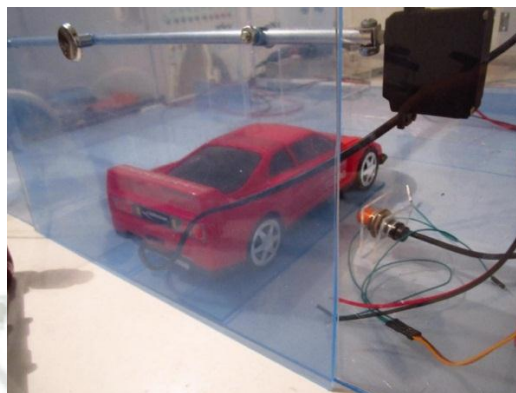


Fig. 3.71 Posicionamiento del sensor inductivo, del auto y del servomotor dentro del garaje.

Fuente: Propia

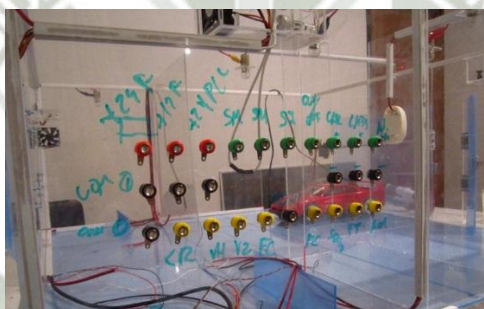


Fig. 3.72 Borneras de entradas y salida.

Fuente: Propia

Son las borneras en su posición final, aquí conectamos la placa de los relay y los fusibles.



Fig. 3.73 Bomba elegida.

Fuente: Propia

Para nuestro sub módulo de nivel de tanque escogimos esta bomba, elegimos la más pequeña puesto que nuestros tanques son pequeños y la altura que elevará la bomba es también pequeña.

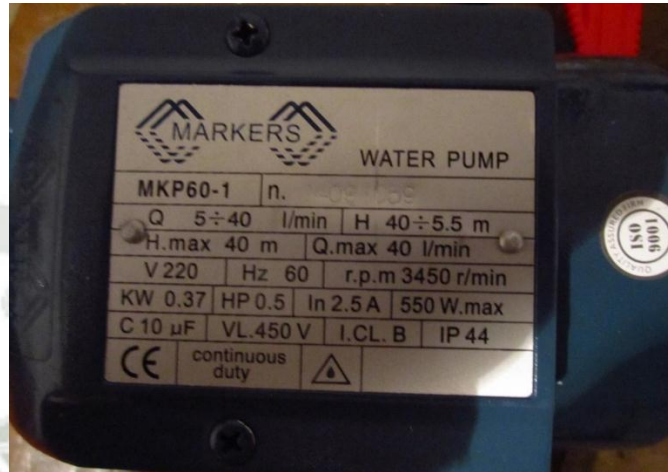


Fig. 3.74 Datos de placa de la bomba elegida

Fuente: Propia

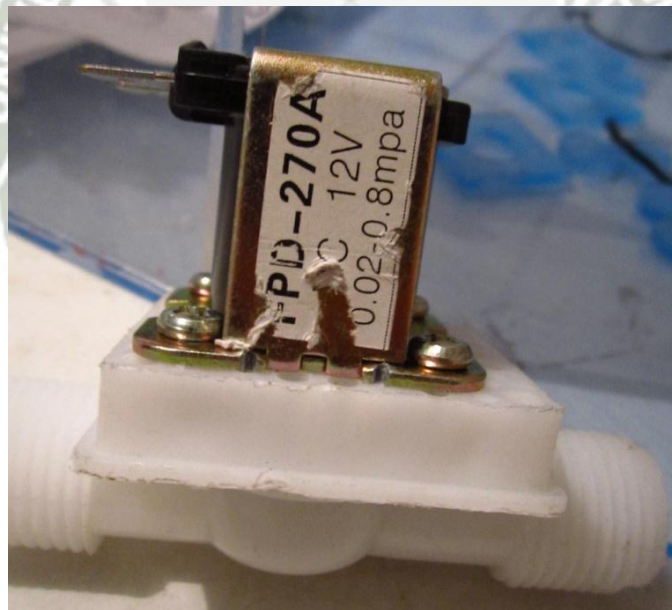


Fig. 3.75 Válvula Solenoide

Fuente: Propia

Elegimos una válvula solenoide para la apertura y cierre de la misma, que sea 12 voltios para que sea activado por la fuente de 12v y 15 A.

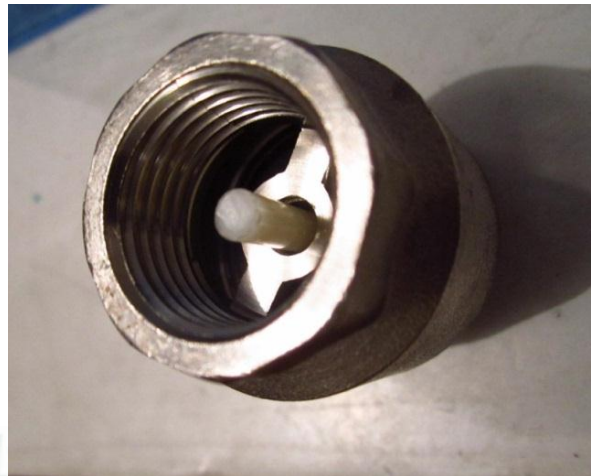


Fig. 3.76 Válvula Check

Fuente: Propia

Para protección de la bomba necesitaremos una válvula Check para el anti retorno del caudal que pase por la tubería.



Fig. 3.77 Pulsadores de INICIO y STOP.

Fuente: Propia

Para los pulsadores de INICIO y STOP utilizamos dos pulsadores, uno Normalmente Abierto y otro Normalmente Cerrado. El pulsador de INICIO da un pulso al presionarse y se activa. El pulsador de STOP al ser normalmente cerrado ya está activado y cuando se presiona corta la línea.



Fig. 3.78 Válvulas manuales y accesorios.

Fuente: Propia

Utilizamos válvulas, codos y tees de ½ pulgada.



Fig. 3.79 Armado de la tubería de succión.

Fuente: Propia

Esta fue la estructura ya diseñada para la succión de la bomba.

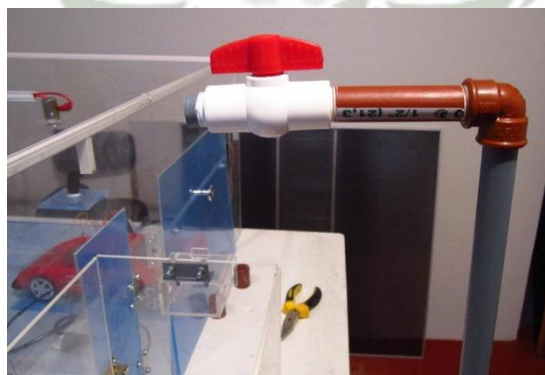


Fig. 3.80 Armado de la tubería de descarga.

Fuente: Propia

Esta fue la estructura ya diseñada para la descarga de la bomba, nótese que se hizo de una altura bastante amplia con respecto a la altura del tanque, para poder ser manipulado fácilmente sin necesidad de desensamblar todo.



Fig. 3.81 Armado de la tubería que conectará los dos tanques con la válvula solenoide.

Fuente: Propia

Ésta fue la estructura para el arreglo de tuberías que conectarán ambos tanques. Se conectó una válvula manual en paralelo a la válvula solenoide en caso de falla.



Fig. 3.82 Elección de nuestros PLC's y HMI.

Fuente: Propia

Con todo lo antes diseño y armado, elegimos nuestros PLC y el HMI. Compramos unos equipos que tenían más de lo que necesitábamos para que también sean de uso para nuestros compañeros con muchas y varias aplicaciones extra.

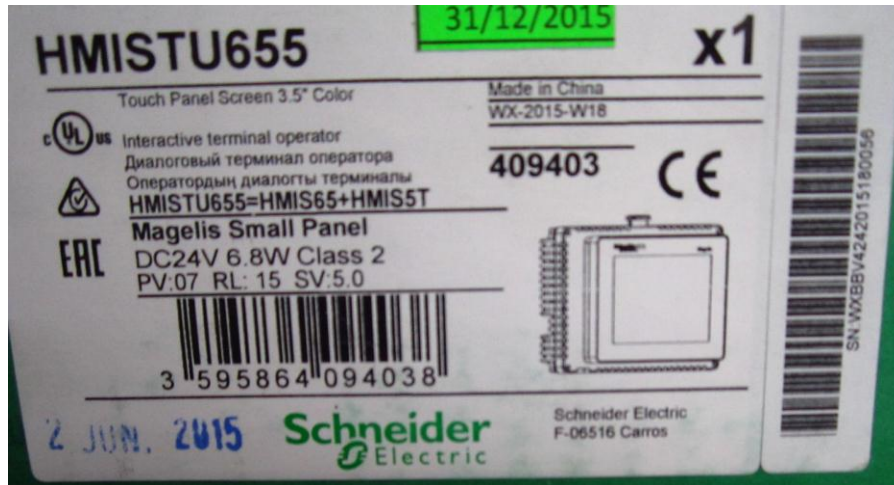


Fig. 3.83 Datos del HMI.

Fuente: Propia

Elegimos este HMI, es un terminal de 3.5" a 64K Colores. La elección fue basada en varios aspectos los cuales son:

- Tiene puerto Serial así como puerto Ethernet multiprotocolo.
- Se puede inclusive descargar programas desde puerto Mini-USB, puerto USB o un teclado USB.
- Un sistema de montaje muy sencillo en comparación de otros modelos y otras marcas que llevan pinzas de sujeción. Éste modelo incorpora un sistema de montaje “push-button” que es de lejos más sencillo y de mayor ajuste que los demás.
- Tienen un software libre y una tecnología abierta para ser modificada o utilizada por cualquier persona.

- Son compatibles con casi todas las marcas de PLC y controladores. Teniendo así una versatilidad muy amplia y su programación es sencilla y gráficamente amigable.
- El reconocimiento de equipo para la comunicación PC-HMI es rápida y no necesita estar manipulando las direcciones.
- La relación calidad-precio es justa. Los precios son accesibles y su utilidad es de enorme práctica.
- Tenemos soporte en nuestra ciudad, tanto en reparación de piezas como en soporte técnico. No es necesario enviar o llamar a otras ciudades para contactarse con algún proveedor.
- Es compatible con todo tipo de accesorios sean de alguna otra empresa o algunos creados por el usuario.
- La tensión de trabajo es pequeña, lo que no representa un riesgo para el usuario de choque eléctrico y puede ser alimentada con cualquier fuente porque su consumo de energía es mínimo.
- La pantalla no es monocromática, lo que aumenta la presentación física y los gráficos son más amigables. Las especificaciones técnicas completas se pueden encontrar en “ANEXOS”.



Fig. 3.84 Datos del PLC.

Fuente: Propia

Elegimos este PLC, tiene 24 E/S digitales con salidas tipo Relay. La elección de los mismos estuvo basada en diferentes aspectos que son:

- Tiene la posibilidad de conectar 2 entradas de tipo analógicas.
- No necesita alguna fuente externa de alimentación, tan solo utilizar la línea doméstica de 220v-230v.
- Con posibilidad de expansión hasta 14 módulos diferentes lo que lo hace un equipo de alta potencia, puesto que puede trabajar con “n” cantidad de variables al mismo tiempo sin afectar su rendimiento.
- Su programación es intuitiva lo que ayuda a comprender mejor el software a los usuarios.
- El software es amigable. Y ante cualquier error te aparece la mejor solución para el mismo y no te permite agregar elementos que no estén permitidos para cada uno de los controladores que se elijan para programar en este software.
- Tiene la opción simulador para registrar los eventos antes de ser descargados al controlador.
- Fácil detección de direcciones y sencilla descarga al controlador lo que lo hace sumamente versátil.
- Posee la posibilidad de incluirle una tarjeta SD y conectarlo a través de un puerto USB.
- Tiene una pila de memoria con un tiempo de vida de 4 años. Para no tener una descarga de datos cada vez que se enciende el controlador.
- Tiene una resolución para entradas analógicas de 10 bits lo que lo hace altamente sensible a los cambios externos que reciban los sensores y que puedan dar una respuesta en señal analógica muy precisa. Siendo el bit menos significativo 0.010 V.
- Tiene una velocidad de respuesta mínima de 5 μ s y máxima de 100 μ s.
- La durabilidad de los relés de salida es bastante amplia. Siendo mayor que 20 000 000 ciclos.

-Soporte técnico en la ciudad de Arequipa. Sin necesidad de viajar o llamar a otras ciudades para encontrar un proveedor para repuestos o soporte.

-La relación calidad-precio es justa. Teniendo con un precio accesible para una utilidad de máxima versatilidad, potencia y eficiencia.

-Tiene varios indicadores luminosos para darnos cuenta de todos los estados en que se encuentre el equipo. Y también indicadores por cada entrada y salida, para saber si está siendo utilizada o activada correctamente.

-Es compatible con cualquier tipo de accesorios. Por lo tanto no tiene una tecnología cerrada para el uso de la misma empresa, sino para el uso de cualquier elemento sea industrial o no que pueda manejar el voltaje de salida correctamente sin sufrir daños. Exceptuando los cartuchos de expansión que si deben ser los recomendados por la empresa de lo contrario pueden sufrir daños los equipos.

Las especificaciones técnicas completas se pueden encontrar en “ANEXOS”.



Fig. 3.85 Vista de los PLC's y el HMI.

Fuente: Propia

Ésta es la vista física de los equipos comprados, nuevos y sin configurar.



Fig. 3.86 Elección de los fusibles.

Fuente: Propia

Luego de elegir los controladores y el terminal, recién supimos la protección que tenían los mismos y en base a ello fue que elegimos este fusible, puesto que los controladores necesitan protección a partir de 2 A. Entonces compramos fusibles para todas las salidas y la alimentación de 2 A.

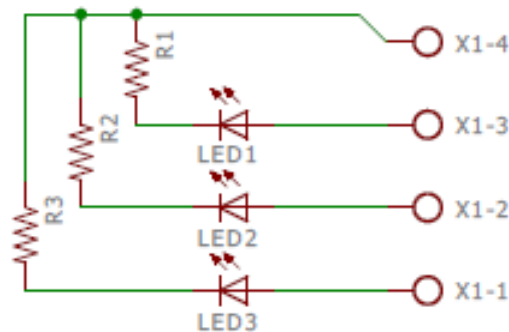


Fig. 3.87 Diseño del circuito de luces de alerta de nivel de tanque

Fuente: Propia

Armamos un circuito de luces para saber el nivel real de agua en el tanque, el led amarillo indica que está todo alimentado, el led verde indica que el nivel de agua es el correcto y por

último el led rojo indica que está en un nivel muy alto de agua y se detendrá la bomba porque el tanque ya está lleno.

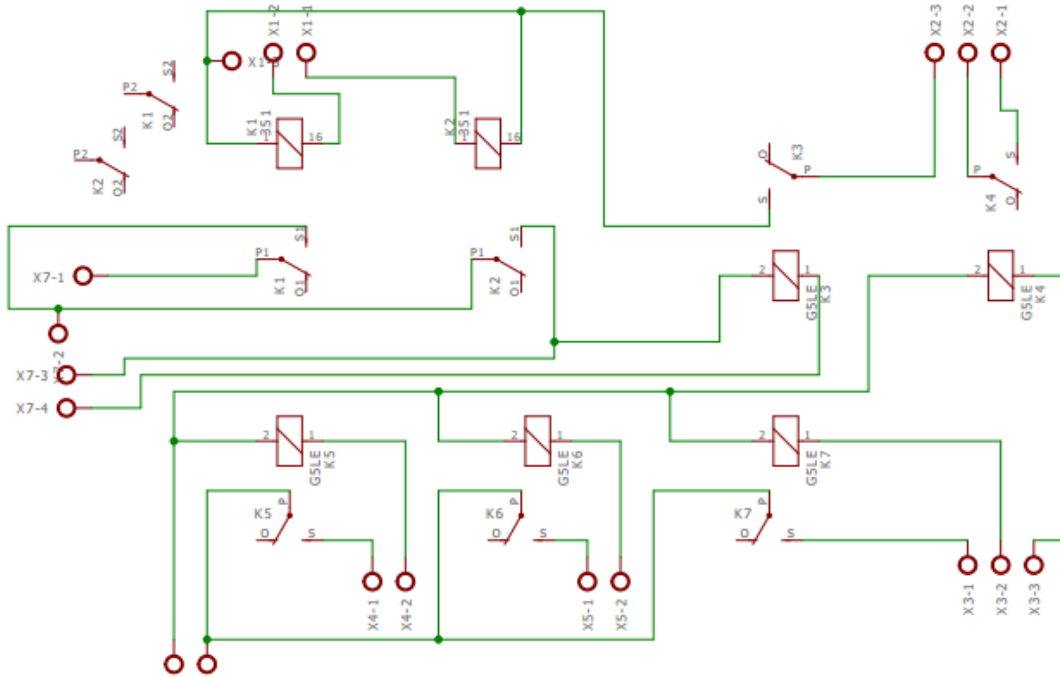


Fig. 3.88 Diseño de los relay del módulo de nivel de tanque

Fuente: Propia

Como información previa sabemos que el PLC tan solo acepta entradas de 24v. Como entradas digitales entonces para convertir la salida de los sensores en 24 v. utilizaremos la fuente interna del PLC por lo cual necesitamos un circuito de relays que separen las salidas analógicas de los sensores, por lo cual medimos las salidas de cada uno y elegimos el relay necesario para cada uno.

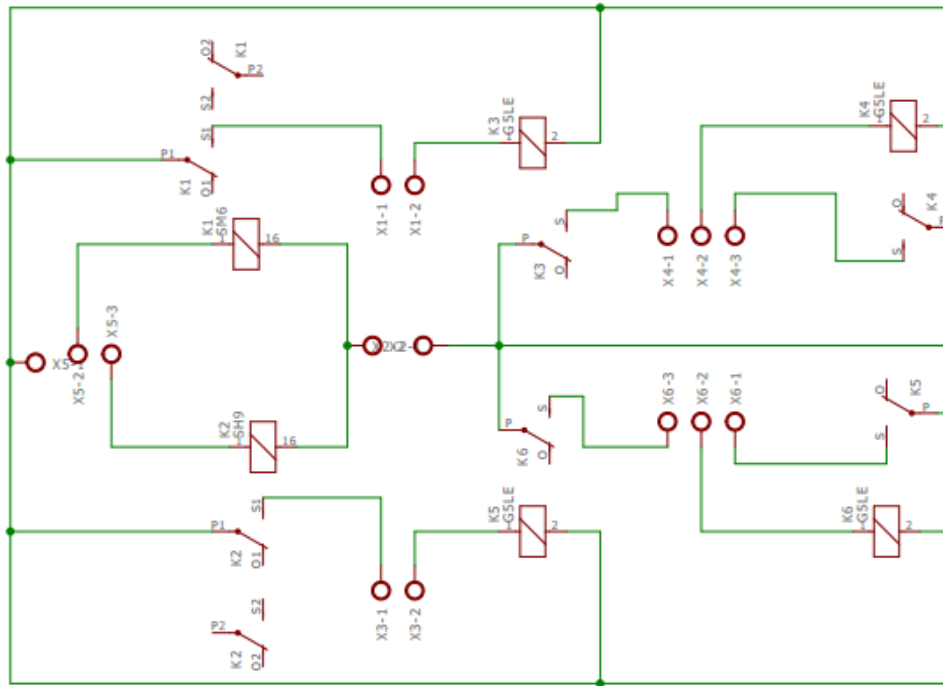


Fig. 3.89 Diseño de los relay de los circuitos alimentados por 12v.

Fuente: Propia

Para una mejor comprensión y orden a la hora de cablear los relay, se dividieron en 3 partes:

- La primera, vista en la Fig. 3.88 que es los Relay del circuito de nivel de tanque.
- La segunda, vista en la Fig. 3.89 está el diseño de los Relay alimentados por la fuente de 12v.
- La tercera, vista en la Fig. 3.90 está el diseño de los Relay alimentados por la fuente de 24v.

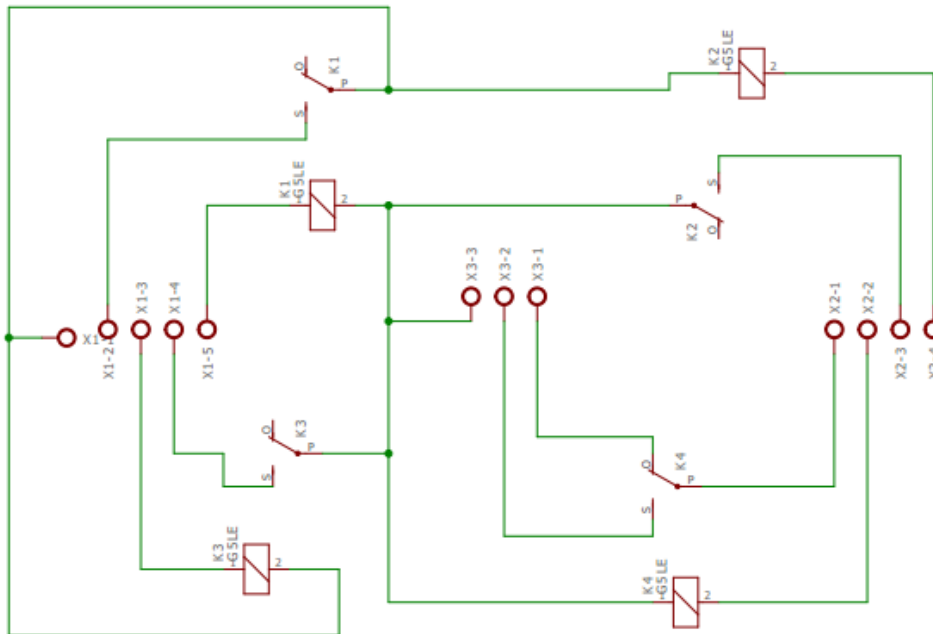


Fig. 3.90 Diseño de los relays de los circuitos alimentados por 24v.

Fuente: Propia

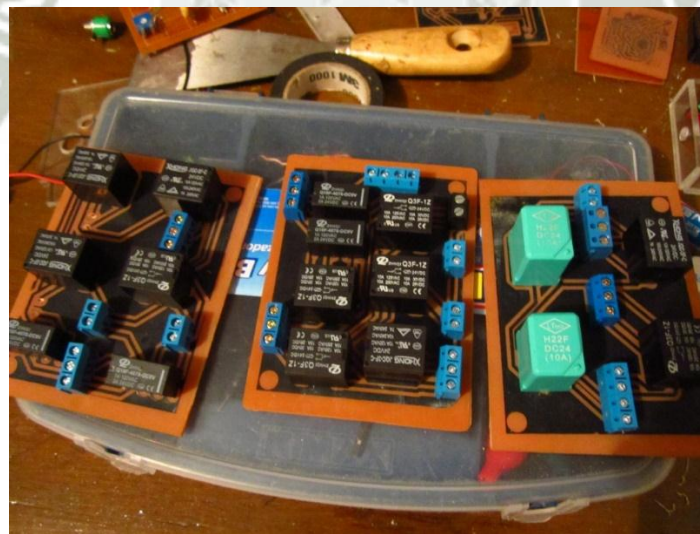


Fig. 3.91 Relays foto de las 3 placas terminadas

Fuente: Propia

Aquí tenemos las placas ya quemadas y soldadas con los Relay respectivos y las borneras.

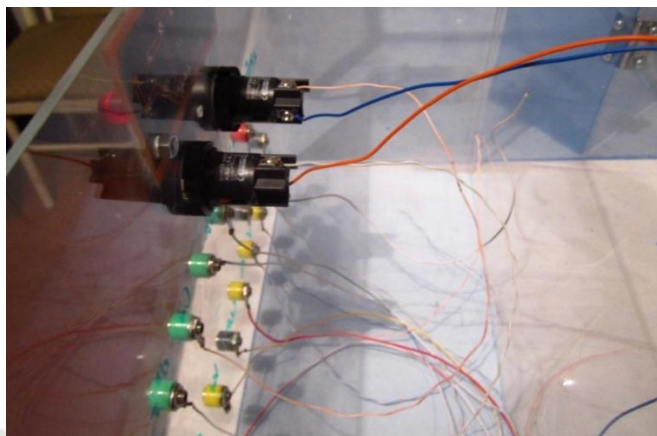


Fig. 3.92 Armado de los terminales circulares del módulo de nivel

Fuente: Propia

Armamos también el módulo de control de nivel tanque con los respectivos terminales circulares, los pulsadores de INICIO y STOP y la placa de luces quemada.

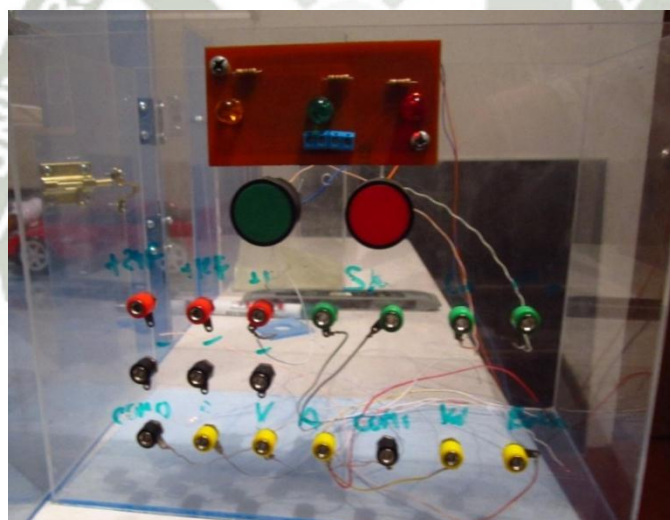


Fig. 3.93 Foto delantera del módulo de nivel

Fuente: Propia

Foto delantera del módulo de nivel parcialmente terminado antes de colocar los Relay.



Fig. 3.94 Todas las placas juntas antes del montaje

Fuente: Propia

Foto de las placas quemadas y terminadas justo antes de ser cableadas.



Fig. 3.95 Switch elegido para la conexión de la red interna

Fuente: Propia

Elegimos este Switch por tener 5 entradas tipo RJ-45 para Ethernet, ya que el HMI sólo tiene una entrada para Ethernet y necesitamos controlar dos esclavos.



Fig. 3.96 Vista exterior del módulo central de control

Fuente: Propia

También elegimos comprar éste módulo para utilizarlo como el centro de control de todo nuestro proyecto, y sobre éste nosotros lo acondicionaremos y armaremos para nuestro proyecto.



Fig. 3.97 Vista interior del módulo central de control

Fuente: Propia

La vista interior del módulo central se ve de ésta manera con dos carriles para insertarle los PLC y la llave general.



Fig. 3.98 Tabla para perforar las entradas en el módulo central

Fuente: Propia

Para perforar la tapa del módulo y adecuarlo a nuestras necesidades se armó una tabla en papel la cual se cuadriculó. Y Cada esquina representa a un orificio que deberá ser taladrado, luego se fijó a la tapa y se martilló para marcarlo para posteriormente taladrarlo.

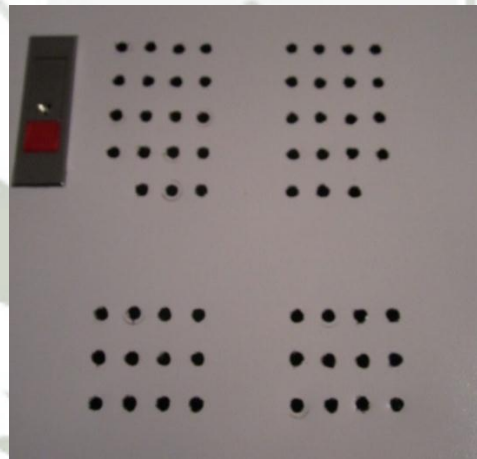


Fig. 3.99 Módulo central ya perforado.

Fuente: Propia

Aquí tenemos ya la disposición que queremos para la tapa de nuestro modulo central. Se separó en 4 partes: La de arriba a la derecha son las entradas del primer PLC, la de arriba a la

izquierda son las entradas del segundo PLC, la de abajo a la derecha son las salidas del primer PLC, y la de abajo a la izquierda son las salidas del segundo PLC.

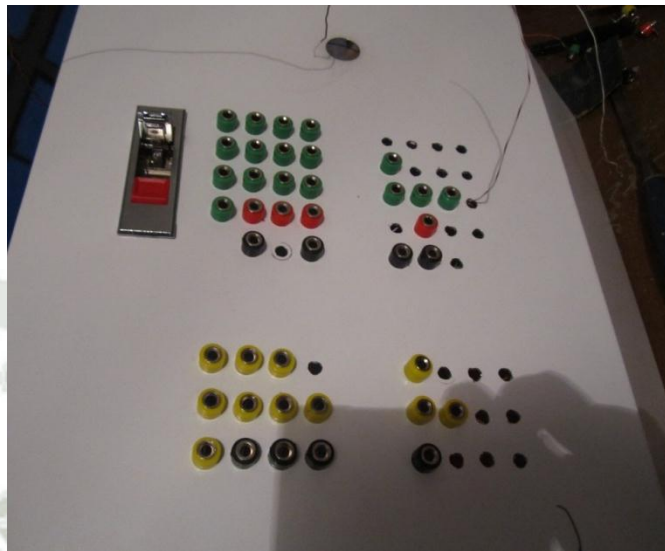


Fig. 3.100 Inserción de los terminales

Fuente: Propia

Luego de perforar empezamos a colocar los terminales circulares de los distintos colores representando entradas, salidas, comunes y voltajes positivos.

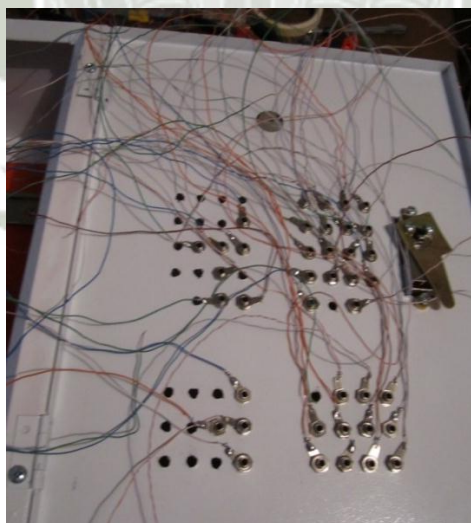


Fig. 3.101 Vista posterior de la puerta del modulo

Fuente: Propia

Por la parte posterior de la tapa soldamos a cada terminal circular, los cables para cablear a los PLC.

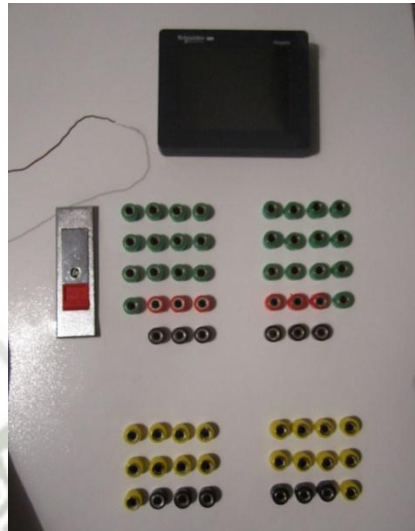


Fig. 3.102 Puerta del módulo terminada con el terminal HMI colocado

Fuente: Propia

Vista terminada de la tapa del módulo central con el HMI ya insertado.

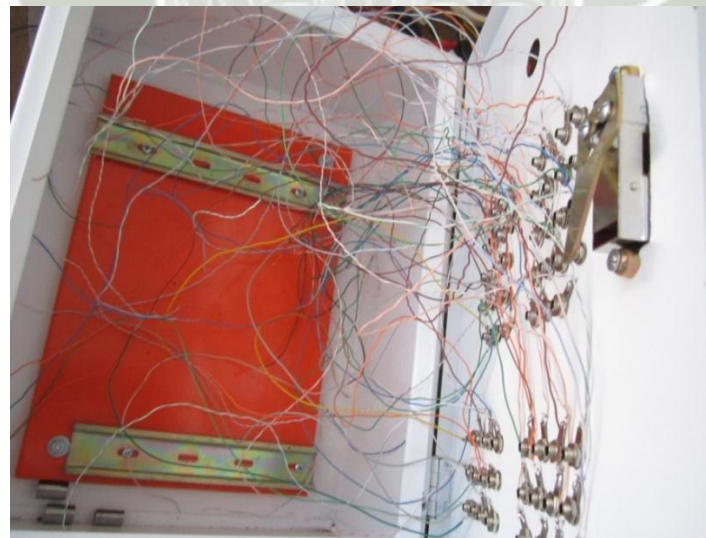


Fig. 3.103 Vista posterior de la puerta terminada

Fuente: Propia

Vista terminada de la parte posterior de la puerta con todos los cables listos para colocar a los PLC.



Fig. 3.104 Posicionamiento interno de los elementos necesarios

Fuente: Propia

Primera disposición interna que se pondrá, calculando el espacio necesario para los elementos que faltan.



Fig. 3.105 Cableado interno con los elementos ya fijos

Fuente: Propia

Luego de ya haber revisado el espacio necesario, se instalaron los cables de alimentación y de conexión Ethernet interna.

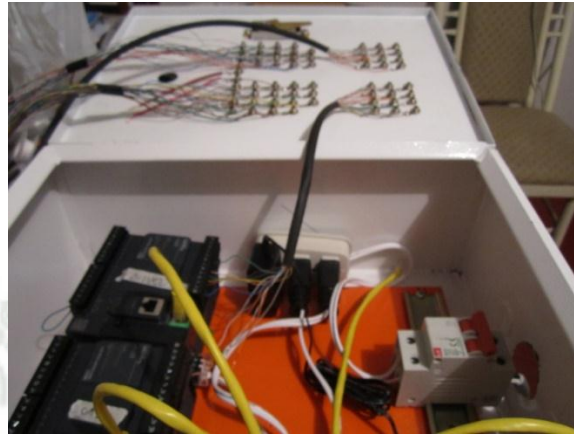


Fig. 3.106 Cableado de los PLC con los terminales circulares.

Fuente: Propia

Luego se cablearon los cables de los terminales de la puerta a cada una de las entradas y salidas de los PLC.

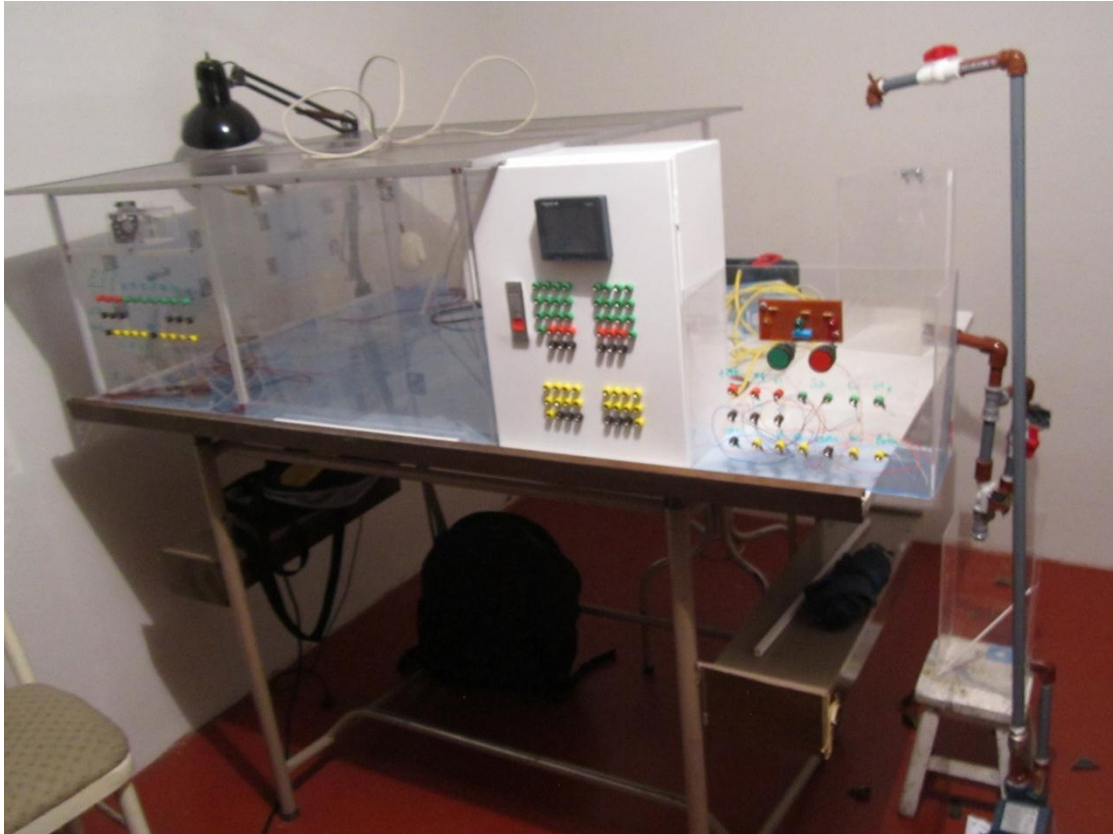


Fig. 3.107 Montaje completo de nuestro módulo.

Fuente: Propia

Y ésta fue la disposición final de nuestro módulo ya terminado y todo cableado. Ésta será la disposición en la cual se entregará al Laboratorio de Control y Automatización.

CAPITULO IV

PROGRAMACIÓN Y COMUNICACIÓN EN LA RED

4.1 Programación de los PLC

Luego de haber terminado el armado de nuestro modulo procederemos a programar los PLC y el HMI y luego a comunicarlos entre sí. Primero programaremos ambos PLC para en base a sus entradas y salidas utilizarlas como entradas para el HMI.

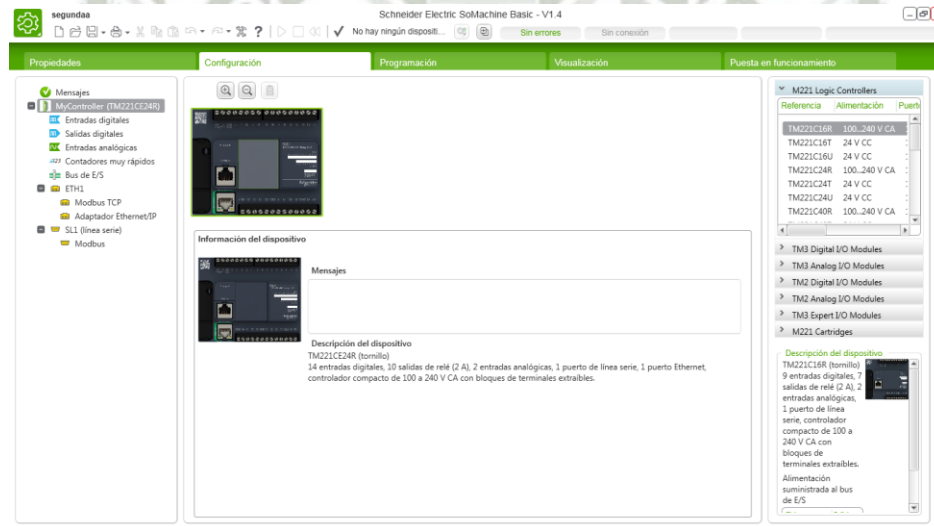


Fig. 4.1 Interfaz de SoMachine Basic con el PLC ya elegido

Fuente: Propia

El primer paso fue insertar el tipo de PLC comprado en el programa para iniciar su programación.

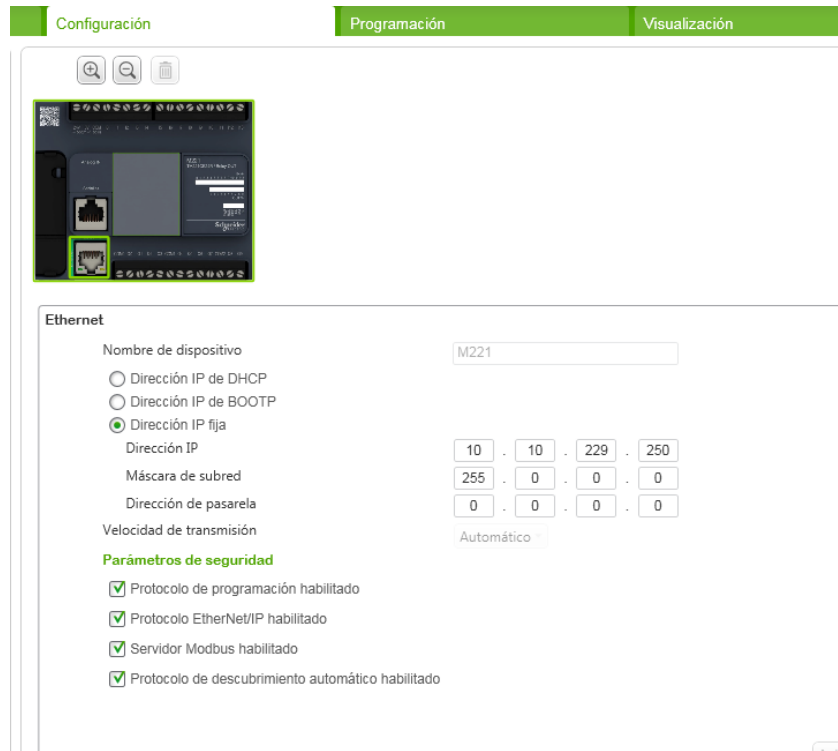


Fig. 4.2 Dirección IP asignada para Ethernet del primer PLC

Fuente: Propia

Asignamos una dirección IP de 10.10.229.250 y una máscara de subred de 255.0.0.0.

Luego se inició la programación en Ladder iniciando con designar las entradas y las salidas como se describe a continuación:

Módulo de Nivel de Tanque:

-Entradas:

- + ON %I0.0.
- + OFF %I0.1.
- + Medio %I0.2.
- + Alto %I0.3.

-Salidas:

- + Amarillo %Q0.0.

- + Verde %Q0.1.
- + Rojo %Q0.2.
- + Válvula Solenoide %Q0.3
- + Bomba %Q0.4.

Y la programación es la siguiente:

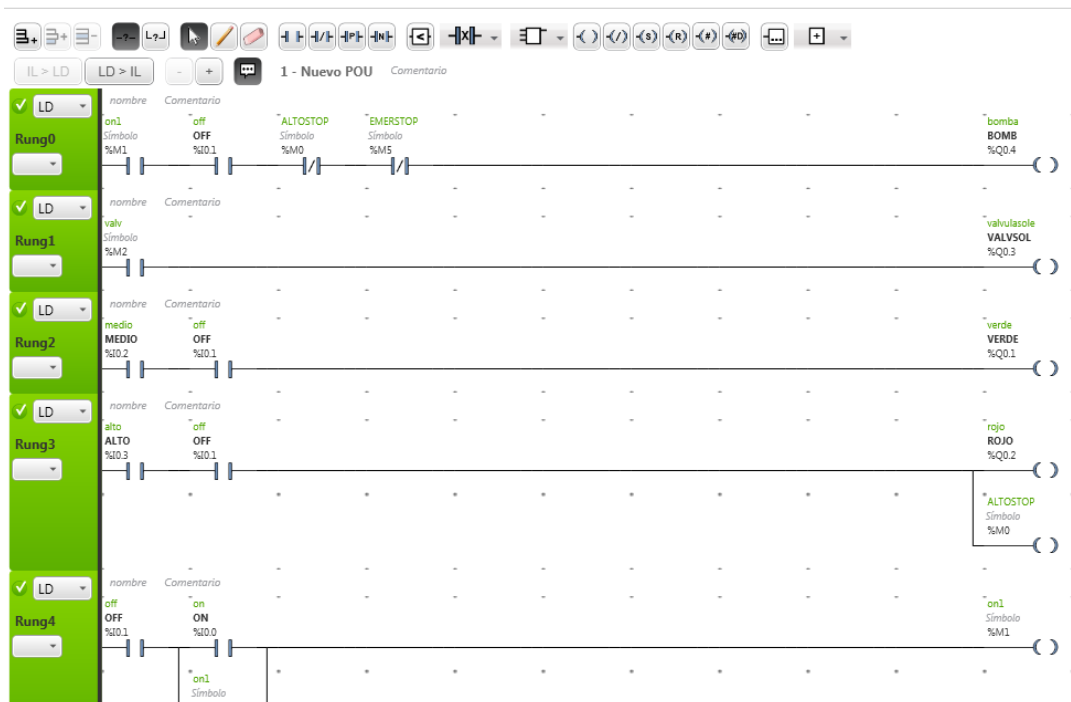


Fig. 4.3 Programa en Ladder para el módulo de nivel de tanque (1).

