

Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Electricidad y Computación



**“AUTOMATIZACIÓN DEL PROCESO DE EMPASTADO DE
LAS REJILLAS PARA LA FABRICACIÓN DE BATERIAS DE
PLOMO-ACIDO”**

TOPICO DE GRADUACIÓN

Previa a la obtención del Título de:

INGENIERO EN ELECTRICIDAD

Especialización:

ELECTRÓNICA Y AUTOMATIZACIÓN INDUSTRIAL

Presentada por:

**Juan Carlos García Tapia
Alfonso Ignacio Mazacón Baño
José Luis Parrales Santistevan**

Guayaquil – Ecuador

**AÑO
2006**

AGRADECIMIENTO

De manera general a DIOS y a todas las personas involucradas directa e indirectamente en la elaboración de esta tesis, en especial a nuestros padres que de forma constante nos ayudaron económica y moralmente, para conseguir nuestros objetivos propuestos.

Un agradecimiento especial para el Ingeniero Alberto Larco.

DEDICATORIA

A nuestros padres
y hermanos

TRIBUNAL DE GRADUACION

Ing. Gustavo Bermúdez
Decano de la FIEC

Ing. Alberto Larco G.
Director de TOPICO

Ing. Holger Cevallos
Miembro del Tribunal

Ing. Sixifo Falcones.
Miembro del Tribunal

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este trabajo, nos corresponde exclusivamente, y el patrimonio intelectual del mismo a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”

Juan Carlos García Tania

Alfonso Ignacio Mazacón Baño

José Luis Parrales Santistevan

RESUMEN

El proyecto a desarrollar consiste en el diseño de un programa de control y monitoreo para el proceso de empastado de las rejillas de plomo para la fabricación de baterías de plomo – ácido. Además en este proyecto se hace un estudio y análisis de una fuente de poder tipo conmutada y del diseño de una tarjeta de adquisición de datos analógicos para interactuar con las señales de este tipo existentes en el proceso.

En el capítulo 1 se procederá a realizar la descripción general del proceso de empastado de las rejillas de plomo, lo cual empieza con la selección de la materia prima que básicamente está conformada por barras de plomo. Estas barras son llevadas hacia un horno de fundición, en donde son sometidas a altas temperaturas dando como resultado el plomo líquido, que mezclado con ácido sulfúrico y agua destilada forma la pasta. Esta pasta es transportada hacia un silo mezclador en donde por medio de un rodillo se procede a empastar las rejillas que son trasladadas desde una banda transportadora. Una vez empastadas las rejillas estas son trasladadas hacia la etapa del horno de secado, el cual posee un sistema de combustión a gas (GLP.), en el cual se controla la temperatura y la presión, dando como resultado las placas de plomo que serán

transportadas hacia la etapa de recepción en donde se procederá a almacenarlas hasta ser utilizadas en los demás procesos.

En el capítulo 2 se desarrollará el monitoreo y control total del sistema de empastado de las rejillas de plomo (placas) para la fabricación de baterías de la empresa “Baterías Lux”, en donde para una mejor visualización se procedió a realizar todas las “pantallas” de control y monitoreo a través de un sistema SCADA (LOOKOUT 5.0 de National Instruments).

Este software es de mucha importancia ya que nos permite establecer una comunicación entre PCs vía Internet, también permite exportar su base de datos histórica (Citadel) hacia otras aplicaciones como EXCEL y SQL Server.

Actualmente el proceso está desarrollado a través de lógica tradicional y controladores de temperatura de la serie MIC2000.

En el capítulo 3 se estudiará y analizará, una fuente conmutada de alta frecuencia de 145 W, con una configuración tipo medio puente (half-bridge), con voltajes de salida de: +5V/ 18A; -5V/ 0.3A; +12V/ 4.2A; -12V/ 0.3A con una alimentación seleccionable de 120 o 240 VAC. Se desarrollará técnicas de ingeniería inversa para lo cual se obtendrá el

diagrama completo de la fuente y se desarrollarán los cálculos que corroboren la selección realizada sobre los elementos de la fuente. Adicionalmente se realizarán análisis mediante el software de simulación PSPICE PROFESIONAL de la familia ORCAD (Candente PSD 14.1 / PSpice). El diseño del circuito impreso de una sola pista será desarrollada usando el ruteador automático de PROTEL (Protel 99 SE Trial / Protel 99 SE), basado en la disposición física de los elementos. Los oscilogramas de las señales reales de las partes más importantes del circuito serán obtenidas con el Osciloscopio Tektronix 2220 y con el Multímetro gráfico Fluke (MODELO 867B). Finalmente se realizará un análisis comparativo entre las señales reales y simuladas.

En el capítulo 4 se procederá a realizar el diseño e implementación de una tarjeta de adquisición de datos a través del microcontrolador 16F877 de MICROCHIP, para lo cual se elaborará una hoja de selección indicando las principales características y funcionamiento de este microcontrolador. La tarjeta de adquisición fue diseñada con el objeto de enlazar el instrumento virtual desarrollado en LabVIEW con un variador de velocidad.

Finalmente, en el capítulo 5 se explica el diseño realizado en LabVIEW 7.1 para el control de la tarjeta de adquisición de datos, esto se realiza a

INDICE GENERAL

INTRODUCCION

CAPITULO 1

1. DESCRIPCION GENERAL DEL PROCESO

1.1 Introducción.....	3
1.1.1 Batería de Níquel – Cadmio.....	5
1.1.2 Batería De Níquel – Metalhidruro.....	5
1.1.3 Batería de Plomo – Acido.....	6
1.1.3.1 Reacción química de las baterías Plomo-Acido.....	8
1.1.3.2 Tipos de falla en baterías Plomo-Acido.....	11
1.1.3.3 Régimen de uso.....	13
1.1.3.4 Profundidad de descarga.....	14
1.1.3.5 Modelos comerciales de las baterías Plomo-Acido...	16
1.1.3.6 Energía almacenada.....	18
1.2 Proceso de fabricación de baterías.....	20
1.2.1 Elaboración de rejillas.....	22
1.2.2 Elaboración de la pasta.....	23
1.2.3 Proceso de empastado.....	24
1.2.3.1 Proceso de pegado de la pasta.....	24
1.2.3.2 Proceso desecado.....	26

1.2.4 Proceso de curado.....	31
1.2.5 Proceso de ensobrado.....	32
1.2.6 Proceso de formación y colocación de celdas.....	33
1.2.7 Proceso de sellado y colocación de bornes.....	34
1.2.8 Proceso de inyección de ácido.....	35
1.2.9 Proceso de carga y control del producto final.....	35

CAPITULO 2

2. DISEÑO DEL PROGRAMA DE MONITOREO EN “LOOKOUT”

2.1 Características y herramientas de LOOKOUT.....	37
2.2 Sistema de monitoreo y comunicación.....	42
2.2.1 El protocolo de comunicación "DDE".....	42
2.2.2 Comunicación del "PLC" con "KEPSERVER".....	44
2.2.3 Comunicación "KEPSERVER" con "LOOKOUT".....	47
2.3 Selección de la aplicación en LOOKOUT.....	49
2.3.1 Configuración de las pantallas.....	53
2.3.2 Edición de las pantallas.....	56
2.3.2.1 Pantalla de la máquina empastadora.....	56
2.3.2.2 Pantalla del horno.....	70
2.3.2.3 Pantalla de recepción de rejillas.....	81
2.3.2.4 Pantalla de la tolva de la máquina empastadora.....	88

2.3.2.5 Pantallas de control y de alarmas.....	92
--	----

CAPITULO 3

3. ANALISIS DE UNA FUENTE DE CONMUTACION

3.1 Introducción.....	104
3.2 Principio básico de las fuentes de conmutación.....	105
3.3 Diseño, cálculos y análisis de una fuente conmutada.....	107
3.3.1 Etapa de rectificación y filtrado en la entrada.....	107
3.3.2 Etapa de conmutación en alta frecuencia.....	112
3.3.3 Análisis de la etapa de rectificación y filtro de salida.....	121
3.3.4 Etapa de control (Modulación de ancho de pulso - PWM)...	129
3.3.5 Análisis de error y comparación entre los datos obtenidos por el simulador y los datos reales de la fuente.....	134

CAPITULO 4

4. DISEÑO DE UNA TARJETA DE ADQUISICIÓN DE DATOS BASADA EN EL MICROCONTROLADOR 16F877.

4.1 Descripción del sistema.....	135
4.1.1 Introducción.....	135
4.1.2 Diagrama de bloques del sistema de adquisición.....	136
4.2 Diseño del hardware.....	138
4.2.1 Características técnicas de la tarjeta.....	139

4.2.1.1 Entradas.....	139
4.2.1.2 Salidas.....	139
4.2.1.3 Alimentación.....	140
4.2.2 Selección del microcontrolador.....	141
4.2.3 Diagrama de bloque de la tarjeta de adquisición.....	142
4.2.4 Diseño de la tarjeta en PROTEL.....	144
4.2.5 Los puertos de E/S y recursos especiales del procesador...	145
4.2.5.1 Puertos de Entradas y Salidas.....	145
4.2.5.2 Módulo de comunicación serial.....	149
4.2.5.3 Generador de baudios.....	151
4.2.5.4 Transmisor Asíncrono.....	155
4.2.5.5 Receptor Asíncrono.....	157
4.2.5.6 Convertidor analógico-digital.....	159
4.3 Diseño del Software para el PIC.....	164
4.3.1 Introducción.....	164
4.3.2 Programación en MPLAB.....	167
4.4 Implementación del sistema.....	168
4.4.1 Análisis del sistema con el software de simulación	
Proteus.....	168
4.4.2 Fotos de la tarjeta de adquisición de datos.....	169
4.4.3 Costos.....	171

CAPITULO 5

5. DISEÑO DEL PROGRAMA DE MONITOREO Y CONTROL DE UN VARIADOR DE VELOCIDAD A TRAVES DE LABVIEW 7.1 STUDENT EDITION.

5.1 Características y Herramientas de LabVIEW.....	174
5.2 Creación Edición y Programación de Instrumentos Virtuales en LabVIEW 7.1.....	178
5.2.1 Creación de Instrumentos Virtuales.....	178
5.2.2 Edición de Instrumentos Virtuales.....	184
5.2.3 Programación de Instrumentos Virtuales.....	186
5.3 Descripción del sistema utilizado para la comunicación entre LabVIEW y la tarjeta de adquisición diseñada.....	190
5.3.1 Protocolos de Comunicación de LabVIEW.....	190
5.3.1.1 Puerto de Comunicación RS232.....	191
5.3.1.2 Instruments Drivers.....	194
5.3.2 Diseño del Control de la Tarjeta de Adquisición.....	196
Conclusiones y recomendaciones.....	207
Índice de figuras.....	210
Índice de tablas.....	220
Índice de anexos.....	222
Abreviaturas.....	223

través del puerto de comunicación serial RS 232. Para esto utilizamos un instrumento virtual de gran ayuda existente en LabVIEW que es el VISA (Virtual Instrument Standar Application), además de las estructuras tradicionales del software antes mencionado.

Glosario.....	229
Bibliografía.....	231
Anexos	

INTRODUCCION

Los sistemas automatizados han tenido un gran desarrollo en los últimos años, gracias a la gran cantidad de ventajas que tiene el usuario, así como a la facilidad de programación para el desarrollo de aplicaciones complejas y la capacidad que tienen estos sistemas para comunicarse hacia terminales remotos a través de los diferentes protocolos de comunicación, permitiendo un control a distancia del proceso.

El proceso de estudio de este proyecto es el empastado de las rejillas de plomo que forma parte del proceso para la fabricación de baterías de plomo – ácido. Se ha analizado el funcionamiento del sistema de transporte de las rejillas de plomo hasta la máquina empastadora, el empastado de las rejillas en dicha máquina y luego la recepción y almacenamiento de las mismas, para su posterior uso.

El control se lleva a cabo por medio de los sensores, el PLC y el programa que reside en él. El sistema de monitoreo se lo realiza a través del sistema SCADA “LOOKOUT 5.0” de National Instruments y se realiza una interacción entre las señales de campo y el programa mencionado anteriormente utilizando el OPC “KEPSERVER” por medio de los protocolos indicados para el caso. Además se realiza una interacción

entre Excel (Versión de Office XP) y Lookout 5.0 utilizando el protocolo DDE.

En gran parte de los procesos automatizados, es necesario manejar fuentes de alimentación DC, actualmente está en auge la utilización de fuentes conmutadas de alta frecuencia debido a su alta eficiencia, voltaje de salida estable y a que ocupa menor espacio físico, entre los dispositivos que generalmente necesitan una fuente de alimentación DC podemos mencionar: Tarjetas de Adquisición de Datos, Sensores que necesiten alimentación externa y Módulos de Ampliación del PLC. La fuente de nuestro estudio es una fuente de PC de 145 W, la cual tiene una configuración similar a las fuentes de conmutación para aplicaciones industriales.

Las tarjetas de adquisición de datos son una gran alternativa en el desarrollo de procesos automatizados, estas tarjetas son generalmente diseñadas para aplicaciones específicas, la tarjeta que construimos trabaja con el microcontrolador PIC16F877, el cual se encarga de recibir las señales analógicas provenientes del campo y a su vez enviar las señales hacia un DAC, el mismo que envía los valores analógicos que necesita el proceso que es objeto de nuestro análisis.