Fuente: Propia

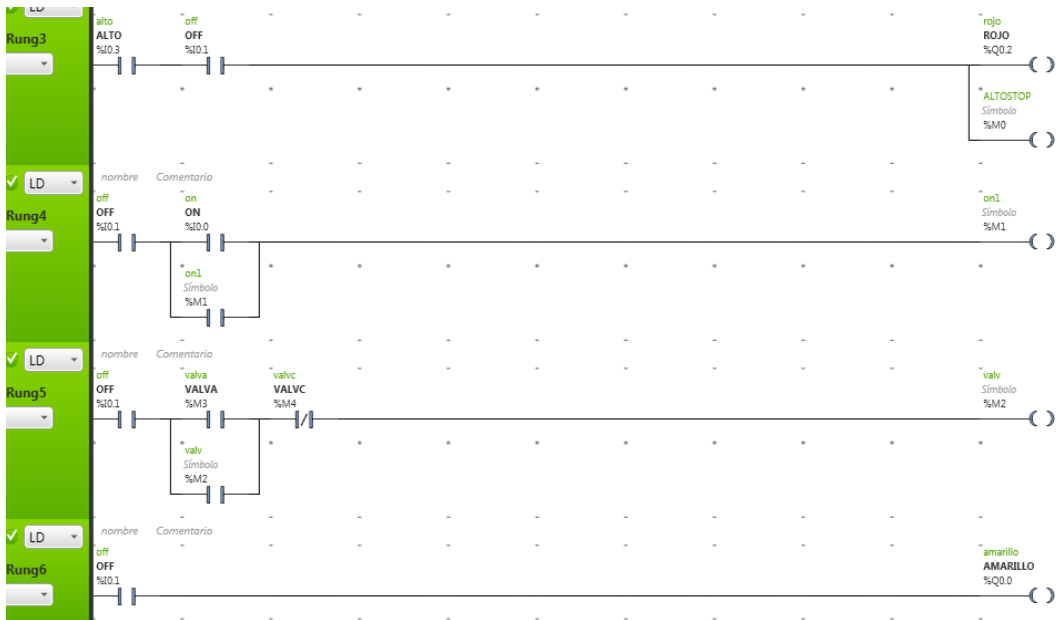


Fig. 4.4 Programa en Ladder para el módulo de nivel de tanque (2).

Fuente: Propia



Fig. 4.5 Dirección IP asignada para Ethernet del segunda PLC

Fuente: Propia

Le asignamos una dirección IP de 10.10.229.249 y una máscara de subred de 255.0.0.0.

Luego iniciamos la programación en Ladder iniciando con designar las entradas y las salidas como se describe a continuación:

Módulo Domótico:

- Entradas:

- + Interruptor %I0.0.
- + Sensor Movimiento %I0.1.
- + Voltaje Actual %IW0.0.
- + Detector Humo %I0.2.
- + Sensor Inductivo %I0.3.
- + Temperatura actual %IW0.1.

- Salidas:

- + Luces Internas %Q0.0.
- + Alarma de Intruso %Q0.1.
- + Luz Entrada %Q0.2.
- + Ventilador de Humo %Q0.3.
- + Cerrar Garaje %Q0.4.
- + Ventilador de Calefactor %Q0.5.
- + Focos Calefactor %Q0.6.
- + Abrir Garaje %Q0.7.

Y el programa en Ladder es el siguiente:

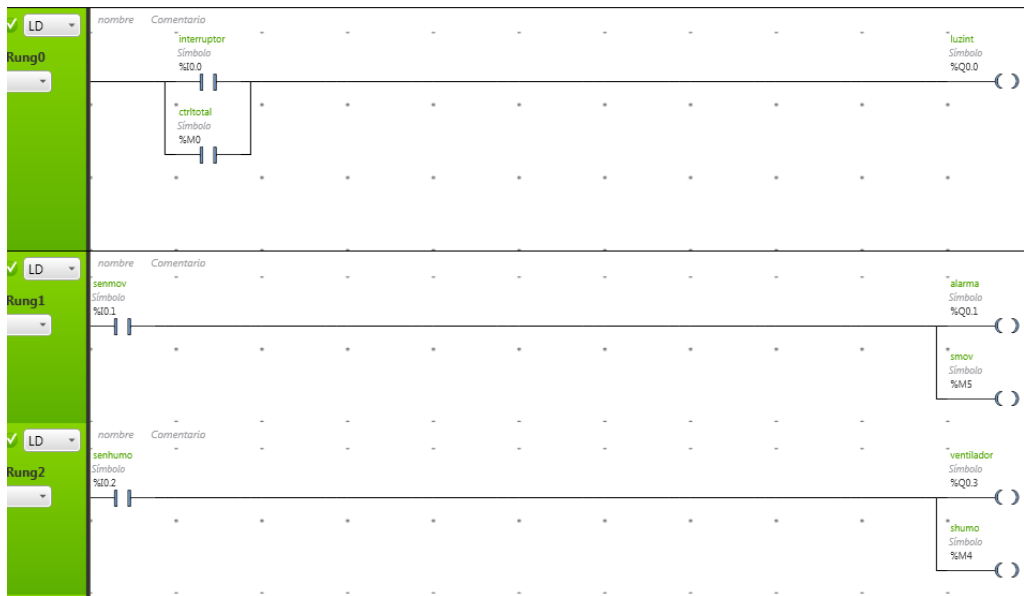


Fig. 4.6 Programa en Ladder para el módulo de domótica (1).

Fuente: Propia

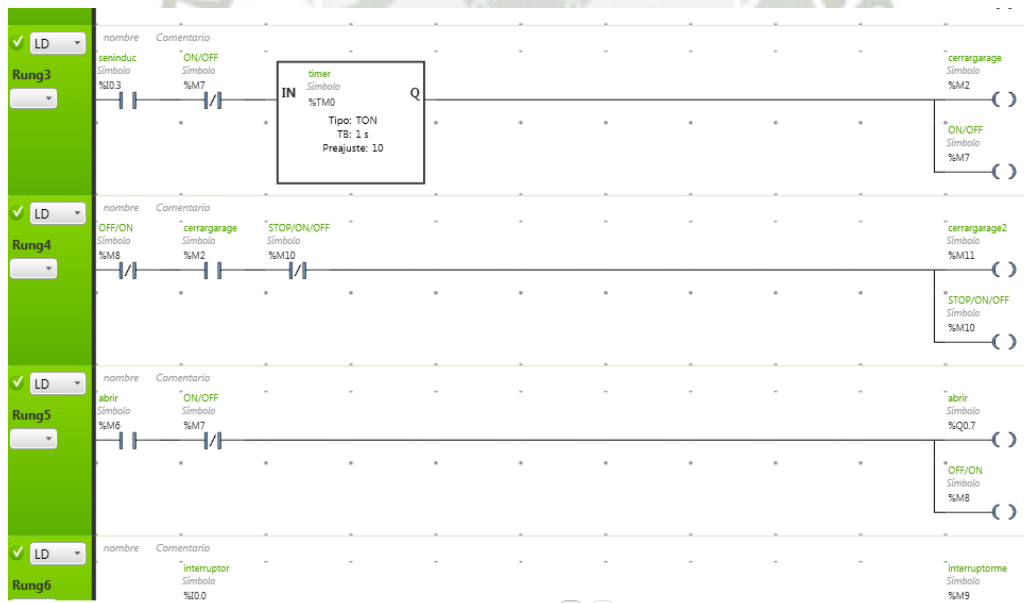


Fig. 4.7 Programa en Ladder para el módulo de domótica (2).

Fuente: Propia

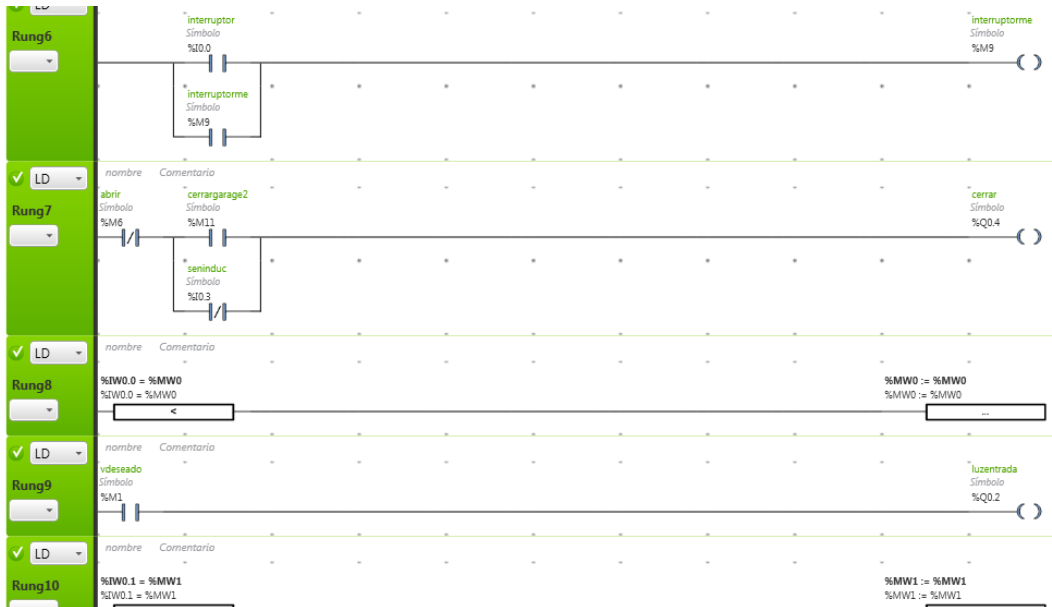


Fig. 4.8 Programa en Ladder para el módulo de domótica (3).

Fuente: Propia

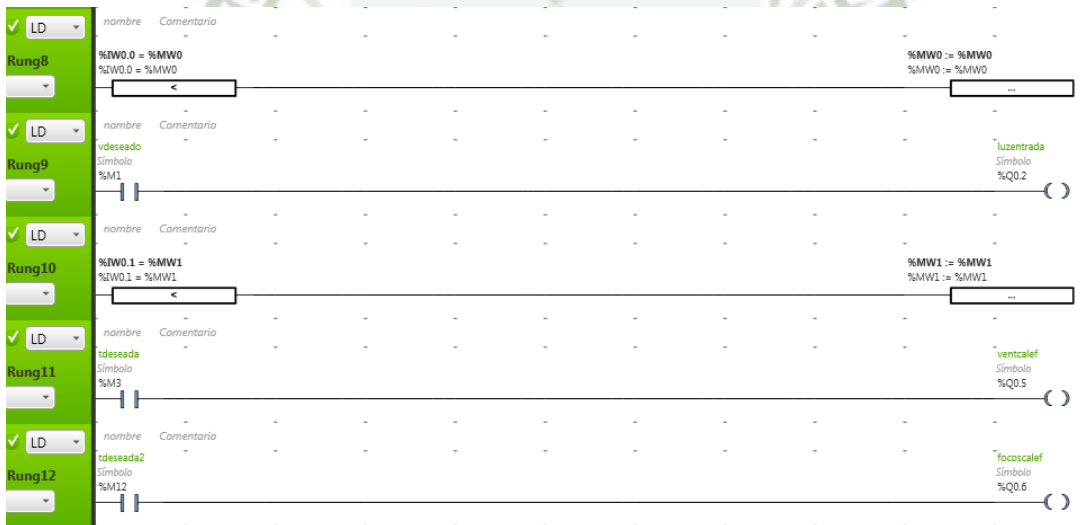


Fig. 4.9 Programa en Ladder para el módulo de domótica (4).

Fuente: Propia

Para la descarga entre la PC y el PLC utilizamos un cable de Ethernet con terminal RJ-45 puesto que nuestro postulado de proyecto necesita este conector.



Fig. 4.10 Terminal RJ-45

Fuente: Propia

Luego de haber cargado el programa al PLC, procedimos a probar las entradas y salidas con la fuente interna del mismo controlador.



Fig. 4.11 PLC ya cargado con el programa detectando entradas y salidas.

Fuente: Propia

4.2 Programación del terminal HMI

Para el Vijeo Designer es necesario primero inicializar el terminal HMI, para ello debemos correr el Vijeo Designer Runtime. Debemos asignarle una dirección IP, en el mismo HMI al prenderlo aparecerá una dirección IP por defecto, se puede tomar ésta o cambiarla por la que se requiera. Nosotros asignamos 10.10.229.251 y de Máscara de subred 255.0.0.0.

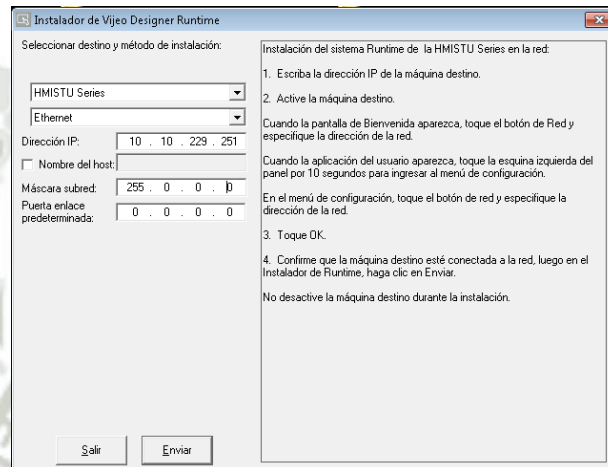


Fig. 4.12 Inicialización de la pantalla de Vijeo Designer.

Fuente: Propia

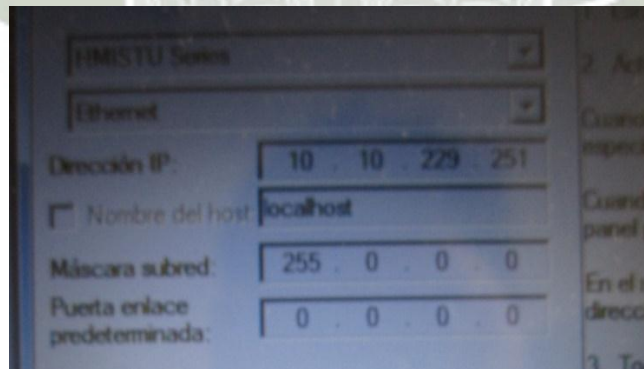


Fig. 4.13 Valores IP asignados para el terminal HMI.

Fuente: Propia

Luego de Enviarlo en el terminal HMI apareció la siguiente pantalla que significa que está conectado también a través de Ethernet.

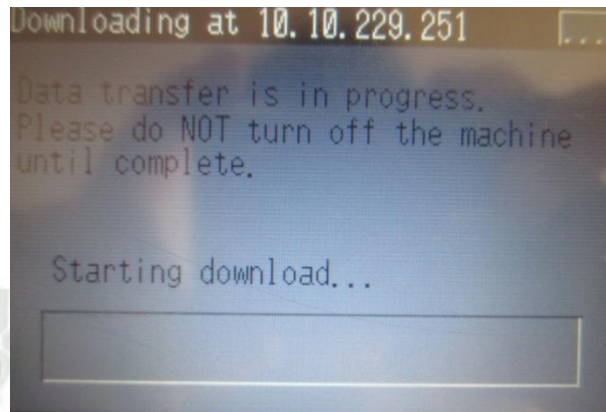


Fig. 4.14 Respuesta del terminal para la inicialización.

Fuente: Propia

En primer lugar identificamos la Interfaz Gráfica del Vijeo Designer, será el software que utilizaremos para programar el HMI.

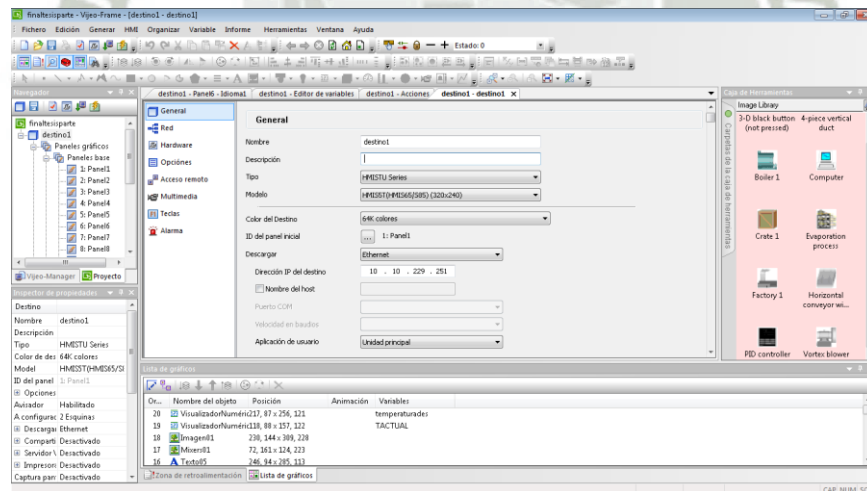


Fig. 4.15 Interfaz de Vijeo Designer para el diseño del SCADA.

Fuente: Propia

Le asignamos la misma dirección IP para el destino que obviamente es el terminal HMI comprado y ya inicializado.

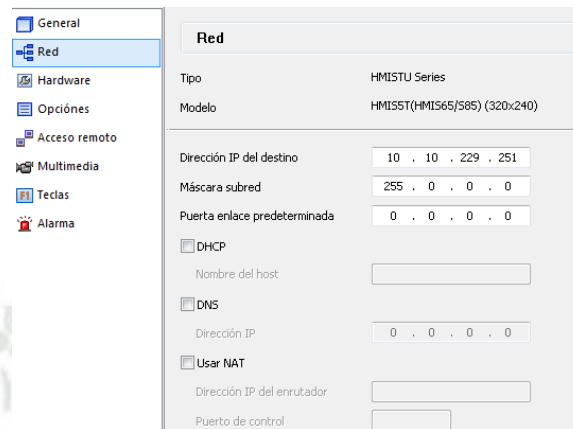


Fig. 4.16 Asignando la IP del terminal para la descarga de datos.

Fuente: Propia

Luego iniciamos el diseño del SCADA, y en las figuras posteriores se presentan los paneles que diseñamos para el control de nuestro módulo.



Fig. 4.17 Panel de entrada.

Fuente: Propia

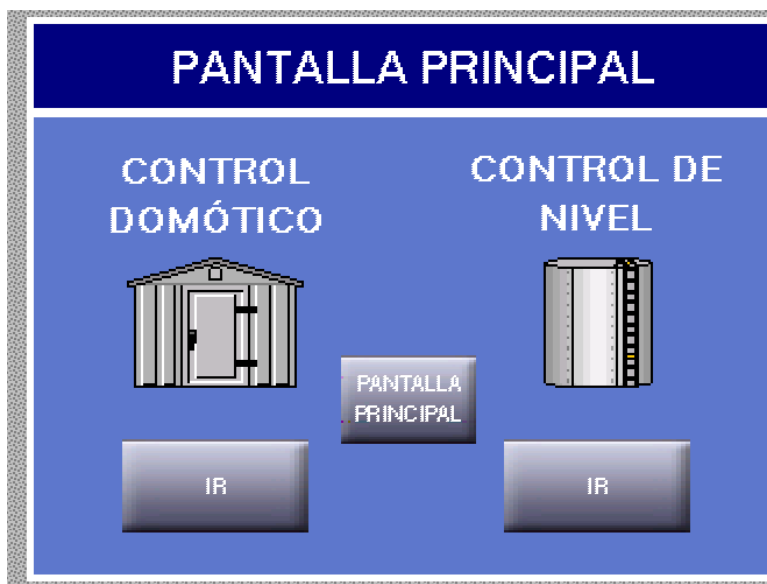


Fig. 4.18 Panel principal.

Fuente: Propia



Fig. 4.19 Panel de control domótico.

Fuente: Propia

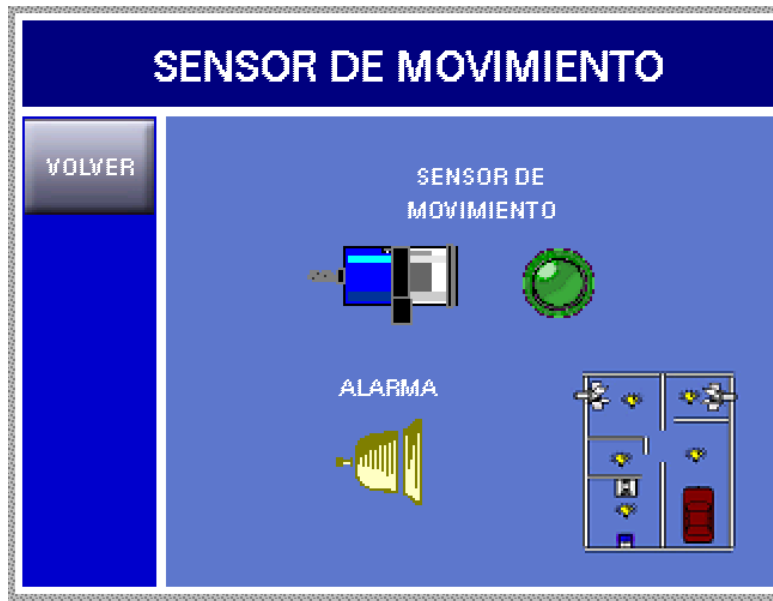


Fig. 4.20 Panel de control del sensor de movimiento.

Fuente: Propia



Fig. 4.21 Panel de control del sensor de humo.

Fuente: Propia

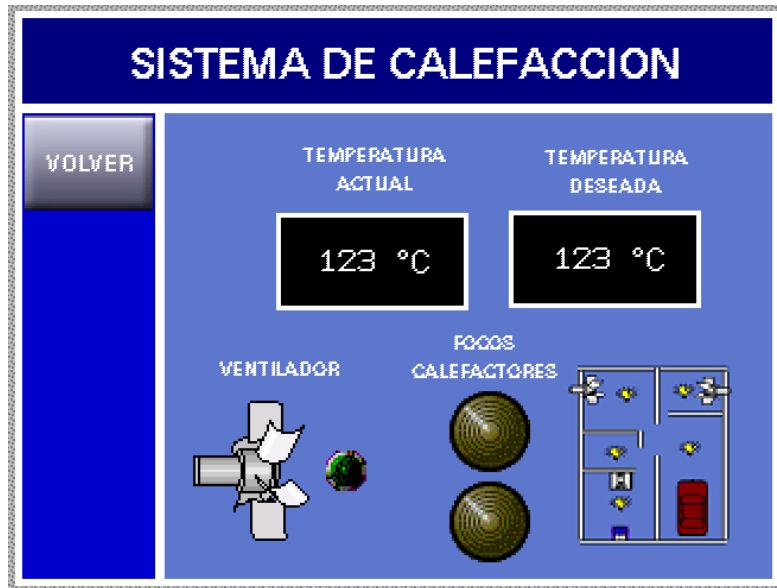


Fig. 4.22 Panel de control del sistema de calefacción.

Fuente: Propia



Fig. 4.23 Panel de control del sistema de alumbrado.

Fuente: Propia



Fig. 4.24 Panel de control de la puerta de garaje.

Fuente: Propia



Fig. 4.25 Panel de control de nivel de tanque.

Fuente: Propia

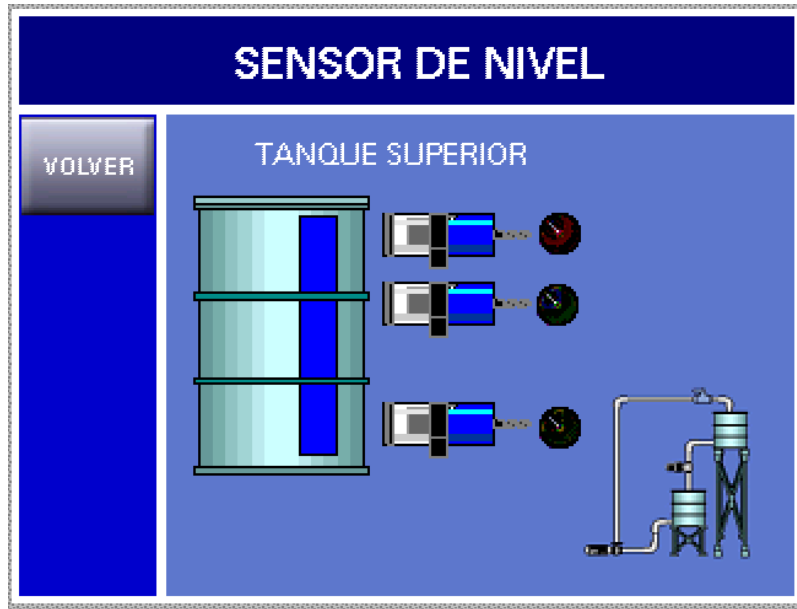


Fig. 4.26 Panel de control de llenado y vaciado de la simulación de un tanque de agua.

Fuente: Propia

Para que las animaciones de relleno y valores analógicos funcionen sin errores es necesario incluirles acciones que ayuden a que no hayan errores en el funcionamiento. Por lo cual decidimos programar en un Script en lenguaje C++ cada animación en la sección “Acciones”.

Acciones				
	Disparador	Propiedad	Enclavamiento	Acciones
1	Periódica	Repetir cada 0.1 seg.	ON1	subir
2	Periódica	Repetir cada 0.1 seg.	VALVS	vaciar
3	Periódica	Repetir cada 1 seg.	temperaturades	temperatura
4	Periódica	Repetir cada 1 seg.	voltajedes	luzentrada

Fig. 4.27 Ventana de acciones

Fuente: Propia

Posteriormente pusimos cada uno de los Script que utilizamos en la creación de acciones para las animaciones.

-SCRIPT PARA EL LLENADO DE TANQUE:

```
//-----
//Script creado el: Apr 20, 2016
//
// Descripción:
//
//-----
// Sustituir esta línea por su script

int subir;
subir=tanque.getIntValue();
if(subir<100)
{
    subir++;
}
else
{
    subir=100;
}
tanque.write(subir);
```

-SCRIPT PARA VACIAR EL TANQUE:

```
//-----  
//Script creado el : Apr 20, 2016  
//-----  
int vaciar;  
vaciar=tanque.getIntValue();  
if(vaciar>0)  
{  
    if(vaciar<10)  
    {  
        ON1.write(true);  
    }  
    else  
    {  
        ON1.write(false);  
    }  
    vaciar--;  
}  
else  
{  
    vaciar=0;  
}  
tanque.write(vaciar);
```

-SCRIPT PARA EL SENSOR DE CALEFACCIÓN:

```
//-----  
//Script creado el : Apr 20, 2016  
  
int tempd;  
tempd = temperaturades.getIntValue();  
  
int tempa;  
tempa=TACTUAL.getIntValue();  
  
if (tempd==(tempa-2))  
{  
  
    FOCOSCALEF.write(false);  
  
    VENTICALEF.write(false);  
  
}
```

```
}  
  
if (tempd==(tempa-1))  
{  
  
    FOCOSCALEF.write(false);  
  
    VENTICALEF.write(false);  
  
}  
  
if (tempd==(tempa+2))  
{  
  
    FOCOSCALEF.write(false);  
  
    VENTICALEF.write(false);  
  
}  
  
if (tempd==(tempa+1))  
{  
  
    FOCOSCALEF.write(false);  
  
    VENTICALEF.write(false);  
  
}  
  
if (tempd==tempa)  
{  
  
    FOCOSCALEF.write(false);  
  
    VENTICALEF.write(false);  
  
}  
  
else  
{
```

```
if (tempd>=(tempa+3))  
{  
    FOCOSCALEF.write(true);  
    VENTICALEF.write(false);  
}
```

```
if (tempd<=(tempa-3))  
{  
    VENTICALEF.write(true);  
    FOCOSCALEF.write(false);  
}  
}
```

-SCRIPT PARA EL SISTEMA DE ILUMINACIÓN:

```
//-----  
//Script creado el : Apr 20, 2016  
//  
// Descripción :  
//  
//-----  
  
// Sustituir esta línea por su script
```

```
int voltd;  
  
voltd = voltajedes.getIntValue();  
  
int volta;  
  
volta=VACTUAL.getIntValue();  
  
if (volta>=(voltd+0.5))  
{  
    LUZENTRADA.write(true);  
}  
else  
{  
    LUZENTRADA.write(false);  
}
```

Ésta es la sección “Variables”. Le asignamos todas las variables necesarias para el funcionamiento de todos los botones, recuadros, imágenes, objetos y luces piloto de nuestro SCADA. Definiendo el tipo de variables y el origen de las mismas.

	Nombre	Tipo de datos	Origen de datos	Grupo de escan...	Dirección de di...	Grupo de alarm...	Grupo de regist...
1	ABRIR	BOOL	Externo	EquipoModbus04	%M6	Desactivado	Ninguno
2	ABRIRTOTAL	BOOL	Externo	EquipoModbus04	%Q0.0.7	Desactivado	Ninguno
3	ALARMA	BOOL	Externo	EquipoModbus04	%Q0.0.1	Desactivado	Ninguno
4	ALTOSTOP	BOOL	Externo	EquipoModbus01	%M0	Desactivado	Ninguno
5	AMARILLO	BOOL	Externo	EquipoModbus01	%Q0.0.0	Desactivado	Ninguno
6	BOMBA	BOOL	Externo	EquipoModbus01	%Q0.0.4	Desactivado	Ninguno
7	CERRAR	BOOL	Externo	EquipoModbus04	%Q0.0.4	Desactivado	Ninguno
8	CTRLTOTAL	BOOL	Externo	EquipoModbus04	%M0	Desactivado	Ninguno
9	EMERSTOP	BOOL	Externo	EquipoModbus01	%M5	Desactivado	Ninguno
10	FOCOSSALIDACALEF	BOOL	Externo	EquipoModbus04	%M12	Desactivado	Ninguno
11	FOCOSSALIDACALEF	BOOL	Externo	EquipoModbus04	%Q0.0.6	Desactivado	Ninguno
12	LUZENTRA	BOOL	Externo	EquipoModbus04	%Q0.0.2	Desactivado	Ninguno
13	LUZENTRADA	BOOL	Externo	EquipoModbus04	%M1	Desactivado	Ninguno
14	LUZINTERNA	BOOL	Externo	EquipoModbus04	%Q0.0.0	Desactivado	Ninguno
15	ON1	BOOL	Externo	EquipoModbus01	%M1	Desactivado	Ninguno
16	ROJO	BOOL	Externo	EquipoModbus01	%Q0.0.2	Desactivado	Ninguno
17	SENINDTIMER	BOOL	Externo	EquipoModbus04	%M2	Desactivado	Ninguno
18	SHUMO	BOOL	Externo	EquipoModbus04	%M4	Desactivado	Ninguno
19	SMOV	BOOL	Externo	EquipoModbus04	%M5	Desactivado	Ninguno
20	TACTUAL	Integer	Externo	EquipoModbus04	%W0.0.1	Desactivado	Ninguno
21	tanque	Integer	Interno			Desactivado	Ninguno

Fig. 4.28 Ventana de variables internas y externas.

Fuente: Propia

Para la comunicación con los controladores hemos asignado variables externas para lo cual configuramos un nuevo controlador en la sección de Administrador de E/S. Para el primer PLC le asignamos la dirección 10.10.229.250 que previamente ya configuramos para éste.

The image shows a software configuration window titled "Configuración del equipo". It contains several sections for setting up a device:

- Dirección del Equipo:** Dirección IP is set to 10 . 10 . 229 . 250. ID de la Unidad is 255 / 255. There is a checkbox for "Conexión secundaria" and a field for "Copia de seguridad del IP" set to 0 . 0 . 0 . 0.
- Protocolo:** Protocolo de IP is set to TCP.
- IEC61131 Sintaxis:** This section is checked. "Modo de dirección" is set to "0-basado (Predetermina...".
- Variables:** "Orden palabra de doble palabra" is set to "Primera palabra alta". "ASCII Mostrar orden de byte" is set to "Primer byte bajo".
- Optimización de la Comunicación:** "Longitud de Trama preferida" is set to "Personalizar" with a value of 120 bytes.
- Administración del Diccionario de datos:** There is a checkbox for "Precargar Diccionario de datos para las modificaciones online".

At the bottom, there are buttons for "Aceptar", "Cancelar", and "Ayuda".

Fig. 4.29 Ventana para la configuración de la comunicación con el primer PLC.

Fuente: Propia

Al configurar el segundo PLC se deberá poner la dirección IP que previamente ya se asignó a dicho controlador. Para lo cual nosotros asignamos 10.10.229.249.

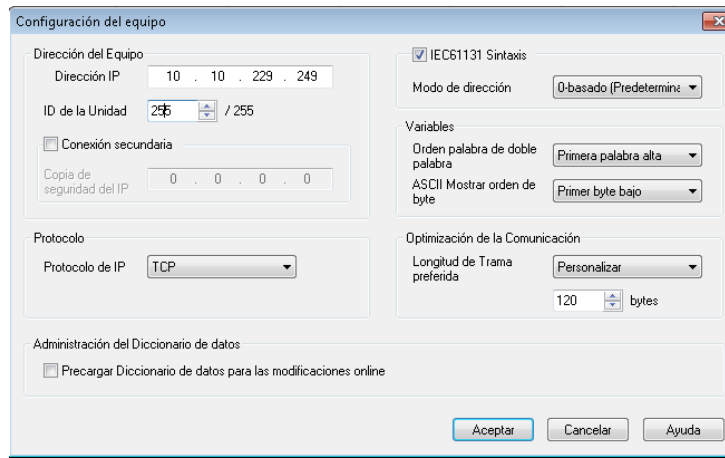


Fig. 4.30 Ventana para la configuración de la comunicación con el segundo PLC.

Fuente: Propia

Luego al descargarlo apareció la pantalla de descarga de información luego de validar la comunicación con el PC.

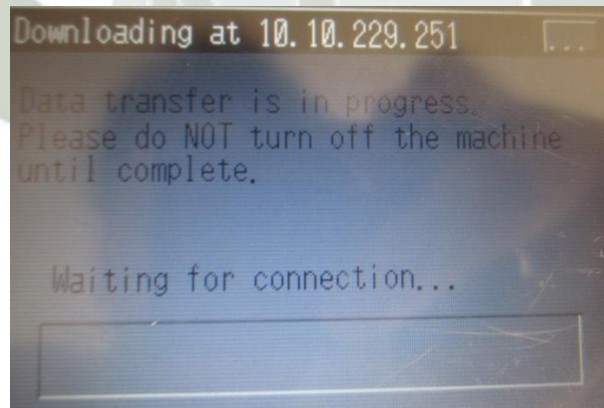


Fig. 4.31 Descarga del Vijeo Designer al terminal HMI Magelis.

Fuente: Propia

Pantalla de cargado al iniciarse el HMI.

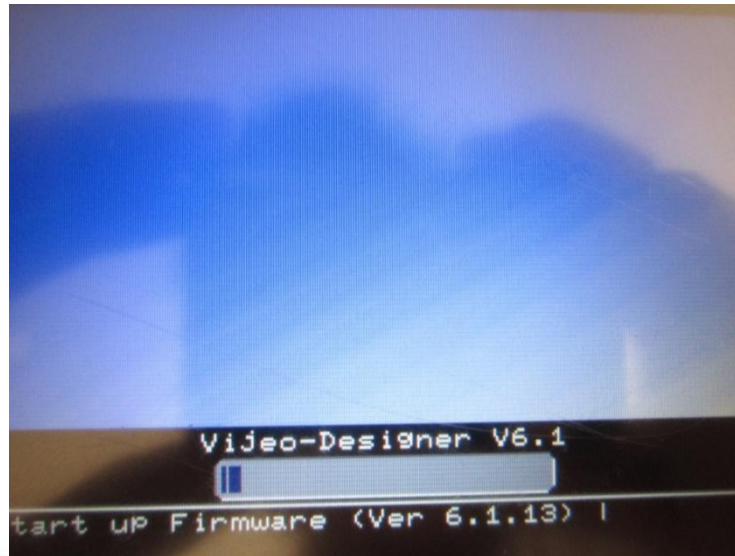


Fig. 4.32 Pantalla de encendido del terminal HMI Magelis.

Fuente: Propia

Finalmente probamos los dos PLC, el terminal HMI y la fuente que lo alimenta encendidos y conectados a través del Switch y listo para las pruebas finales.



Fig. 4.33 Pruebas de comunicación en la red.

Fuente: Propia

Y está listo el programa ya cargado para trabajar con los PLC y los sensores y actuadores del módulo.



Fig. 4.34 Programa ya cargado en el terminal HMI Magelis.

Fuente: Propia

CAPITULO V

PRUEBAS Y RESULTADOS FINALES

En este capítulo vamos a realizar las pruebas y veremos cómo responden a ellas nuestros dispositivos.

En primer lugar debemos tener nuestros programas ya en los PLC's y en el HMI, una vez funcionando los programas y fijados los IP's correspondientes a cada dispositivo debemos proceder realizar las conexiones de red con los cables Ethernet.



Fig. 5.1 HMI conectado a través del cable Ethernet

Fuente: Propia



Fig. 5.2 PLC's conectados a través del cable Ethernet

Fuente: Propia

Conectamos los 3 dispositivos mediante un Switch al cual también podríamos conectar nuestra PC y por lo tanto podríamos programar cualquiera de los dispositivos y monitorearlos estando en la misma red y conociendo sus números IP.

En caso de haber cualquier error de comunicación los dispositivos nos avisan de distintas formas, en el caso del HMI podemos observar en la Figura 5.3 como aparecen los íconos de error en las distintas entradas y salidas.



Fig. 5.3 Error de comunicación.

Fuente: Propia

Entonces, una vez que se arreglan los errores de comunicación y el programa empieza a funcionar correctamente, hay que empezar las pruebas verificando todas las variables del módulo y compararlas sus respectivas representaciones en el terminal HMI. Por ejemplo podemos observar en la Figura 5.4 que las luces de entrada en el módulo físico se encuentran apagadas, por lo tanto al buscar su representación en el terminal HMI en la pestaña de Control doméstico deberíamos encontrarla apagada y eso es lo que podemos observar en la Figura 5.5.

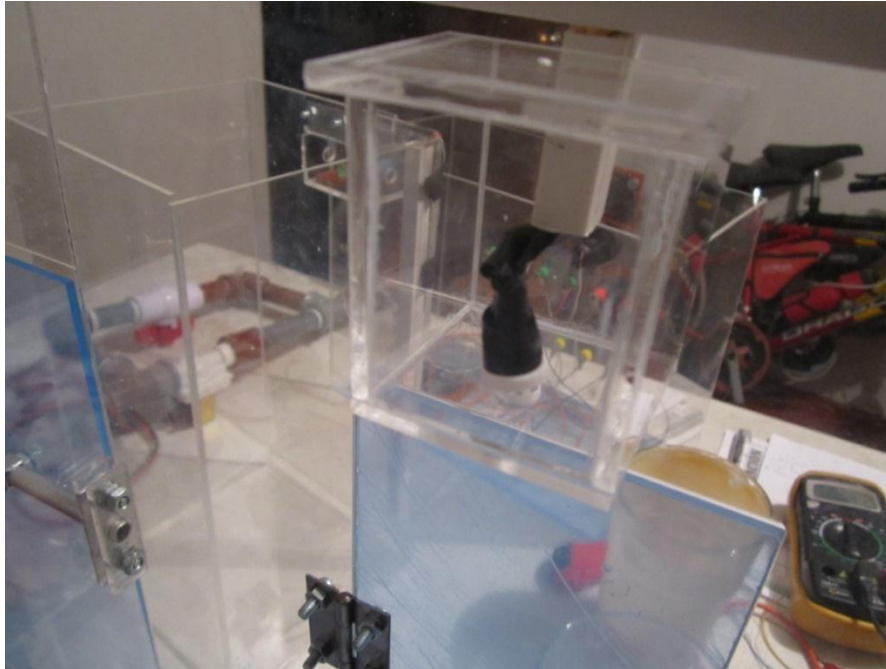


Fig. 5.4 Luces de Entrada apagadas

Fuente: Propia



Fig. 5.5 Activando las luces de entrada en el HMI

Fuente: Propia

Así mismo, esta misma variable debe corroborarse con la salida del PLC esclavo que es el intermediario entre el HMI y la luz del módulo, porque la coincidencia podría deberse a errores de conexión o pérdida de señal.

Una vez visto esto, procedemos a encender la luz de entrada desde nuestro terminal HMI e inmediatamente podemos escuchar y observar cómo se activa la salida correspondiente en el PLC esclavo correspondiente al módulo de domótica como se muestra en la Figura 5.6.

A su vez podemos observar la luz de entrada en el módulo encenderse como muestra la Figura 5.7.



Fig. 5.6 Respuesta del PLC a la activación de Luces de Entrada

Fuente: Propia

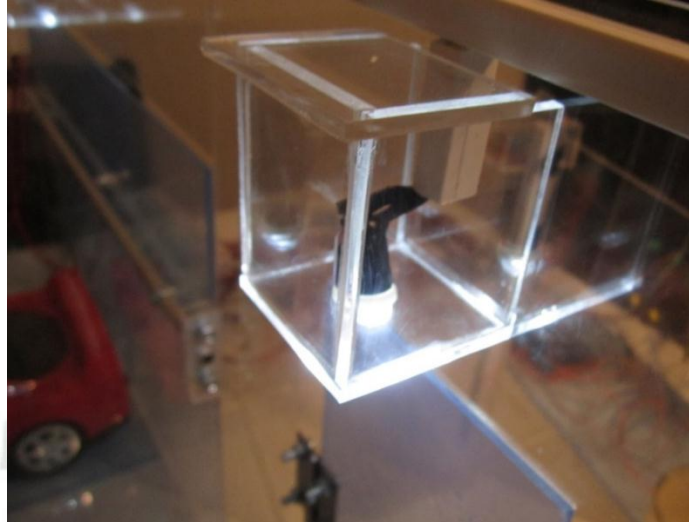


Fig. 5.7 Luces de Entrada encendidas

Fuente: Propia

Nuestra siguiente prueba será con las luces internas que como podemos observar están apagadas en un primer momento, y luego dando la orden desde el terminal HMI podemos encenderlas remotamente.



Fig. 5.8 Luces Internas apagadas

Fuente: Propia

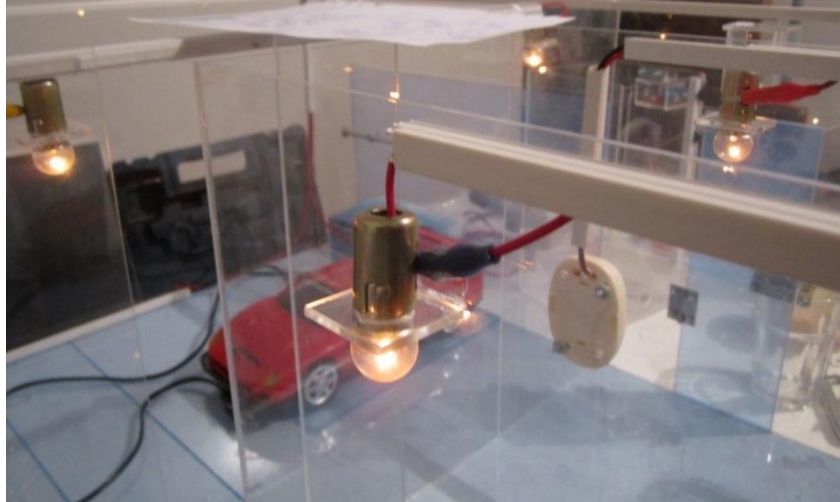


Fig. 5.9 Luces Internas encendidas

Fuente: Propia

Luego podemos empezar a probar otras partes del módulo, en este caso podremos probar nuestro sistema de detección de Humo, para el cual tenemos el sensor que implementamos anteriormente y como actuador un ventilador pequeño.

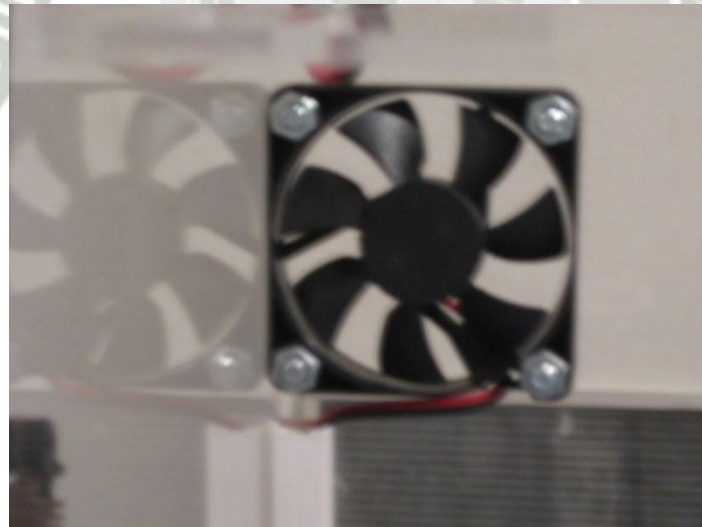


Fig. 5.10 Ventilador de Humo apagado.

Fuente: Propia

Podemos ver como en reposo nuestro ventilador se encuentra apagado, pero al crear una perturbación simulando humo en el sensor, manda una señal que activa la entrada correspondiente en el PLC y gracias al programa de este, se activa la salida correspondiente al ventilador como podemos observar en la Figura 5.11.

Inmediatamente, el PLC comunica a nuestra terminal HMI este suceso y los cambios de estado tanto del sensor como del actuador y podemos observar como conmutan sus respectivas variables y sus representaciones en la interfaz gráfica en la Figura 5.12. Y finalmente en la Figura 5.13 ya podemos observar el ventilador físico encendido.



Fig. 5.11 Respuesta del PLC a la activación del sensor

Fuente: Propia



Fig. 5.12 Respuesta del HMI a la activación del sensor

Fuente: Propia



Fig. 5.13 Ventilador de Humo encendido.