CAPITULO I

1. DESCRIPCION GENERAL DEL PROCESO

1.1 Introducción.

Sin entrar en mayores detalles técnicos el acumulador o batería es un dispositivo capaz de transformar energía potencial química en energía eléctrica. Se compone esencialmente de dos electrodos sumergidos en un electrolito donde se producen las reacciones químicas debidas a la carga y la descarga.

Erróneamente se suele hablar de “pilas recargables”, cuando este concepto está equivocado. Las pilas pueden ser salinas o alcalinas y no son recargables. Las baterías son aquellos acumuladores que se pueden recargar (aunque tengan forma de pilas).

El sentido de la utilización de las baterías está en la adaptación de los diferentes ritmos de producción y demanda de energía eléctrica, almacenando energía en los momentos en los que la producción es mayor que la demanda y cediendo energía en los momentos en que ocurre la situación contraria.

Esencialmente, una batería es un recipiente de químicos que transmite electrones. Es una máquina electro-química, o sea, una maquina que crea electricidad a través de reacciones químicas.

Las baterías tienen dos polos, uno positivo (+) y otro negativo (-). Los electrones (de carga negativa) van del polo negativo hacia el polo positivo, o sea, son recogidos por el polo positivo. A no ser que los electrones corran del polo negativo hacia el polo positivo, la reacción química no ocurre. Esto significa que la electricidad solo es generada cuando se le enchufa una carga, como un teléfono móvil, y la batería casi no se gasta si es guardada en un cajón.

Existen diversas formas de medir la capacidad de las baterías. En general se utilizan los Amperes que en forma constante es capaz de entregar la batería en un ciclo de 20 horas, así una batería de 55 A/h provee 2,75 A durante 20 horas ($2,75 * 20 = 55$).

Los tipos de baterías que normalmente encontramos en el mercado son:

1.1.1 Batería de níquel – cadmio.

Proporciona una tensión de 1.2 V. Contienen cadmio, un metal pesado que representa un peligro ecológico. Exteriormente tienen la misma forma y tamaño de las pilas. Interiormente tienen dos electrodos, el de cadmio (negativo) y el de hidróxido de níquel (positivo), separados entre sí por un electrolito de hidróxido de potasio. Llevan también un separador situado entre el electrodo positivo y la envoltura exterior y un aislante que las cierra herméticamente.

1.1.2 Batería de níquel – metalhidruro.

No tienen metales pesados como el cadmio y por eso son menos perjudiciales para el medio ambiente. Además de ser menos contaminantes proporcionan tensiones de 1.3 voltios y tienen una capacidad mucho mayor por lo tanto duran más que las de Níquel-Cadmio y dan más energía. No tienen efecto memoria, de modo que se pueden recargar aunque no se hayan agotado al completo. Las mejores llegan a soportar hasta 1.000 procesos de carga.

1.1.3 Batería de plomo – ácido.

Las baterías de ácido de plomo tienen un voltaje nominal de alrededor de dos voltios. La mayoría de baterías de coche se componen de seis celdas y por ello tienen una tensión de doce voltios.

Las primeras baterías de plomo-ácido (acumuladores de plomo), fueron fabricadas a mediados del siglo XIX por Gastón Planté. Hoy en día todavía son uno de los tipos de baterías más comunes. Se descubrió que cuando el material de plomo se sumergía en una solución de ácido sulfúrico se producía un voltaje eléctrico el cual podía ser recargado.

Este tipo de baterías es único en cuanto que utiliza el plomo, material relativamente barato, tanto para la placa positiva como para la negativa.

El material activo de la placa positiva es óxido de plomo (**PbO₂**).

El de la placa negativa es plomo puro esponjoso y el electrolito está disuelto en ácido sulfúrico (**H₂SO₄**).

Cuando hablamos de material activo en las baterías de ácido de plomo, nos referimos al óxido de plomo y al plomo esponjoso.

Pila convencional (LeClanche):

<u>Usos:</u>	Son las pilas que utilizamos para aparatos como walkman, juguetes, etc...
<u>Química:</u>	Anodo: Zn; Catodo: Pasta de NH ₄ Cl, ZnCl ₂ y MnO ₂ ;
<u>Ventajas:</u>	----
<u>Desventajas:</u>	Corta vida y produce algunos gases que ocasionan una bajada de voltaje en las pilas.

Pila alcalina:

<u>Usos:</u>	Son las pilas que utilizamos para aparatos como walkman, juguetes, etc...
<u>Química:</u>	Anodo: Zn; Catodo: Mn;
<u>Ventajas:</u>	No produce gases.

Pilas de mercurio:

<u>Usos:</u>	Son las pilas que utilizamos para aparatos como calculadoras, relojes, etc... (de boton)
<u>Química:</u>	Anodo: Zn; Catodo: Hg;
<u>Ventajas:</u>	pequeña
<u>Desventajas:</u>	Contamina mas que las demas el medio ambiente.

Baterías para acumular grandes cantidades de energía:**Baterías de Plomo:**

<u>Usos:</u>	Son las baterías que se usan en los coches.
<u>Química:</u>	Anodo: Pb poroso; Catodo: Pb comprimido; inmersos en H ₂ SO ₄ (aq.).
<u>Ventajas:</u>	recargable, proporciona una gran corriente
<u>Desventajas:</u>	grande y pesada, en comparacion poca energia

Baterías de Ni/Cd:

<u>Usos:</u>	Son las pilas que se usan para dispositivos portátiles.
<u>Química:</u>	Anodo: Cd; Catodo: Ni;
<u>Ventajas:</u>	recargable, prácticamente constante voltaje, ligera
<u>Desventajas:</u>	----

Tabla 1.1. Cuadro comparativo de los usos y ventajas de las clases de baterías.

Actualmente existen dos tipos de baterías recargables que dominan el mercado: las baterías de plomo y las de níquel-cadmio.

Las baterías de plomo reinan en nuestros automóviles pero sólo destinadas a cubrir las necesidades de arranque, iluminación e ignición (no tienen suficiente energía para mover el coche).

Las baterías de níquel-cadmio a falta de mejores baterías, se emplean en artículos de electrónica de consumo como videocámaras y ordenadores o teléfonos móviles.

1.1.3.1 Reacción química de las baterías plomo-ácido.

La batería es un acumulador de energía que pone en marcha generalmente motores de combustión. La acumulación de energía se realiza por medio de un proceso químico entre placas de plomo y un líquido llamado electrolito formado por agua y ácido sulfúrico.

La reacción química que se produce en la batería plomo ácido ($\text{PbO}_2 + \text{Pb} + 2\text{H}_2\text{SO}_4 \leftrightarrow 2\text{PbSO}_4 + 2\text{H}_2$)

+ $1/2 O_2$) genera energía eléctrica y además, es reversible. Alimentando a la batería con una determinada tensión se consigue recombinar los elementos químicos que regeneran la carga de la batería. Todo este proceso de carga y descarga origina un desgaste de las placas internas.

El ácido se agota en la descarga y se regenera en la recarga. Se forman hidrógeno y oxígeno durante la descarga y carga flotante (debido a que la carga flotante está neutralizando la auto-descarga). En baterías sumergidas, estos gases escapan y por tanto se debe agregar agua periódicamente. En baterías reguladas por válvula, Plomo-Acido, los gases hidrógeno y oxígeno se recombinan para formar agua.

Adicionalmente, en baterías VRLA (selladas), el ácido está inmovilizado por una lámina de fibra de vidrio absorbente (AGM) o en un gel. La lámina es semejante al aislamiento de fibra de vidrio usando en las casas. Atrapa el hidrógeno y el oxígeno formado durante la descarga y les permite migrar en forma tal que

reaccionan nuevamente para formar agua. Es por esto que el tipo VRLA nunca necesita que se agregue agua, comparado con las baterías Plomo-Acido sumergida (húmeda, ventilada).

Internamente la batería tiene placas positivas y negativas en forma alternada, separadas por caucho micro-poroso en las sumergidas tipo Plomo-Acido; por láminas de fibra de vidrio absorbente en las tipo VRLA; por ácido gelatinoso en baterías gel VRLA o láminas de plástico en la tipo NiCd. Todas las placas de polaridad se sueldan entre sí y al terminal apropiado. En el caso de celdas VRLA, se ejerce alguna presión a modo de sandwich entre placa-lámina-placa para mantener un buen contacto entre ellos. Existe además una válvula de alivio de presión (PRV) con auto-resellado, para ventilar los gases cuando ocurre una sobre presurización.

1.1.3.2 Tipos de falla en las baterías de plomo

– ácido

Cada batería tiene varios tipos de fallas, algunos de las cuales son más frecuentes que otros. En baterías Plomo-Acido sumergida, los tipos de falla comunes son los listados a continuación:

Corrosión de la rejilla o terminal positivo

Formación de sedimento (desprendimiento)

Corrosión de cables de salida superior

Sulfatación de placas

Sedimentación (trozos de pasta)

Algunos de estos tipos de fallas se manifiestan por sí mismos con eventos tales como la formación de sedimentos debido a un excesivo número de ciclos de operación repetitivos. Otros ocurren naturalmente tales como el crecimiento de la rejilla positiva (oxidación). Solo es cuestión de tiempo antes de que falle una batería. El mantenimiento y las condiciones ambientales pueden incrementar o disminuir el riesgo de falla prematura de la batería.

Los separadores porosos no son conductores, y evitan cortocircuitos, cada grupo forma una celda con un voltaje algo superior a los 2 voltios. El voltaje de cada celda es el mismo sin importar su tamaño y el número de placas. Para lograr voltajes mas altos las celdas se deben conectar en serie (por ejemplo 6 celdas producirán 12 voltios).

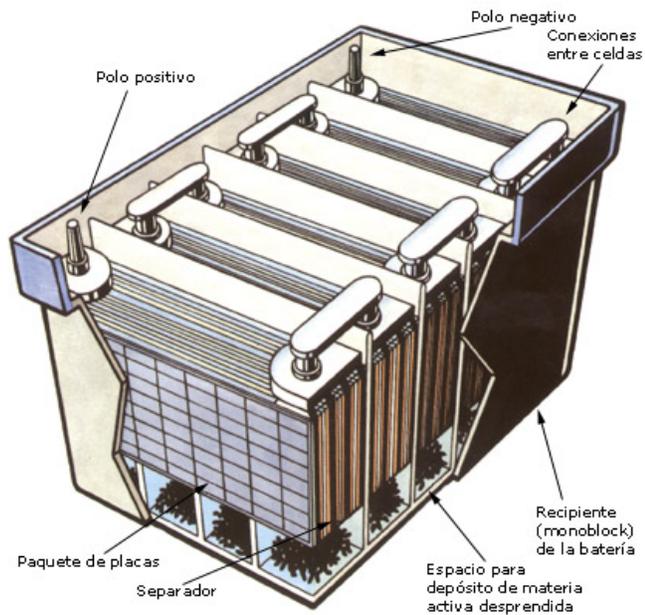


Figura 1.1. *Arquitectura interna de una batería de 12 Voltios.*

1.1.3.3 Régimen de uso.

El modelo de batería usado en los automotores está diseñado para sostener corrientes elevadas (200 a 350 A) por muy breves instantes (segundos) durante el arranque del motor. El resto del tiempo la batería está siendo cargada o permanece inactiva. La batería de un sistema solar, por el contrario, debe ser capaz de sostener corrientes moderadas (una decena de amperes), durante horas. Además, en muchas aplicaciones deberá permanecer activa sin recibir carga alguna (servicio nocturno). Normalmente los períodos de reposo son nulos, ya que está siendo cargada o descargada. Diferentes requerimientos de uso sólo pueden satisfacerse con diseños distintos.

Las placas de una batería de Pb-ácido para automotor están hechas con plomo esponjoso. Esta técnica de fabricación hace posible el obtener un máximo de superficie activa con un mínimo de peso. La reducción en la cantidad de plomo contribuye, en forma substancial, a abaratar su costo. El incremento de la

superficie activa disminuye la densidad de corriente (A/cm^2), permitiendo alcanzar niveles muy altos para la corriente de carga, por breves instantes. Las baterías de automotor llamadas de “mantenimiento nulo” tienen electrodos con aleación de calcio lo que contribuye a minimizar la **gasificación** durante la carga. Esto hace que se reduzca la pérdida de agua en el electrolito. El agregado de esta aleación disminuye, asimismo, las pérdidas por autodescarga, permitiendo la retención de la carga durante largos períodos de inactividad.

1.1.3.4 Profundidad de descarga.

La batería de automotor está diseñada para soportar una leve profundidad de descarga (PD). Para ilustrar este aspecto, asumiremos que en invierno la corriente de arranque alcanza 350A y dura 3 segundos. Durante el arranque la batería habrá entregado 0,29Ah ($350 \times 3/3.600$). Como la capacidad típica de estas baterías es de 80Ah, los 0,29Ah representan una PD de sólo 0,36%. Tomaría tres arranques consecutivos para que la PD llegase al 1%. La batería solar permite una PD máxima del 80%, cientos de veces, a niveles de

corriente moderados. Es por ello que a estas baterías se las denomina de ciclo profundo (BCP).