Fuente: Propia

Nuestra siguiente prueba es el sistema de calefacción y aire acondicionado, es un sistema simulado sencillo, para probar el sistema de control distribuido. En este sistema se tienen como actuadores un par de focos de alta potencia como calefactor y otro ventilador para refrescar el

aire. Y como sensor se tiene un LM35, un sensor analógico de temperatura que envía señales de voltaje. Podemos observar que en estado de reposo a una temperatura promedio confortable, ambos actuadores se encuentran apagados, pero el sensor siempre estará en funcionamiento. En el terminal HMI podemos colocar cual será esa temperatura que deseamos en nuestro ambiente y podemos observar la temperatura que lee nuestro sensor.



Fig. 5.14 Focos de calefactor apagados

Fuente: Propia

Si la temperatura baja considerablemente por debajo de nuestra temperatura deseada, podemos observar como el PLC correspondiente responde según la programación hecha en la Figura 5.15, y a su vez este comunica este problema al HMI que responde como podemos ver en la Figura 5.16 donde podemos ver e ingresar la temperatura que deseamos al igual que podemos solo ver la temperatura actual que indica el sensor LM35 y para poder corregir este problema se encienden los focos que funcionan como calefactores como observamos en la Figura 5.17.



Fig. 5.15 Respuesta del PLC a los focos de calefactor encendidos.

Fuente: Propia



Fig. 5.16 Respuesta del HMI al sensor LM35 con temperatura bajas

Fuente: Propia

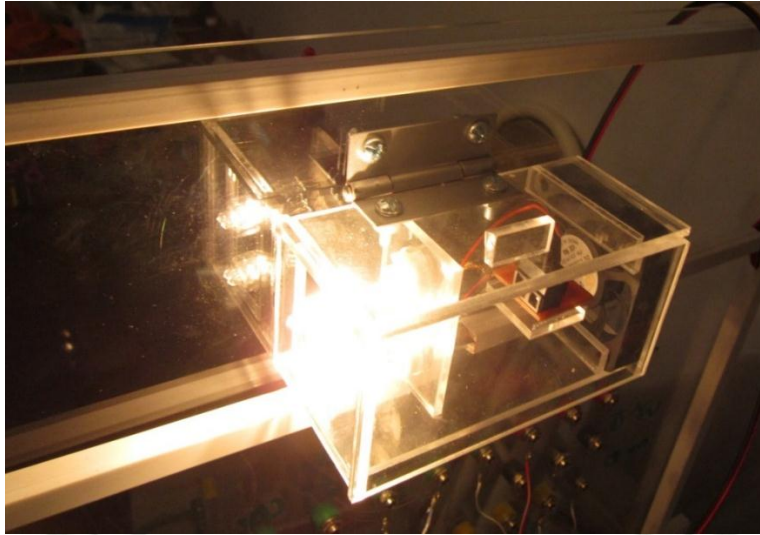


Fig. 5.17 Focos de calefactor encendidos.

Fuente: Propia

En caso de que la temperatura sea considerablemente mayor a la deseada lo que ocurre es lo contrario, se enciende el ventilador, y el comportamiento de los dispositivos de control podemos verlos en las Figuras 5.18, 5.19, 5.20 y 5.21.



Fig. 5.18 Ventilador de calefactor apagado.

Fuente: Propia



Fig. 5.19 Respuesta del PLC a la activación del ventilador de calefactor.

Fuente: Propia



Fig. 5.20 Respuesta del HMI al sensor LM35 con temperaturas altas.

Fuente: Propia

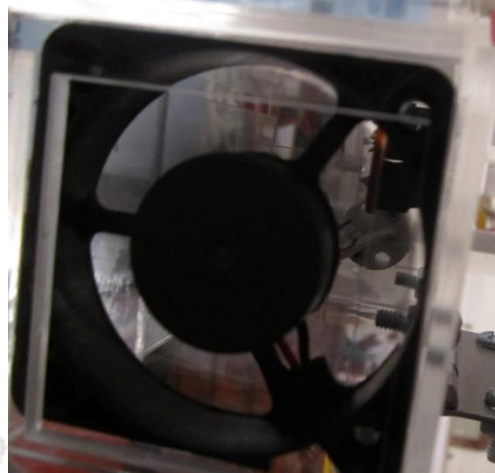


Fig. 5.21 Ventilador de calefacción encendido.

Fuente: Propia

El siguiente proceso de control domótico sería nuestra alarma de intrusos. Para el cual tenemos el sensor de presencia implementado anteriormente y como actuador una simple alarma que podría ser mejorada desde los PLC por ejemplo haciendo que se envíe un mensaje de texto al dueño.



Fig. 5.22 Respuesta del HMI a alarma de intruso apagada.

Fuente: Propia

En la Figura 5.22 podemos ver el estado de reposo del sensor y del actuador, pero en caso de que esta sea activada, podemos observar en las Figuras 5.23 y 5.24 como responden a esta perturbación, tanto el control HMI como el PLC.



Fig. 5.23 Respuesta del PLC a alarma de intruso encendida.

Fuente: Propia



Fig. 5.24 Respuesta del HMI a alarma de intruso encendida.

Fuente: Propia

Hasta aquí termina el trabajo de nuestro primer PLC esclavo que está encargado de controlar la parte domótica del módulo.

La siguiente parte es la de control de nivel que es controlada por otro PLC esclavo pero que mediante la comunicación Ethernet podrá monitorearse y controlarse remotamente desde nuestro terminal HMI

En la Figura 5.25 podemos observar la pantalla del control de nivel que se encuentra en el caso en el que la bomba está, pero la válvula solenoide está apagada.



Fig. 5.25 Respuesta del HMI a la activación de la bomba.

Fuente: Propia

Ahora procederemos a drenar el tanque, para esto es necesario apagar la bomba, que para evitar que nuestros tanques se rebalsen, un sensor de nivel indica cuando se ha llegado a un nivel máximo. Y luego debemos encender la válvula solenoide. Este estado podemos observarlos en la Figura 5.26.



Fig. 5.26 Luego del llenado del tanque, la respuesta del HMI a la activación de la válvula solenoide.

Fuente: Propia

Y en otra pantalla, podemos observar en tiempo real como el tanque es vaciado (Figura 5.27)

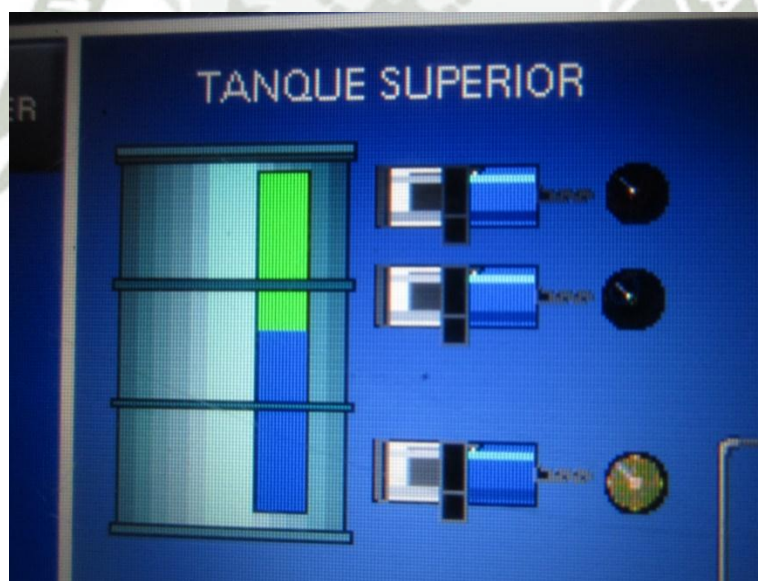


Fig. 5.27 Simulación en el HMI del drenado del tanque.

Fuente: Propia

Finalmente podemos observar la pantalla de inicio de nuestro programa, esta es la pantalla que se ve al encender nuestro módulo donde podemos encontrar los datos de los autores y de la institución para la cual fue diseñada. (Figura 5.28)



Fig. 5.28 Presentación final lista.

Fuente: Propia

CONCLUSIONES

1. Se logró diseñar e implementar el módulo didáctico de un sistema de control distribuido con un terminal de dialogo HMI como maestro y dos controladores lógico programables como esclavos comunicados a través de una red Ethernet.
2. Para una mejor comprensión en las variables se logró diseñar e implementar un sistema tipo SCADA para la interfaz humano-maquina (HMI), que permite modificar y manipular las variables de manera sencilla y entendible sin necesidad de conocer sobre programación, siendo éste aporte de mucha necesidad en industria puesto que puede implementarse para ser controlados desde la comodidad de un terminal HMI instalado en nuestra oficina y poder manipular cualquier variable sin la necesidad de estar presente en la planta.
3. Se realizó una buena comunicación Ethernet en nuestro módulo debido a su bajo flujo de información pero en otras industrias el uso de switches introduce un retado ya considerable. Por lo que en nuestro módulo solo se prueba el correcto funcionamiento de una red más no el óptimo.
4. Se diseñó y se implementó correctamente el módulo físico de control al igual que los dos sub módulos de prueba donde encontramos las maquetas de nuestros procesos simulados.

5. Se pudo diseñar las respectivas guías de práctica en base al formato usado en la Universidad Católica Santa María de Arequipa y tienen como objetivo ayudar a los alumnos a manejar el módulo de control distribuido.

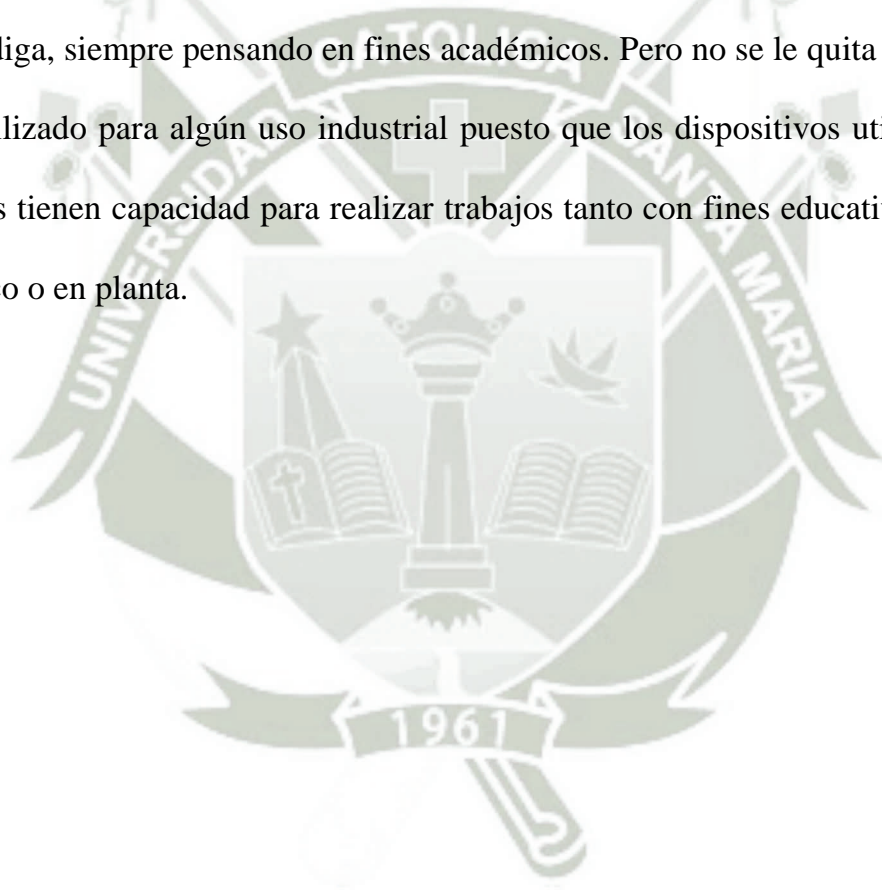


RECOMENDACIONES

1. Dejamos como posibilidad de ampliación y como recomendación agregar 1 PLC como esclavo al módulo y probar el funcionamiento del mismo al igual que no disminuirá la velocidad de respuesta al añadirse uno más a la red.
2. Se puede colocar un router inalámbrico para poder crear una red inalámbrica donde se podría monitorear y hasta controlar variables desde un dispositivo inteligente como un Smartphone o una Tablet siempre y cuando estén conectadas a la red. Así mismo podría agregarse una conexión de Internet al router lo que permitiría monitorear y hasta controlar variables desde casi cualquier lugar del mundo siempre y cuando se tenga un dispositivo conectado a internet. Hasta incluso se podría desarrollar una app específica para éste uso.
3. En cuanto al terminal HMI es crucial alimentarlo con 24v DC con un margen de +4v y -4v. Si el voltaje de alimentación es mayor o menor que estos valores el equipo se apagará o puede dañarse. Como norma de seguridad si se tiene una fuente de 24v AC es obligatorio colocarle un diodo de protección.

4. Se recomienda utilizar fusibles en las salidas del PLC sino se aislarán las mismas. Al ser tipo Relay enviarán un único pulso que de no ser tomado con precisión por el circuito siguiente pueden dañar el equipo. Y como recomendación deberán ser fusibles de 1 o 2 Amperios.

5. Como módulo funciona bastante bien, pero como sugerencia para guías de práctica a mis compañeros se podrá utilizar el módulo central para programar y diseñar en base a lo que el tutor diga, siempre pensando en fines académicos. Pero no se le quita valor en caso pueda ser utilizado para algún uso industrial puesto que los dispositivos utilizados en la presente tesis tienen capacidad para realizar trabajos tanto con fines educativos, como en uso doméstico o en planta.



BIBLIOGRAFIA

Creus A. (2005), *Instrumentación Industrial 7ma Edición*, Madrid, España: S.A. Marcombo

Morales O. (2011), *Circuitos Electricos I*, Lima, Perú: Ciencias S.R.Ltda.

Perez M. (2003) *Instrumentación Electrónica*, Madrid, España: Ediciones Paraninfo

Hart D. (2001) *Electrónica de Potencia*, Madrid, España: Prentice Hall

Tocci R. (2007) *Sistemas Digitales. Principios y Aplicaciones 10ma Edición*, Mexico DF,
Mexico: Pearson

Mandado E. (2010) *Autómatas Programables y Sistemas de Automatización*, Barcelona, España:
S.A. Marcombo

Ambardar A. (2002) *Procesamiento de señales analogicas y digitales 2da Edición*, Mexico DF,
Mexico:Thomson Learning

Romero E. (2000). *Redes de Comunicaciones Industriales*. Recuperado de
<http://neutron.ing.ucv.ve/revista-e/No4/RCI.html>

Universidad de Valencia (s. f.). *Redes de Comunicación Industriales*. Recuperado de
http://www.uv.es/rosado/courses/sid/Capitulo3_rev0.pdf

Universidad de Oviedo (s. f.). *Comunicaciones Industriales*. Recuperado de
<http://isa.uniovi.es/docencia/iea/teoria/comunicacionesindustrialesdocumento.pdf>

De La Cruz L. (2012, mayo 7). De: Ethernet, Aplicaciones en la Industria. Recuperado de
<http://uhu.es/antonio.barragan/content/7-aplicaciones-industria>

Emerson Process Management (2002). *Comprendiendo Ethernet*. Recuperado de
http://www2.emersonprocess.com/siteadmincenter/PM%20Central%20Web%20Documents/EngSch-Buses_102_es.pdf

Romero D. (2005, octubre). *Introducción a Ethernet Industrial*. Ponencia presentada para la IEEE en coordinación con Schneider Electric en Argentina. Recuperado de
<http://www.ieee.org.ar/downloads/romero-eth-ind.pdf>

Granier F. (2004). *La importancia de la red en la optimización de procesos*. Revista Electro industria. Recuperado de <http://www.emb.cl/electroindustria/articulo.mvc?xid=80&tip=7>

CREA (2012). *Planear y Construir borradores*. Recuperado de
http://sitios.ruv.itesm.mx/portales/crea/planear/como/planteamiento_tesis.htm

Schneider Electric (2014). *Controladores programables y terminales de dialogo*. Recuperado de http://www.schneiderelectric.com.pe/documents/local/lista_de_precios_2015/06_controladores_programables_y_terminales_de_dialogo.pdf

Villajulca J. (2012, noviembre 3). De: Control de nivel de agua en un tanque por medición por sensor de resistencia. Recuperado de <http://www.instrumentacionycontrol.net/cursos-libres/automatizacion/curso-de-plcs-avanzado/item/673-programación-analógica-en-plc-aplicación.html>

MatrikonOPC (s. f.). *Servidor MatrikonOPC para MODBUS*. Recuperado de <http://www.matrikonopc.es/drivers/opc-modbus.aspx>

Alejandro J. (2009, abril). *Sistema de control domótico de una vivienda*. (Tesis de Pregrado). Universidad Rovira y Virgili. Terragona, España.

Arias J. A. y Marulanda A. (2010, marzo). *Control y medida de nivel liquido por medio de un sensor de presión diferencial*. (Tesis de grado). Universidad Tecnológica de Pereira. Risaralda, Colombia.

Pérez R. (2013, julio). *Simulación y control mediante PLC de una vivienda* (Tesis de grado). Universidad de La Rioja. La Rioja, Argentina.

- Molano S. (2013, diciembre). *Medición y control de un sistema de nivel de tanques con PLC Allen Bradley*. Recuperado de <https://prezi.com/nqinzc3yltyg/medicion-y-control-de-un-sistema-de-nivel-de-tanques-con-pl/>
- Loxone (s. f.). *Smart Home*. Recuperado de <http://www.loxone.com/eses/smart-home/funcionalidades/persianas.html>
- ONI (2008). *Controlador Lógico Programable PLC*. Recuperado de http://www.oni.escuelas.edu.ar/2008/SANTA_FE/1376/INFORME%20FERIA%20DE%20CIENCIAS%20Y%20TECNOLOGÍA%202008.htm
- Schneider Electric (2013, mayo 31). De: ¿Cuál es la diferencia entre SCADA y HMI? Recuperado de <http://www.indusoft.com/blog/2013/05/31/cual-es-la-diferencia-entre-scada-y-hmi/>
- Schneider Electric (2012). *Vijeo Design" Air & Air Plus Tutorial*. Recuperado de http://www.schneider-electric.com/pe/ls/download/document/Vijeo%2BDesign%2BAir%2B%2526%2BAir%2BPlus%2BTutorial?_downloadcenter_WAR_downloadcenterRFportlet_documentId=668763624&_downloadcenter_WAR_downloadcenterRFportlet_documentDetailsBackURL=%2Fpe_ls%2Fschneider%2Fdownloads%2Fresults%2F0%2F0%3Fau_i_3_4_0_1_166%3Dvijeo%26keywordForm%3Dvijeo

Schneider Electric (s. f.). *HMI (Terminales y PC Industriales)*. Recuperado de
<http://www.schneider-electric.com/products/pe/lis/2100-hmi-terminales-y-pc-industriales/>

Molina (s. f.). *¿Qué es un sensor?* Recuperado de
http://www.profesormolina.com.ar/tecnologia/sens_transduct/que_es.htm

Universidad de Guadalajara (s. f.). *Morfología de Actuadores*. Recuperado de
<http://proton.ucting.udg.mx/materias/robotica/r166/r68/r68.htm>

Vildósola E. (s. f.). *Actuadores*. Ponencia presentada para la empresa Soltex Chile S.A. Chile.
Recuperado de <http://www.aie.cl/files/file/comites/ca/abc/actuadores.pdf>

CCM (2016). *Introducción a Ethernet*. Recuperado de <http://es.ccm.net/contents/672-ethernet>

Orellana B. J. (2015). *Diseño e implementación de un sistema de control de dispositivos Maestro-Eslavo basados en la red industrial AS-I para el laboratorio de automatización industrial*. (Tesis de Pregrado). Recuperado de
<http://dspace.ups.edu.ec/handle/123456789/10421>

Lozano Areiza N. M. (s. f.). *Control de Nivel para tanques de agua con realimentación*.
Recuperado de <http://docentes.uni.edu.ni/fec/Alejandro.Mendez/Nivel1.pdf>

Cancel Vélez E. (2006). *Diseño y construcción de un sistema inteligente de control de presión, nivel, temperatura y flujo en un sistema de agua de uso doméstico* (Tesis de Grado).

Recuperado de

http://www.academia.edu/3262667/Dise%C3%B1o_y_construcci%C3%B3n_de_un_sistema_inteligente_de_control_de_presi%C3%B3n_nivel_temperatura_y_flujo_en_un_sistema_de_agua_de_uso_dom%C3%A9stico

Chavez J. (2014, diciembre). *Informe de diseño de control de nivel para un tanque*. Abstract

recuperado de

http://www.academia.edu/9842224/INFORME_DE_DISE%C3%91O_DE_CONTROL_DE_NIVEL_PARA_UN_TANQUE

PROMETEC (s. f.). *El modulo Bluetooth HC-05*. Recuperado de <http://www.prometec.net/bt-hc05/>

Ruben J. (2014, febrero 21). *Bluetooth HC-05 y HC-06 Tutorial de Configuración*. Recuperado de <http://www.geekfactory.mx/tutoriales/bluetooth-hc-05-y-hc-06-tutorial-de-configuracion/>

Gómez Flores V. A. (2014). *Sistema de control de iluminación con protocolo de control doméstico estandarizado* (Tesis de Pregrado). Universidad Nacional Autónoma de México. México D.F.

Álvarez Barreno C, Holguín Cabezas D. O. y Serrano Guevara E. (2007). *Diseño de una instalación domótica en un condominio para el control de seguridad e iluminación mediante la tecnología LonWorks* (Tesis de Grado). Escuela Superior Politécnica del Litoral. Guayaquil, Ecuador.

Schneider Electric (2012). *Vijeo Designer Tutorial*. Recuperado de http://www.infopl.net/files/descargas/schneider/infoPLC_net_VijeoDesigner_Tutorial_ESP.pdf

Luque J. M. (2012). *Simulación Pantalla Vijeo Designer con Simulador PLC de Unity*. Recuperado de <http://plc-hmi-scadas.com/013.php>

Hurtado Banguero A. F. [Andrés Felipe Hurtado Banguero]. (2013, noviembre 26). *Vijeo Designer clase 4* [Archivo de video]. Recuperado de <https://www.youtube.com/watch?v=JsVP3H5mpbY&nohtml5=False>

Schneider Electric España (2010). *Manual Vijeo Designer*. Recuperado de http://www.equiposdidacticos.com/pdf/catalogos/Manual_formacion_Vijeo_Designer.pdf

Duchozny D. (2006, enero 26). De: Asociación dirección IP con dirección MAC. Recuperado de <http://soporte.miarroba.es/332796/4128145-sobre-la-asociacion-de-direccion-ip-y-direccion-mac/>

David (2012, diciembre 3). De: Fundamento de redes y comunicaciones. Recuperado de <http://frc-hramos.blogspot.pe/2012/12/direccionamiento-ip.html>

Unidad de Tecnologías de la Información Argentina (s. f.). *Dirección IP y MAC address (dirección física)*. Recuperado de <http://red.agro.uba.ar/direccionip>

Grup F. (s. f.). *Cómo calcular la cantidad de luz necesaria para una habitación*. Recuperado de http://bricolaje.facilísimo.com/blogs/electricidad/como-calcular-la-cantidad-de-luz-necesaria-para-una-habitacion_880812.html

Gomrei F. (2014, mayo 12). De: ¿Cuánta potencia lumínica necesito? Recuperado de <http://nergiza.com/cuanta-potencia-luminica-necesito/>

Schneider Electric (s. f.). *Controlador Lógico Modicon M221*. Recuperado de <http://www.schneider-electric.es/es/product-range-selector/62128-controlador-logico---modicon-m221/?filter=business-1-automatizacion-y-control-industrial&parent-category-id=3900>

Schneider Electric (s. f.) *Magelis STO, STU*. Recuperado de <http://www.schneider-electric.es/es/product-range-selector/5774-magelis-sto--stu/?filter=business-1-automatizacion-y-control-industrial&parent-category-id=2100>

Schneider Electric [PT Schneider Indonesia]. (2012, diciembre 18). *HMI STU Configuration*

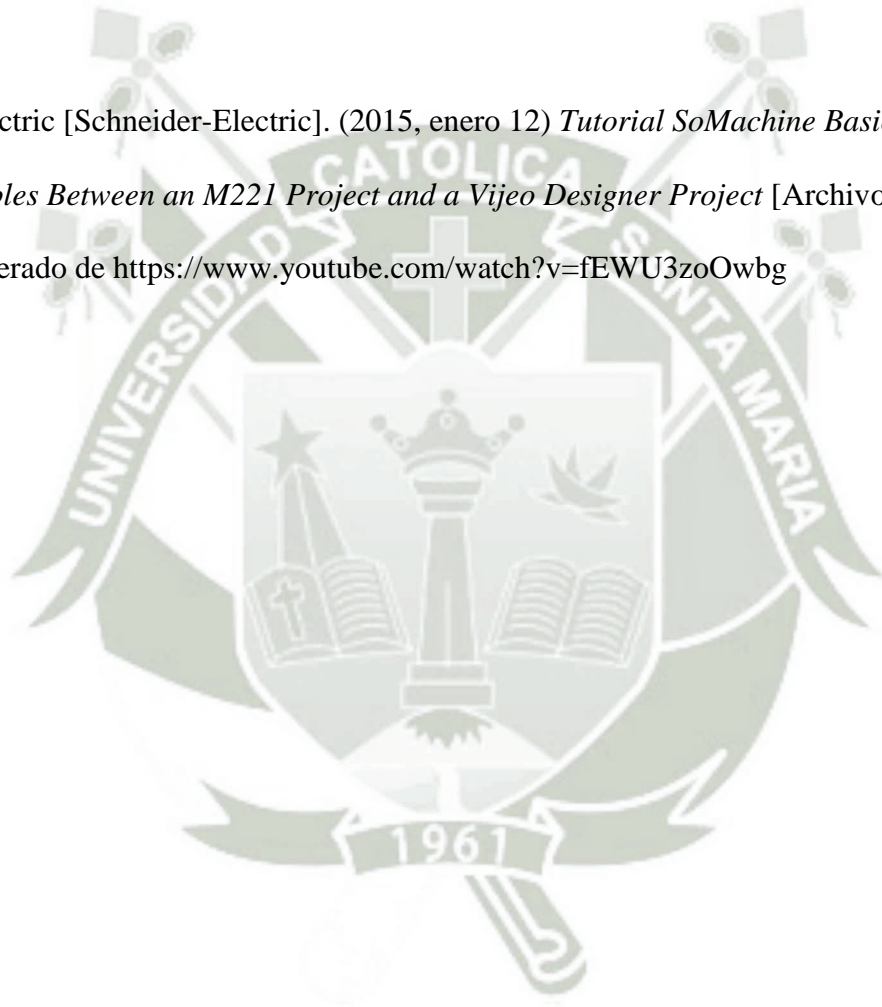
[Archivo de video]. Recuperado de https://www.youtube.com/watch?v=m6SAiQel2_Y

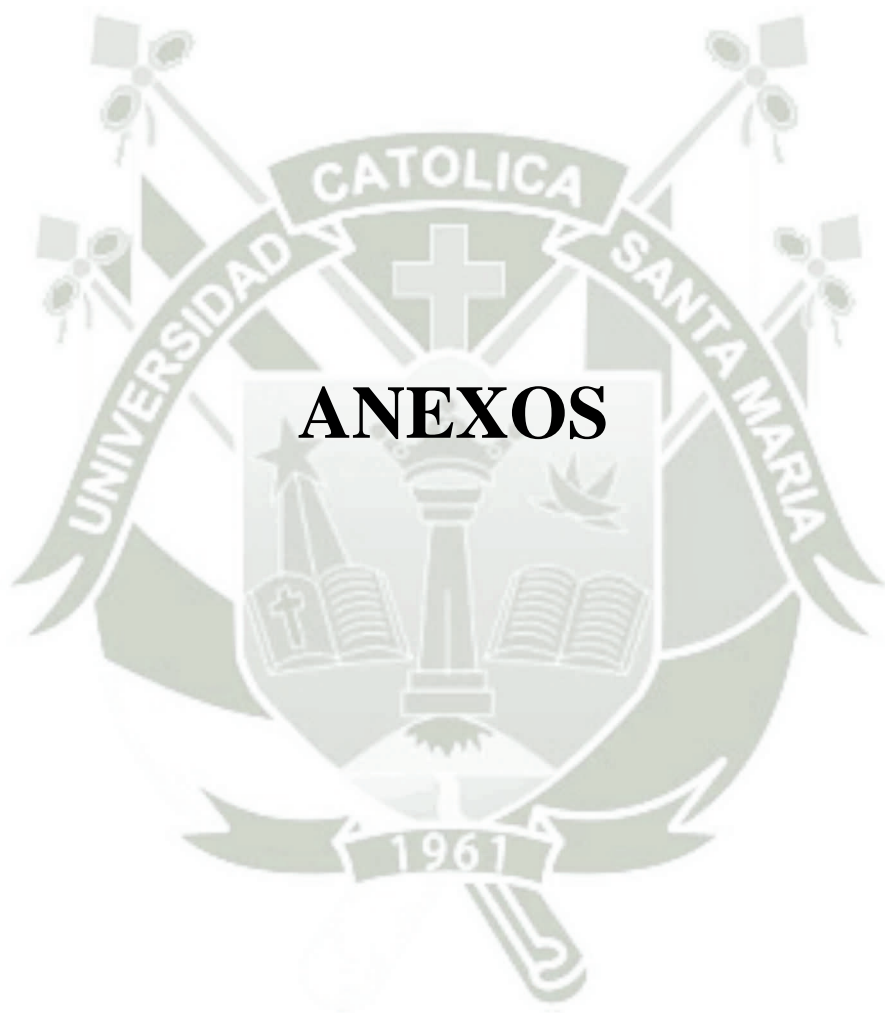
Schneider Electric [Schneider-Electric]. (2015, enero 12). *Tutorial SoMachine Basic - Running the M221 Simulator and the HMI Simulation at the Same Time* [Archivo de video].

Recuperado de <https://www.youtube.com/watch?v=IeanTDY1zds>

Schneider Electric [Schneider-Electric]. (2015, enero 12) *Tutorial SoMachine Basic - Sharing Variables Between an M221 Project and a Vijeo Designer Project* [Archivo de video].

Recuperado de <https://www.youtube.com/watch?v=fEWU3zoOwbg>





MANUAL DE USO DE MODULO DIDACTICO

I. OBJETIVOS:

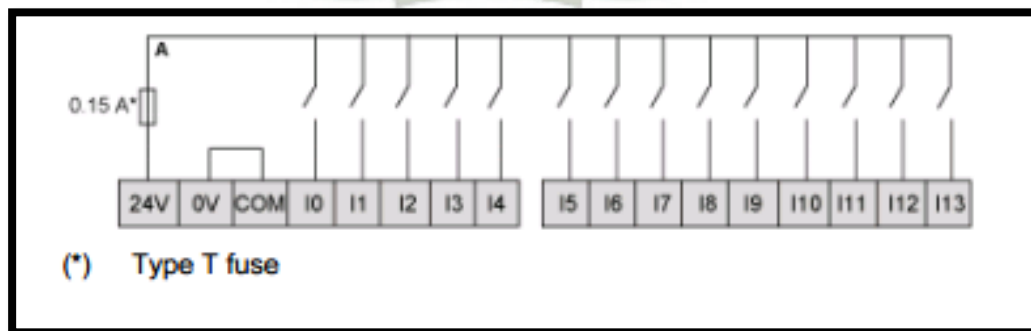
- Conocer el funcionamiento del Módulo Didáctico.
- Interacción entre Módulo Central, Módulo Domótico y Módulo de Control de Nivel.
- Probar el funcionamiento real de los PLC y el HMI.

II. MATERIALES Y/O EQUIPOS A UTILIZAR:

- Módulo de Domótica.
- Módulo de Control de Nivel
- Módulo Central
- Cables con terminal banano-banano

III. DESARROLLO DEL MANUAL

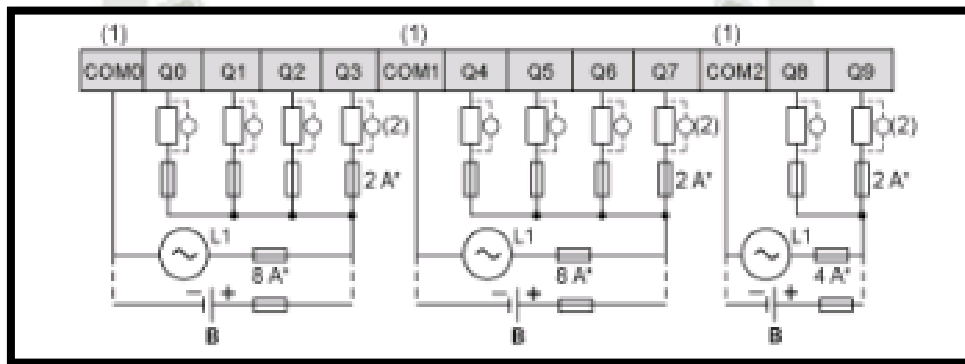
1. Comprobar que no esté conectado el módulo a la red eléctrica, para evitar choques eléctricos.
2. Comprobar que la palanca de la llave principal de 50 A, que está ubicada detrás del módulo de control de nivel esté apagada o baja.
3. Utilizar los bananos para conectar las salidas del módulo central al módulo domótico, recordar que las conexiones de las entradas del PLC son de la siguiente manera:



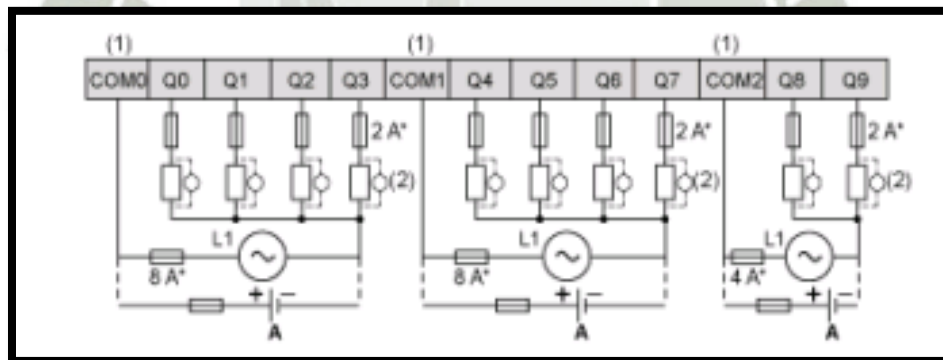
Disposición de las Entradas Digitales

Por lo tanto se debió proteger con relay las entradas del PLC. Entonces se puenteo internamente 0v y COM y se llevó hasta un terminal COM en el tablero. Por lo tanto se deberá llevar desde 24v F (rojo) de la fuente interna del PLC a través de un cable banano hasta 24v PLC (rojo) y dejar libre el COM (negro) debajo de 24v PLC de ambos módulos, al igual que el COM (negro) debajo de 24v F del tablero central sin conectar.

- Recordar que las salidas tipo relay tienen la siguiente estructura interna:



Disposición Negativa.



Disposición Positiva.

De la manera en la que se quiera conectar está bien. Recordar que ésta disposición deberá contar con una fuente externa para acceder a los relay. Por lo cual recomendamos puentear COM 0, COM 1 Y COM 2 de ambos PLC para trabajar bajo un mismo voltaje y llevarlo por un cable banano al terminal de la fuente de 24v FUENTE (rojo) del módulo

de domótica. Y a su vez puentear COM 0, COM 1 Y COM 2 de ambos módulos y llevarlo por un cable banano al terminal COM (negro) debajo de la fuente de 24v FUENTE del módulo de domótica.

5. Realizar las siguientes conexiones para conectar los sensores y actuadores al sistema:

MODULO DE DOMOTICA (PLC 1)

ENTRADAS DIGITALES

<i>I0 : ON/OFF</i>	<i>I1 : S MOVIMIENTO</i>
<i>I2 : S HUMO</i>	<i>I3 : S INDUCTIVO</i>

ENTRADAS ANALOGICAS
(respetar positivo rojo y negativo negro)

<i>IW0 : V ACTUAL</i>	<i>IW1 : T ACTUAL</i>
-----------------------	-----------------------

SALIDAS DIGITALES

<i>Q0 : LUZ INTERNA</i>	<i>Q1 : ALARMA</i>
<i>Q2 : LUZ ENTRADA</i>	<i>Q3 : VENTILADOR HUMO</i>
<i>Q4 : SERVOMOTOR</i>	<i>Q5 : VENTILADOR CALEFACTOR</i>
<i>Q6 : LUZ CALEFACTOR</i>	

MODULO DE NIVEL (PLC 2)

ENTRADAS DIGITALES

<i>I0 : ON</i>	<i>I1 : OFF</i>
<i>I2 : BOT</i>	<i>I3 : TOP</i>

SALIDAS DIGITALES

<i>Q0 : AMBAR</i>	<i>Q1 : VERDE</i>
<i>Q2 : ROJO</i>	<i>Q3 : VALVULA</i>
<i>Q4 : BOMBA</i>	

6. Revisar de nunca conectar de ninguna manera los terminales de las fuentes positivo y negativo de manera directa a través de un cable:
 - Los terminales de la fuente interna de los PLC rojo y negro 24v F y COM respectivamente (PUEDE CAUSAR DAÑOS INTERNOS AL PLC). Se recomienda dejar el terminal COM (negro) del tablero central de control sin conexiones a menos que se sepa cómo usarlo correctamente sin dañar a los equipos.
 - Los terminales de la fuente de 12v FUENTE y COM de ninguno de los módulos.
 - Los terminales de la fuente de 24v FUENTE y COM de ninguno de los módulos.
 - Los terminales de alguna FUENTE o PLC con el terminal COM de alguna de las fuentes.
7. Para alimentar el HMI se deberá conectar directamente con un cable banano desde el terminal rojo de 24v FUENTE del módulo de domótica hasta el terminal 24v del módulo central, y de la misma manera desde el COM que se encuentra debajo del terminal 24v FUENTE del módulo de domótica hasta el terminal COM del módulo central.
8. Luego de esto revisar que la llave de 10 A del módulo central esté encendida.
9. Proceder a conectar el enchufe de la bomba (Amarillo) al tomacorrientes doble ubicado detrás del módulo de control de nivel. Nótese que el orden de mismo es la primera toma para línea y la segunda toma para la bomba.
10. Conectar a través de un cable con enchufe doble el módulo de control de nivel y en tomacorrientes cuádruple que va a la llave principal.
11. Conectar los enchufes de ambas fuentes (Verde) al tomacorrientes cuádruple.
12. Conectar el enchufe del módulo central (Amarillo) al tomacorrientes cuádruple

13. Conectar el enchufe principal que sale de la llave principal de 50 A (Blanco), a la toma de 220v de la red eléctrica.
14. Luego de estar seguros de haber conectado correctamente todos los terminales en su lugar y de no haber podido conectar algún cable banano que pueda hacer corto circuito, finalmente subir la palanca de la llave principal de 50 A.
15. Observar que el terminal HMI esté encendido y en funcionamiento y no presente ningún error.



Errores de conexión HMI-PLC

De aparecer estos símbolos se deberá volver a conectar el cable Ethernet En los PLC y el HMI para estar seguros de estar enviando y recibiendo datos correctamente.

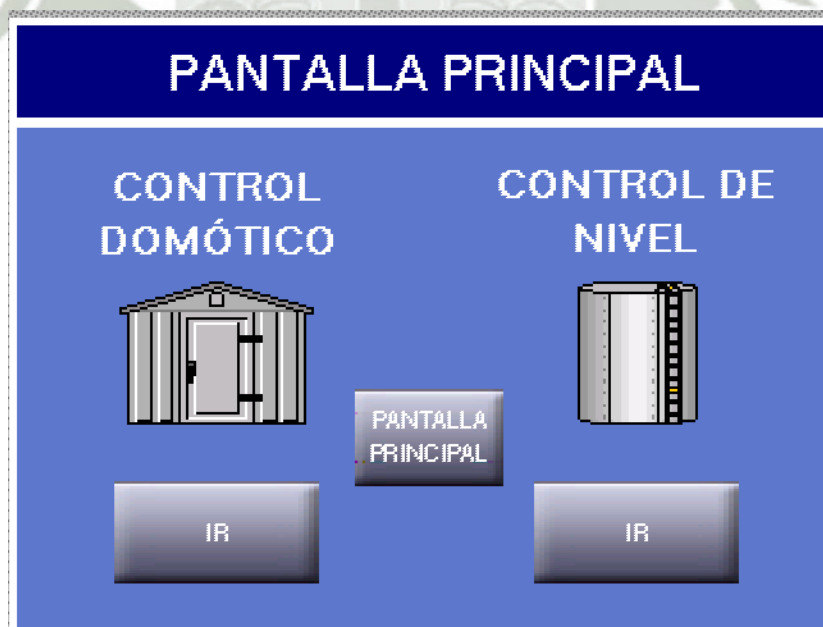
16. Abrir el módulo central y observar que los PLC estén en funcionamiento. Y no presenten encendidas las luces de error.
17. Los pasos para reconocer el programa cargado son los siguientes:

-Primera Pantalla:



Ésta es la pantalla principal que deberá aparecer por defecto cada vez que se encienda el modulo. Pulsar en “INICIAR” para comenzar con el programa.

-Segunda Pantalla:



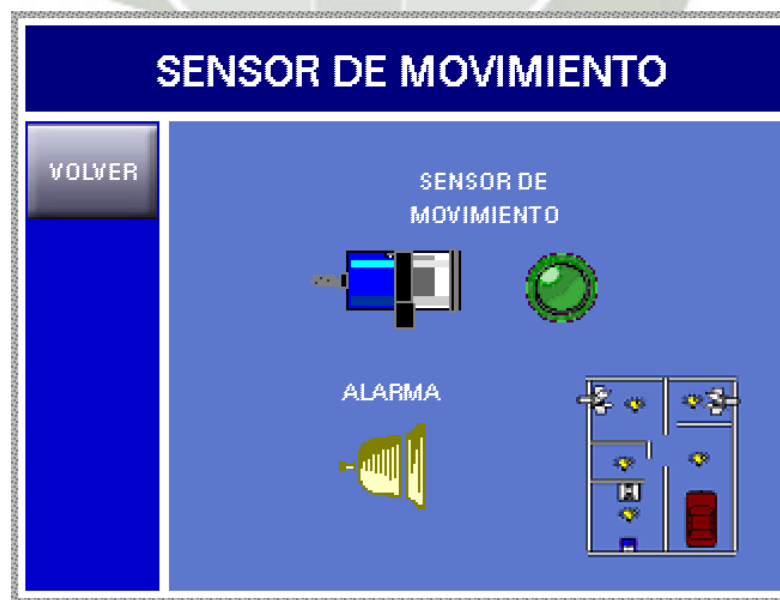
Pulsar el botón “IR” que aparece debajo del grafico de una casa.

-Tercera Pantalla:



Ésta es la pantalla desde donde se podrá acceder a todas las funciones del módulo de domótica. Si pulsamos en “Ir a Modulo de Nivel” nos llevará a la pantalla de control del módulo de nivel. Si pulsamos en “Control Total de Luz” o si accionamos el interruptor dentro de la vivienda, encenderemos o apagaremos todas las luces dentro de la casa. Luego pulsamos en “Sensor de Movimiento”.

-Cuarta Pantalla:



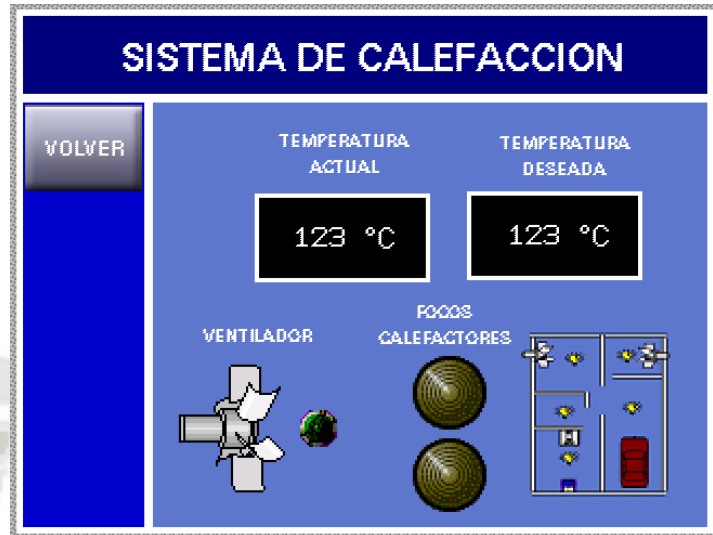
Aquí se detectará variación en el caso de ser accionado el sensor de movimiento en la casa. En la maqueta se deberá pasar por delante o atravesar la puerta y se accionará una alarma y aquí se encenderá la luz piloto del sensor y deberá parpadear la imagen de alarma cuando ésta se encienda.

-Quinta Pantalla:



Para acceder a ésta pantalla, se deberá pulsar el botón “Detector de Humo” desde la pantalla de control. Al acceder aquí se observa dos luces piloto que representan el accionamiento del sensor detector y del ventilador. En la casa se verá una luz que envía un led hasta un LDR al interrumpir éste haz de luz con cualquier objeto, simulará que el humo de la cocina en caso haya algún incendio o se esté quemando algo, interrumpe la detección de luz por lo tanto activará el sensor de humo que a su vez activará el ventilador encendiéndolo.