Se considera que una BCP ha completado todos los ciclos de carga y descarga cuando, al ser cargada nuevamente, la máxima energía que puede almacenar se reduce al 80% de su valor inicial. El número de ciclos de carga/descarga depende de la PD. Cuando ésta disminuye, el número de ciclos aumenta. Para una dada PD, la batería más robusta proporciona el mayor número de ciclos. La siguiente tabla muestra el número de ciclos y las PD para dos baterías distintas.

BATERIA	NUMERO DE CICLOS	%PD
217 Ah – 30 Kg	543	80
	757	50
	1100	30
	1800	20
350 Ah – 65 Kg	693	80
	1068	50
	2050	30
	3050	20

Tabla 1.2. *Tabla comparativa del número de ciclos y %PD entre dos baterías.*

1.1.3.5 Modelos comerciales de las baterías

Plomo-ácido.

Las versiones con mayor aceptación son las de 6 y 12V nominales. Baterías de 6V, con una capacidad de unos 200Ah, son utilizadas en sistemas de mediana capacidad de reserva, donde pasan a formar parte de un banco de baterías con conexión serie o serie-paralelo, a fin de satisfacer los valores de voltaje y corriente del sistema. Esta versión ofrece el mejor

compromiso entre peso (facilidad de manejo) y número de Ah de reserva. Como los sistemas FVs de bajo consumo son sistemas de 12V nominales, los requerimientos de reserva pueden ser satisfechos con la versión de 12V, la que tiene una capacidad de unos 100Ah. Baterías de más de 250Ah resultan convenientes cuando se trabaja con sistemas de alto consumo. Se fabrican asimismo baterías de ciclo profundo con capacidad de reserva mucho más grandes. Un modelo de 24 V, diseñado para ser usado en un sistema comunal (varias casas conectadas a un sistema FV) tiene una capacidad de 1.493Ah y pesa 1.200 Kgs.

La doble conversión de energía que toma lugar en una batería resulta en una eficiencia total η_b que está dada por la expresión:

$$\eta_b = \eta_c \times \eta_d$$

Donde η_c es el valor de la eficiencia de carga y η_d el de descarga.

Ambos valores varían con la temperatura del electrolito y el estado de carga de la batería, ya que la resistencia

interna de la misma genera estas pérdidas. Las pérdidas durante el proceso de carga fuerzan un incremento en la capacidad de generación del sistema, a fin de restablecer el balance energético del mismo.

1.1.3.6 Energía almacenada.

La capacidad de almacenamiento de una batería de Pb-ácido varía con la temperatura del electrolito, la que, en la práctica, está determinada por la temperatura ambiente del lugar donde ésta será instalada. La Tabla 1.3 muestra esta interdependencia.

TEMPERATURA (°C)	CAPACIDAD ALMACEN. (%)	EFICIENCIA
30	105	1,05
25	100	1,00
16	90	0,90
4	77	0,77
-7	63	0,63
-18	49	0,49

Tabla 1.3. *Variación de la energía almacenada de una batería con respecto a la temperatura de operación.*

Cuando se usan baterías de Pb-ácido es importante mantener la temperatura del electrolito cercana a los 25°C, ya que a esta temperatura se alcanza el balance óptimo entre la eficiencia y la vida útil de este componente.

1.2 Proceso de fabricación de baterías.

El proceso de fabricación de Baterías de plomo ácido empieza desde la obtención de la materia prima, pasando por la elaboración de las rejillas de plomo, el “empastado” con una mezcla química de estas rejillas, el proceso de curado de las mismas, el ensobrado y la formación de las celdas generadoras de energía, el sellado térmico, la colocación de los bornes, la inyección de ácido, hasta llegar finalmente al proceso de carga.

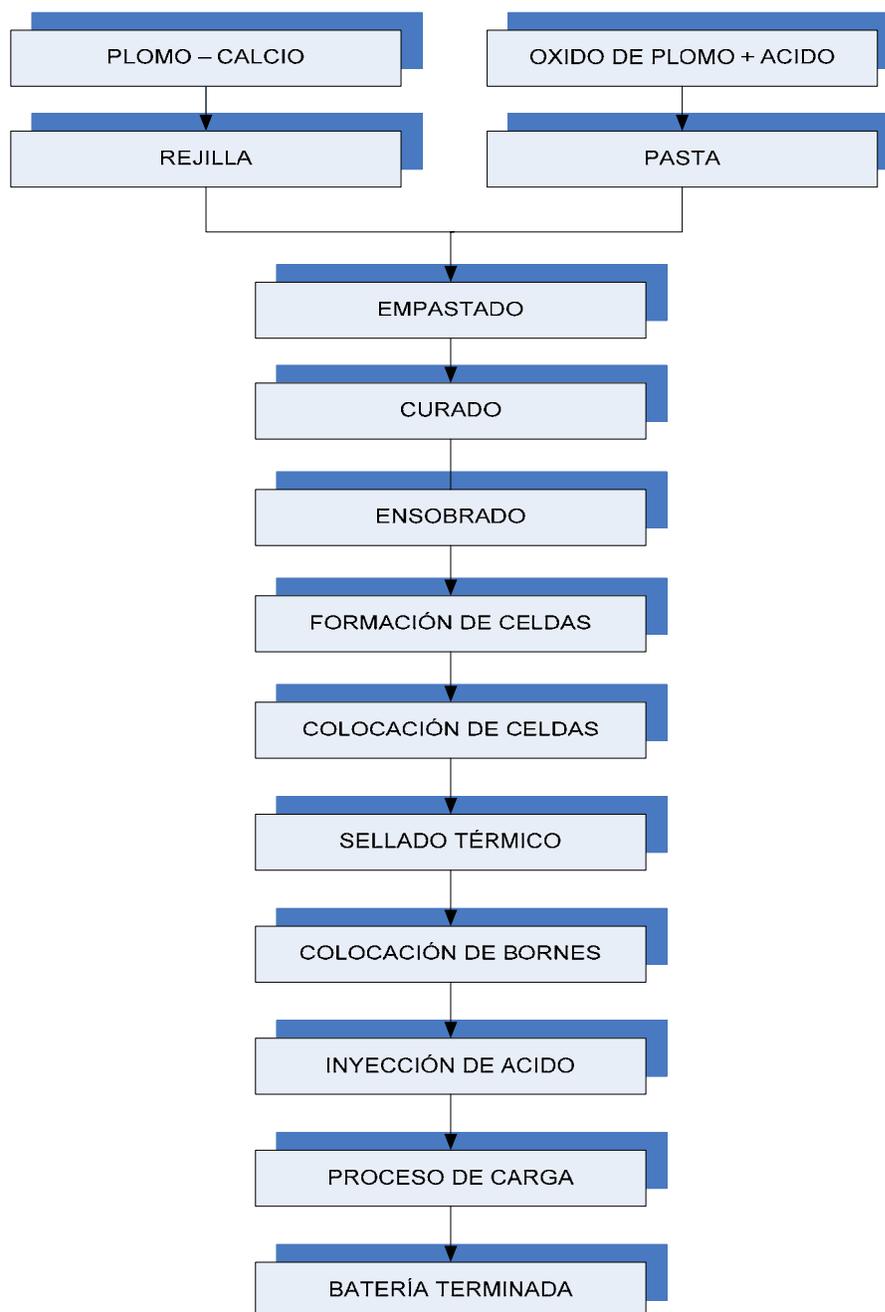


Figura 1.2. *Diagrama de Bloques del Proceso de Fabricación de Baterías de Plomo-Ácido.*

1.2.1 Elaboración de rejillas.

La materia prima básicamente está formada por barras de plomo, del cual un porcentaje es importado y el resto es nacional. El plomo que se usa es generalmente una aleación con calcio, antimonio o plata según la calidad de batería a fabricar.

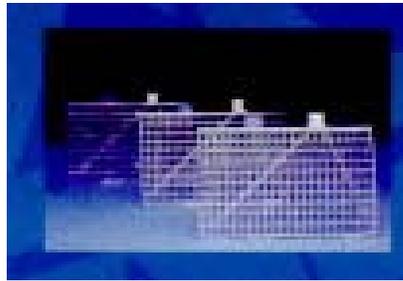


Figura 1.3. *Rejillas de plomo para fabricación de baterías.*

El proceso de obtención de las rejillas empieza cuando se coloca las barras de plomo en estado sólido en un horno de fundición, en el cuál las barras se someten a altas temperaturas hasta llevarlas a estado líquido, desde el horno a través de una bomba se traslada el plomo hacia las máquinas rejilladoras. Cada máquina genera rejillas de diferentes dimensiones y pesos (según el tipo de baterías que se fábrica).

1.2.2 Elaboración de la pasta.

La materia prima para la elaboración de la pasta se encuentra conformada por plomo puro en barra, solución de ácido sulfúrico y agua destilada.

La pasta se obtiene cuando se mezcla óxido de plomo, ácido sulfúrico y agua destilada. El óxido de plomo se lo obtiene de la siguiente manera:

Las barras de plomo puro son colocadas en un horno que se encuentra a una temperatura superior a 500° C, en este horno el plomo se encuentra en estado líquido, de este lugar es bombeado a una máquina llamada “revolver” que se encarga de convertir el plomo en pequeñas barras cilíndricas de 1cm de altura por 1cm de diámetro, del revolver las barras son transportadas hacia el molino del cual se obtiene el óxido de plomo.

El óxido de plomo es trasladado hacia la tolva mezcladora en la cual se le agrega ácido sulfúrico y agua en porcentajes ya “seteados” por el controlador; una vez lista la pasta, ésta se vacía sobre la tolva de empastado.

1.2.3 Proceso de empastado.

El proceso de empastado se divide en dos sub-procesos, el primero es el proceso de pegado de la pasta en las rejillas y el segundo el proceso de secado de las rejillas ya empastadas.

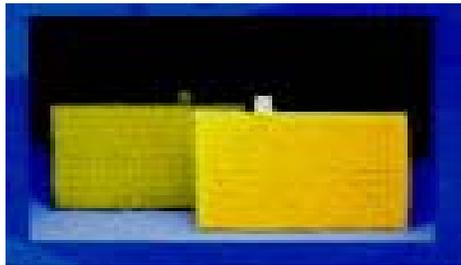


Figura 1.4. *Vista frontal de las rejillas una vez empastadas.*

1.2.3.1 Proceso de pegado de la pasta.

Las rejillas son apiladas de manera vertical en una banda transportadora (Banda1), que es accionada por el motor MB1. Por medio de una máquina llamada “Grid Picker” se traslada de manera unitaria cada una de las rejillas hacia una segunda banda transportadora (Banda2), que es accionada por el motor MB2, en donde su posición cambia a horizontal. El “Grid Picker” es accionado por medio del motor MGP, el cual es de velocidad variable, es decir, que éste determina la

cantidad de rejillas que pasan a la Banda2; entonces la velocidad del motor es proporcional a la cantidad de rejillas que pasan por esta banda transportadora.

A través de la Banda2 se transportan las rejillas hacia la tolva de empastado en donde se encuentra la pasta que es mezclada uniformemente por unos agitadores internos de la tolva. La pasta desciende hasta un rodillo que se encuentra en la parte inferior de la tolva; éste compacta la pasta en la rejilla a través del contacto con ésta. Luego que pasa por la tolva, las rejillas empastadas (placas) son transportadas hacia el horno de secado.

Además en la entrada de la tolva de empastado existen dos “microswitch”. El primero es para que detecte si hay o no presencia de rejillas para ser empastadas, y el segundo es para detectar si hay dos rejillas pegadas y asegurarse de que solo pase una rejilla a la vez para que sea empastada. En el momento en que se accionan cualquiera de los dos, estos envían una señal de error hacia el panel operador del proceso.

1.2.3.2 Proceso de secado.

Las rejillas empastadas son transportadas, por medio de una banda transportadora que es accionada por el motor MBH, esta banda pasa a través del horno de secado el cuál posee un sistema de combustión a gas (GLP) en el cuál se controla la temperatura y la presión.

Para la temperatura existen tres controladores digitales, ubicados en el panel de mando, uno para el límite máximo de temperatura al interior del horno, otro para el control del lazo cerrado de la temperatura en el interior del horno y por último un indicador de temperatura en el interior de la cámara de combustión.

Indicadores de presión, existen los siguientes: uno para la presión del gas en la tubería principal del sistema de suministro de combustible, otro para medir la caída de presión en la tubería de gas piloto. También presenta un indicador de presión en la cámara de combustión y por último, un indicador de la presión en el horno. Además existen “switchs” de presión colocados en los

sitios donde se encuentran instalados los indicadores mencionados anteriormente.

También posee un foto sensor colocado en la salida del horno, el cual sirve para detectar la presencia de placas en el horno así como también para llevar un control de la producción diaria de placas.

Los sistemas indicados anteriormente, junto con las botoneras de marcha y paro del sistema de control del horno, sirven para controlar los siguientes actuadores:

Motores:

- ✓ **Oven Conveyor (MBH).**- Trifásico, rotor Jaula de Ardilla de 5 Hp. Se encarga de mover la banda transportadora que atraviesa el horno.

- ✓ **Combustión Fan (M2).**- Trifásico, rotor Jaula de Ardilla de 3 Hp que acciona el ventilador de la cámara de combustión que impulsa el aire caliente hacia el horno.