-Sexta Pantalla:



Para acceder a ésta pantalla, se deberá pulsar “Sistema de Calefacción” desde la pantalla de control. Aquí un sensor LM35 sensará la temperatura del ambiente. Al pulsar el recuadro de “Temperatura Deseada” se elegirá el valor que se quiere y en base a éste valor se calculará la diferencia de temperaturas con un rango de ± 2 ; De ser mayor la temperatura actual a la deseada se encenderá el ventilador para ventilar el ambiente, en caso contrario se encenderán los focos calefactores para que calienten el ambiente.

-Séptima Pantalla:



Para acceder a ésta pantalla, se deberá pulsar “Sistema de Alumbrado” desde la pantalla de control. Aquí un sensor LDR sensará la luz del exterior y lo expresará en un voltaje el cual se comparará con el “Voltaje Deseado” que al igual que “Temperatura Deseada” ingresará a través de un teclado que aparecerá tras pulsar el recuadro correspondiente. La comparación de estos dos valores encenderá o apagará automáticamente la iluminación de entrada.

-Octava Pantalla:



Para acceder a ésta pantalla, se deberá pulsar “Control de Garaje” desde la pantalla de control. Aquí tendremos varias opciones, en primer lugar por defecto la puerta estará cerrada. El pulsador de “ABRIR” Activará el servomotor de la maqueta luego de estar abierta tras 5 segundos activará la luz de abierto para saber que está abierto el garaje. Le incluimos un botón de “SENSOR INDUCTIVO” en caso el sensor inductivo incluido en la maqueta falle puesto que es muy sensible a subidas de corriente o corto circuito. Al pulsarlo cumple la misma función que el sensor en la maqueta avisar que hay un objeto en nuestro garaje. Mientras el botón de “ABRIR” esté pulsado la puerta no cerrará a pesar

que el sensor o el botón de sensor estén activados. Al volver a presionar o apagar el botón “ABRIR”, si el botón “SENSOR INDUCTIVO” está activado o si el sensor inductivo de la maqueta envió señal que está abierto y mientras no haya interrupción de ésta señal, tras luego de 10 segundos la puerta se cerrará y la luz piloto de “Abierto” se apagará y se encenderá la luz piloto de “Cerrado”. Éste arreglo se hizo por la misma razón que un garaje real. No deberá cerrar por la seguridad del usuario en ningún momento mientras no se dé la confirmación de que todo está listo para cerrarse. En éste caso si sigue activado “ABRIR” nunca se cerrará y si se desactiva se cerrará automáticamente si y solo si “Sensor Inductivo” está encendido y si pasó 10 segundos sin ninguna modificación de señal cualquier cambio de ésta señal evitará que la puerta se cierre puesto que se detecta como si el auto está saliendo de nuevo.

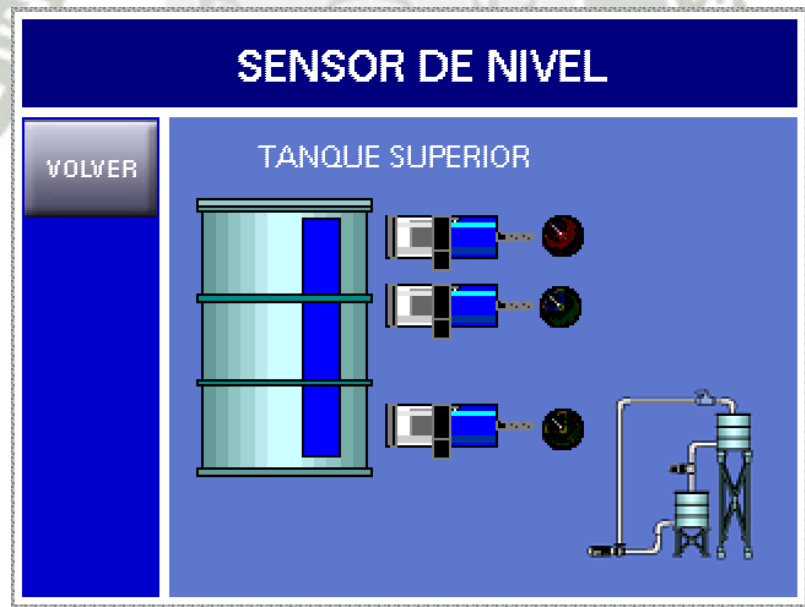
-Novena Pantalla:



En la segunda pantalla pulsar el botón “IR” que aparece debajo del grafico de un tanque de agua. Ésta es la pantalla desde donde se podrá acceder a todas las funciones del módulo de control de nivel. Si pulsamos en “Ir a Modulo Domótico” nos llevará a la

pantalla de control del módulo domótico. Si pulsamos en “Abrir Válvula”, entonces la Válvula Solenoide que ésta en la instalación se abrirá y si pulsamos “Cerrar Válvula” se cerrará y a su vez encenderá la luz piloto debajo del gráfico de válvula solenoide para indicar que está encendida. Aparecen también en ésta pantalla 3 luces piloto más que indicaran el nivel en el que se encuentra el tanque. Que también están ubicadas en el módulo de control de nivel. La luz amarilla se enciende cuando todo el sistema está conectado. La luz verde cuando el tanque está siendo llenado por la bomba y pasa el nivel bajo del tanque. La luz roja enciende cuando el tanque está al límite y a su vez detiene la bomba puesto que puede desbordar. Finalmente el botón “Parada de Emergencia” detiene todo lo que se encuentre activo en ése momento sea la bomba o la válvula. Presionamos “Sensor de Nivel”.

-Décima Pantalla:



Finalmente ésta es la última pantalla que es una representación gráfica del llenado del tanque. Aquí también vemos las luces piloto del nivel en que se encuentre el tanque.

DISEÑO DE LAS PRÁCTICAS A DESARROLLAR

PRÁCTICA N° 1

INTRODUCCIÓN A LA PROGRAMACION EN SOMACHINE BASIC

I. OBJETIVOS:

- Aprender a crear nuevos proyectos en el software SoMachine Basic.
- Identificar el entorno de programación del mismo software.

II. MATERIALES Y/O EQUIPOS A UTILIZAR:

- Computador
- Software SoMachine Basic

III. DESARROLLO DE LA PRÁCTICA

1. Ingresamos al programa SoMachine Basic
2. Hacemos clic en “Crear un nuevo proyecto”

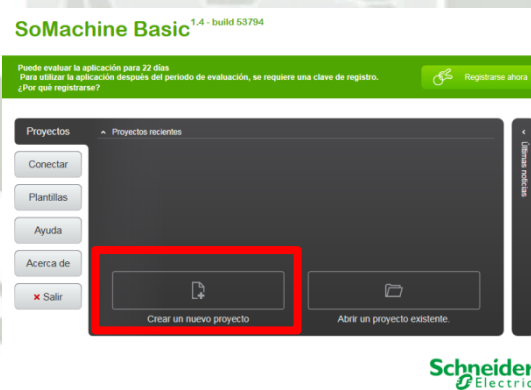


Figura 1. Creación de nuevo proyecto

3. Buscamos el PLC con el que trabajaremos en la lista que tenemos a la derecha de la ventana que en nuestro caso es el TM221CE24R, una vez que lo encontramos lo arrastramos hacia el espacio central como muestra la figura 2.

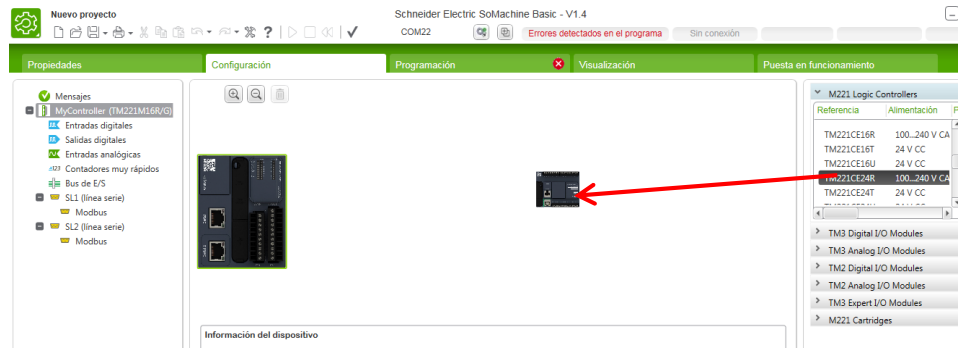


Figura 2. Selección de controlador

- Confirmamos que queremos reemplazar el PLC que esta por defecto.

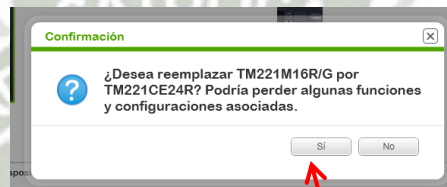


Figura 3. Confirmación de reemplazo

- Si es el caso, se puede colocar módulos de expansión o cartuchos que se hayan incluido en nuestro PLC físico para que sean reconocidas sus entradas y salidas simplemente arrastrándolos de las listas de la derecha pero en nuestro caso no es necesario ninguno.



Figura 4. Adición de módulos o cartuchos

- Ahora que tenemos nuestro controlador listo, podremos reconocer sus características en el panel de la izquierda el panel inferior.

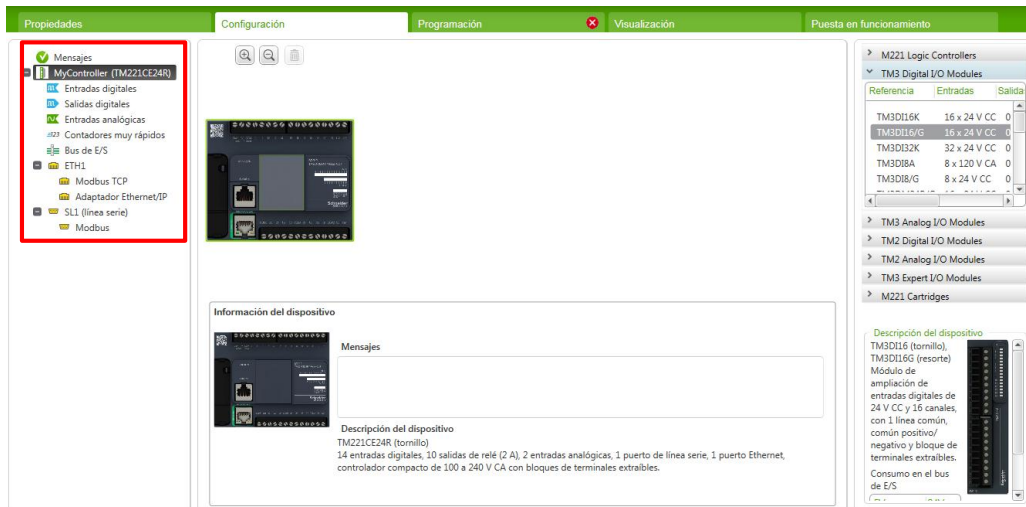


Figura 5. Visualización de características

Entradas Digitales:

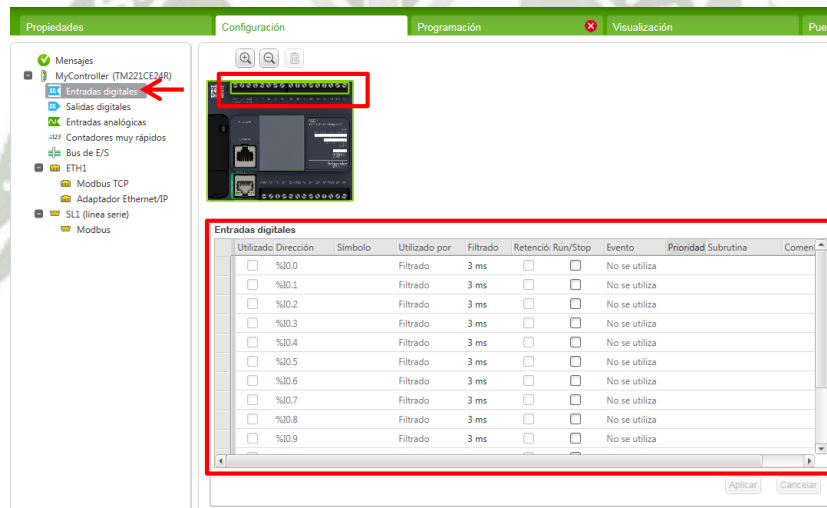


Figura 6. Entradas digitales

Salidas digitales:

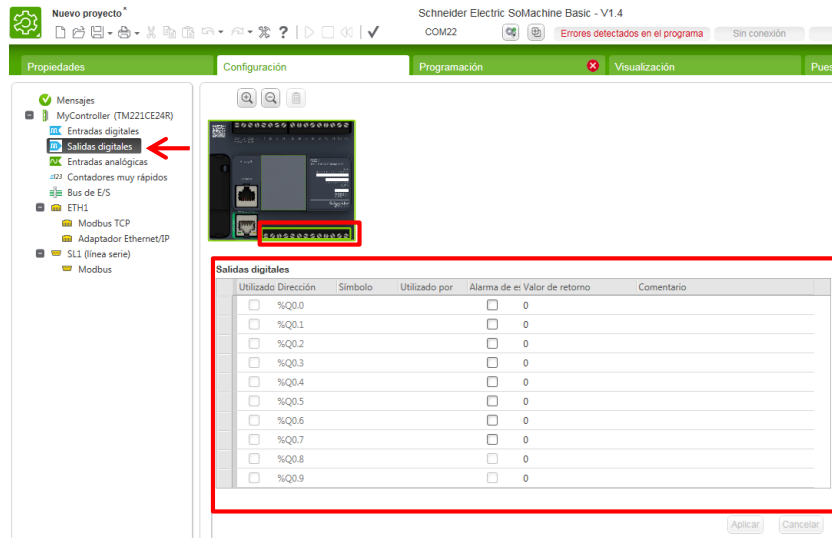


Figura 7. Salidas digitales

Y así igualmente las entradas analógicas, contadores muy rápidos, bus de E/S, Modbus TCP, Ethernet y Modbus.

- Una vez reconocidas todas las características y las direcciones de las entradas y salidas podemos pasar a la programación haciendo clic en la respectiva pestaña. Aquí encontraremos la opción de programar en lenguaje Ladder (LD) o en Lista de Instrucciones (IL)

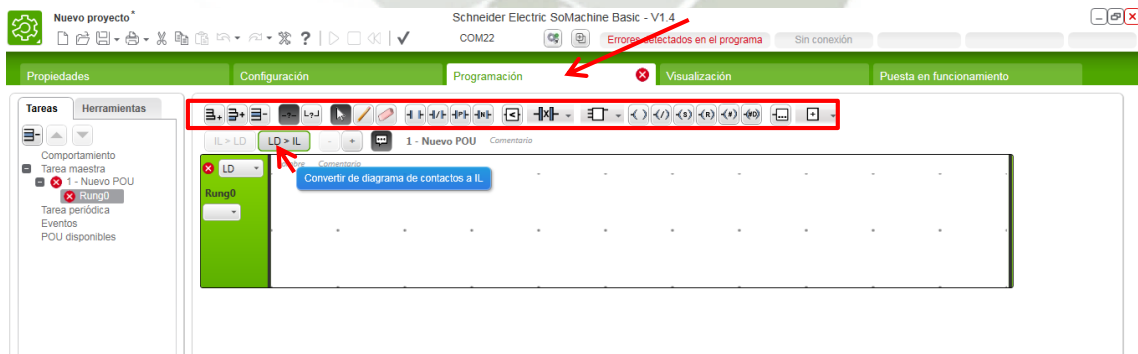


Figura 8. Entorno de programación

8. En la imagen anterior podemos también observar las distintas herramientas de programación de Ladder como son el añadir escalones, añadir contactos, contactos negados, bobinas, bloques de función, dibujar líneas de conexión, etc. Bastará con dejar el cursor sobre cada opción para leer su función.
9. En el lado izquierdo de la ventana podemos las tareas y herramientas de nuestro programa, incluso podemos agregar y eliminar POU's y en esta versión de software podemos agregar un POU en lenguaje Grafcet.



Figura 9. Tareas y herramientas de programa

10. Para empezar a programar vamos a colocar un contacto (entrada) y una bobina (salida) arrastrándolos desde sus iconos en la pantalla.
11. Luego de colocarlos, en caso de que no se una automáticamente, dibujaremos su línea de unión haciendo clic en el icono del lápiz.
12. Una vez hecho esto pasamos a darle la dirección al contacto (%I0.0) y la bobina (%Q0.0).

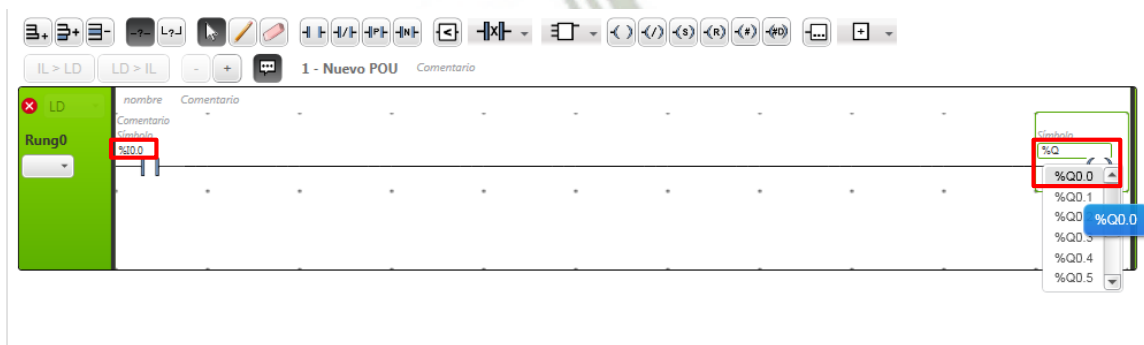


Figura 10. Direccionamiento de variables

13. Luego compilamos el programa haciendo clic en el icono del Check y corregimos los errores en caso de haberlos

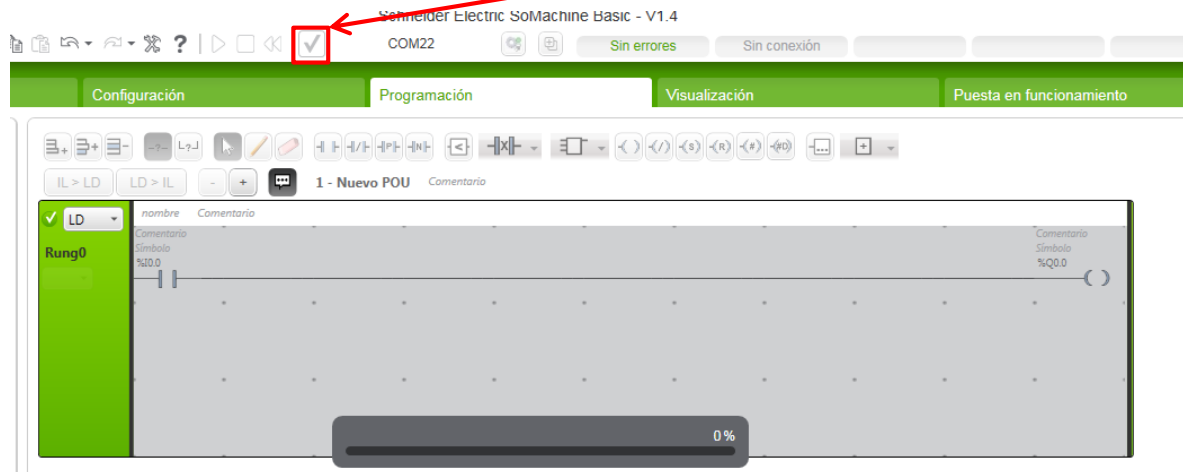


Figura 11. Compilación de programa

14. Para probar nuestro programa vamos a la pestaña “Puesta en funcionamiento” y hacemos clic en iniciar simulador.

15. Nos deberá aparecer un cuadro que podemos observar en la Figura 12.

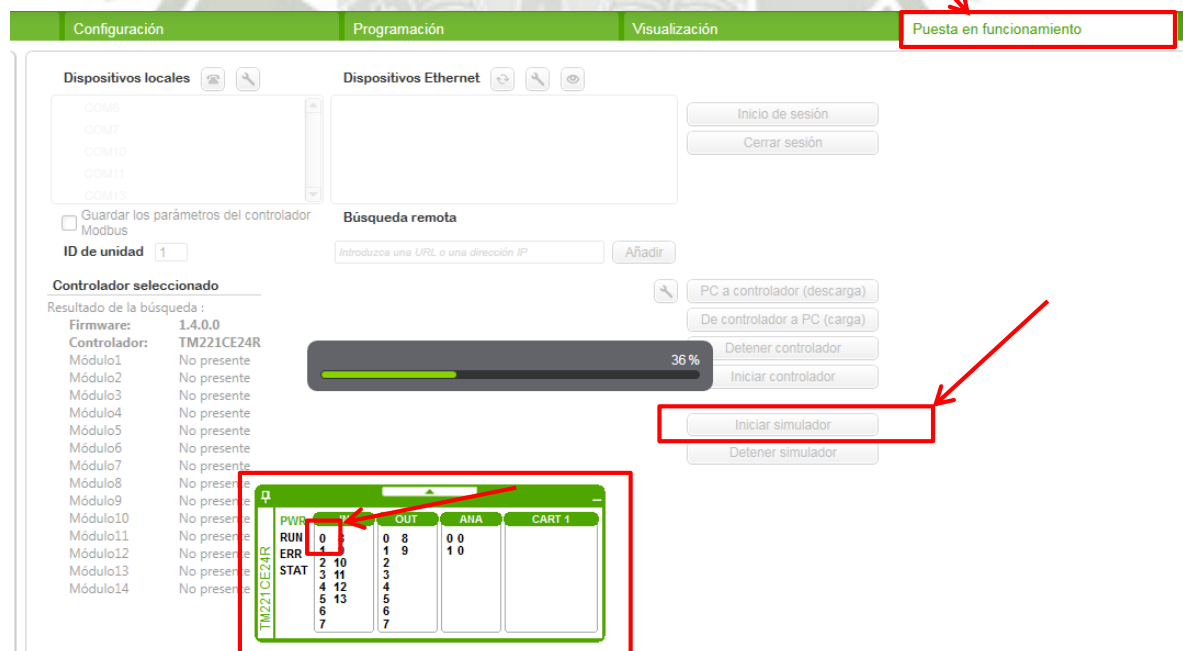


Figura 12. Inicio de Simulador

16. Para nuestra simulación volvemos a la pestaña “Programación” y le damos clic en el botón de “RUN” para correr el programa.
17. Luego hacemos clic en la entrada 0 de la tabla que nos apareció al iniciar el simulador y podremos observar cómo cambia el valor de nuestro contacto y a su vez el control de la salida.

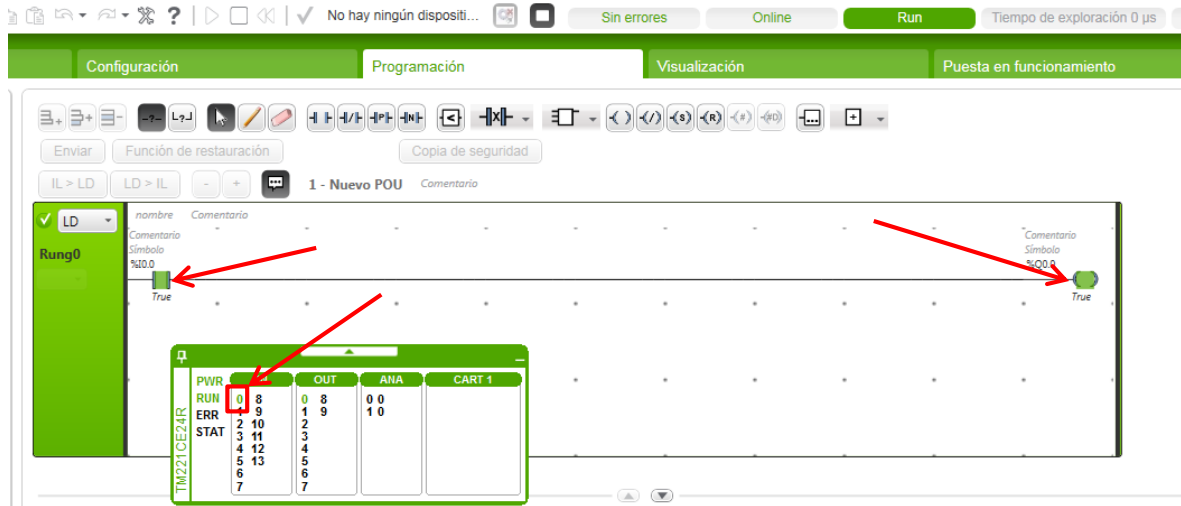


Figura 13. Simulación

18. Para poder volver a programar tenemos que terminar nuestra simulación haciendo clic en el icono de “STOP” que está al lado de “RUN” y luego en la pestaña de “Puesta en funcionamiento” hacemos clic en “Detener simulador”
19. Ahora realizaremos y simularemos el programa mostrado en la Figura 14, teniendo en cuenta que se debe añadir manualmente los escalones con el icono “Añadir un nuevo escalón” (Ctrl + R) de la barra de herramientas.



Figura 13. Programa 2

20. Describa la función que tiene la variable %M0

21. Describa la función que tiene %I0.1

IV. CUESTIONARIO FINAL

- 1) ¿Qué ventajas y desventajas tiene el lenguaje Ladder (LD) comparado con la Lista de Instrucciones (IL)?
- 2) ¿Qué son las variables %M_ y cuantas tiene nuestro PLC?
- 3) ¿Cuáles son las ventajas y desventajas de usar un sistema de enclavamiento virtual comparado con uno físico?
- 4) ¿En qué caso se debería colocar un contacto NA como stop en el enclavamiento de nuestro programa?

V. OBSERVACIONES Y CONCLUSIONES

PRÁCTICA N° 2

TEMPORIZADORES EN SOMACHINE BASIC

I. OBJETIVOS:

- Aprender a usar temporizadores en el software SoMachine Basic.

II. MATERIALES Y/O EQUIPOS A UTILIZAR:

- Computador
- Software SoMachine Basic

III. MARCO TEORICO

Los temporizadores dentro de la programación para PLC's son bloques de función que permiten regular la conexión o desconexión de una bobina (salida) después de que se ha programado un tiempo.

En SoMachine su direccionamiento es %T_{Mi} donde i se reemplaza por un número entre el 0 y 254 en el caso del PLC que se tiene en el laboratorio (TM221CE24R).

Los temporizadores tienen 3 tipos de funcionamiento:

- TON: permite gestionar retardos en la conexión.
- TOF: permite gestionar retardos en la desconexión.
- TP: permite elaborar un pulso de duración precisa.

IV. DESARROLLO DE LA PRÁCTICA

1. Ingresamos al programa SoMachine Basic
2. Creamos un nuevo proyecto
3. Seleccionamos el PLC que vamos a utilizar
4. Vamos a la pestaña de programación

- Colocamos un temporizador dentro del icono “Bloques de funciones” (F8 / Alt+F) como vemos en la Figura 1.

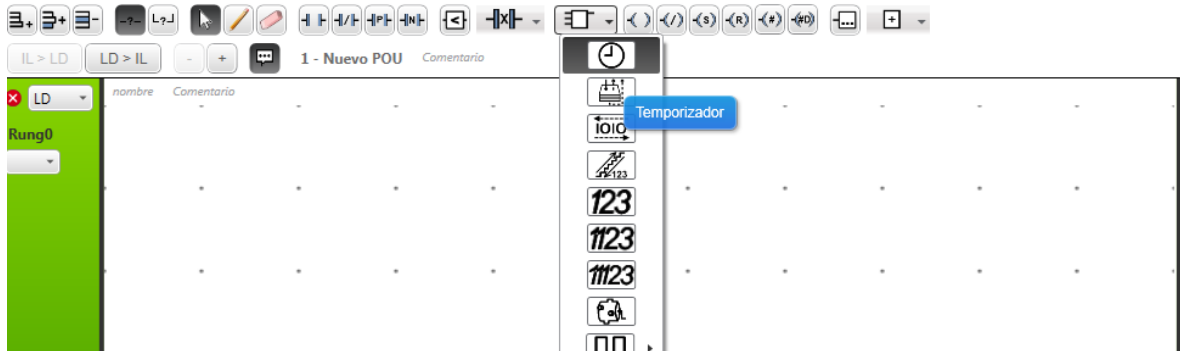


Figura 1. Selección de Temporizador

- Luego le colocamos un contacto (entrada) y una bobina (salida) y las unimos con el temporizador de manera adecuada como se muestra en la Figura 2.
- Debemos direccionar el contacto y la bobina, el temporizador viene direccionado por defecto en %TM0 y si colocamos más irán ascendiendo (%TM1, %TM2, etc.)

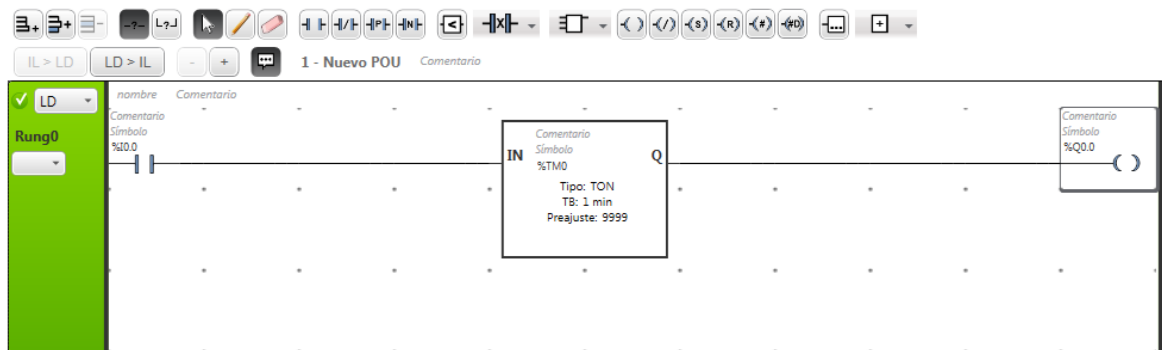


Figura 2. Selección de Temporizador

- Para configurar el temporizador le damos doble clic y nos aparecerán en la parte inferior las Propiedades de Temporizador donde encontraremos todos los temporizadores que tiene nuestro PLC y podremos ver cuales están activos en nuestro programa.

9. Por defecto los temporizadores están configurado tipo TON, con un tiempo base de 1 min y un pre ajuste de 9999. Debemos cambiar el tiempo base a 1s y el pre ajuste a 5 haciendo doble clic sobre los mismos y finalmente le damos al botón aplicar

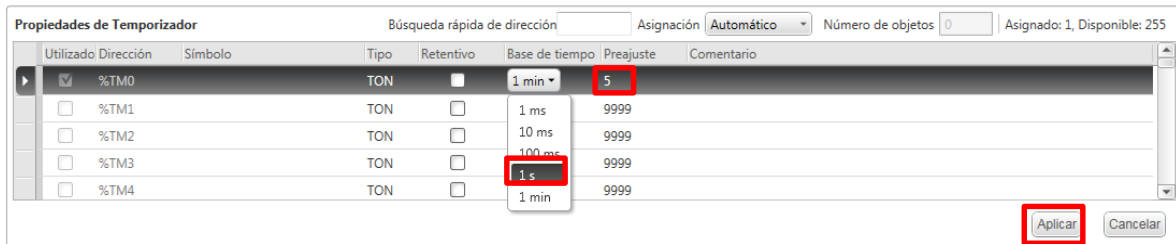


Figura 3. Configuración de Temporizador

10. Una vez hecho esto podemos pasar a la simulación y ver el funcionamiento de nuestro programa.
11. Una vez que realice este programa, cambie el tipo del temporizador a TOF y explique la diferencia.
12. Haga lo mismo con el tipo TP.
-
13. Realice la secuencia de un semáforo que permanezca 8 segundos en verde, 4 en ámbar y 12 en rojo, y una vez que acabe la secuencia vuelva a empezar. Considere un pulsador de Start y Reset para iniciar y resetear el funcionamiento.
14. Modifique su programa de forma que otras 3 luces en otra dirección de tránsito también puedan ser controladas. Realice este diseño usando el menor número posible de temporizadores.

Semáforo 1	ROJO		VERDE	AMBAR
Semáforo 2	VERDE	AMBAR	ROJO	
	8 segundos	4 segundos	8 segundos	4 segundos

Tabla 1. Secuencia de 2 semáforos

V. CUESTIONARIO FINAL

- 1) Describa el funcionamiento de los 3 tipos de temporizadores y mencione algunas aplicaciones para cada tipo.
- 2) Dibuje los diagramas de tiempo de cada tipo de temporizador.
- 3) ¿Se pueden tener 2 o más temporizadores con la misma dirección en un solo programa?
¿Si es posible que aplicaciones tendría?
- 4) ¿Podemos decir que un temporizador “TP” tiene memoria? ¿Por qué?

VI. OBSERVACIONES Y CONCLUSIONES

PRÁCTICA N° 3

CONTADORES EN SOMACHINE BASIC

I. OBJETIVOS:

- Aprender a usar contadores en el software SoMachine Basic.

II. MATERIALES Y/O EQUIPOS A UTILIZAR:

- Computador
- Software SoMachine Basic

III. MARCO TEORICO

Los contadores dentro de la programación para PLC's son bloques de función que permiten almacenar y contar los impulsos que recibe en la entrada destinada a tal efecto.

En SoMachine su direccionamiento es %Ci donde i se reemplaza por un número entre el 0 y 254 en el caso del PLC que se tiene en el laboratorio (TM221CE24R).

Los contadores tienen 2 tipos de funcionamiento:

- CTU: conteo progresivo.
- CTD: conteo regresivo.

En el caso de nuestro PLC que se programa en SoMachine Basic ambos están juntos en un solo bloque contador.

IV. DESARROLLO DE LA PRÁCTICA

1. Ingresamos al programa SoMachine Basic
2. Creamos un nuevo proyecto
3. Seleccionamos el PLC que vamos a utilizar
4. Vamos a la pestaña de programación

- Colocamos un contador dentro del icono “Bloques de funciones” (F8 / Alt+F) como vemos en la Figura 1.

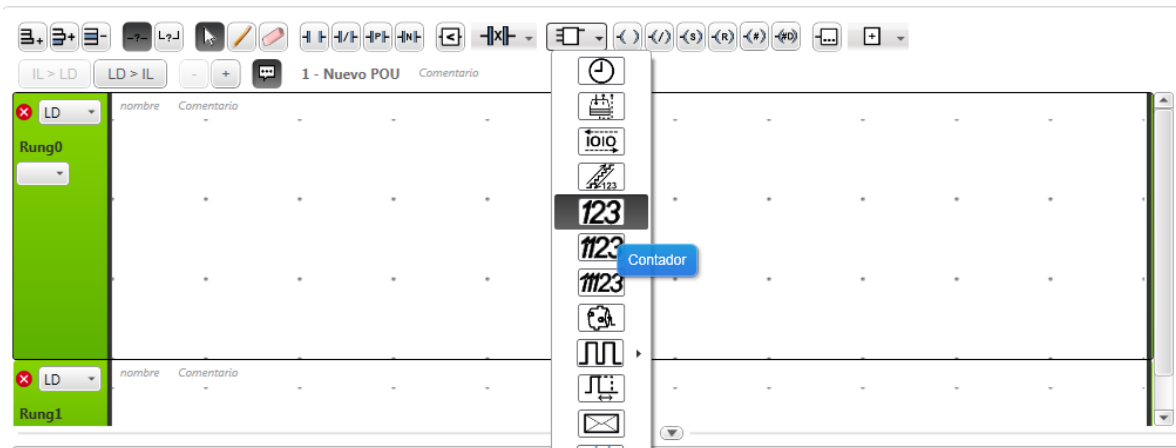


Figura 1. Selección de Contador

- Luego colocamos un contacto en cada entrada del contador y una bobina en cada salida del contador.
- Direccionamos todas como se observa en la Figura 2. (El contador viene direccionado en %C0 por defecto)

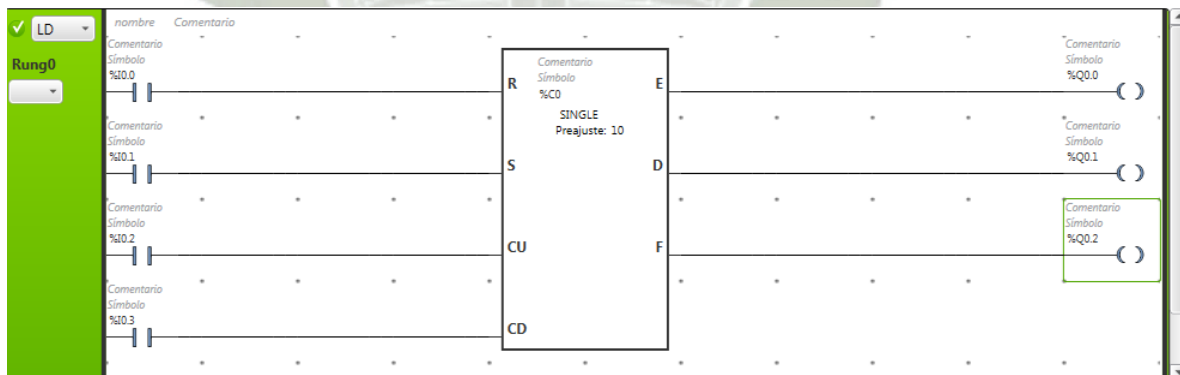


Figura 2. Entradas y Salidas del Contador

8. Hacemos doble clic en el bloque de contador para configurarlo en una ventana que nos aparecerá debajo del programa.
9. En la ventana Propiedades de Contador cambiamos el pre ajuste del %C0 a 10 haciendo doble clic sobre él y luego le damos a aplicar

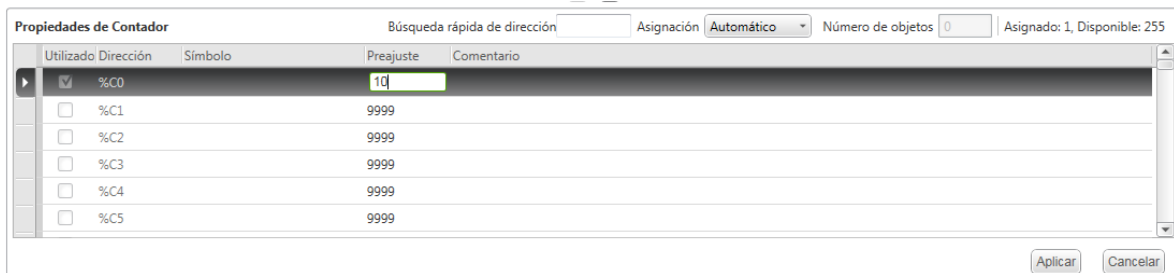


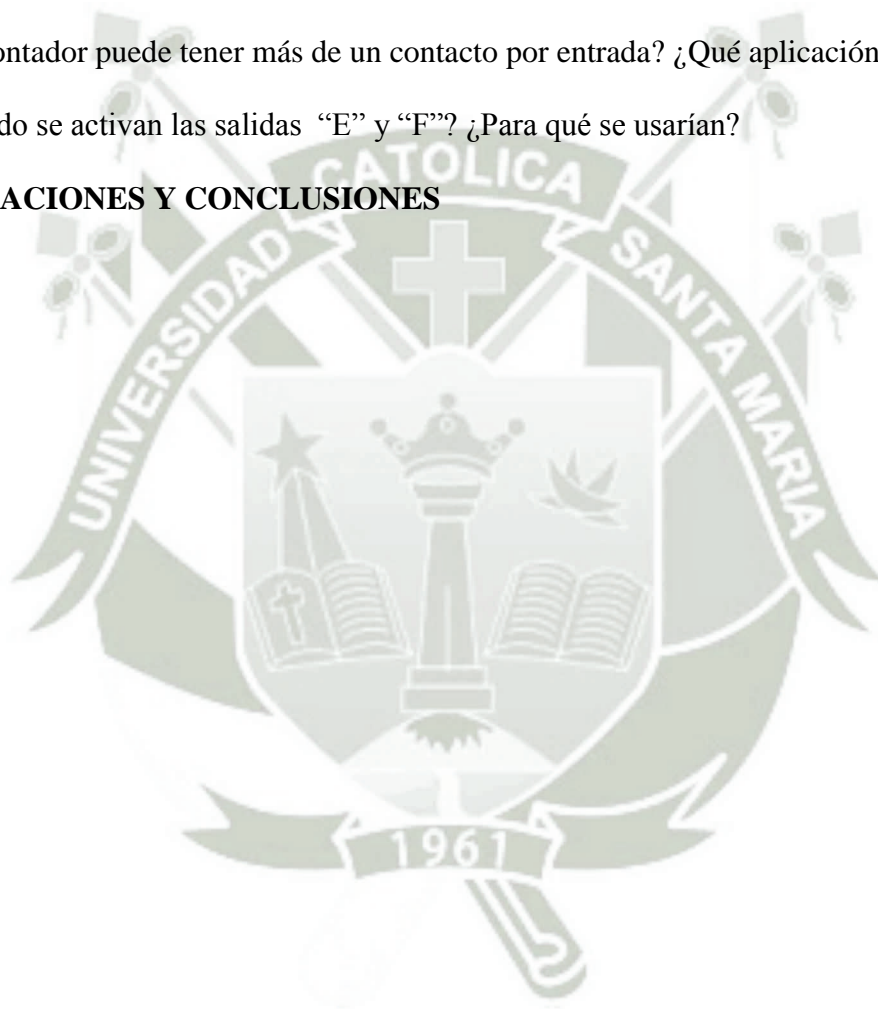
Figura 3. Propiedades del Contador

10. Compilamos y corremos la simulación.
11. Explicamos que ocurre cuando le llega un pulso a la entrada:
 - CU: _____
 - CD: _____
 - R: _____
 - S: _____
12. Se desea implementar un programa para el parpadeo de una lámpara de seguridad con un intervalo de 3 segundos para encendido y 2 segundos para apagado. Si la lámpara se activó 10 veces, esta deberá quedar permanentemente encendida.
Deberá considerar un pulsador de START y STOP para iniciar o parar el funcionamiento
Indique las variables de entrada y salida.

V. CUESTIONARIO FINAL

- 1) Mencione y describa el funcionamiento de los tipos de contadores que existen en cualquier PLC y mencione algunas aplicaciones para cada tipo.
- 2) Dibuje los diagramas de tiempo de cada tipo de contador.
- 3) ¿Se pueden tener 2 o más contadores con la misma dirección en un solo programa? ¿Si es posible que aplicaciones tendría?
- 4) ¿Un contador puede tener más de un contacto por entrada? ¿Qué aplicación tendría?
- 5) ¿Cuándo se activan las salidas “E” y “F”? ¿Para qué se usarían?

VI. OBSERVACIONES Y CONCLUSIONES



PRÁCTICA N° 4

ENTRADAS ANALÓGICAS EN SOMACHINE BASIC

I. OBJETIVOS:

- Aprender a leer y escalar datos analógicos en el software SoMachine Basic.

II. MATERIALES Y/O EQUIPOS A UTILIZAR:

- Computador
- Software SoMachine Basic

III. MARCO TEORICO

Cuando una señal (tensión o corriente) ingresa a la entrada analógica, la señal se convierte de analógica a digital por un convertidor analógico-digital (A/D o ADC). El valor convertido que es proporcional a la señal analógica se envía a la CPU del PLC. Luego que el CPU haya procesado la información según el programa de usuario, el CPU da salida a la información a través de un convertidor digital-analógico (D/A o DAC) por medio del módulo de salida analógica.

En nuestro PLC TM221CE24R vienen incorporadas solo dos entradas analógicas de 0-10V, pero se pueden agregar módulos extras con entradas y salidas de corriente o voltaje.

El rango de conversión de nuestro PLC viene escalado por defecto de la siguiente manera:

Voltaje	Representación Decimal
0 a 10 V	0 a 1000

Tabla 1. Escala por defecto

La dirección de las entradas analógicas es %IWa.b y de las salidas %QWa.b siendo “a” el módulo y “b” el número de pin.

Para poder guardar estos datos analógicos y luego escalarlos debemos hacer uso de los bloques de operación para los cuales necesitaremos conocer algunas instrucciones básicas

Operador	Sintaxis
:=	[Op1 := Op2] Op1 toma el valor de Op2

Tabla 2. Asignación de Palabras

En nuestro caso Op1 será una Palabra de Memoria tipo entero (%M_{Wi}) que guardará el dato de la entrada analógica %IW0.1 (Op2). Ambas palabras son valores de tipo entero y para poder tener aproximaciones decimales tenemos que convertirlas a valores de tipo flotante mediante la instrucción de la tabla 3

Operador	Sintaxis
INT_TO_REAL	[Op1 := INT_TO_REAL(Op2)]

Tabla 2. Conversión de entero a real

Nuestro Op1 será una Palabra de Memoria tipo flotante (%M_{Fi}) que guardará el dato del Op2 que será la Palabra de Memoria tipo entero (%M_{Wi}) donde guardamos el dato de la entrada analógica %IW0.1

Finalmente para escalar deberemos usar los operadores básicos de adición, sustracción, multiplicación y división en base a la ecuación de la recta ($y=ma+b$) que deseemos tenga nuestra escala.

Operador	Sintaxis
+, -, *, /	[Op1 := Op2 operador Op3]

Tabla 3. Operaciones básicas

IV. DESARROLLO DE LA PRÁCTICA

1. Ingresamos al programa SoMachine Basic
2. Creamos un nuevo proyecto
3. Seleccionamos el PLC que vamos a utilizar
4. Vamos a la pestaña de programación y colocamos un Bloque de Operación como vemos en la Figura 1.

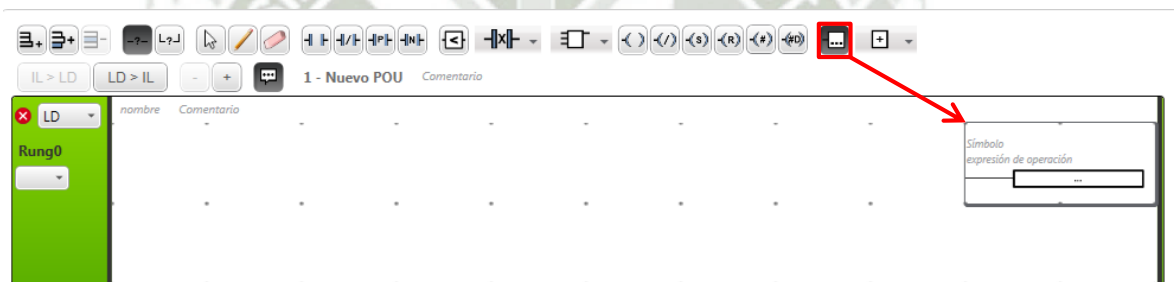


Figura 1. Bloque de Operación

5. Hacemos doble clic en “expresión de operación y escribimos la siguiente instrucción:
%MW0 := %IW0.0
6. Colocamos luego otros 2 bloques de operación a los cuales les colocamos las siguientes instrucciones:

%MF0 := INT_TO_REAL(%MW0)

%MF1 := %MF0 / 100.0

- Dibujamos una línea que una nuestros 2 últimos bloques con la línea del 1ro como vemos en la Figura 2.

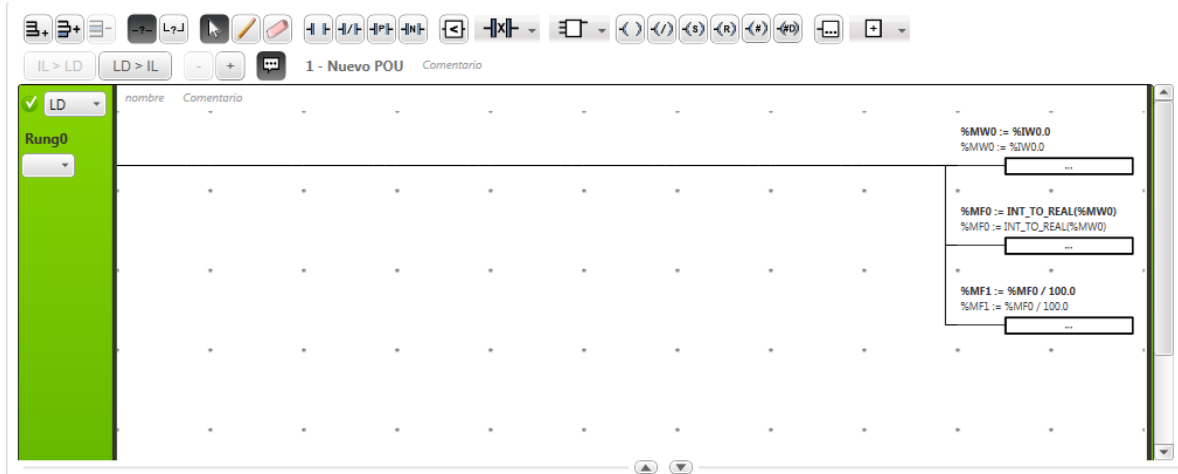


Figura 2. Unión de los Bloques de Operación

- Para tener una salida visible debemos comparar nuestro valor escalado con un valor dado por el usuario, para ello colocamos un “Bloque de comparación” en un nuevo escalón como muestra la Figura 3.

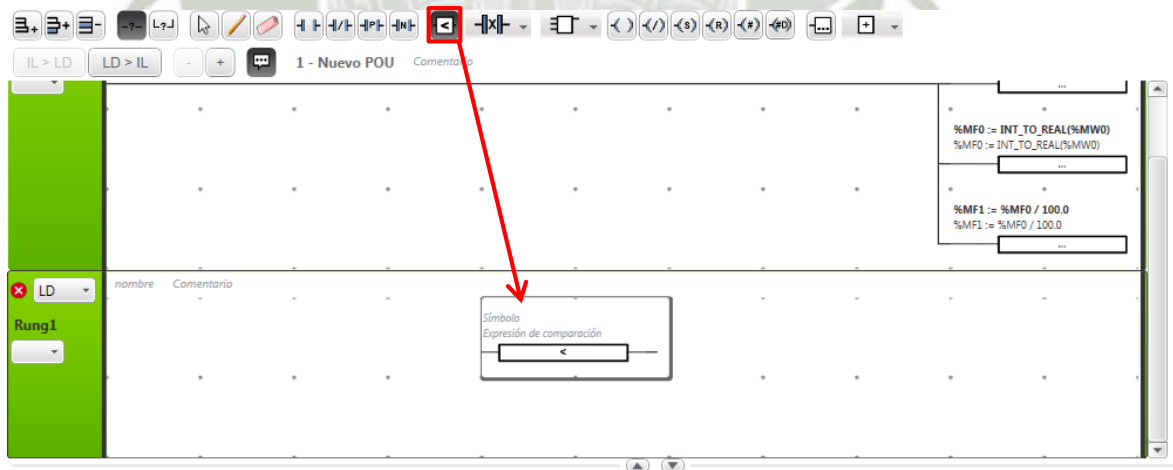


Figura 3. Bloque de Comparación

- Damos doble clic en Expresión de comparación y escribimos la siguiente instrucción:

$\%MF1 > 5.0$

10. Luego colocamos una bobina a la salida del bloque de comparación y la direccionamos a la salida %Q0.0.

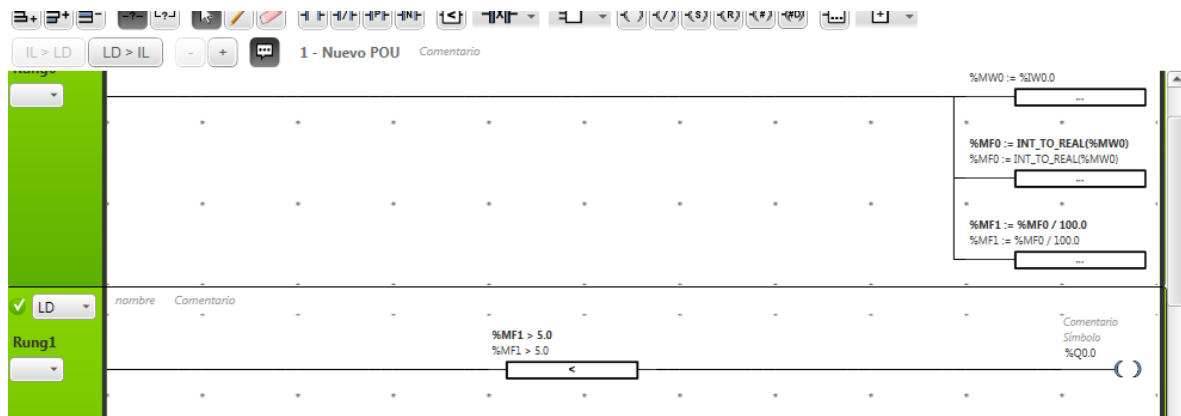


Figura 4. Salida a %Q0.0

11. Finalmente compilamos y probamos nuestro programa. En la ventana de simulación damos doble clic en la entrada analógica 0 y se abrirá una nueva ventana llamada “Establecer entradas analógicas” donde podemos modificar el valor de las entradas analógicas.

12. Podremos observar que al ser la entrada analógica mayor a 500, esta se escala a un valor mayor a 5 y por lo tanto se activa nuestra salida %Q0.0.

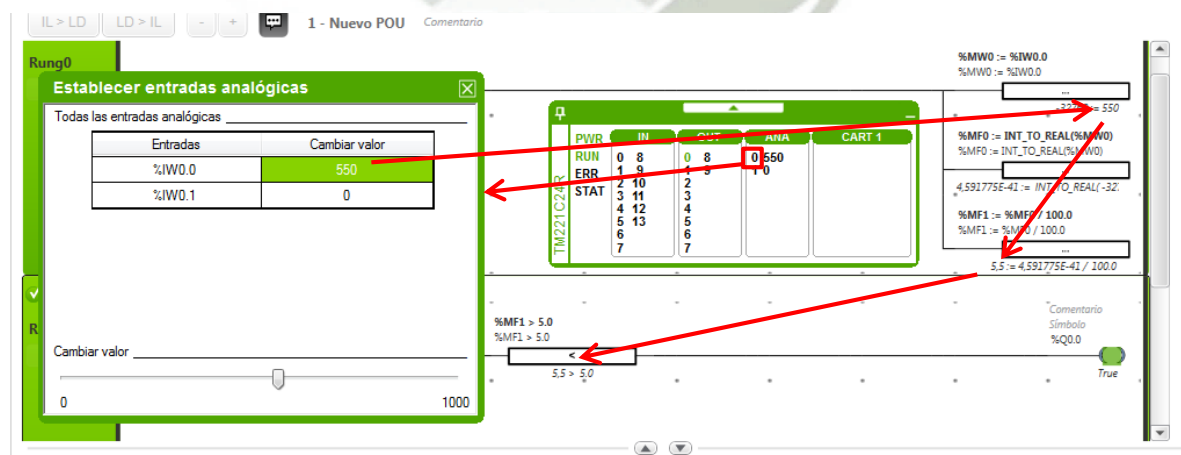


Figura 5. Simulación de entrada analógica

13. Implementar un programa con instrucciones matemáticas que permita monitorear la temperatura de un proceso mediante un transductor de temperatura con salida de 0 a 10 VDC conectado en el terminal de conexión 1 de las entradas analógicas del PLC. La señal de voltaje del transductor es proporcional al rango de 100° a 500°C. La temperatura del proceso debe permanecer entre 200° y 400°C. Si la temperatura se desvía de este rango se activan dos lámparas intermitentes.

- Si la temperatura es mayor a 400°C se activa la salida %Q0.1 a intervalos de 250 ms de prendido por 250ms de apagado (Lámpara HIGH).
- Si la temperatura está en el rango de 200° a 400°C se activa la salida %Q0.2, esta permanecerá permanentemente prendida (Lámpara NORMAL).
- Si la temperatura es menor a 200°C se activa la salida %Q0.3 a intervalos de 500 ms de prendido por 500ms de apagado (Lámpara LOW).

V. CUESTIONARIO FINAL

- 1) ¿Qué ocurre cuando se tiene un voltaje negativo en la entrada analógica?
- 2) ¿Qué ocurre cuando se tiene un voltaje mayor a 10V en la entrada analógica?
- 3) Represente en un plano cartesiano la recta de escalamiento del punto 13 de la práctica donde el eje X son los valores de voltaje y el eje Y los valores de temperatura.
- 4) Mencione todos los estándares de señales analógicas y explique sus ventajas y desventajas.
- 5) En el ejemplo de la práctica, ¿Dónde colocarías un START y un STOP?

VI. OBSERVACIONES Y CONCLUSIONES

PRÁCTICA N° 5

INTRODUCCIÓN A LA INTERFAZ DE VIJEEO DESIGNER v6.1

IV. OBJETIVOS:

- Reconocer la interfaz del software Vijeo Designer v6.1.
- Interactuar con la variedad de herramientas que nos otorga el software.

V. MATERIALES Y/O EQUIPOS A UTILIZAR:

- Computador
- Software Vijeo Designer v6.1

VI. DESARROLLO DE LA PRÁCTICA

1. Ingresamos al programa Vijeo Designer v6.1.
2. Hacemos clic en “Crear un nuevo proyecto” y luego en “Siguiente”.

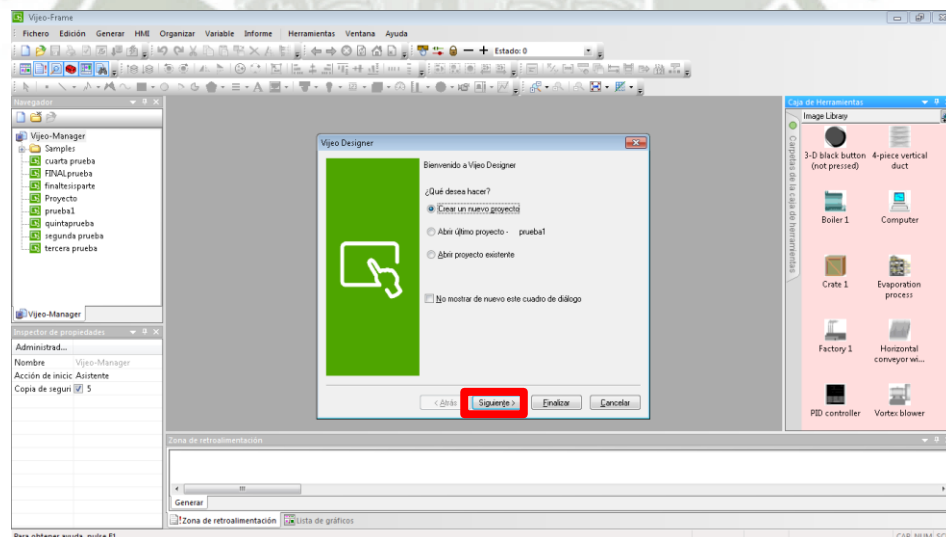


Figura 1. Creación de nuevo proyecto

3. Asignarle un nombre al proyecto, elegir la cantidad de destinos y si se desea una contraseña de protección de archivos aceptar y poner la contraseña.

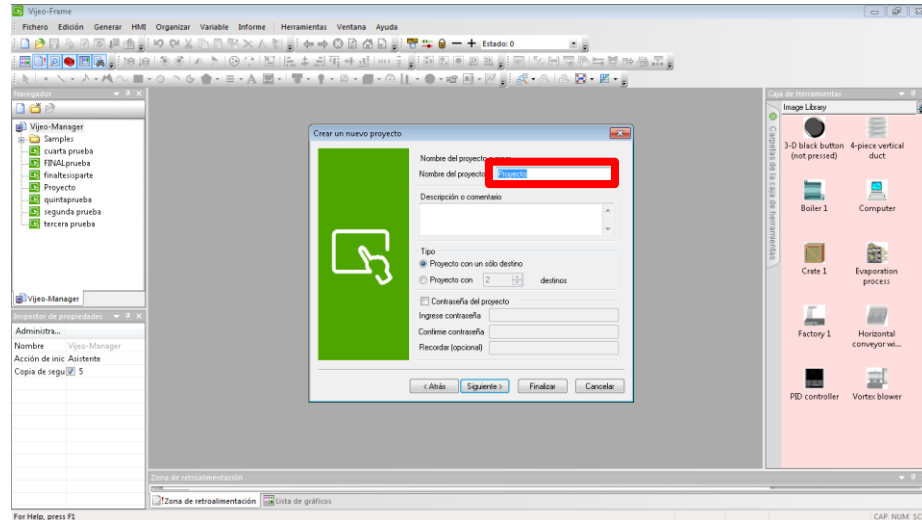


Figura 2. Nombre, descripción y destinos.

4. Luego de darle a siguiente asignarle un nombre a el destino o el nombre del proyecto del HMI, luego elegir el tipo de componente que se va a conectar.

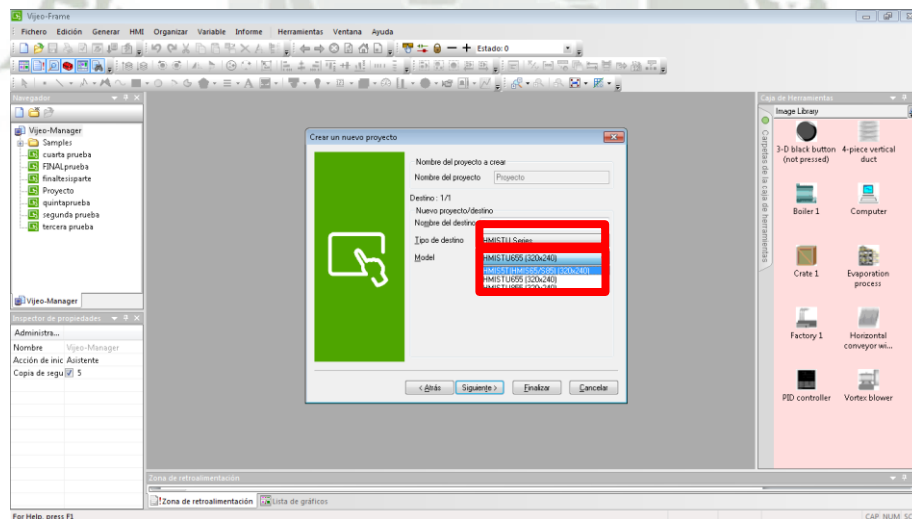


Figura 3. Elección del tipo de destino.

5. Luego asignarle una dirección IP valida y una máscara de subred que deberá ser la misma del HMI destino, en el caso que se requiera alguna función adicional a la que el programa

use que diese el proveedor del equipo se debe activar la casilla auditoría para tener una asesoría online.

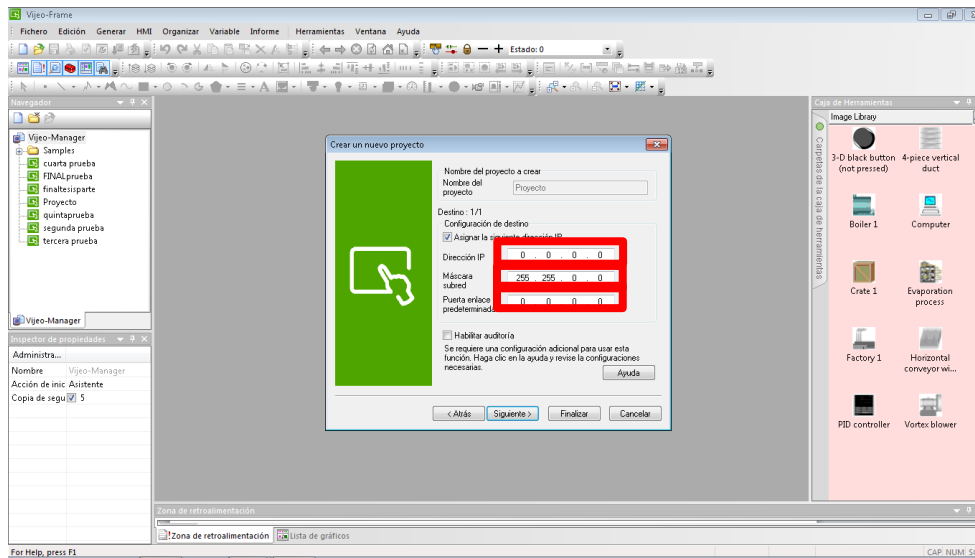


Figura 4. Asignación de la dirección IP y Máscara de Subred.

- En la última parte de la creación de un nuevo proyecto se podrá añadir distintos drivers o equipos para conectar desde el HMI destino como PLC's, y darle click a finalizar.

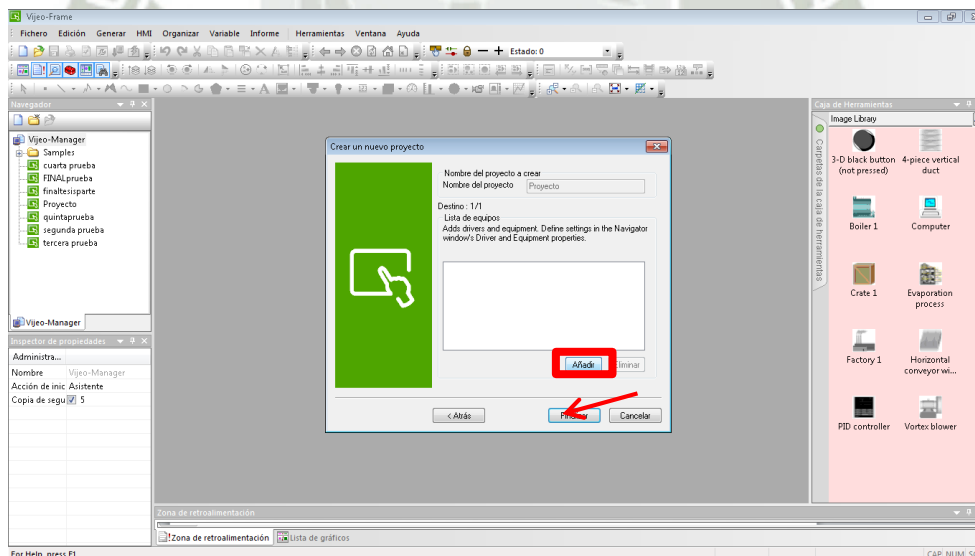


Figura 5. Añadir Drivers y Controladores.

7. Una vez terminado esto hay que familiarizarnos con la interfaz del software.

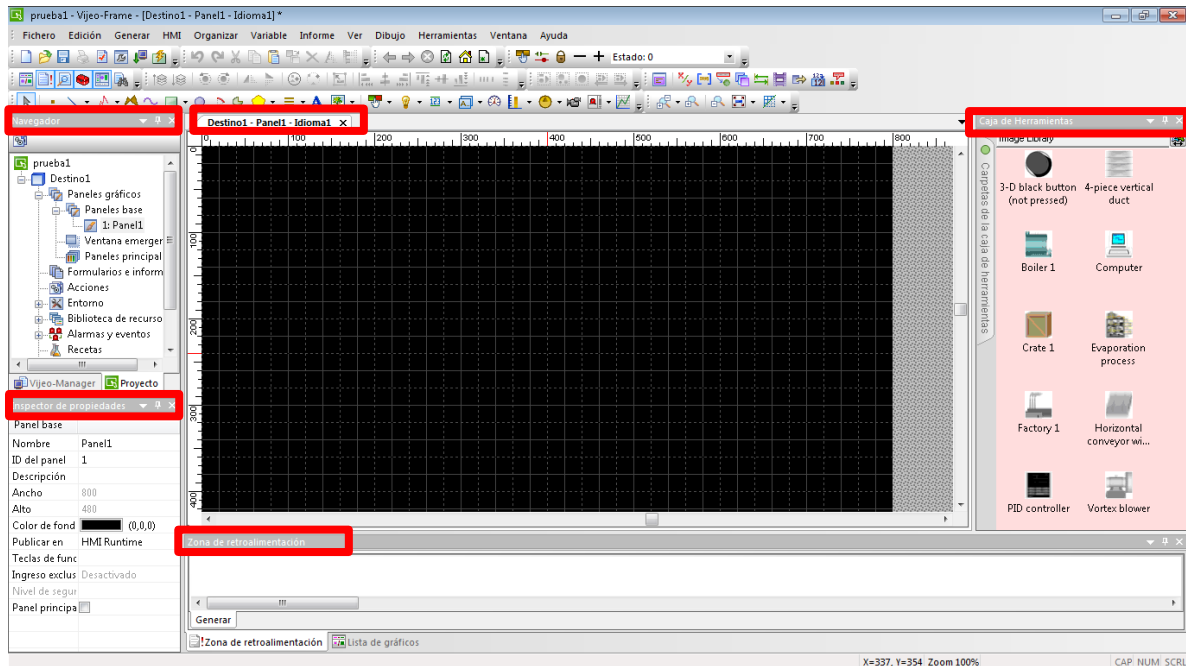


Figura 6. Interfaz de Vijeo Designer.

8. Al hacer click sobre Destino1 se abre el panel de propiedades del mismo. Se puede modificar nombre, descripción, tipo, modelo, etc. Dese aquí se pueden modificar todos los parámetros en la pestaña general, red, hardware, etc. Dese aquí se pueden modificar todos los parámetros en la pestaña general, red, hardware, etc.

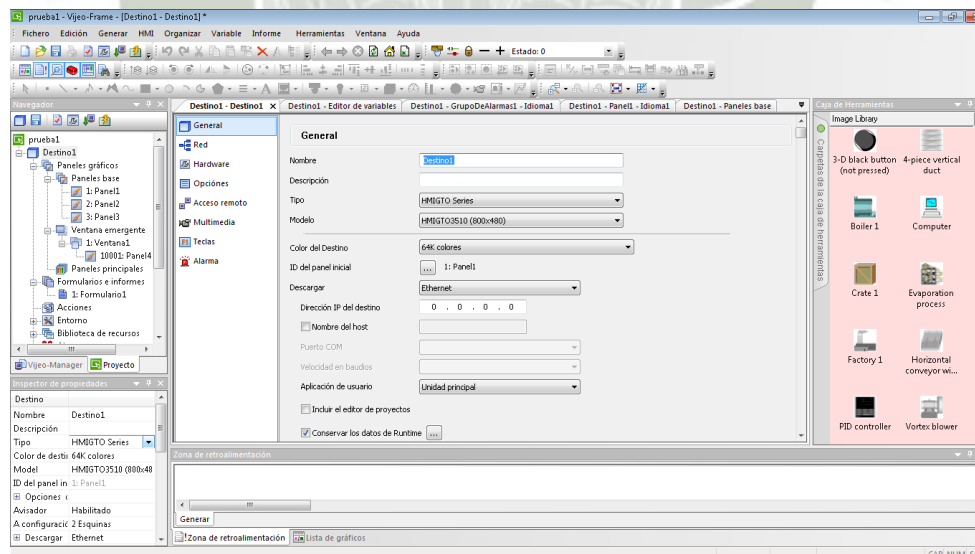


Figura 7. Propiedades del Destino1.

9. Todos los gráficos, opciones y animaciones se trabajan en los paneles, para crear un nuevo panel se debe hacer click derecho y luego click en “Nuevo panel” y luego asignarle un nombre.

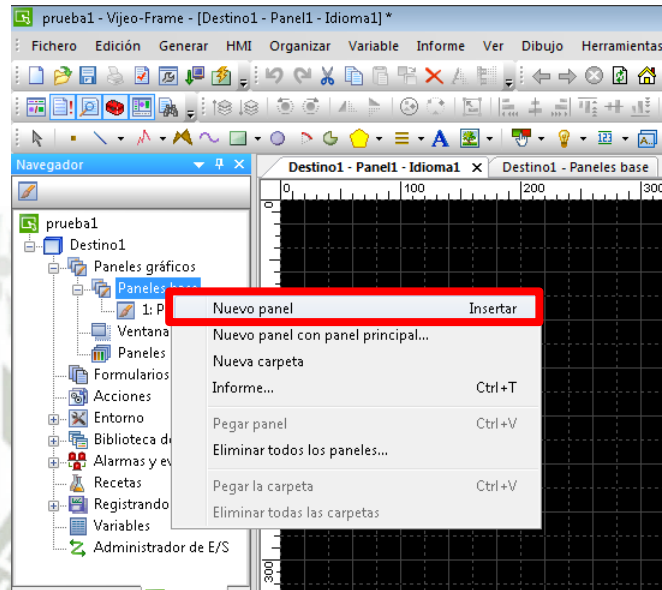


Figura 8. Creación de un Nuevo Panel.

10. Las ventanas emergentes son paneles pequeños que aparecerán sobre los paneles en el caso de habilitarse alguna opción.

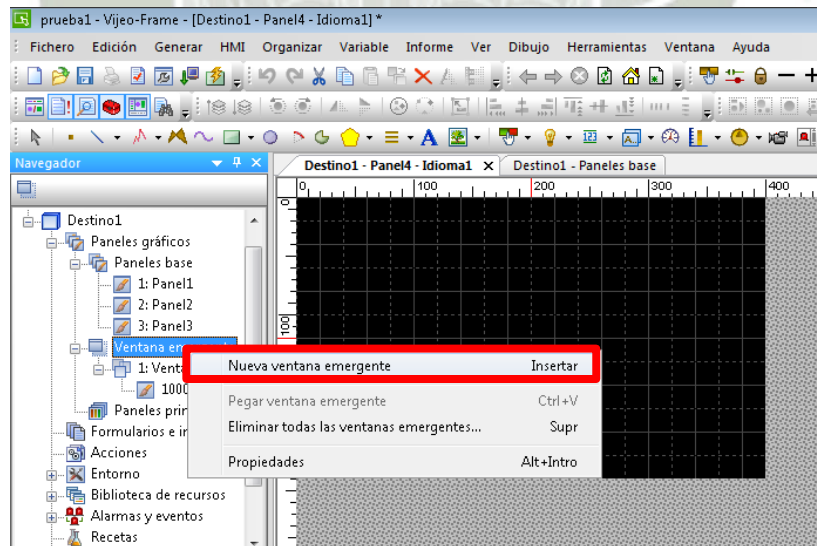


Figura 9. Creación de Nueva ventana emergente.

11. En la opción acciones es donde se podrá programar cada una de las acciones a realizar por los gráficos insertados en los paneles base. Darle click a “Nueva acción” y luego elegir el tipo de disparador o sea como se activará la acción, al activar la casilla “Activar Enclavamiento” se elige una variable que habilitará el inicio de la acción, en frecuencia se elige el tiempo en el que se repetirá la acción.

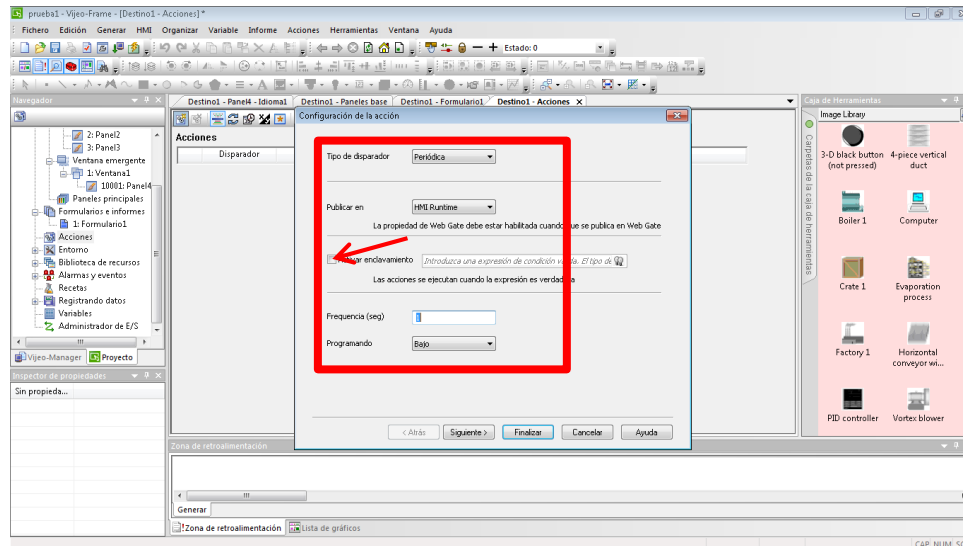


Figura 10. Configuración de una Nueva acción.

12. Luego de darle click a siguiente se elegirá el tipo de operación que tendrá la acción. La mayoría de las operaciones nos enseña cómo realizarlas sin mayor dificultad así como que valores deberán ser asignados para cada variable, la opción más completa y la más complicada es la opción de script. La cual nos permite programar en lenguaje C++ acciones más variadas y completas para ello elegimos “Script” en el menú de operaciones.

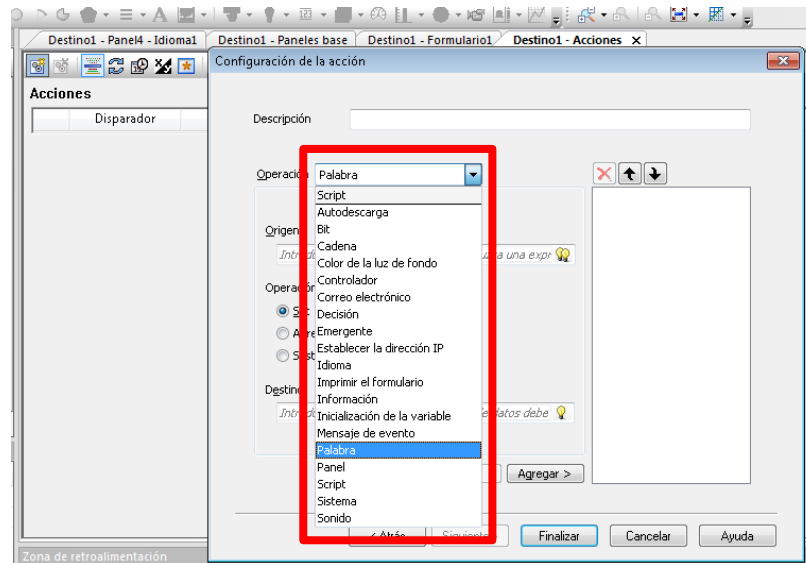


Figura 11. Operaciones de Variables.

13. Y nos sale la ventana de Script en donde con toda facilidad podemos programar en C++, luego le damos a validar para ver si no existe algún error en donde muy explícitamente nos explicará si existe o no error y como solucionarlo y finalmente le damos click a agregar.

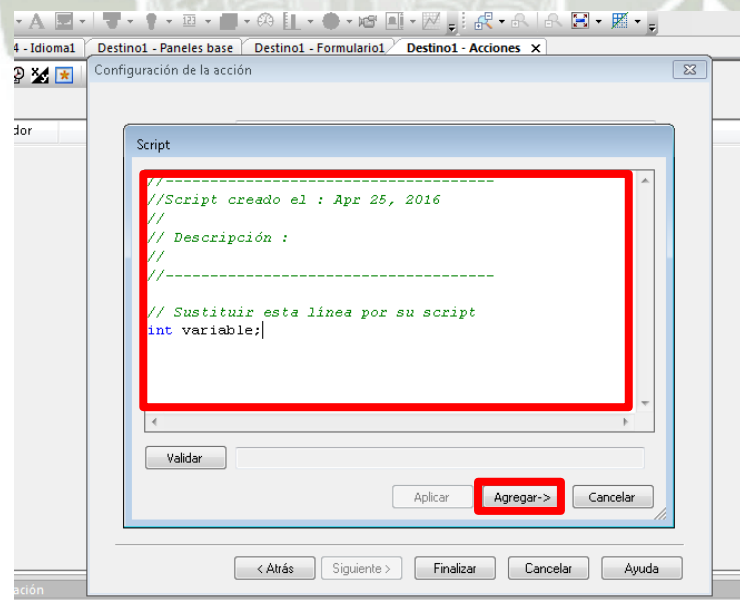


Figura 12. Script.

14. En la opción “Alarmas y eventos” es para activar alguna alarma o algún evento que se requiere en algún momento que alguna variable sea activada parcial o totalmente.

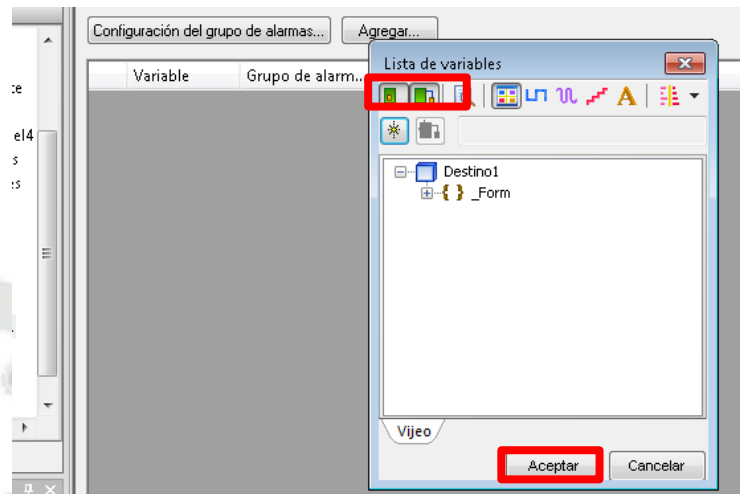


Figura 13. Configuración de Alarmas.

15. En la opción de variables al darle click a “Nueva Variable” nos aparece otra ventana en donde podemos especificar el nombre, descripción y tipo de datos que se almacenaran en variables, tenemos 4 tipos notables: Booleano, Entero, Flotante y Texto. Luego se deberá elegir si será una variable interna o externa, interna será cuando es una variable que el propio software trabaja y externa cuando la variable proviene de un componente externo, es aquí en donde trabajaremos con variables que provengan de un PLC.

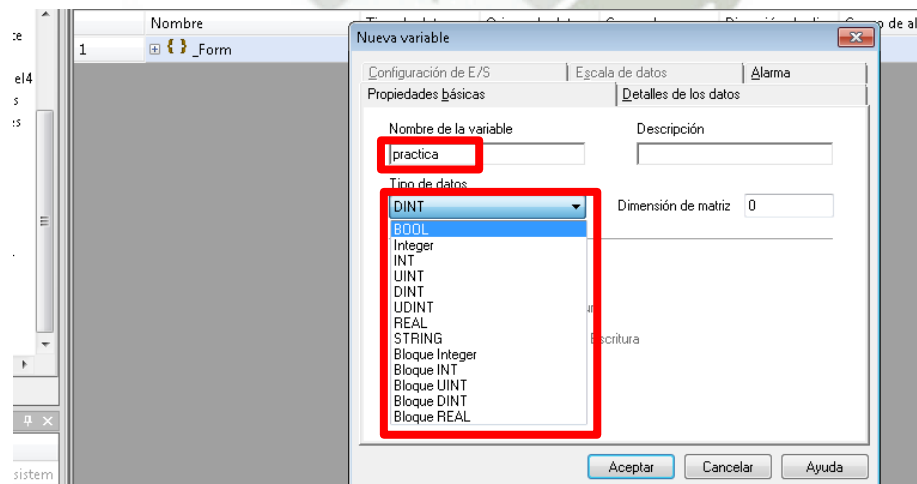


Figura 14. Configuración de Nueva variable.

16. En el caso de elegir que el origen de datos es externo, nos aparecerá activas 3 opciones:

La primera es grupo de escaneo en donde elegiremos de donde provendrá la variable externa. La segunda es dirección de dispositivo, aquí elegiremos cual es la dirección física que tendrá nuestra variable en el componente de donde provenga, por ejemplo %Q0.0.1. Y la tercera es una dirección opcional que se nos da en caso tenga una etiqueta en el componente de donde provenga la variable.

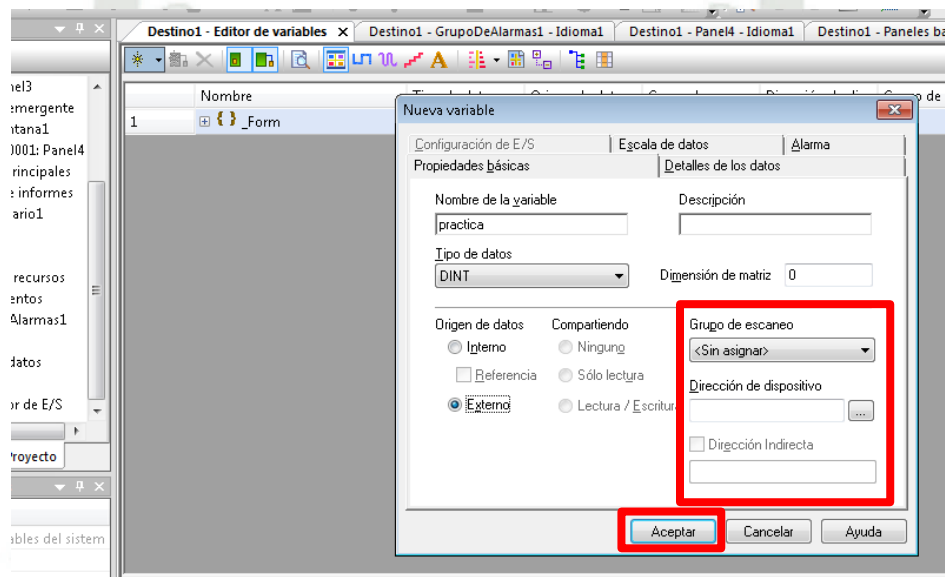


Figura 15. Configuración de Nueva variable externa.

17. Para poder comunicar el terminal HMI con algún controlador, se debe de ir a la sección de “Administrador de E/S” luego click derecho y luego click en “Nuevo controlador”.

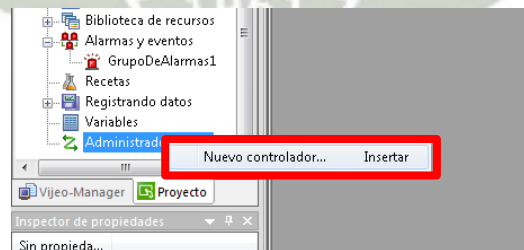


Figura 16. Configuración de un nuevo controlador.

18. Aquí se seleccionara el tipo de controlador, el nombre del fabricante y el tipo de equipo y luego click en “Aceptar”.

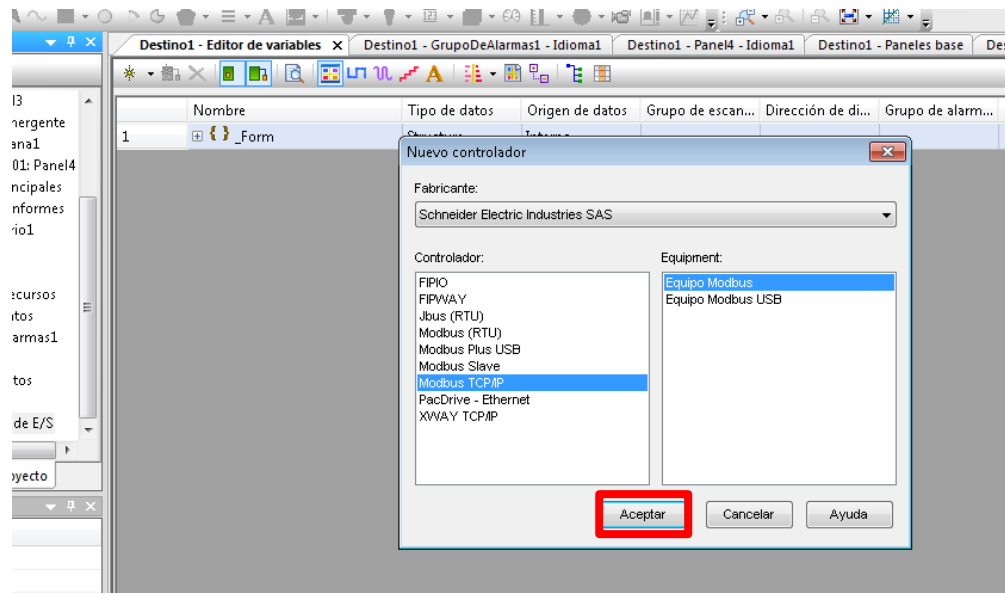


Figura 17. Elección de un nuevo controlador.

19. Y para finalizar se elegirá la dirección IP del equipo a conectar y se dará click en “Aceptar”.

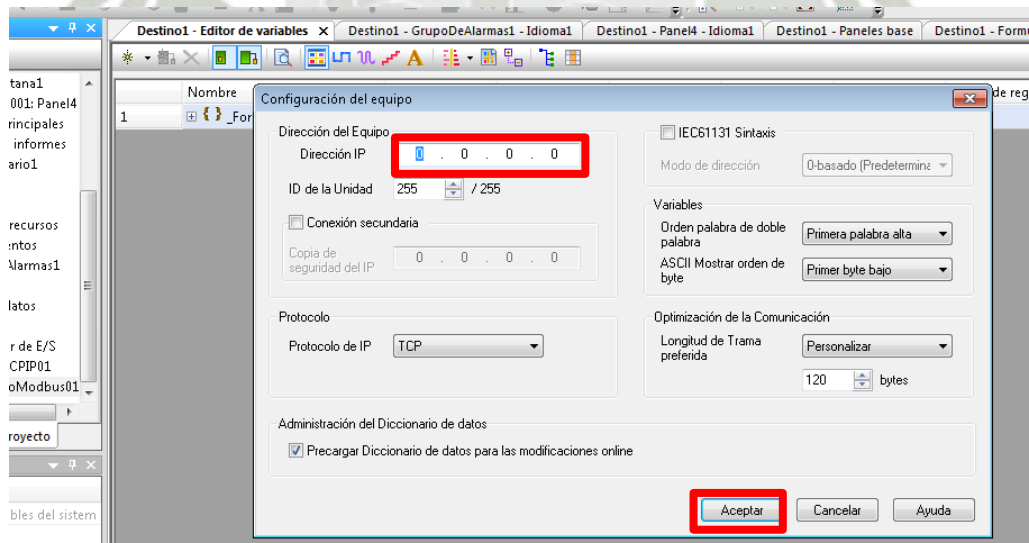


Figura 17. Asignación de Dirección IP del nuevo controlador.