- ✓ **Recircle Fan (M3).**- Trifásico, MIJA de 15 Hp. Acciona el ventilador que se encarga de hacer recircular el aire en el interior del horno.

- ✓ **Exauhst Fan (M4).**- MIJA trifásico de 1 Hp. Mueve el ventilador colocado en el chimenea del horno. Es de menor potencia que M3, y es el encargado de sacar el aire caliente del horno (así evitamos que el horno tenga en su interior presiones elevadas que lo harían explotar).

Válvulas:

- ✓ **Blocking Valve.**- Válvula solenoide colocada en la tubería de gas principal, para permitir el paso de flujo de gas a través de esta.

- ✓ **Pilot Valve.**- Válvula solenoide que permite el ingreso de gas por la tubería de gas piloto.

- ✓ **Damper.**- Actuador proporcional que controla el flujo de gas que circula por la tubería de gas principal.

El modo de operación del horno es como sigue:

Al pulsar la botonera de marcha se encienden los motores "M3" y "M4", los cuales purgan el interior del horno de cualquier residuo en el aire dejado en procesos anteriores. Esto lo hacen durante 60 segundos, luego del cual se enciende el motor "M2".

Luego por medio del pulsador "Ignition Start" (Inicio de combustión) se inicia la combustión del gas, a través de un módulo de encendido electrónico, que controla las válvulas "Pilot Valve" y "Blocking Valve". Primero solo se energiza "Pilot Valve", ya que por medio de esta válvula se determina el momento en el cual el flujo de gas es el adecuado para permitir el paso del combustible por la válvula principal "Blocking Valve".

En ese momento se enciende el motor de la banda transportadora y se inicia el control de lazo cerrado de temperatura y presión que actúa sobre el "Damper", el

cual regula gradualmente el paso del gas a través de la tubería principal.

Si sucediera un incremento en las variables controladas superior a los valores “seteados” (de configuración), se activa una alarma y se apaga todo el sistema de combustión (válvulas solenoides y Motores), en este caso debe reiniciar el proceso (pulsando la botonera de paro) y revisar los valores configurados en el panel de mando.

El rodillo posee un **sistema mecánico** que lo desplaza verticalmente para ajustar la cantidad de pasta colocada en la rejilla, esto se debe a que la cantidad de pasta varia según el tipo de rejilla a procesarse (acordémonos que existían varios tipos), así también para cumplir con las especificaciones de peso que deben tener las placas una vez que salen del horno, para cumplir con esta especificación, actualmente existe un lazo de control muy singular, el cual lo narramos a continuación:

Una placa de las que salen del horno es seleccionada por un operador ubicado a la salida del horno, para pesarla en una balanza de precisión, si el peso es superior al deseado, el operador toca un timbre luminoso colocado a la izquierda de los operadores de la empastadora, entonces estos operadores desplazan el rodillo hacia abajo, si el timbre que tocan desde el horno es el derecho, los operadores desplazan el rodillo hacia arriba. Este proceso se realiza en el transcurso de la operación normal de la planta.

1.2.4 Proceso de curado.

Se llama así al proceso en el cual se somete a determinados valores de temperatura y humedad a las placas procedentes del proceso de empastado, esto permite una mayor adherencia de la pasta a la rejilla, este proceso consiste básicamente en lo siguiente:

Cierta cantidad de placas, procedentes de la empastadora, son colocadas en un lugar llamado “**Cuarto de Curado**”, en este lugar empieza el proceso que consiste en dos partes:

Las placas son sometidas a valores altos de humedad y temperatura durante 18 horas aproximadamente (varía según el tipo de placas).

Los valores de humedad y temperatura disminuyen, este proceso dura aproximadamente 5 horas.

Las placas son retiradas del cuarto de curado, Luego, son almacenadas hasta el momento en que sean trasladadas hacia el lugar donde se encuentran las maquinas ensobradoras.

1.2.5 Proceso de ensobrado.

Las placas provenientes del cuarto de curado son trasladadas hacia las máquinas ensobradoras, las cuales se encargan de fabricar el “sobre” en el que va la placa, además de colocar dichas placas en el interior de éste.



Figura 1.5. *Vista de las placas una vez que son colocadas en el sobre.*

El material utilizado para la fabricación de los sobres es polietileno, con lo cual se elimina el riesgo de posibles cortocircuitos y mejoran la conductividad eléctrica.

A la salida de la ensobradota existe una banda que se encarga de apilar de diez en diez las placas ya ensobradas, para que de esta manera queden listas para el siguiente proceso.

1.2.6 Proceso de formación y colocación de celdas.

Para este proceso existen dos líneas de producción, cada línea posee su respectiva máquina totalmente automatizada llamadas “cangrejo” y “pulpo” respectivamente, el proceso inicia cuando el operador coloca las placas provenientes de las ensobradoras (en grupos de diez) en las maquinas anteriormente nombradas.



Figura 1.6. *Vista interna de las celdas de una batería.*

Estas máquinas se encargan de soldar las placas (formar las celdas), agregarle un terminal de salida a las celdas y dejarlas listas para ser colocadas en las cajas. La colocación en las cajas se la realiza de manera manual a la salida del “pulpo” y el “cangrejo”.

1.2.7 Proceso de sellado y colocación de bornes.

Las cajas provenientes del proceso anterior son llevadas hacia unas máquinas que se encargan de soldar las celdas y sellar la tapa que cubre y protege la caja, todo este proceso se lo realiza a altas temperaturas.

Luego las cajas ya selladas son trasladadas, por medio de una banda transportadora, hacia donde se encuentran los operadores encargados de colocar los bornes, este proceso se lo realiza a través de la colocación de un molde en cuyo interior se coloca el plomo derretido y de esta manera se forman los bornes en la batería.

1.2.8 Proceso de inyección de ácido.

Una vez selladas y con sus respectivos bornes, las baterías son trasladadas hacia el cuarto de colocación de ácido en donde existe una máquina automatizada, la cual detecta la presencia de las baterías para empezar el proceso de inyección de ácido en cantidades previamente ajustadas por el operador, estas cantidades de ajuste dependen del tipo de batería a la que se le va a inyectar el ácido sulfúrico.

1.2.9 Proceso de carga y control del producto final.

Las baterías semi listas provenientes del cuarto de colocación de ácido, son colocadas en las líneas de carga (120 en total), cada línea de carga tiene capacidad para quince baterías del mismo tipo, para que la carga sea uniforme. Todas las líneas de carga se encuentran automatizadas y se “setea” los valores de carga (voltaje, tiempo de carga y amperios-horas), desde el computador ubicado en el cuarto de carga; en dicho cuarto se encuentra el control de todas las líneas.