20. Describa que función cumple la casilla de activación IEC61131 Sintaxis.

21. Averigüe para qué sirve la opción Formularios.

22. Explique la función de la Zona de Retroalimentación.

VII. CUESTIONARIO FINAL

- 1) ¿Qué es el protocolo IP?
- 2) ¿Qué significa Modbus?
- 3) ¿Qué función tiene la Lista de Gráficos?
- 4) ¿Cuál es la dirección de una variable externa que proviene de la salida 5 de un PLC?

VIII. OBSERVACIONES Y CONCLUSIONES

PRÁCTICA N° 6

CREACION DE OBJETOS Y COMUNICACIÓN EN VIJEO DESIGNER v6.1

I. OBJETIVOS:

- Aprender a crear objetos, figuras y botones en el software Vijeo Designer v6.1.
- Aprender a crear acciones y variables.
- Descargar información al terminal HMI a utilizar.

II. MATERIALES Y/O EQUIPOS A UTILIZAR:

- Computador
- Software Vijeo Designer v6.1

III. DESARROLLO DE LA PRÁCTICA

1. Ingresamos al programa Vijeo Designer v6.1 y creamos un nuevo proyecto.
2. Crearemos un rectángulo con la opción rectángulo.

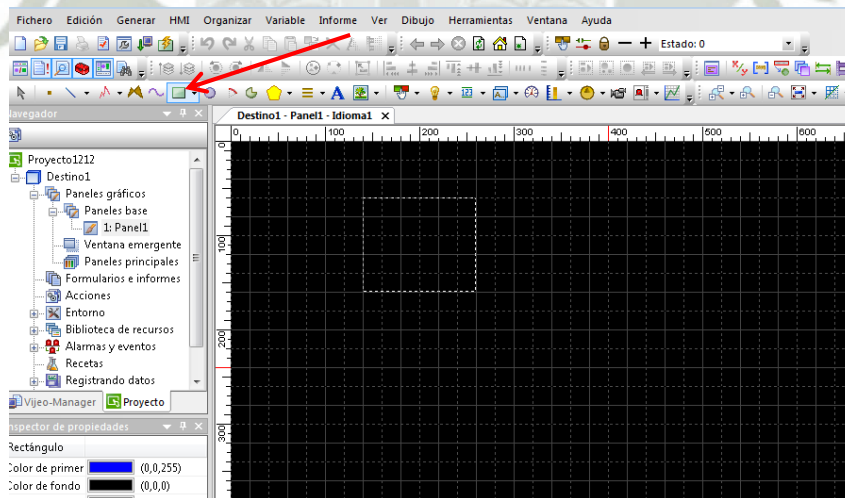


Figura 1. Creación de un rectángulo.

3. Al crearlo el rectángulo saldrá la ventana de propiedades, donde podremos seleccionar la animación a realizar. Dentro de las opciones tenemos color, relleno, tamaño, posición, girar, pulsación y visual. Describiendo cada una de ellas:

-Color.- Cambia el color de relleno del rectángulo al activarse la variable designada.

-Relleno.- Aumenta o disminuye el total del relleno de forma vertical o de forma horizontal al activarse la variable designada.

-Tamaño.- Cambia el tamaño de la figura al activarse la variable designada.

-Posición.- Cambia la posición de la figura en el panel de trabajo al activarse la variable designada.

-Girar.- Gira la figura al activarse la variable designada.

-Pulsación.- Se despliegan varias opciones de igual manera que en el menú de acciones.

-Visual.- Activa animaciones visuales como aparecer o desaparecer y parpadear.

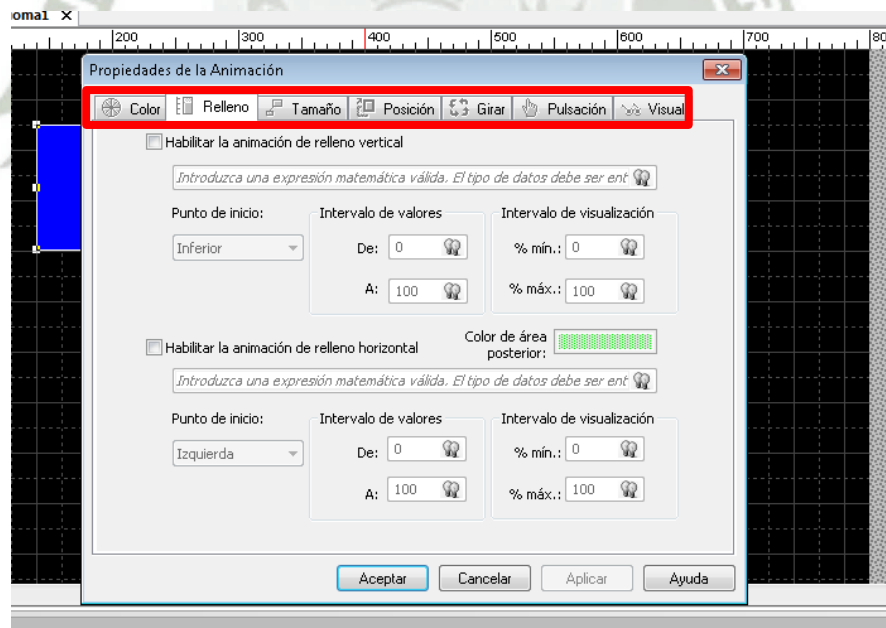


Figura 2. Propiedades de la Animación de un objeto.

- Para insertar una imagen se presiona en el icono de imagen y se abre la ventana “Abrir” en donde elegiremos la imagen que deseamos importar.

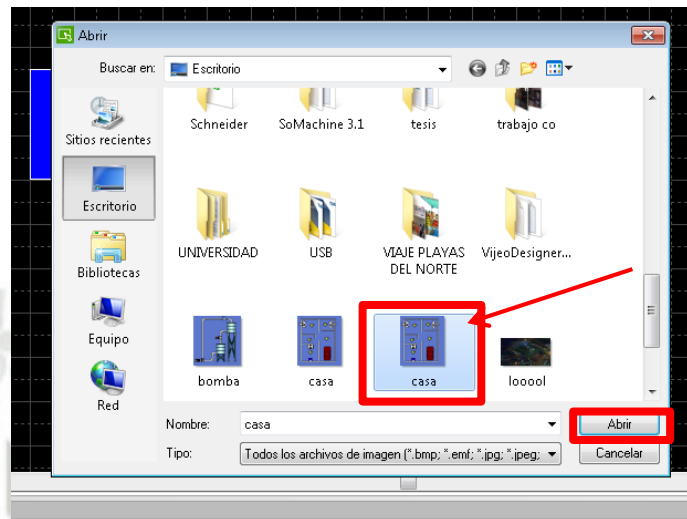


Figura 3. Importando una imagen.

- Crearemos también un botón o interruptor, tras crearlo nos aparecerá la ventana de configuración del interruptor en donde hay varias pestañas, en la pestaña general en la sección de al pulsar se elegirá la acción que ocurrirá tras pulsarlo, durante se pulsa y al liberar el botón.

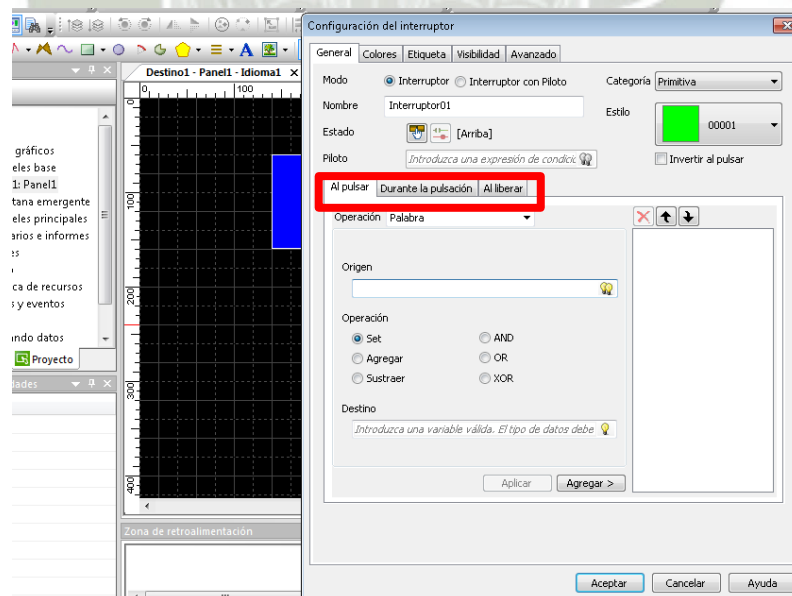


Figura 4. Configuración de un Interruptor.

- En la pestaña de Etiqueta, al activarse Tipo de etiqueta Estático se elegirá el texto que aparecerá sobre el botón.

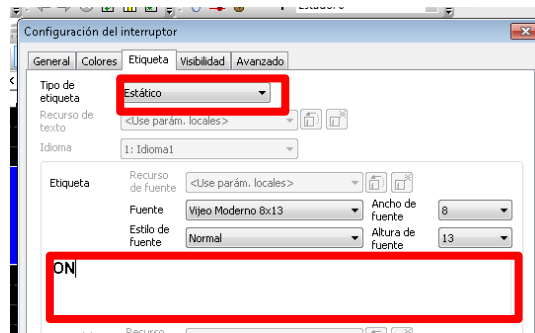


Figura 5. Pestañas de la configuración de un botón.

- Ahora jalaremos las imágenes de la caja de herramientas, en ésta sección de la interfaz se pueden elegir entre diversas imágenes ya predefinidas para poder hacer más atractivo nuestro SCADA, tan solo es necesario jalar la imagen sin soltar el click y ponerlo en el lugar que lo queramos tener.

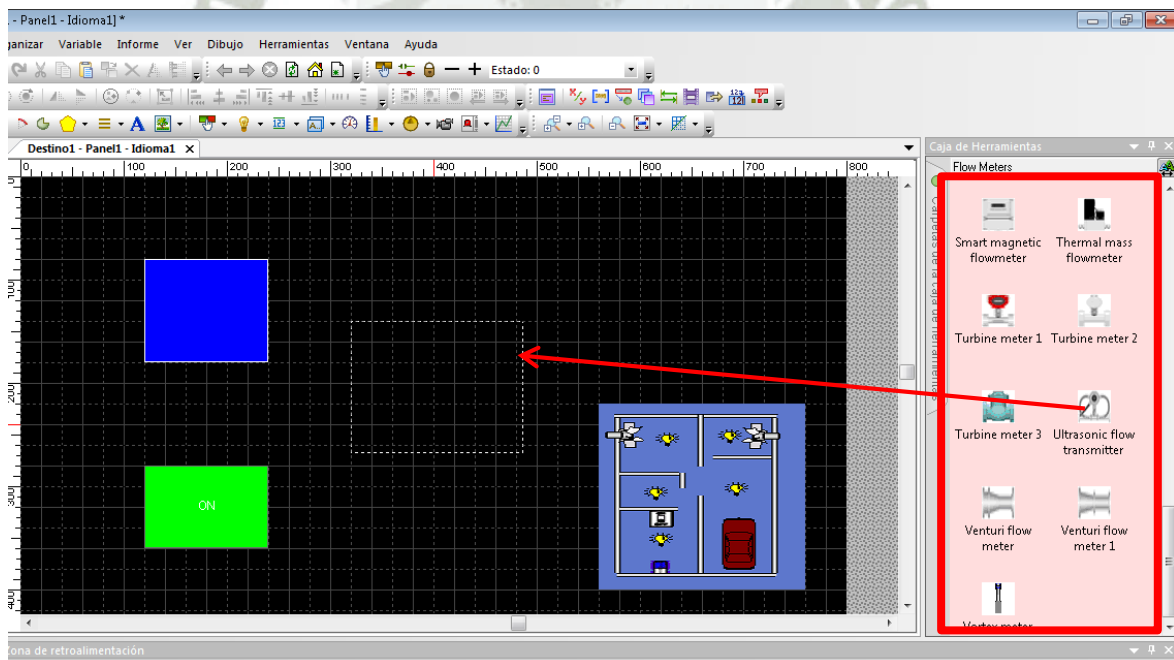


Figura 6. Insertando imágenes prediseñadas.

8. Ahora pondremos animaciones a lo que hemos creado. Para el rectángulo le insertaremos una animación de color, cambiaremos su color al activar una variable llamada ON. Cuando salga el Teclado editor de expresiones elegiremos crear “nueva variable” y luego le pondremos el nombre “ON” de tipo “BOOL” e “Interna”.

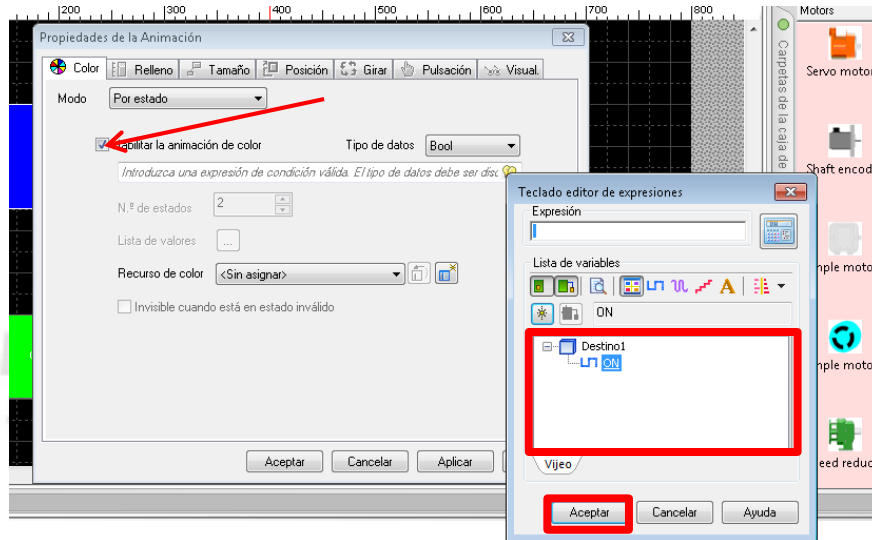


Figura 7. Asignando una variable de activación para la animación.

9. En Recurso de color, le daremos asignaremos un nuevo recurso llamado “RecursoDeColor001”.

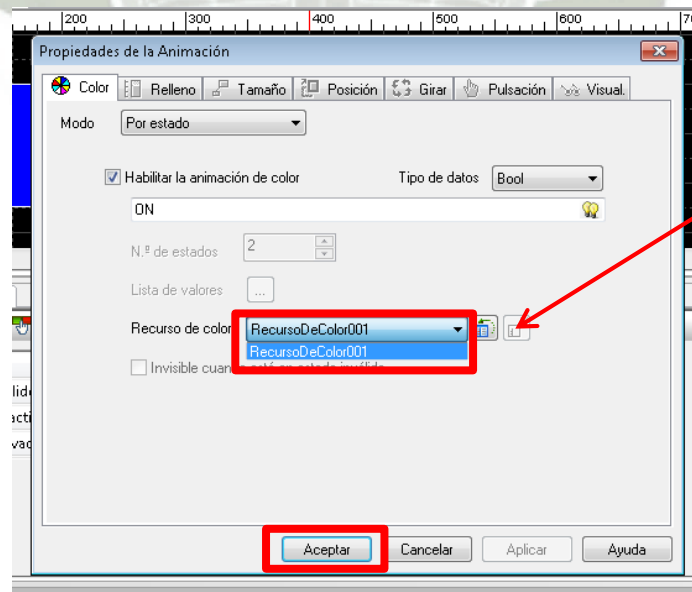


Figura 8. Propiedades de la animación.

10. Para animar el rectángulo es necesario tener un botón de activación de animación, el cual fue creado anteriormente. En la sección “Al Pulsar” seleccionaremos en operación “Bit”, y “Alternar” y en Destino seleccionaremos la variable “ON” y luego le daremos click a “Agregar” sino se diera click a “Agregar” no se carga la operación.

Al presionar el botón activa la variable “ON” que activa el cambio de color del rectángulo.

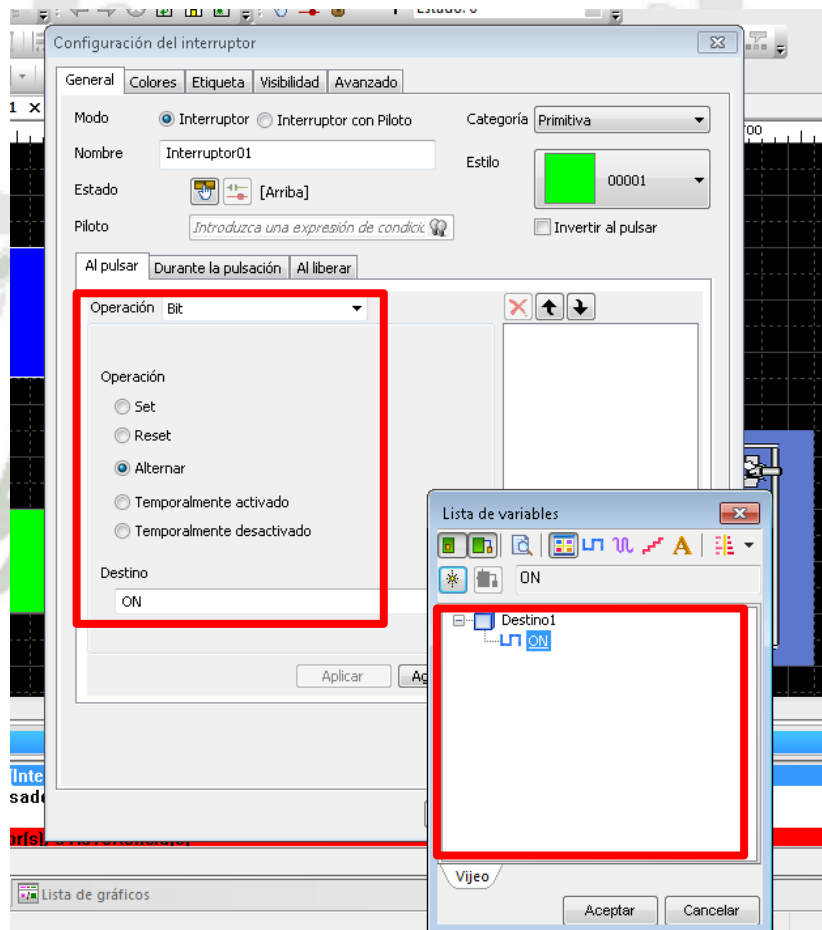


Figura 9. Configuración del Interruptor para encendido.

11. Para la figura que hemos traído desde la Caja de Herramientas, le pondremos la animación de parpadeo, entonces cuando se active la variable ON al pulsar el botón verde, activará la opción de parpadeo que hará parpadear la imagen hasta que se desactive la señal en ON.

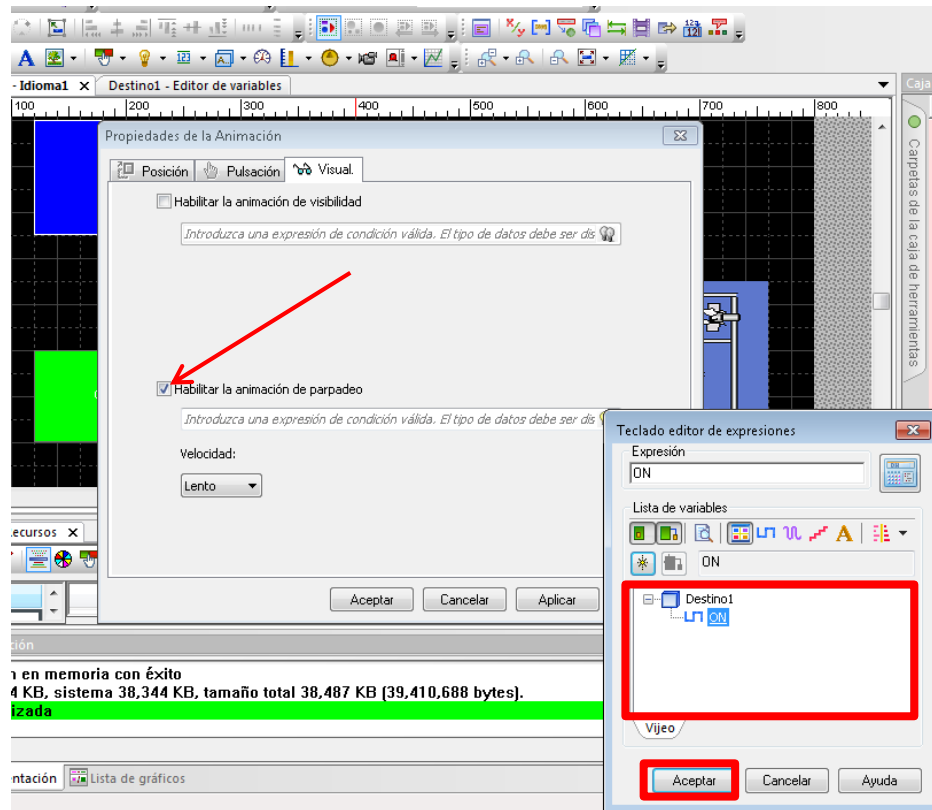


Figura 10. Habilitando la animación de parpadeo.

12. Y para la imagen traída desde nuestros archivos, también le pondremos una animación visual, pero en éste caso será de visibilidad. Entonces cuando se active la variable ON aparecerá la imagen y cuando se desactive desaparecerá.

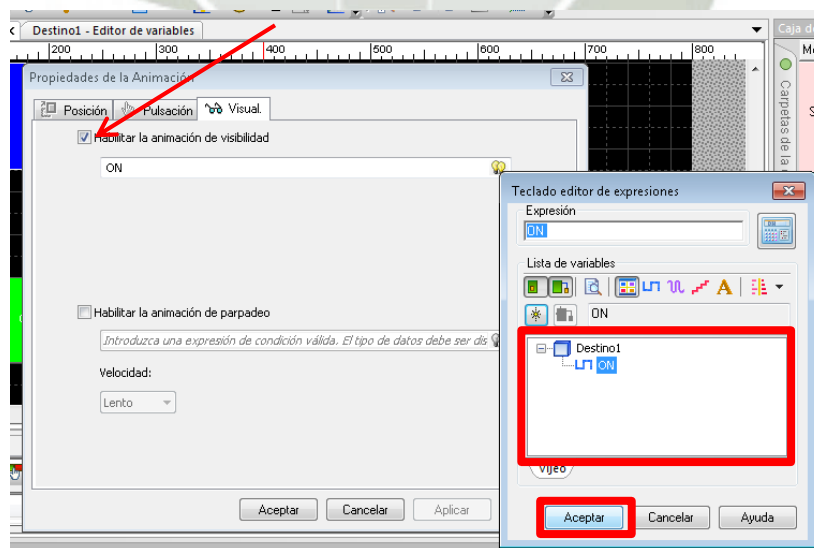


Figura 11. Habilitando la animación de visibilidad.

13. Ahora crearemos un nuevo panel, como en la práctica anterior. En este panel probaremos una acción.

14. Crearemos un botón con la etiqueta “LLENAR”, ésta llenará un tanque.

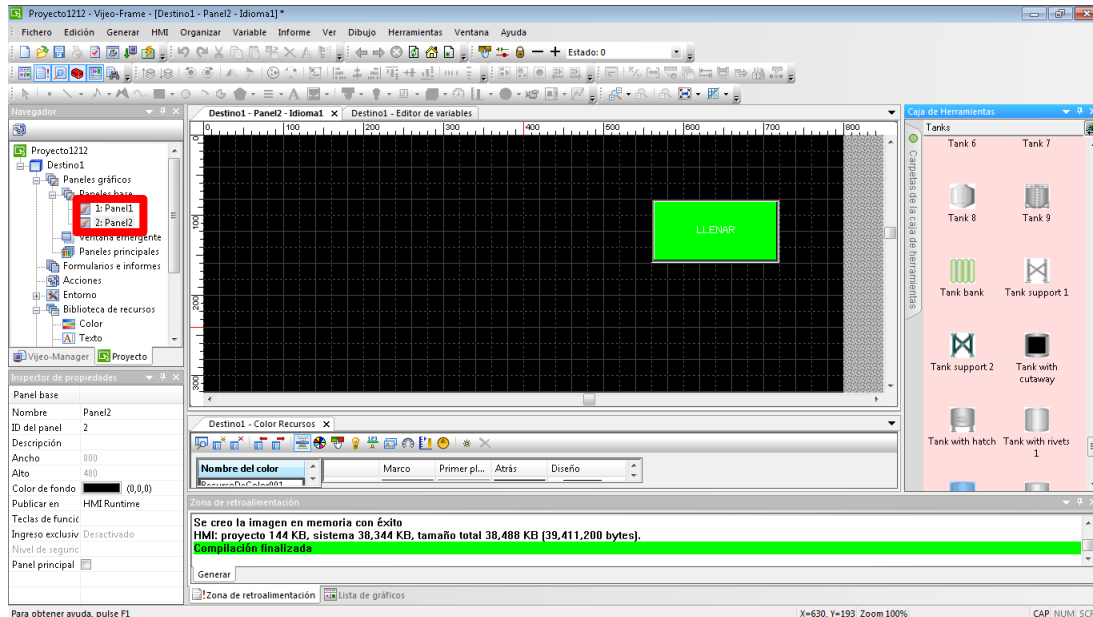


Figura 12. Panel 2

15. Y como ya hemos visto, le asignaremos una acción, en este caso crearemos una nueva variable llamada “LLENAR” de tipo “BOOL” e “Interna”.

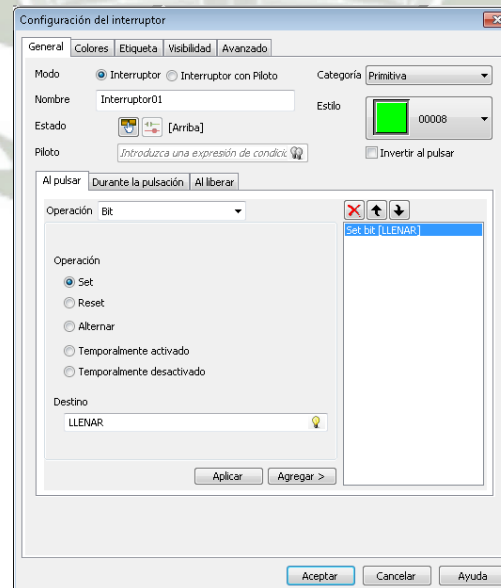


Figura 13. Configurando un botón en el Panel 2.

16. Ahora jalaremos otra imagen de la “Caja de herramientas” en éste caso pondremos una imagen de un tanque.

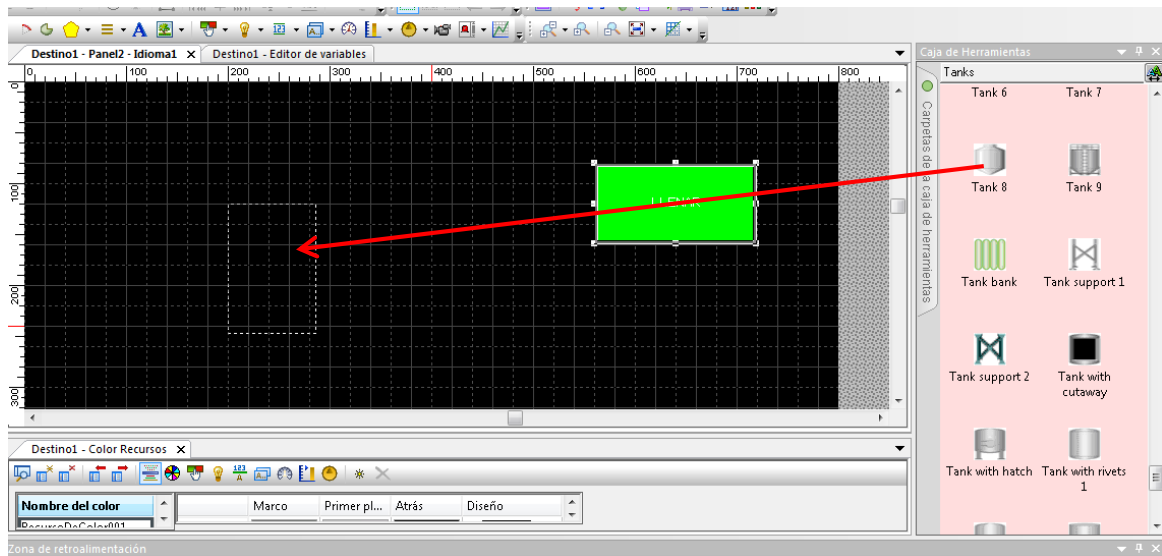


Figura 14. Insertando imagen del tanque.

17. Ahora sobre el tanque crearemos un rectángulo para simular el nivel del tanque.

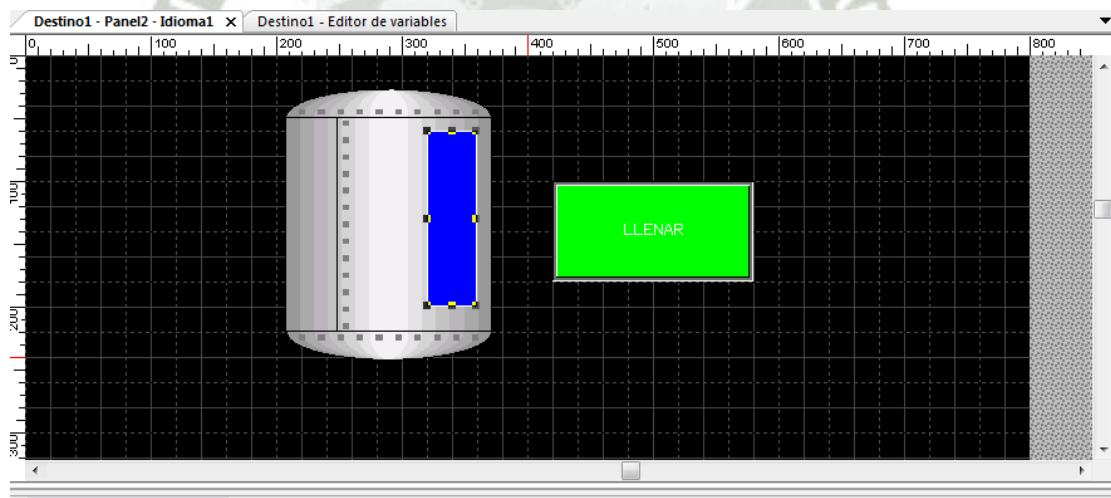


Figura 15. Vista del tanque y botón.

18. Ahora al rectángulo se le asignará la animación de “Relleno”. Y el valor que lo activará será una nueva variable pero de tipo “INT” llamada “TANQUE”.

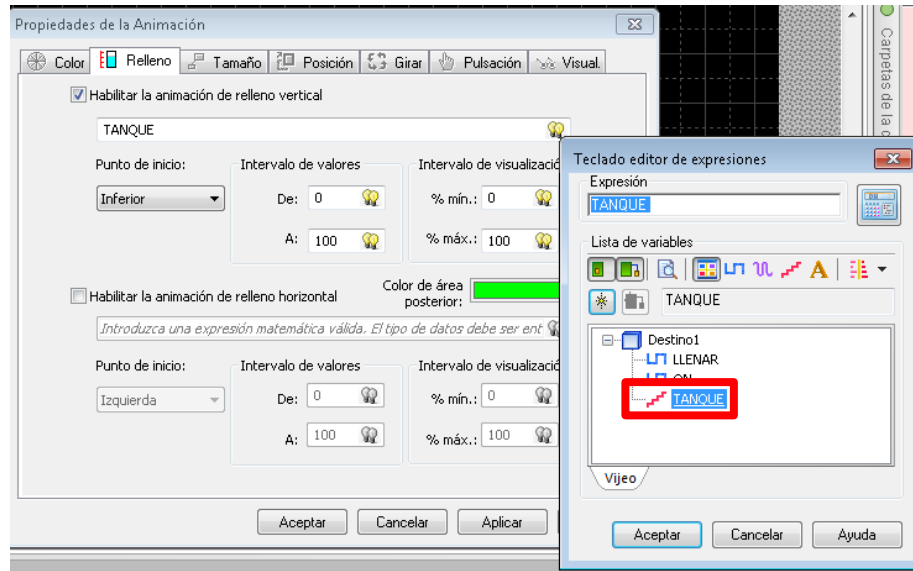


Figura 16. Habilitando la animación de relleno.

19. Pero de por si no actuará puesto que necesita una acción que desencadene la animación relleno. Por lo cual iremos a la sección de “Acciones” y crearemos una nueva acción, será una acción “Periódica” puesto que se repetirá a lo largo del tiempo, se activará la pestaña “Activar enclavamiento” para designar a la variable que activara la animación de llenar el tanque.

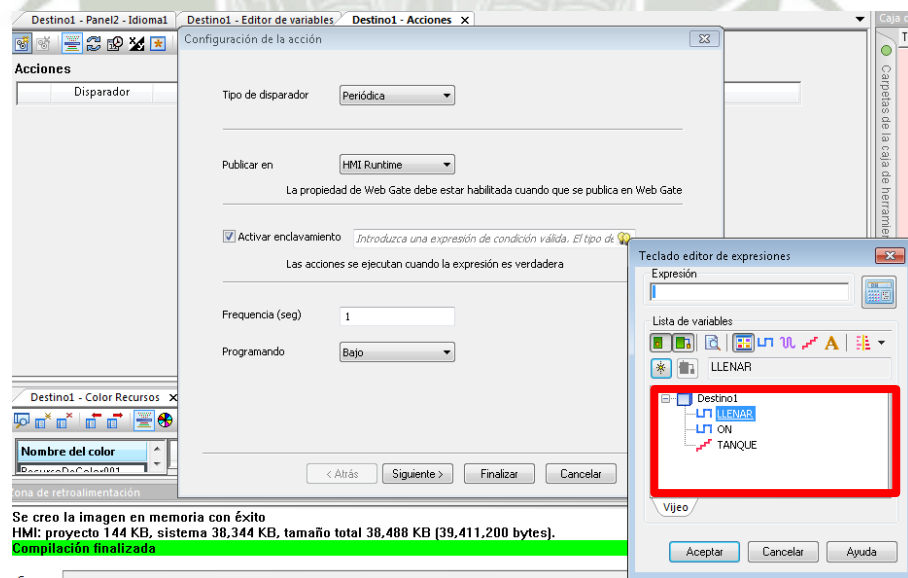


Figura 17. Configuración de la acción.

20. Luego en la siguiente parte, se elegirá la operación de “Script”, con el siguiente código.

-Primero declaramos una variable de tipo entero llamada llenar.

-Luego tomará los valores de la variable de tipo “INT” de TANQUE.

-Luego se creará una condición “if-else” que nos dice “Si el valor de llenar es menor que 100 entonces seguirá aumentando el valor de llenar”, “Sino tendrá el valor de 100”.

-Finalmente en la variable TANQUE se escribirá el valor de la variable declarada “llenar”.

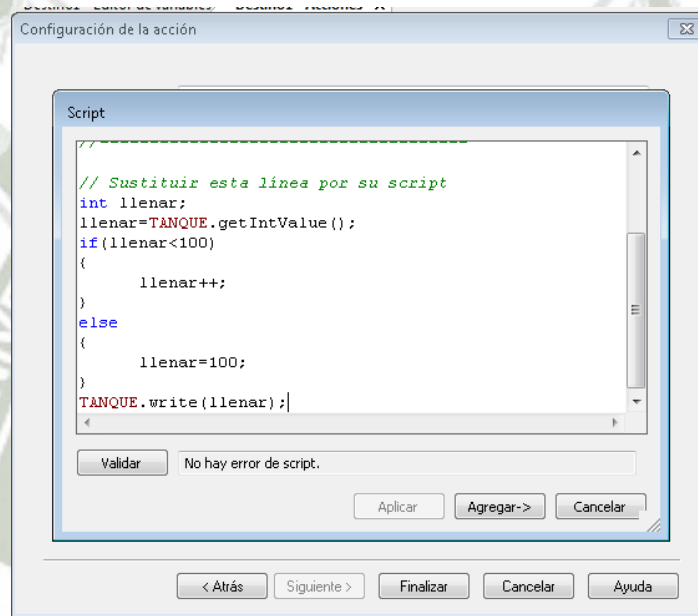


Figura 18. Script del llenado de tanque.

21. Así debería aparecer la sección de Variables.

	Nombre	Tipo de datos	Origen de datos	Grupo de escan...	Dirección de di...	Grupo de alarm...	Grupo de regist...
1	LLENAR	BOOL	Interno			Desactivado	Ninguno
2	ON	BOOL	Interno			Desactivado	Ninguno
3	TANQUE	INT	Interno			Desactivado	Ninguno

Figura 19. Tabla de variables.

22. Pero para llevar a cabo dicha animación se deberá primero crear un interruptor que nos lleve del “Panel1” al “Panel2” por lo cual crearemos un nuevo botón y le pondremos de etiqueta “PANEL 2”. Pero para que lleve a cabo la acción de cambiar de panel, en la sección de Operación se deberá elegir “Panel” y en ID del panel escribiremos “2” o buscaremos dicho panel y luego le daremos click a “Agregar” y “Aceptar”.

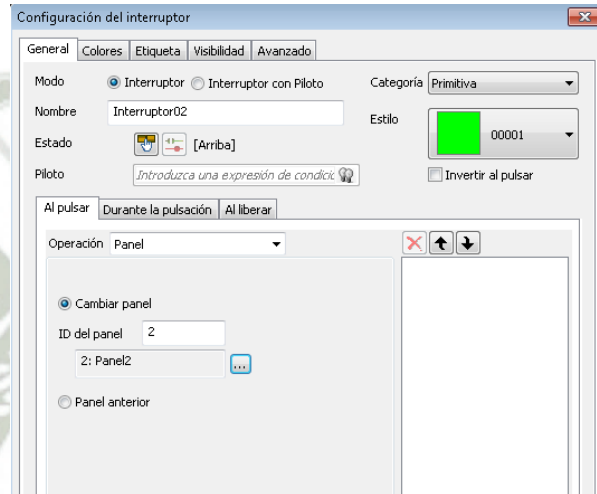


Figura 20. Configuración de un tercer botón.

23. Luego para simularlo se le da click derecho a “Destino1” se valida y luego se da click a “Iniciar simulación de dispositivo”.

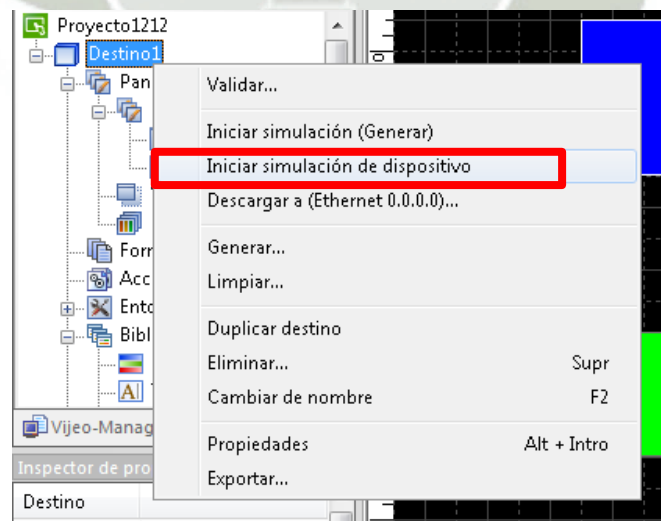


Figura 21. Iniciando simulación.

24. Y al aparecernos la nueva ventana deberíamos poder activar el botón “ON” y el botón “PANEL2”.

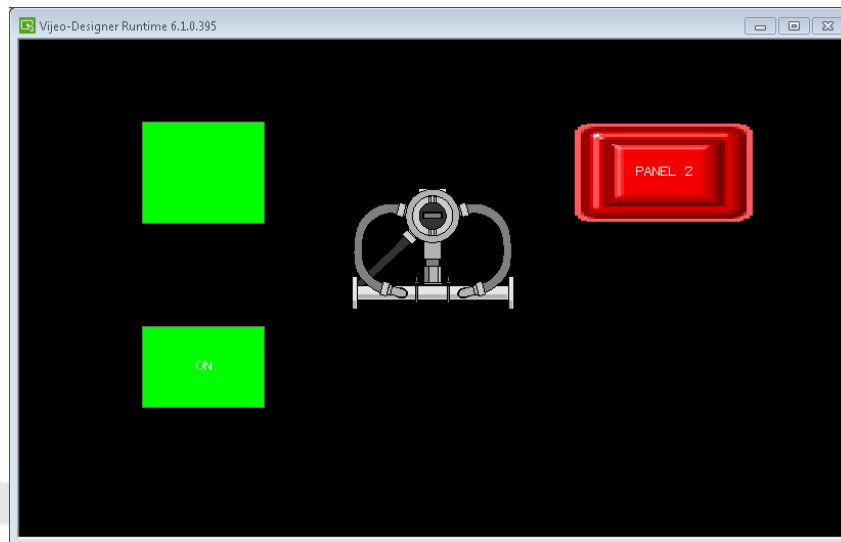


Figura 22. Simulación activada.

25. Y al presionar el botón “ON” deberá cambiar el color del rectángulo, deberá aparecer la imagen traída de nuestros archivos y deberá parpadear la imagen traída de Caja de Herramientas.

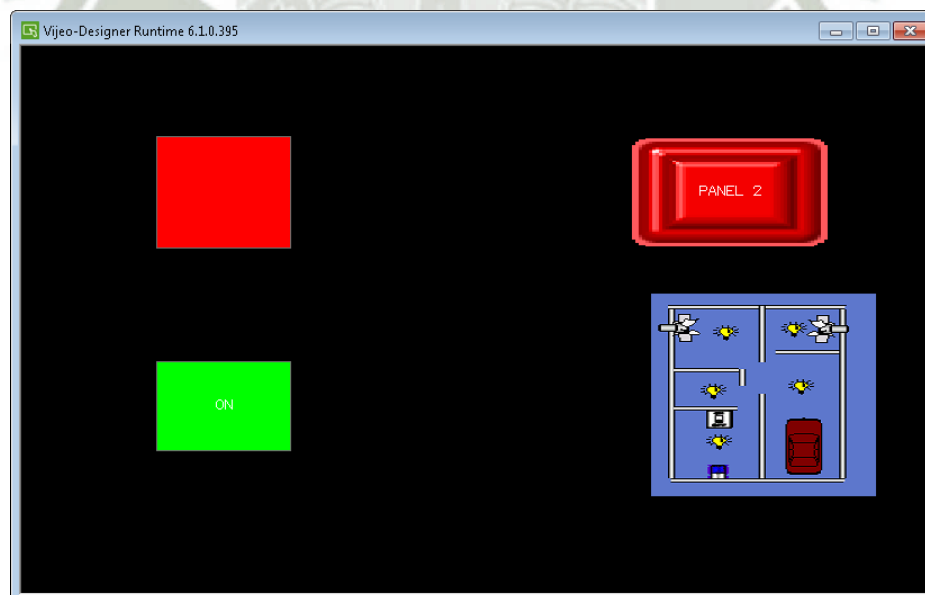


Figura 23. Botón “ON” activado.

26. Al darle click al botón “PANEL 2” nos manda al siguiente panel donde estará la simulación de nuestro tanque, y al presionar “LLENAR” deberá empezar a subir el relleno del rectángulo azul simulando el llenado del tanque.

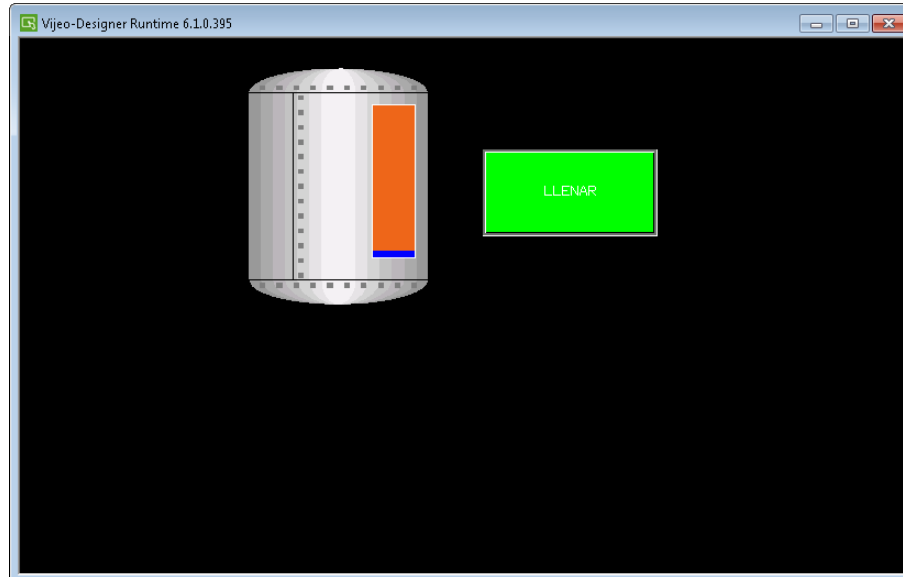


Figura 24. Panel 2 con la simulación del llenado de tanque.

27. Para descargar nuestro proyecto al terminal HMI conectado, deberá revisarse en las propiedades de “Destino1” que la dirección IP y la Máscara de subred estén bien asignadas, de lo contrario asignarlo manualmente.

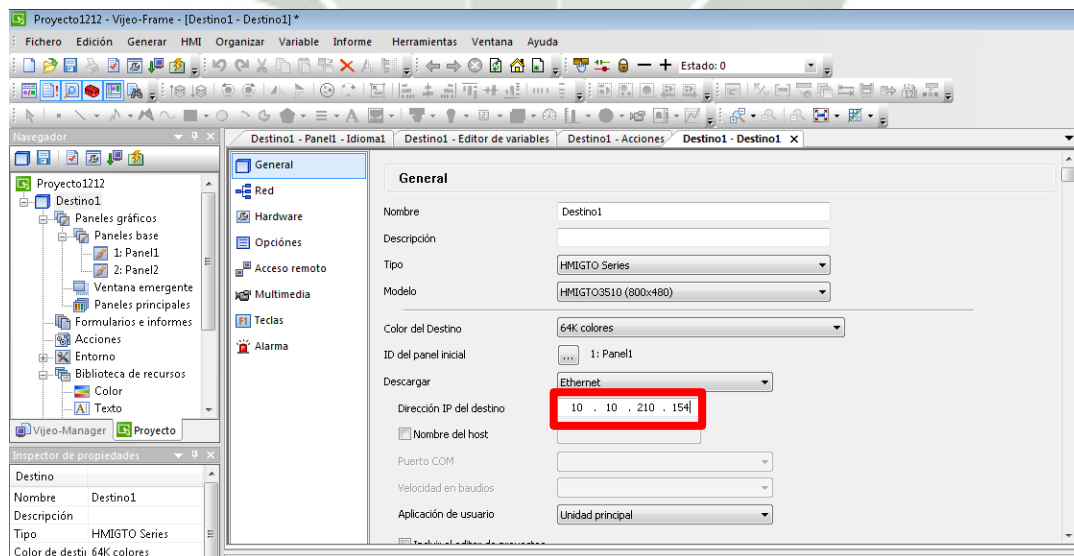


Figura 25. Configurando la dirección IP del destino.

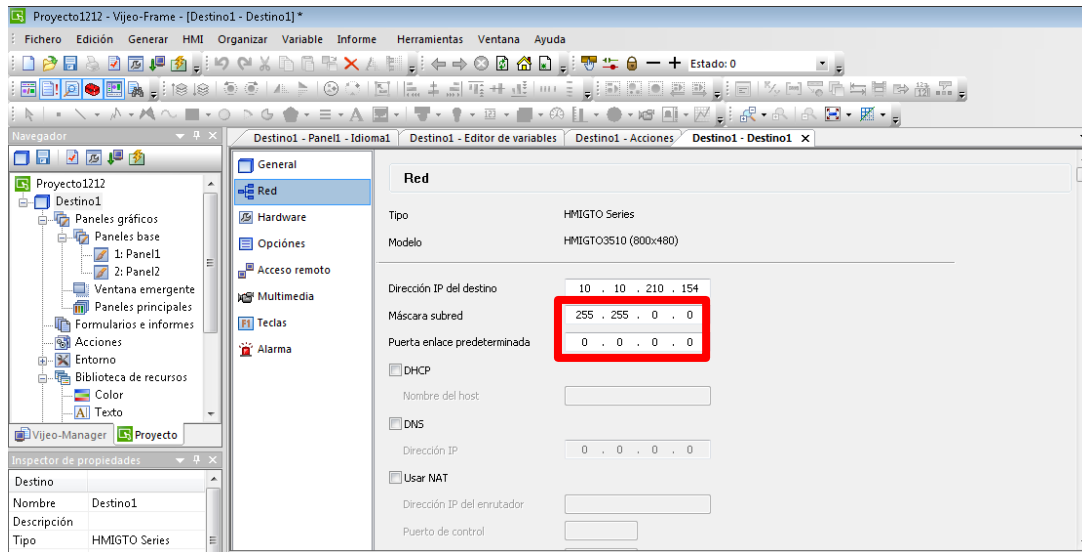


Figura 26. Configurando la Máscara de Subred y la Puerta de enlace predeterminada.

28. Luego se dará click derecho a “Destino1” y finalmente click a “Descargar a (Ethernet 10.10.210.154)...” En mi caso esta es la dirección pero deberá aparecer la dirección que usted le asigne.

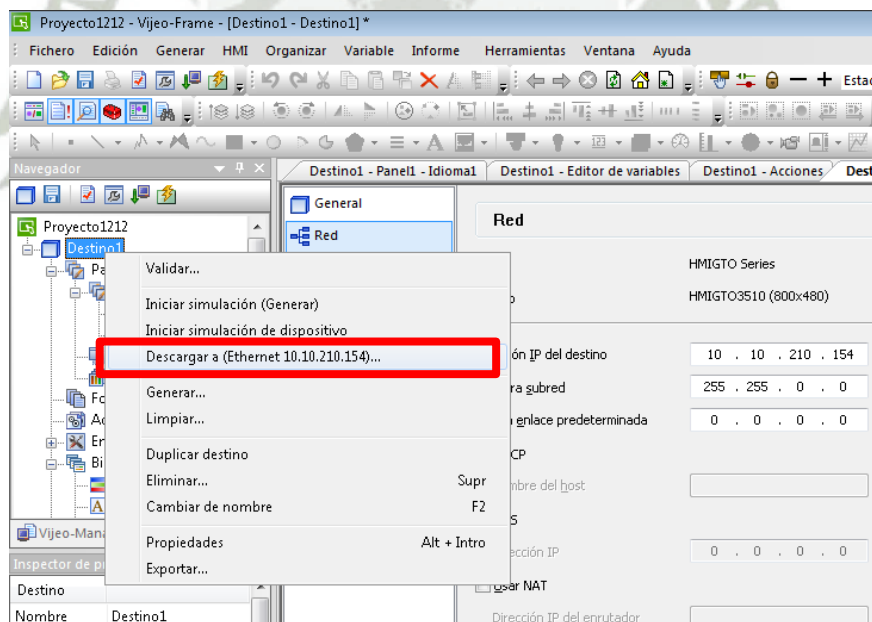


Figura 27. Descargando a terminal HMI.

29. Describa como se puede Exportar un proyecto sin perder las variables externas.

30. Averigüe todas las funciones de Operación de un botón.

31. Convierta el llenado del tanque en un drenado del mismo.

IV. CUESTIONARIO FINAL

- 1) ¿Qué es “Enlace de Puerta Predeterminada”?
- 2) ¿Qué significa NAT?
- 3) ¿Cuál es la diferencia entre DHCP y DNS?
- 4) ¿Qué significa “host”?
- 5) ¿Cuál es la diferencia entre datos Booleanos, Enteros y Flotantes?

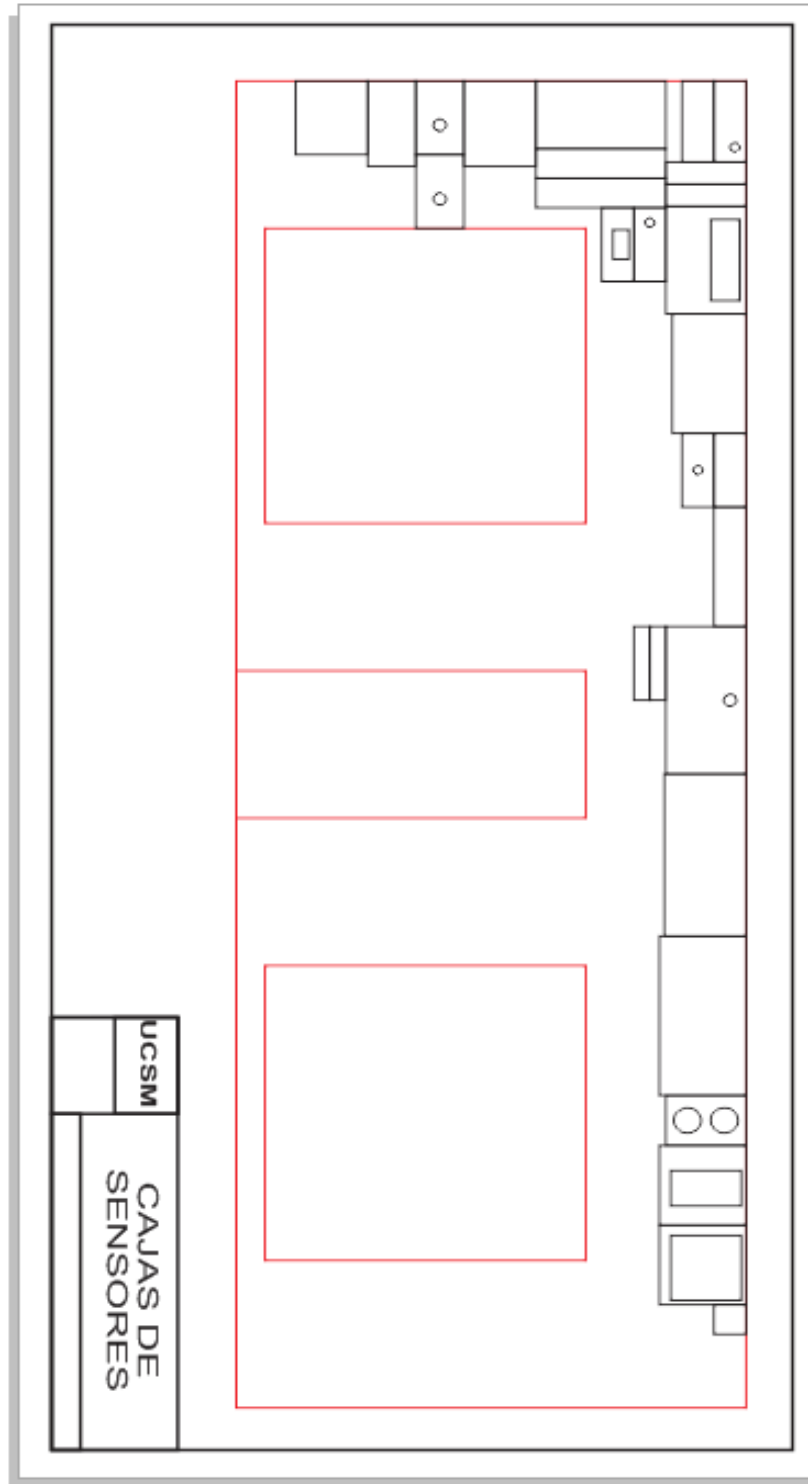
V. OBSERVACIONES Y CONCLUSIONES

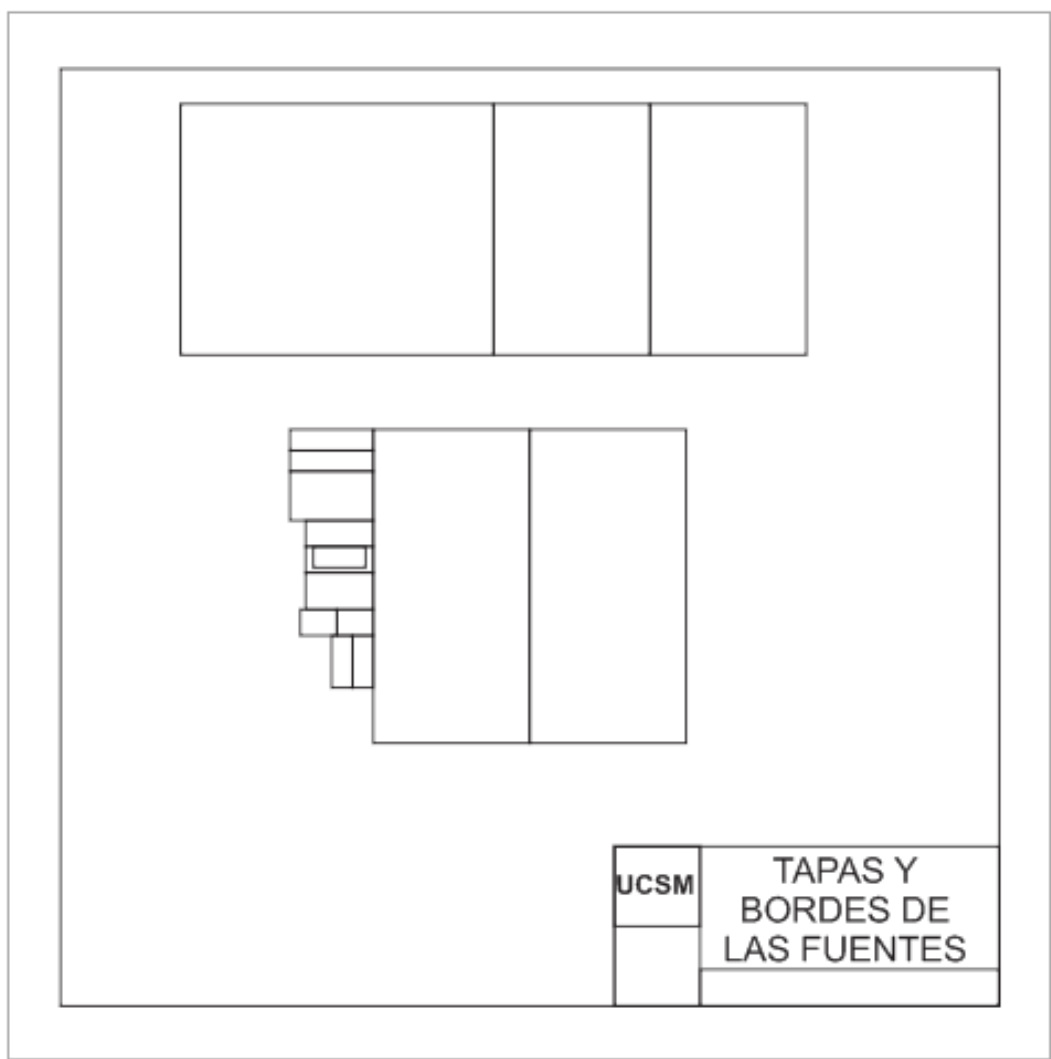
PRESUPUESTO FINAL

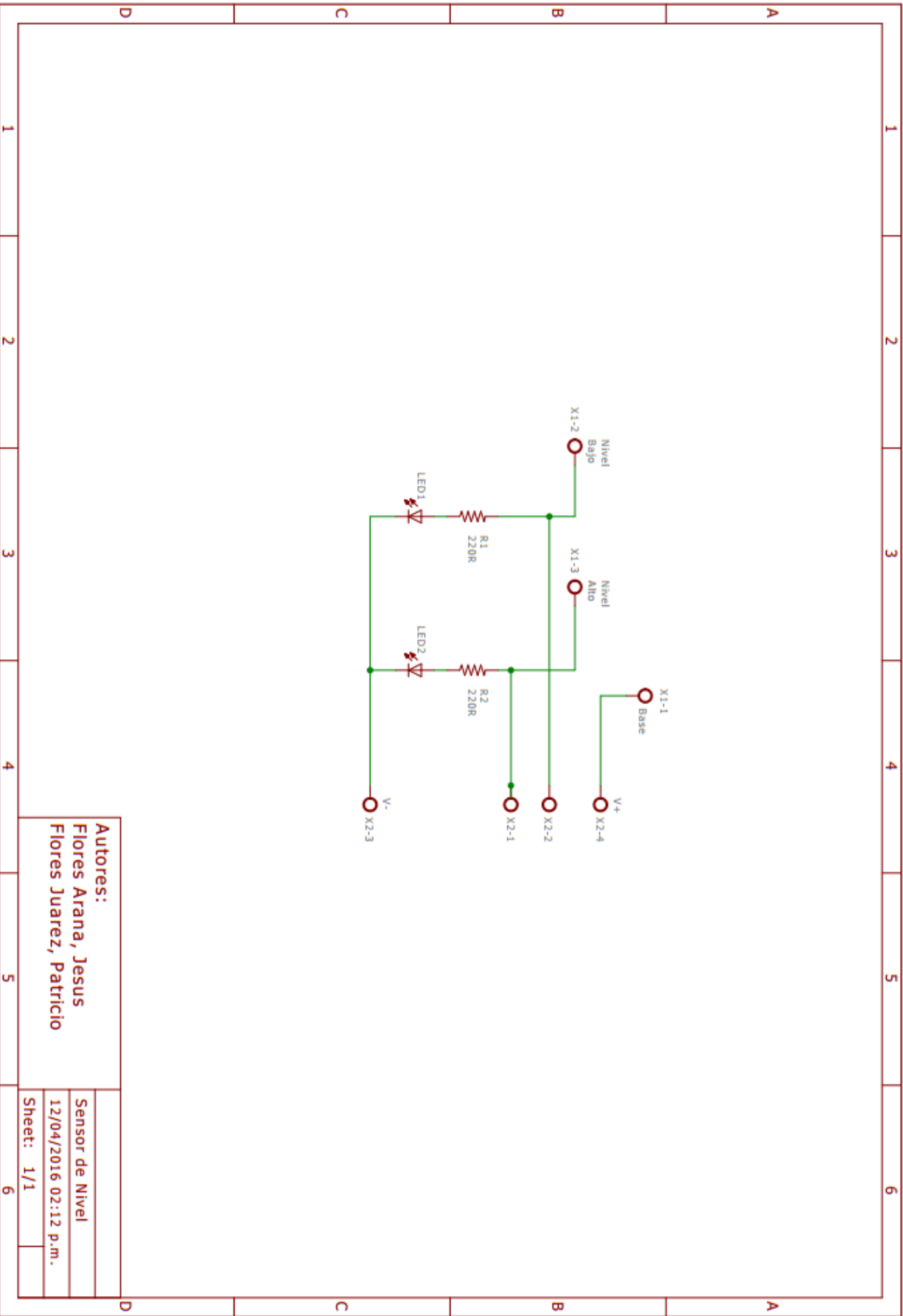
Cantidad		Precio en S/. por unidad	Precio Total en S/.
1	Terminal HMI Magelis	2 700.80	2 700.80
2	Controladores Lógico Programables	2620.80	5 241.60
1	Switch	50.00	50.00
1	Componentes electrónicos	300.00	300.00
5	1m. de Cables de seguridad con sus enchufes		
1	Sensor Inductivo	13.00	65.00
2	Extensiones	15.00	30.00
100	Cables conectores varios	1.00	100.00
1	Interruptor termo magnético	80.00	80.00
1	Fuente 12 v	120.00	120.00
1	Fuente 24 v	120.00	120.00
1	Actuador para simulación de procesos	60.00	60.00
20	Dispositivos de seguridad	5.00	100.00
1	Tablero de montaje	200.00	200.00
1	Estructura de sujeción bomba	150.00	150.00
1	Bomba Hidráulica	150.00	150.00
1	Válvula Solenoide	110.00	110.00
1	Varios componentes Hidráulicos	130.00	130.00
3	Plancha de acrílico	80.00	240.00
10	1m perfil en U de aluminio	11.00	110.00
2	1m perfil en H de plástico	15.00	30.00
2	Plancha de policarbonato	70.00	140.00
1	Otros	1 000.00	1 000.00
		Total:	11 227.40

Tabla 1 Tabla de precios y costos

PLANOS

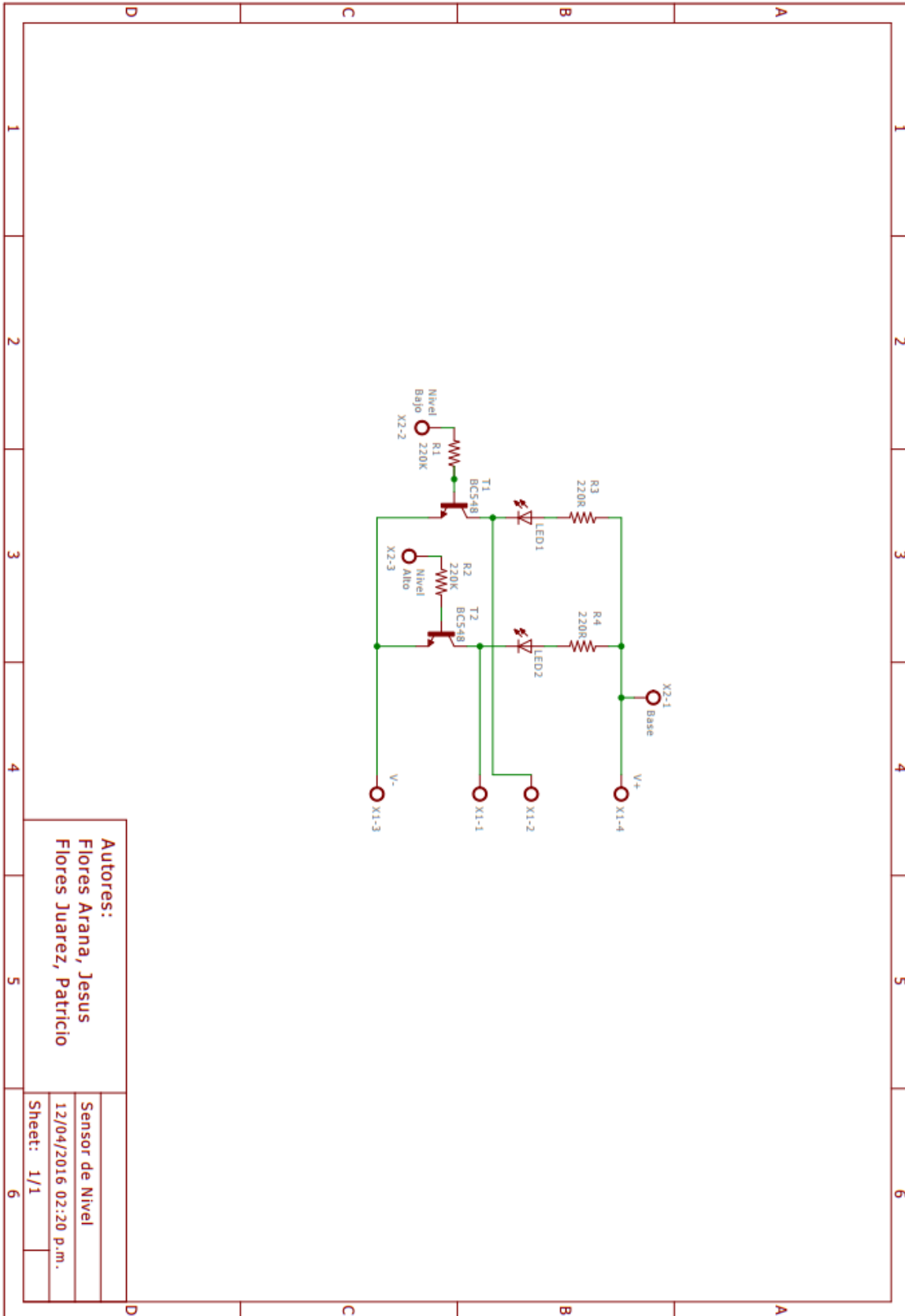






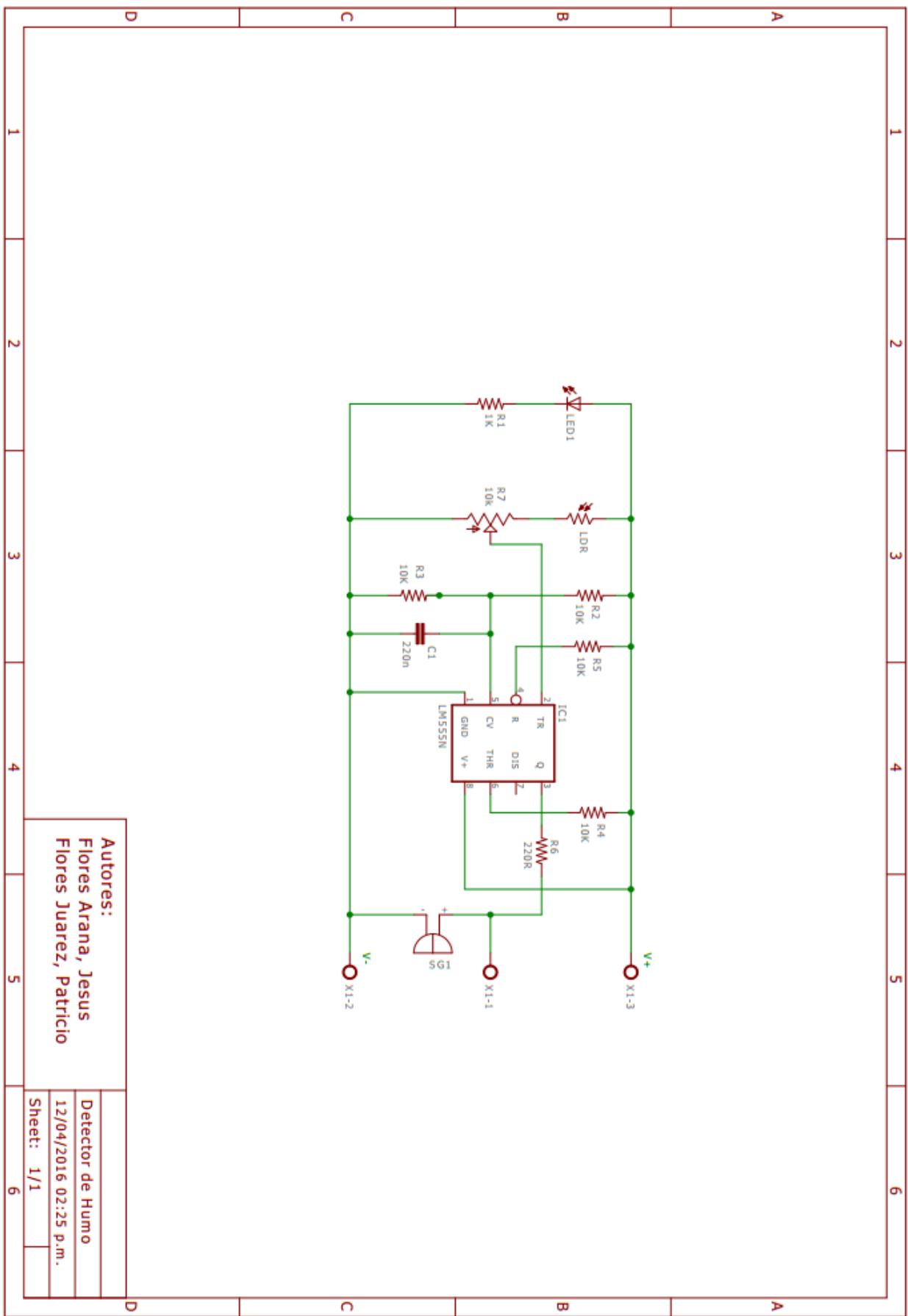
Autores:
Flores Arana, Jesus
Flores Juarez, Patricio

Sensor de Nivel
12/04/2016 02:12 p.m.
Sheet: 1/1



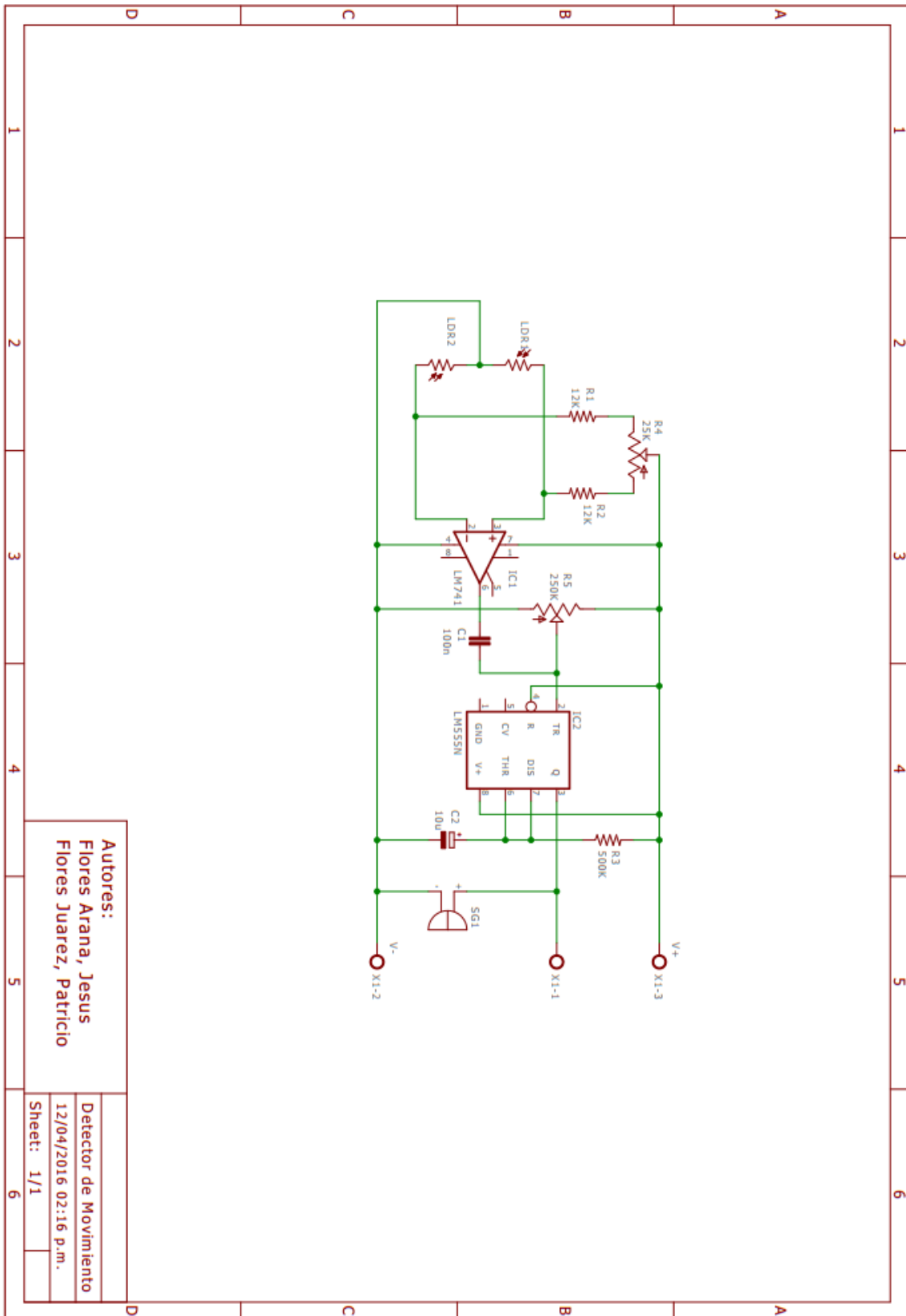
Autores:
Flores Arana, Jesus
Flores Juarez, Patricio

Sensor de Nivel
12/04/2016 02:20 p.m.
Sheet: 1/1



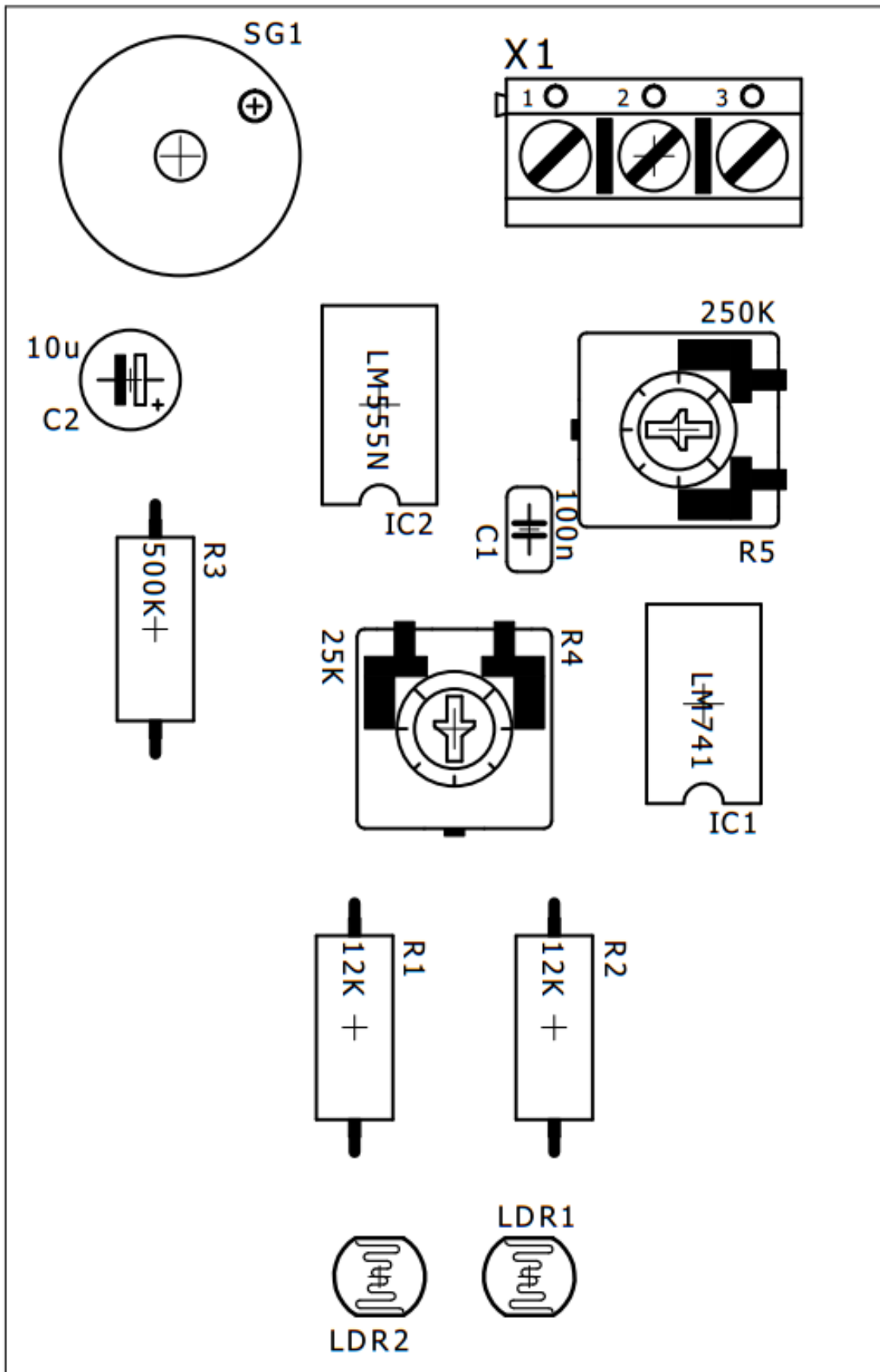
Autores:
Flores Arana, Jesus
Flores Juarez, Patricio

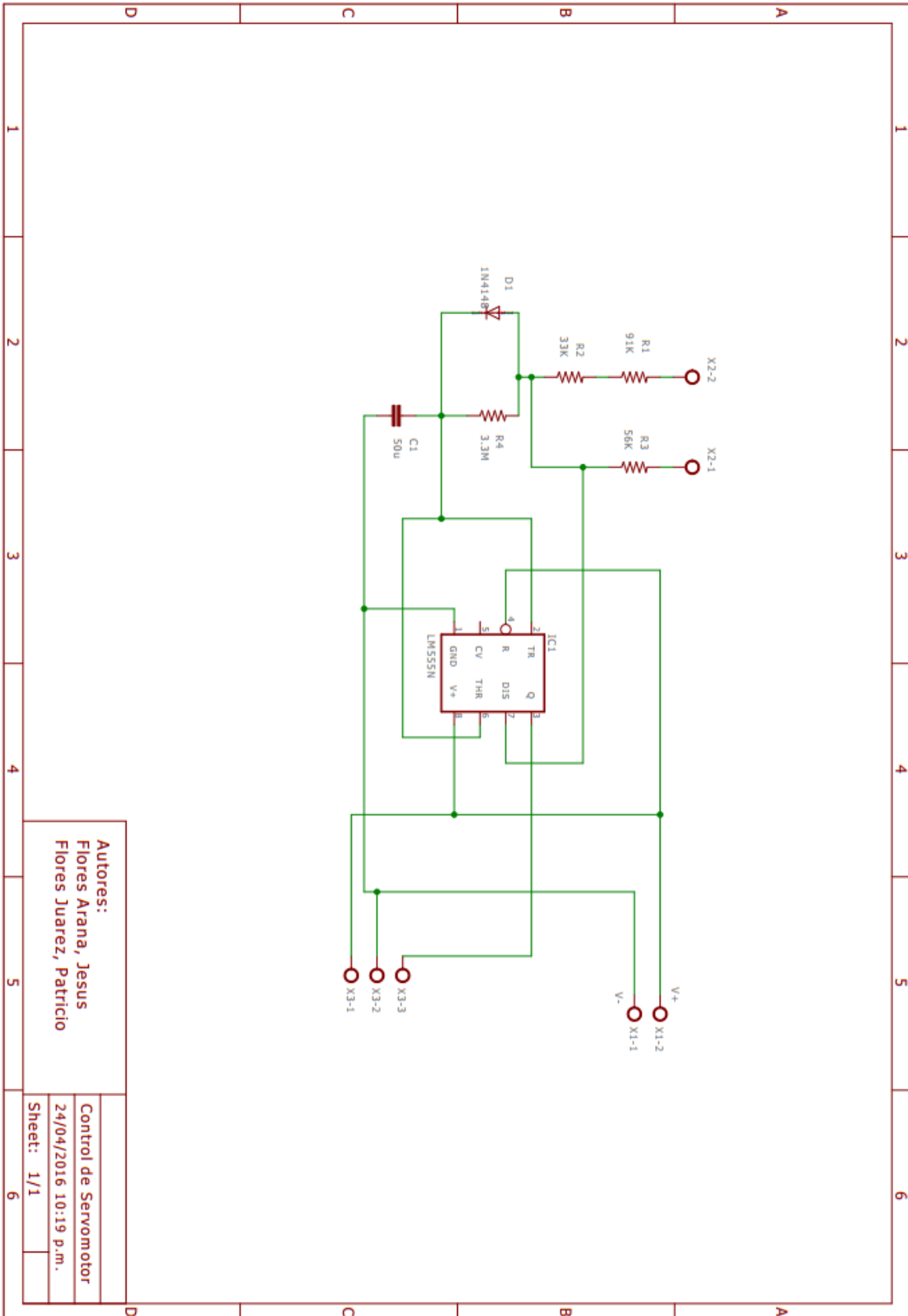
Detector de Humo
12/04/2016 02:25 p.m.
Sheet: 1/1



Autores:
Flores Arana, Jesus
Flores Juarez, Patricio

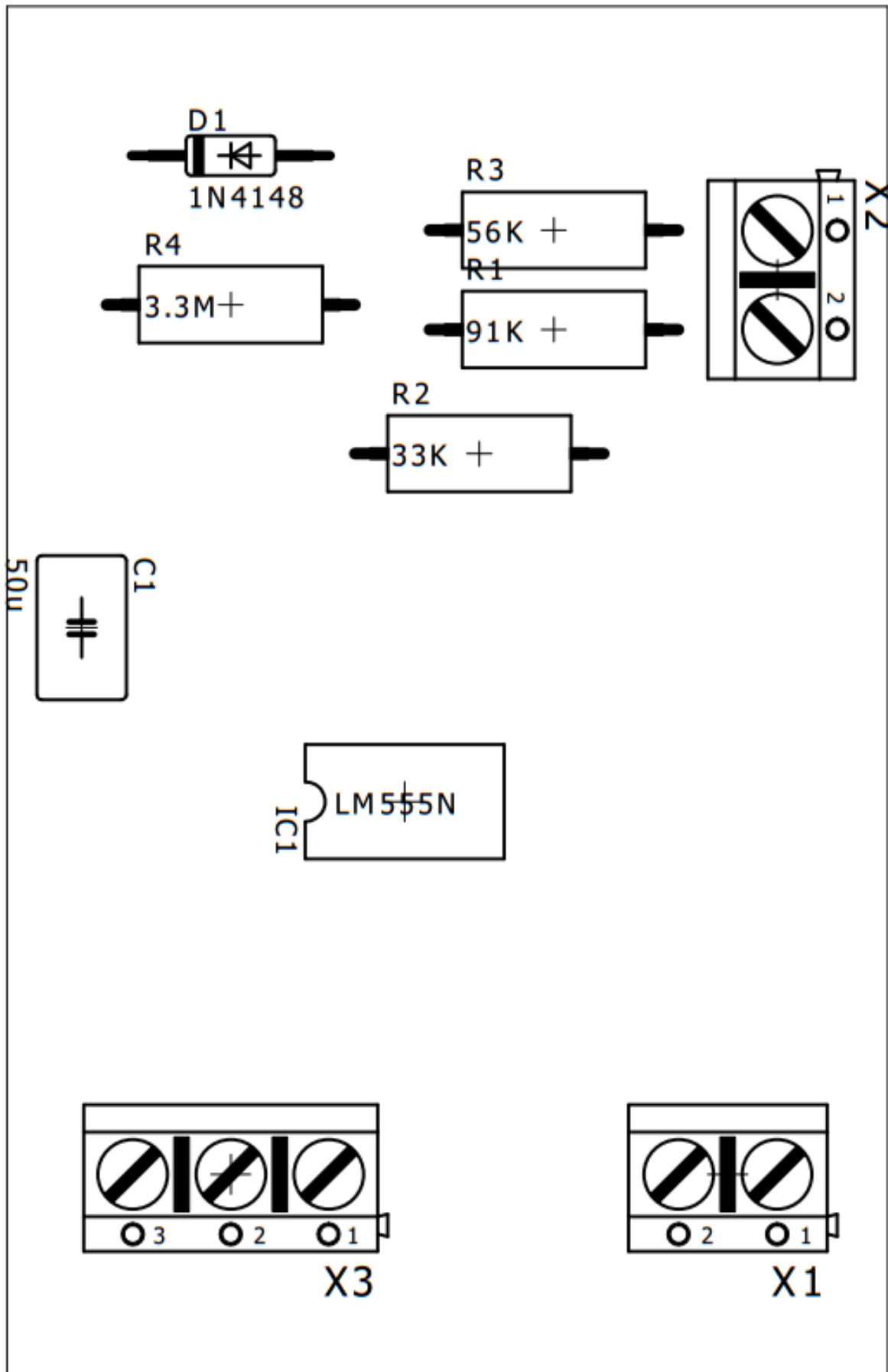
Detector de Movimiento
12/04/2016 02:16 p.m.
Sheet: 1/1

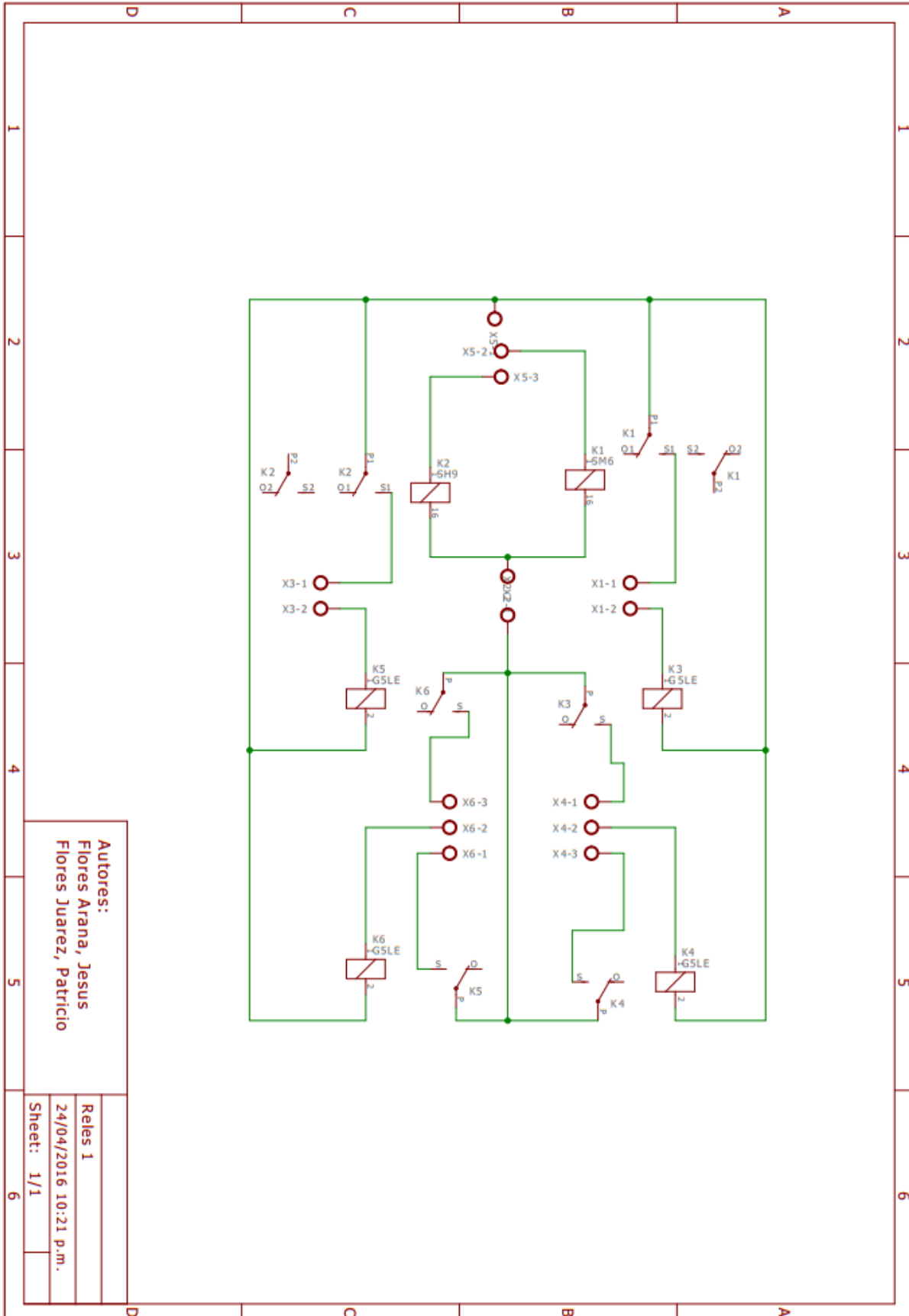


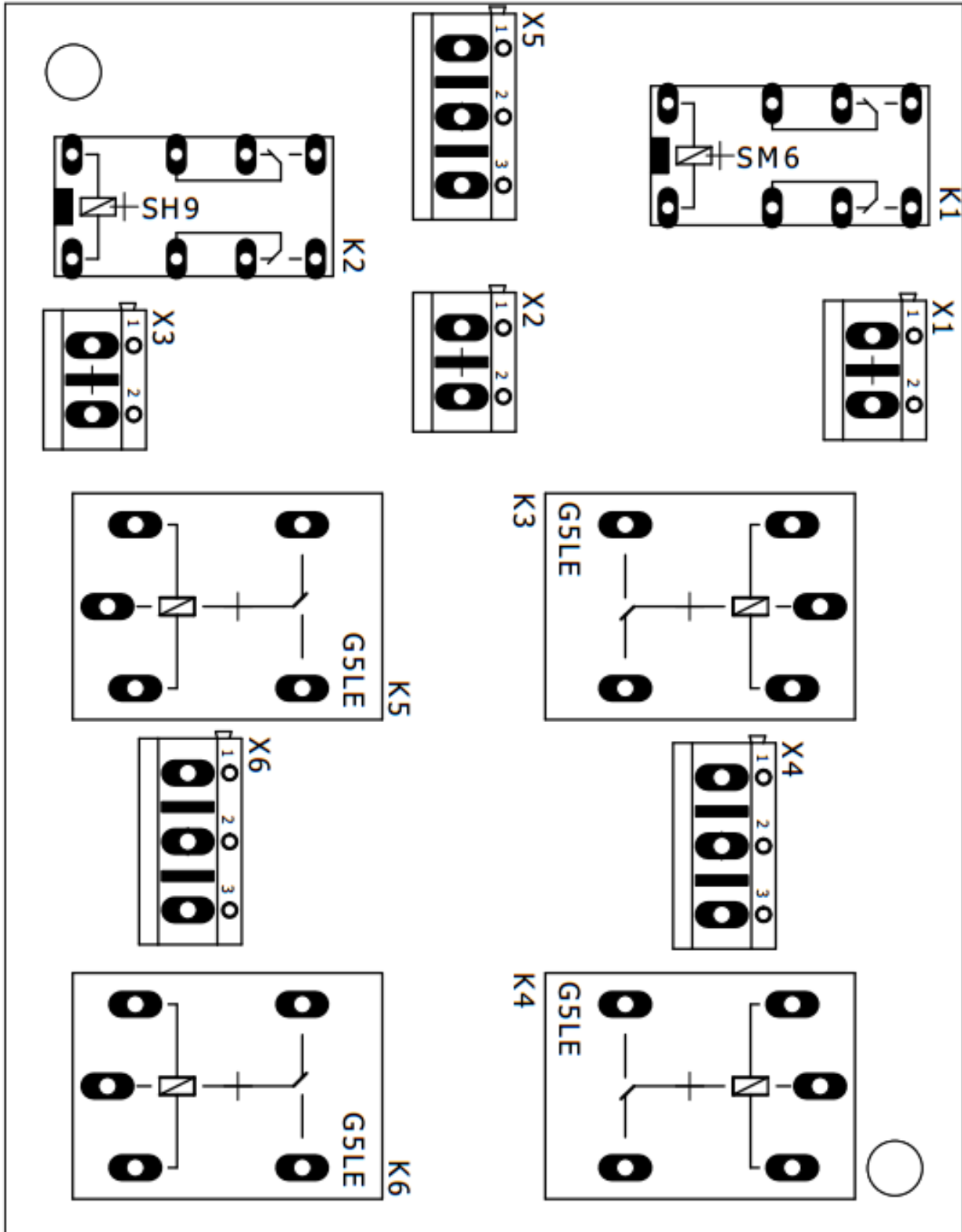


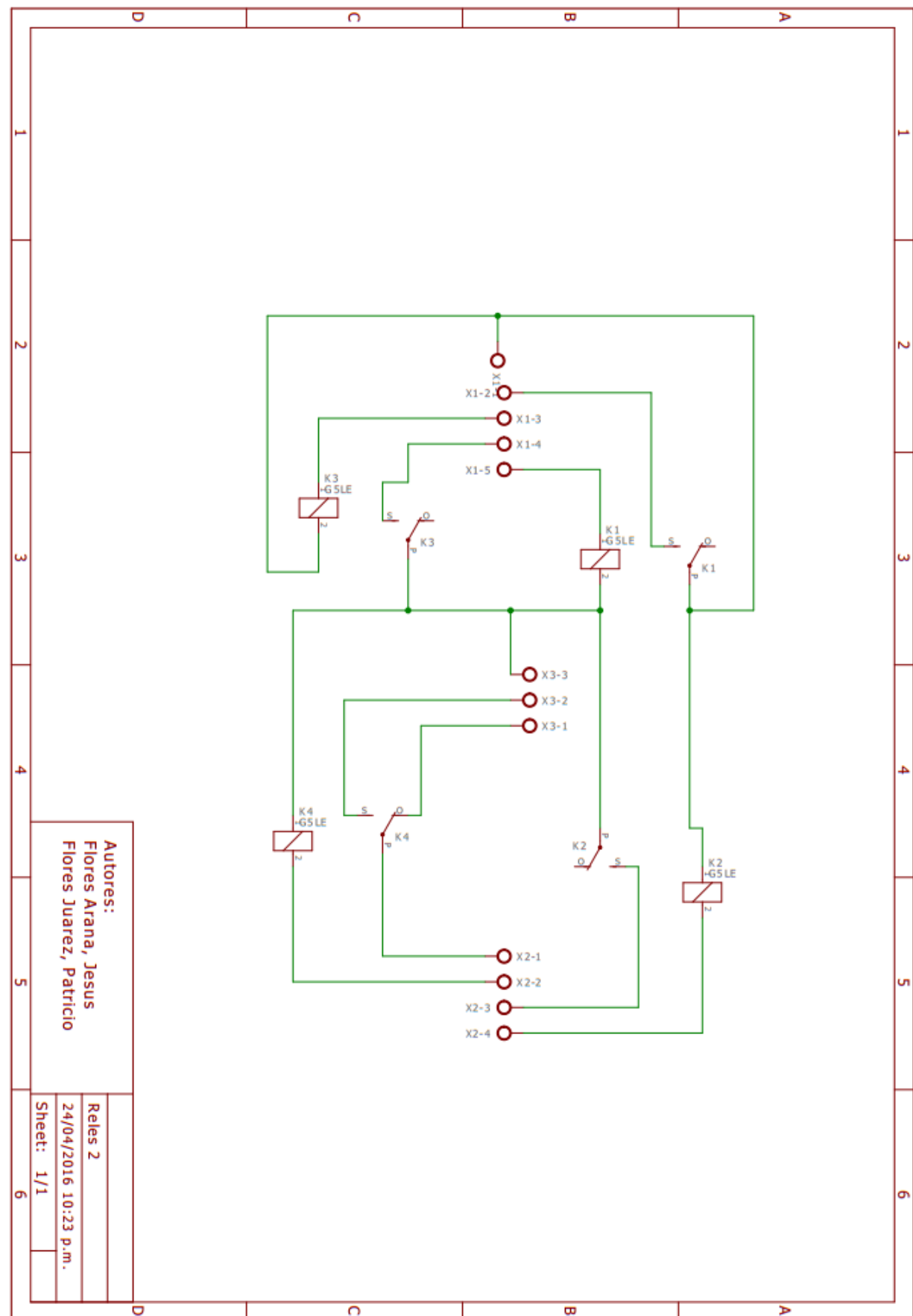
Autores:
Flores Arana, Jesus
Flores Juarez, Patricio

Control de Servomotor
24/04/2016 10:19 p.m.
Sheet: 1/1



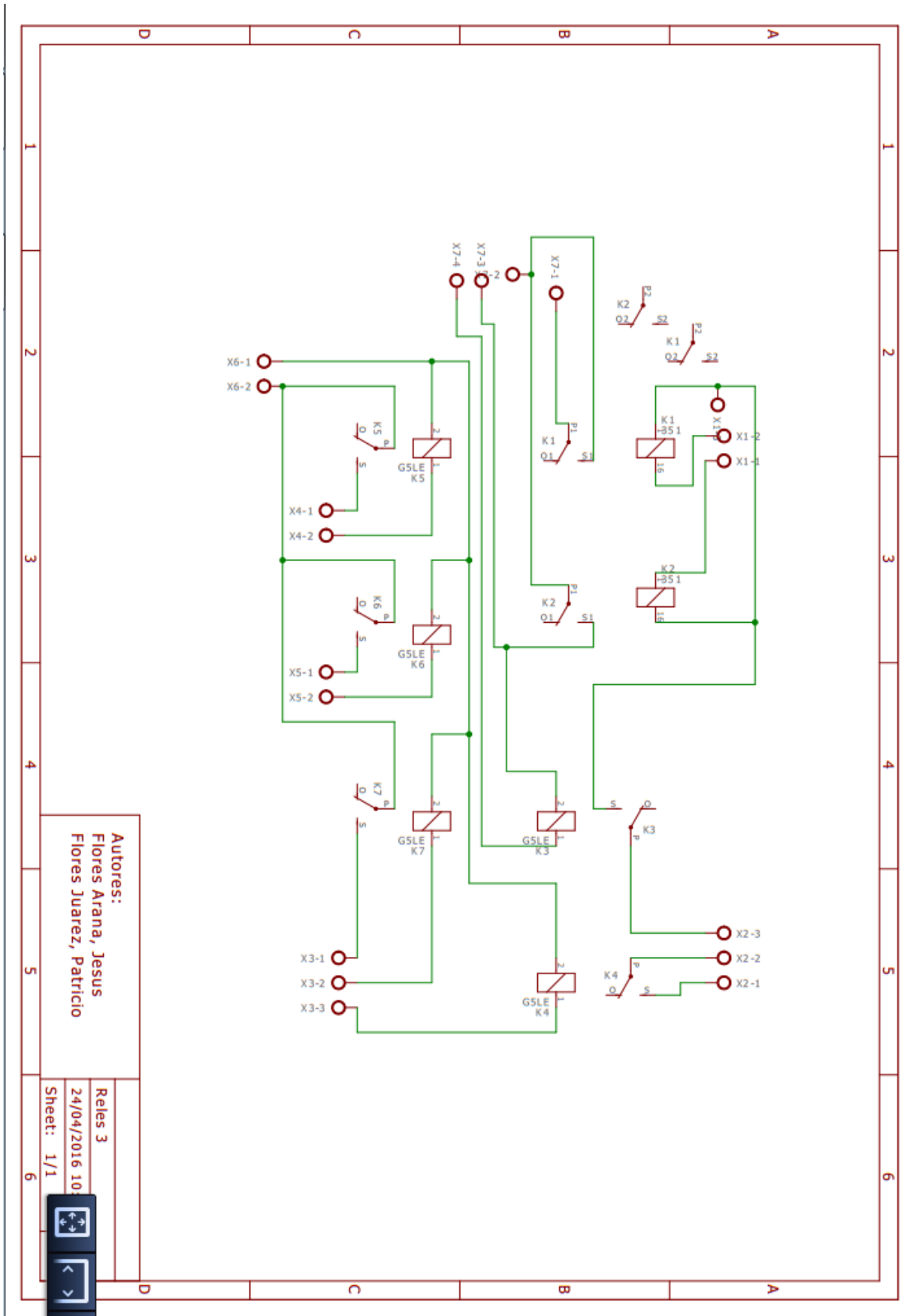


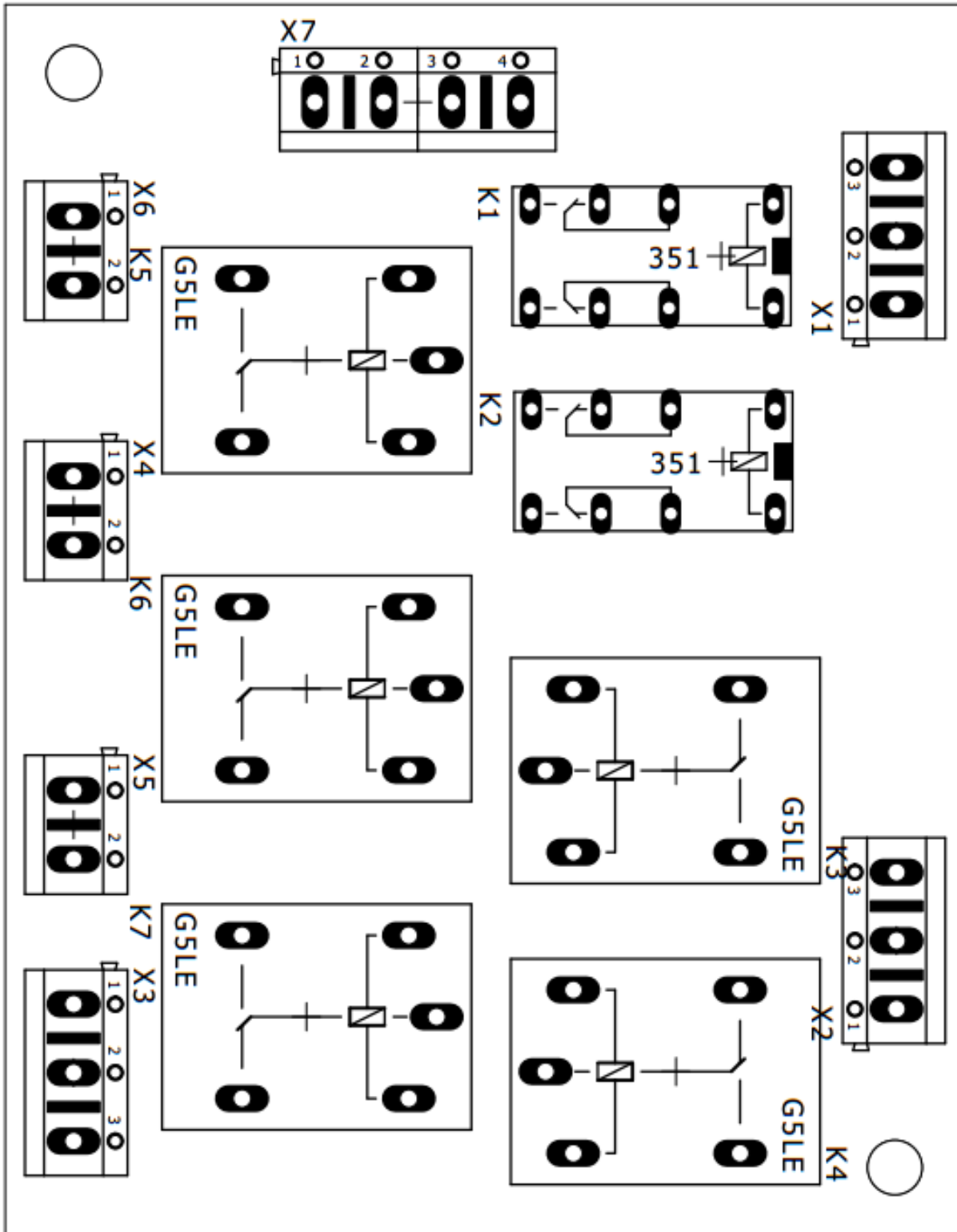


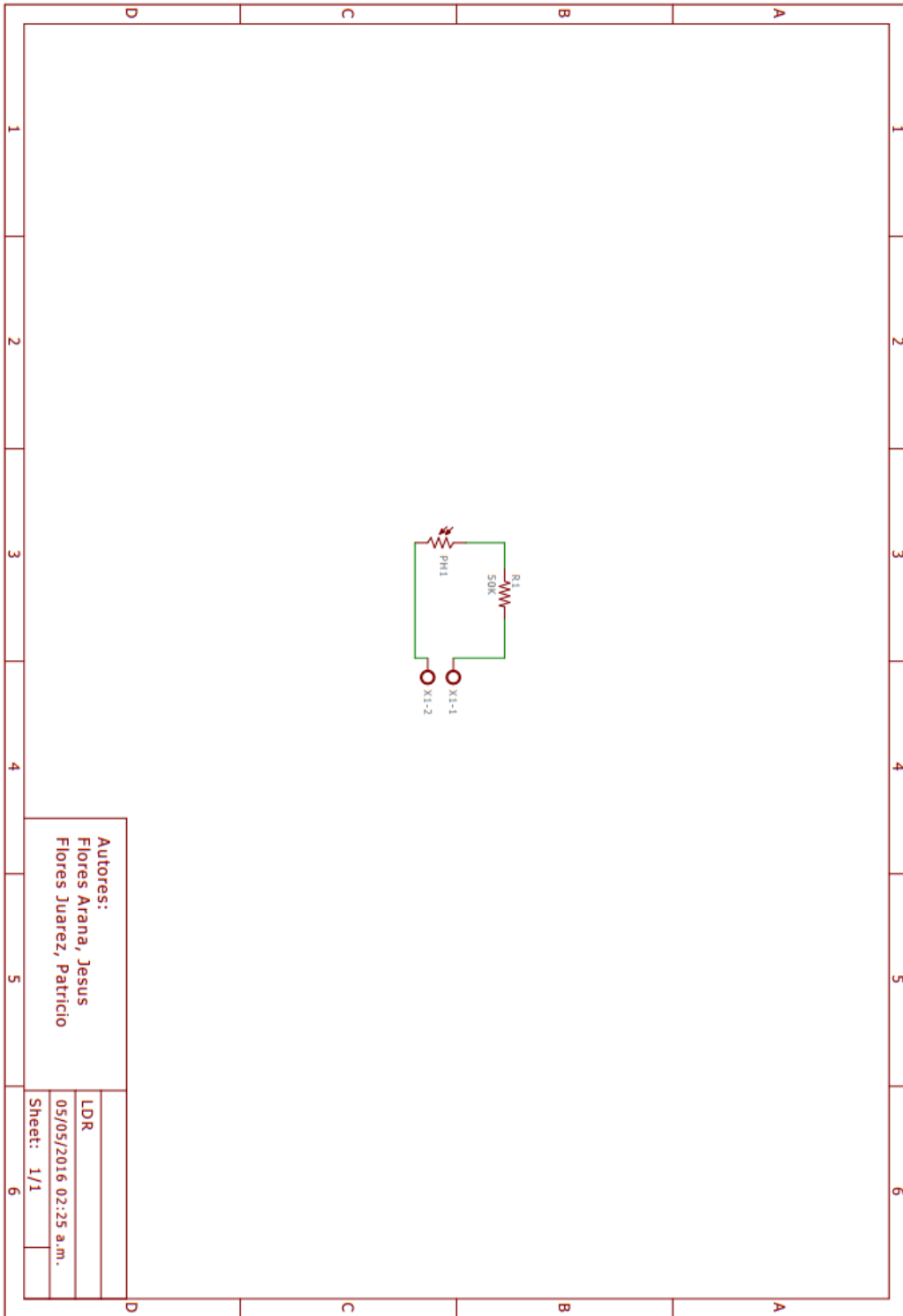


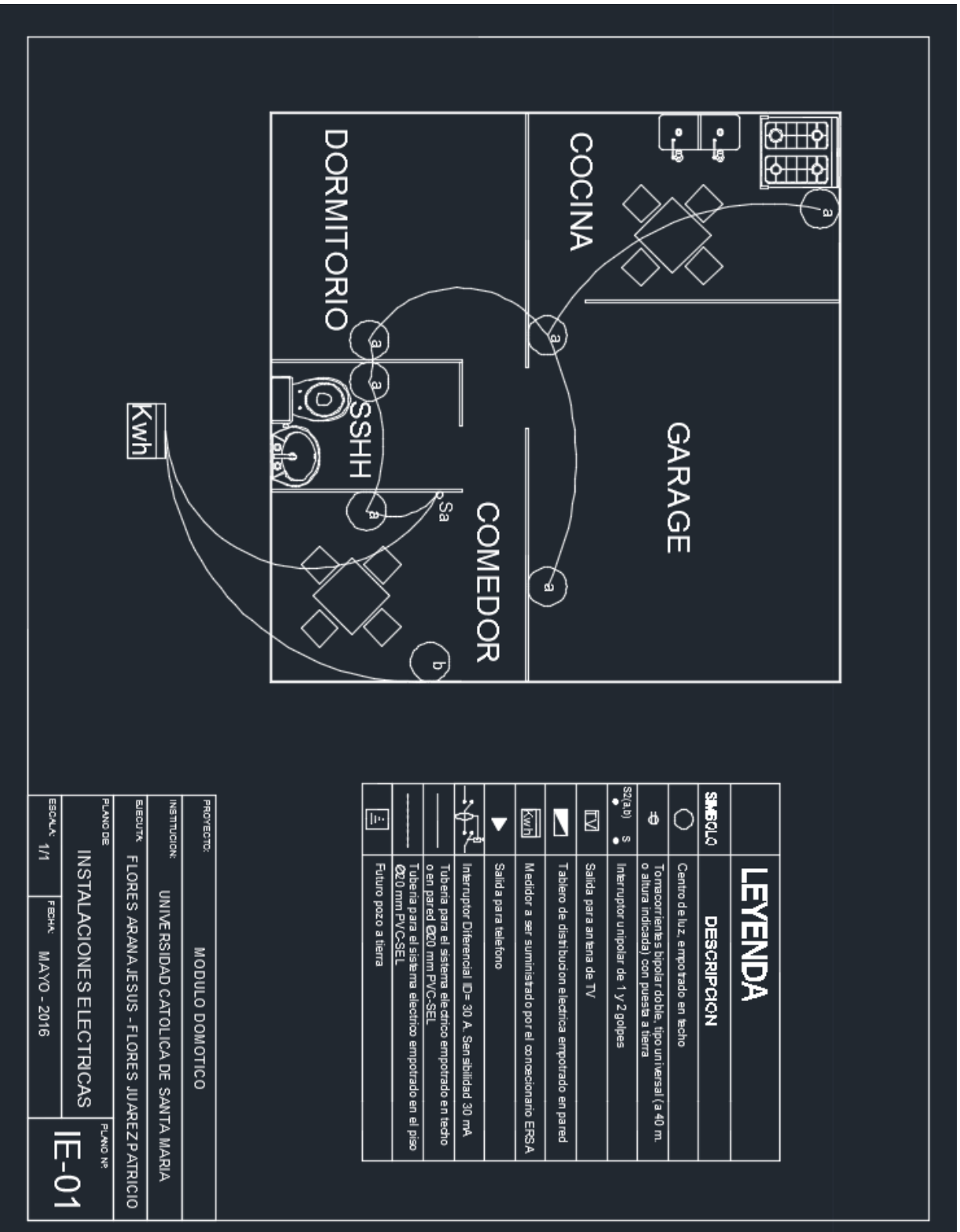
Autores:
Flores Arana, Jesus
Flores Juarez, Patricio

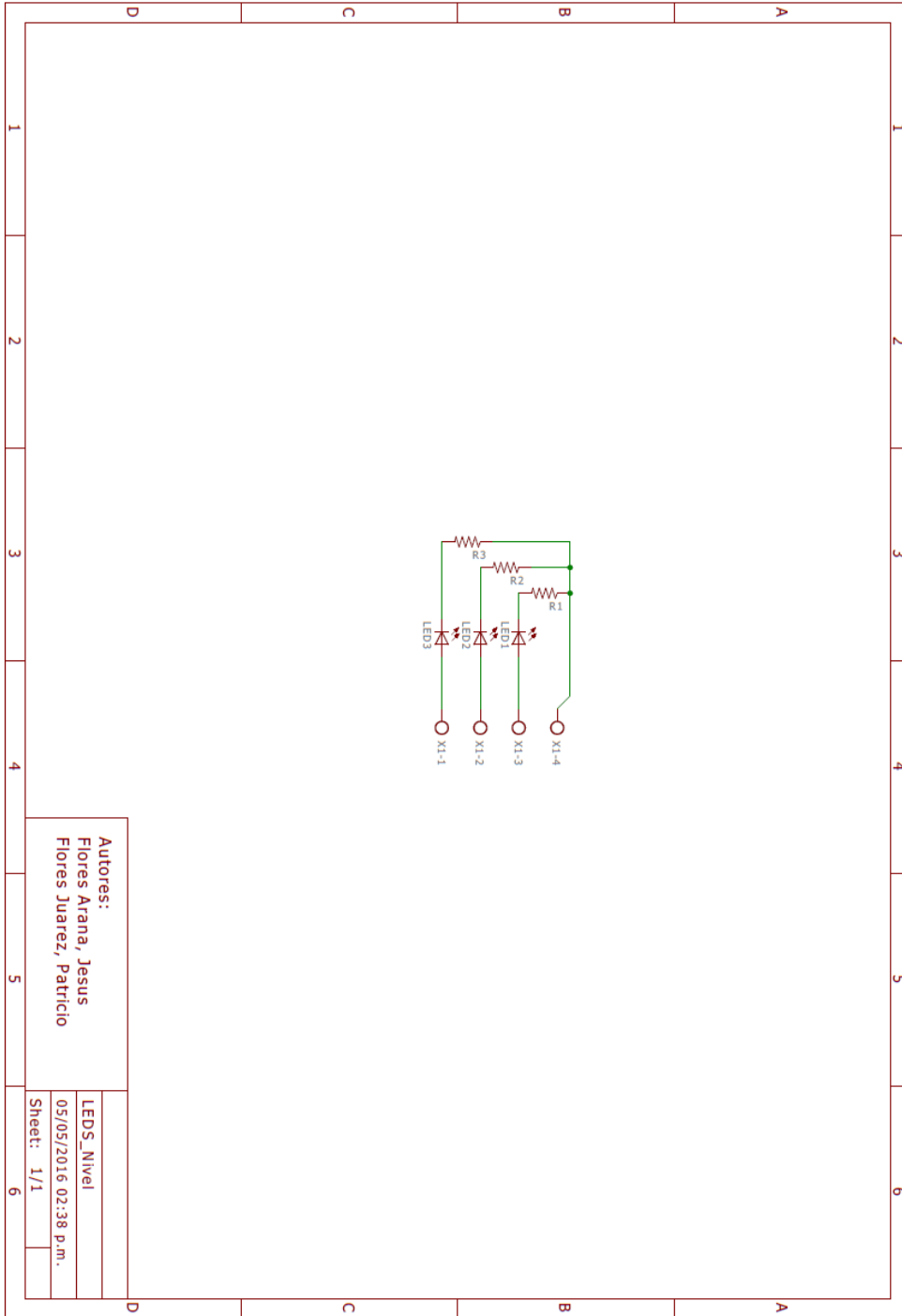
Relés 2
24/04/2016 10:23 p.m.
Sheet: 1/1











Ficha de producto
Características

TM221CE24R
Controlador M221 24 E/S relé Ethernet



Principal

Estatus comercial	Comercializado
Gama de producto	Modicon M221
Tipo de producto o componente	Autómata programable
[Us] tensión de alimentación nominal	100...240 V AC
De pie conducto	14 entrada discreta de acuerdo con IEC 61131-2 tipo 1
Número de entrada analógica	2 en el rango de entrada: 0...10 V
Tipo de salida digital	Relé normalmente abierto
Número de salidas discretas	10 relé
Tensión de salida	5...250 V CA 5...125 V CC
Montado en la pared del conducto	2 A

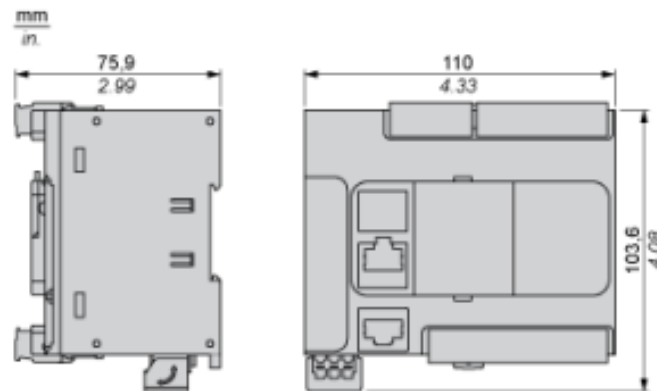
Complementario

Número de E/S digitales	24
Numero de E/S del módulo de expansión	<= 7 para salida del relé <= 7 para salida transistor
Límites tensión alimentación	85...264 V
Frecuencia de red	50/60 Hz
Corriente de entrada	<= 40 A
Consumo de potencia en VA	<= 35 VA en 100...240 V sin módulo de expansión E/S <= 58 VA en 100...240 V módulo de expansión con número máximo de E/S
Corriente de salida fuente de alimentación	0.16 A en 24 V para bus de expansión 0.52 A en 5 V para bus de expansión
Entrada lógica	Receptor o suministro (positivo/negativo)
Tensión de entrada digital	24 V
Tipo de voltaje entrada discreto	CC
Resolución de entrada analógica	10 bits
Valor LSB	10 mV
Tiempo conversión	1 ms por canal + 1 controlador del ciclo de tiempo para entrada analógica
Sobrecarga permitida em entradas	+/- 13 V CC para entrada analógica permanente +/- 30 V CC para entrada analógica con 5 min máximo
Estado de tensión 1 garantizado	>= 15 V para entrada
Posición de conexión libre	>= 4.2 mA para entrada digital >= 2.6 mA para entrada rápida
Estado de tensión 0 garantizado	<= 5 V para entrada
Transmisión de datos codificada	<= 0.6 mA para entrada rápida <= 1.3 mA para entrada digital
Corriente de entrada discreta	5 mA para entrada rápida 7 mA para entrada digital
Tapa de conexiones trasero	4.9 kOhm para entrada rápida 100 kOhm para entrada analógica

Tiempo respuesta	100 μ s turn-off operación para entrada; otros terminales terminal 5 μ s turn-off operación para entrada rápida; I0, I1, I6, I7 terminal 35 μ s turn-on operación para entrada; otros terminales terminal 5 μ s turn-on operación para entrada rápida; I0, I1, I6, I7 terminal 10 ms turn-off operación para salida 10 ms turn-on operación para salida 35 μ s turn-off operación para entrada; I2...I5 terminal
Tiempo filtro configurable	12 ms para entrada 3 ms para entrada 0 ms para entrada
Límites de tensión de salida	277 V CA 125 V CC
Elevación	7 A en COM 1 terminal 7 A en COM 0 terminal 4 A en COM 2 terminal
Error de precisión absoluta	+/- 1 % de la escala total para entrada analógica
Durabilidad eléctrica	Inductivo (L/R = 7 ms) DC-13, 24 V/ 7.2 W : 300000 ciclos Inductivo (L/R = 7 ms) DC-13, 24 V/ 24 W : 100000 ciclos Resistivo DC-12, 24 V/ 16 W : 300000 ciclos Resistivo DC-12, 24 V/ 48 W : 100000 ciclos Inductivo AC-14, (cos phi = 0.7) 240 V/ 72 VA : 300000 ciclos Inductivo AC-14, (cos phi = 0.7) 120 V/ 36 VA : 300000 ciclos Inductivo AC-14, (cos phi = 0.7) 240 V/ 240 VA : 100000 ciclos Inductivo AC-14, (cos phi = 0.7) 120 V/ 120 VA : 100000 ciclos Inductivo AC-15, (cos phi = 0.35) 240 V/ 36 VA : 300000 ciclos Inductivo AC-15, (cos phi = 0.35) 120 V/ 18 VA : 300000 ciclos Inductivo AC-15, (cos phi = 0.35) 240 V/ 120 VA : 100000 ciclos Inductivo AC-15, (cos phi = 0.35) 120 V/ 60 VA : 100000 ciclos Resistivo AC-12, 240 V/ 160 VA : 300000 ciclos Resistivo AC-12, 120 V/ 80 VA : 300000 ciclos Resistivo AC-12, 240 V/ 480 VA : 100000 ciclos Resistivo AC-12, 120 V/ 240 VA : 100000 ciclos
Frecuencia de conmutación	20 operaciones de conmutación/minuto con carga máxima
Durabilidad mecánica	>= 20000000 ciclos para salida del relé
Carga mínima	1 mA en 5 V CC para salida del relé
Tipo de protección	Sin protección en 5 A
Hora de rearme	1 s
Capacidad de memoria	256 kB para variables internas RAM 256 kB para aplicación de usuarios y datos RAM con 10000 instrucciones
Orejetas terminales de anillo	256 kB memoria flash integrada para copia de seguridad de la aplicación y de los datos
Mantenido Ti24	2 GB Tarjeta SD opcional
Tipo de batería	BR2032 litio no-recargable, vida batería: 4 yr
Tiempo de backup	1 año en 25 °C por interrupción de fuente de alimentación
Tiempo de ejecución para 1 Kinstrucción	0.3 ms para evento y tarea periódica
Soporte angular	0.2 μ s Booleano
Soporte adaptable	60 μ s tiempo de respuesta
Tamaño máximo de las áreas de objeto	512 %M bits de memoria 8000 %MW palabras de memoria 512 %KW palabras constantes 255 %TM temporizadores 255 %C contadores
Reloj en tiempo real	Donde
Deriv. reloj	<= 30 s/mes en 25 °C
Lazo de regulación	Regulador PID ajustable hasta 14 lazos simultáneos
Número de entrada de contaje	4 entrada rápida (modo HSC) (contando frecuencia: 100 kHz), contando capacidad: 32 bits
Función de contaje	A/B Impulso/Dirección Monofásico
Pares de nueces	Ethernet con conector RJ45 Enlace serie sin aislar "serie 1" con conector RJ45 e interfaz RS232/RS485 Porta USB con conector USB 2.0 mini B
Suministro	Serie fuente de alimentación de enlace serie en 5 V 200 mA

Velocidad de transmisión	480 Mbit/s - protocolo comunicación: USB 1,2-115,2 kbit/s (115,2 kbit/s por defecto) para long bus de 3 m - protocolo comunicación: RS232 1,2-115,2 kbit/s (115,2 kbit/s por defecto) para long bus de 15 m - protocolo comunicación: RS485
Protocolo de puerto de comunicaciones	: Ethernet protocolo Enlace serie sin aislar : Modbus protocolo maestro/esclavo - RTU/ASCII o Red SoMachine Puerto USB : USB protocolo - SoMachine-Red
Puerto Ethernet	10BASE-T/100BASE-TX 1 puerto con 100 m cable cobre
Servicio de comunicación	Cliente DHCP Ethernet/Adaptador IP Servidor Modbus TCP Cliente Modbus TCP Dispositivo esclavo Modbus TCP
Señalizaciones en local	Link de reed Ethernet amarillo para Link (Link estado) Actividad de red Ethernet verde para ACT 1 LED verde para SL 1 LED por canal verde para estado de E/S 1 LED rojo para BAT 1 LED verde para tarjeta SD de acceso (SD) 1 LED rojo para error de módulo (ERR) 1 LED verde para RUN 1 LED verde para PWR
Consecutivo, seguido, continuo, adosado	USB 2.0 mini B conector para un terminal de programación Conector, 4 terminales para entradas analógicas Bornero, 3 terminales para conexión de la fuente de alimentación de 24 V CC Desmontable bornero de tornillo para salidas Bornero de tornillo extraíble para entradas
Longitud de cable	<= 1 m cable sin apantallar para entrada analógica <= 30 m cable sin apantallar para entrada digital <= 30 m cable sin apantallar para salida <= 10 m cable apantallado para entrada rápida
Aislamiento	2300 V CA entre el suministro y la fuente de alimentación del sensor 500 V CA entre el terminal Ethernet y la lógica interna 2300 V CA entre la salida y la lógica interna 500 V CA entre alimentación del sensor y la lógica interna 2300 V CA entre el suministro y la lógica interna 1500 V CA entre la salida y la tierra 500 V CA entre la entrada y la tierra 500 V CA entre alimentación y masa del sensor 1500 V CA entre el suministro y la tierra Sin aislamiento entre las entradas analógicas Sin aislamiento entre la entrada analógica y la lógica interna 500 V CA entre la entrada y la lógica interna
Marca	CE
Fuente de alimentación de detector	CC en 250 mA suministrado por el controlador
Soporte de montaje	Placa o panel con juego de fijación Tipo de tapón TH35-7.5 perfil de acuerdo con IEC 60715 Tipo de tapón TH35-15 perfil de acuerdo con IEC 60715
Altura	70 mm
Profundidad	70 mm
Anchura	110 mm
Peso del producto	0,395 kg

Dimensions





Principal

Estatus comercial	Comercializado
Gama de producto	Magelis STO & STU
Tipo de producto o componente	Pantalla táctil
Designación de software	Vijeo Designer
Sistema operativo	Magelis
Nombre del procesador	CPU ARM9

Complementario

Tamaño de pantalla	3.5 inch
Tipo de pantalla	Pantalla táctil acolor QVGA TFT
Color de pantalla	65536 colores
Resolución en píxeles	320 x 240 pixels
Panel táctil	Analógico
Vida útil de la luz posterior	50000 horas
Brillo	16 niveles
Fuente del carácter	ASCII Chino (chino simplificado) Japonés (ANK, kanji) Coreano Taiwanés (chino tradicional)
[Us] Tensión de alimentación	24 V CC
Alimentación	Fuente de alimentación externa
Límites tensión alimentación	20.4...28.8 V
Corriente de entrada	<= 30 A
Consumo de potencia en W	6.5 W
Número de páginas	Limitado por capacidad de memoria interna
Frecuencia de procesador	333 MHz
Descripción de memoria	Copia seg. datos, 64 kB Memoria de aplicaciones, 16 MB
Pares de nueces	Alimentación blq term rosca extrbls Enlace serie COM1 RJ45 hembra, veloc transmisión <= 115,2 kbits/s (RS232C/RS485) 1 puerto USB (V2.0) USB tipo A 1 puerto USB (V2.0) mini B USB
Reloj en tiempo real	Acceso al reloj de tiempo real del PLC
Protocolos descargables	Modbus Modbus TCP/IP Protocolos de terceros Uni-TE
CHC SCEW	Orificio de 22 mm Ø
Material del envoltente	PC/PBT
DESC	CE
Altura	98.15 mm

Entorno

Inmunidad a microcortes	<= 10 ms
Normas	EN 61131-2 FCC Class A IEC 61000-6-2 UL 1604 UL 508
Certificaciones	C-Tick CULus UL clase 1 Div2 T4A ó T5 Marine
Temperatura ambiente de trabajo	0...50 °C
Temperatura ambiente de almacenamiento	-20...60 °C
Humedad relativa	0...85 % sin condensación
Altitud máxima de funcionamiento	<= 2000 m
Grado IP	IP65 (panel frontal) de acuerdo con IEC 60529 IP20 (panel trasero) de acuerdo con IEC 60529
Grado de protección NEMA	NEMA 4X panel frontal (uso interior)
Resistencia a los choques	15 gn para 11 ms de acuerdo con IEC 60068-2-27
Resistencia a las vibraciones	1 gn (f = 9...150 Hz) de acuerdo con IEC 60068-2-6 +/-3,5 mm (f = 5...9 Hz) de acuerdo con IEC 60068-2-6
Resistencia a campos electromagnéticos	10 V/m de acuerdo con IEC 61000-4-3

Sostenibilidad de la oferta

Estado de la Oferta sostenible	Producto no Green Premium
RoHS	Compliant - since 1001 - Schneider Electric declaration of conformity Declaración de conformidad de Schneider Electric
REACH	Reference not containing SVHC above the threshold
Perfil ambiental del producto	Disponible Descargar Perfil Medioambiental

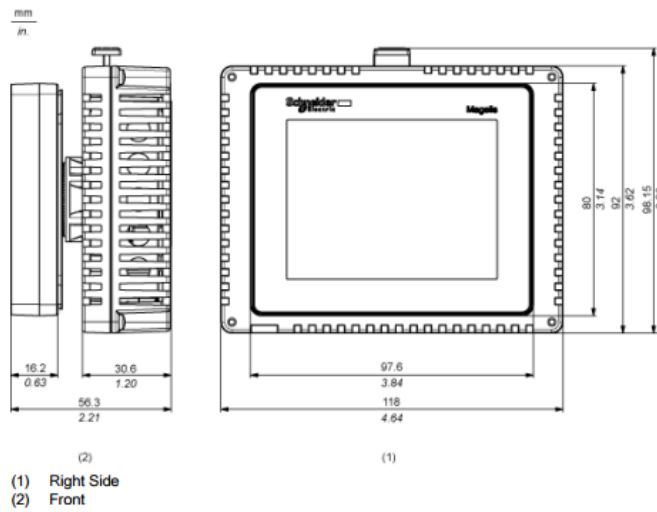


Ficha de producto
Dimensions Drawings

HMISTU655

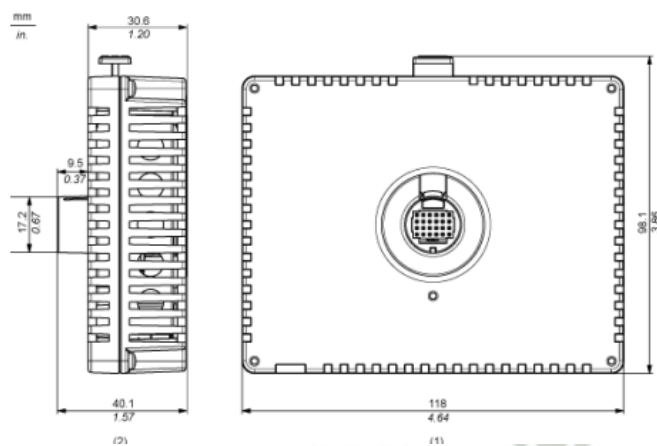
Display and Rear Modules

Dimensions



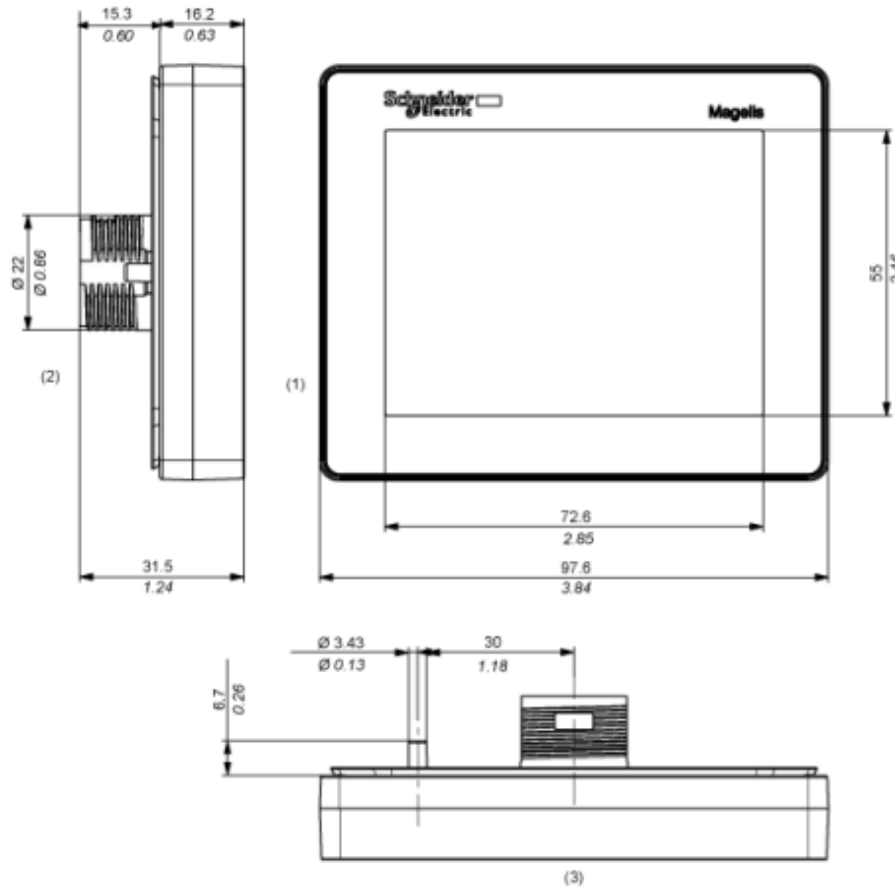
Rear Module

Dimensions



Dimensions

mm
in.



- (1) Front
- (2) Right Side
- (2) Top