Figura 1.7. *Batería lista. Producto terminado.*

Cuando las líneas de carga completan el tiempo de carga de sus valores “seteados”, las baterías son llevadas por el personal de cuadrilla hacia una banda transportadora que las lleva a través de un cuarto de limpieza donde se las lava con agua, y luego pasan hacia el cuarto de control del producto final donde se realiza un control estadístico y de calidad de la cantidad de baterías producidas.

Cabe indicar que en todos los procesos mencionados existe un riguroso control de calidad.

CAPITULO II

2. DISEÑO DEL PROGRAMA DE MONITOREO EN “LOOKOUT”.

2.1 Características y herramientas de Lookout.

Lookout es un programa para automatización industrial que opera bajo los modos de Interface-Hombre-Maquina, y Supervisión-Control y Adquisición de Datos (HMI y SCADA respectivamente, por sus siglas en ingles). Lookout trabaja bajo Windows y se comunica con campos de I/O (Entradas/Salidas) a través de hardware de control. Típicas aplicaciones de Lookout incluyen monitoreo, supervisión y control continuo del proceso, manufacturación discreta, aplicaciones conjuntas, y sistemas de telemetría remota.

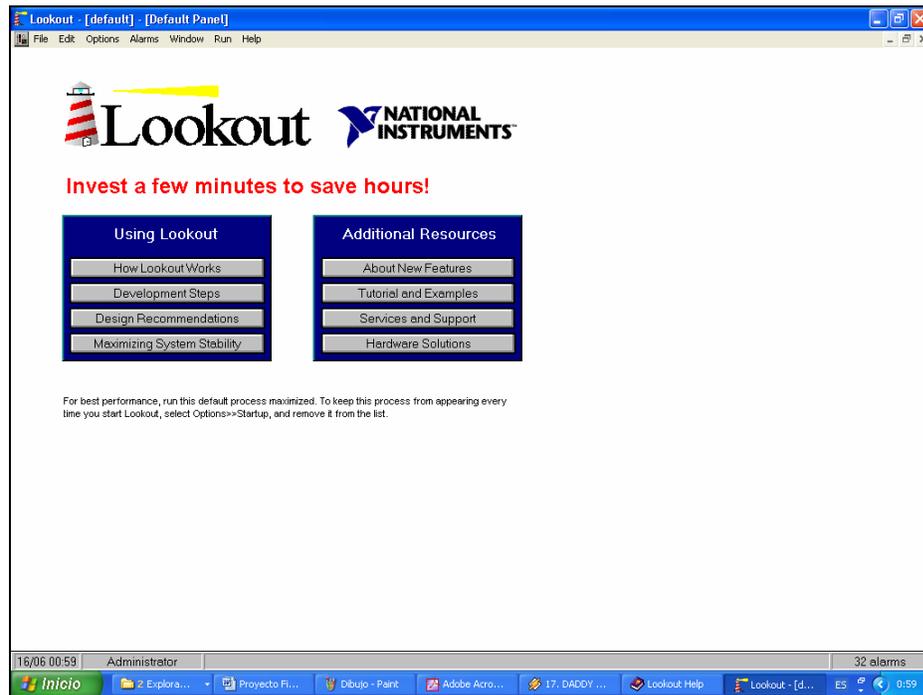


Figura 2.1. Pantalla inicial del “*Default Panel*” que aparece al iniciar Lookout.

Lookout ofrece las siguientes características y capacidades.

- Puerto de comunicación serial – Se puede comunicar a ciertos “hardwares” de control a través de los puertos seriales de la computadora. Lookout usa los puertos seriales de manera arbitraria. Refiérase al Capítulo 4, “*Serial Port Communication*”, del “Lookout Developer’s Manual” para mas información.
- Gráficos – Lookout tiene una extensa librería de gráficos estándar, como el “*Image Navigator*” y su librería de gráficos, que usted puede

usar para construir su interface de usuario. Usted también puede crear sus propios gráficos personalizados para usarlos en Lookout. Refiérase al Capítulo 6, “*Graphics*”, del “Lookout Developer’s Manual” para mayor información.

- Alarmas – En Lookout usted puede generar, mostrar, guardar, filtrar, agrupar e imprimir alarmas. Refiérase al Capítulo 9, “*Alarms and Events*”, del “Lookout Developer’s Manual” para mayor información.
- Multimedia – Lookout permite reproducir archivos de sonidos. También, se puede crear animaciones en sus paneles de Lookout usando objetos como el “*Animator, Multistate y Pipe*”. Refiérase a los tópicos de esos objetos en “*Lookout Help*” para más información acerca de la implementación de esas características.
- Seguridad - Lookout proporciona un sistema de seguridad local y de red. Usted puede configurar la red de seguridad, control de seguridad, vistas de seguridad y acción de seguridad. Usted puede especificar el nivel de acceso y control para cada operador o maquina de la red. Refiérase al Capítulo 10, “*Security*”, del “Lookout Developer’s Manual” para más información acerca de seguridad.

- Guardar Alarmas y Eventos – Usted puede almacenar en tiempo real información en archivos ASCII, o usted puede guardar un histórico de datos en la base de datos Citadel. Lookout guarda los comandos del operador, al cerrar los archivos o hacer cambios en ellos. Con cada evento, Lookout guarda el nombre de la cuenta (operador), fecha y tiempo del evento, nombre del objeto ajustado, y los parámetros anteriores y posteriores del objeto. Refiérase al Capítulo 8, “*Logging Data and Events*”, del “Lookout Developer’s Manual” para mayor información.
- Soporte ODBC – Debido al soporte de conectividad (ODBC) de la base de datos de Lookout, usted puede usar otras aplicaciones, como Microsoft Access, para trabajar con los datos históricos de Citadel. Refiérase al Apéndice B “*Structured Query Language*”, del “Lookout Developer’s Manual”, para más información acerca de ODBC.
- Soporte DDE – Lookout puede enviar los valores de sus procesos activos hacia otras aplicaciones, y puede recibir en tiempo real valores de otras aplicaciones. Lookout puede actuar como un cliente DDE o como un servidor DDE. Refiérase al Capítulo 11, *DDE*, del “Lookout Developer’s Manual” para mas información.

- Red – Lookout provee de una rápida velocidad en una red cliente-servidor a través del uso del TCP/IP. Usted puede monitorear y controlar sus procesos desde múltiples estaciones de trabajo en una red. Refiérase al Capítulo 13, “*Networking and Deploying an Application*”, del “Lookout Developer’s Manual” para mayor información acerca de redes de computadoras en Lookout.
- OPC – Lookout es un completo cliente y servidor OPC, debido a la gran cantidad de protocolos de comunicación industrial con los que puede interactuar. Para mayor información, refiérase al Capítulo 3 del “Lookout Developer’s Manual”
- Redundancia – Usted puede configurar dos computadoras por redundancia, proporcionando una transferencia automática del monitoreo y control si una de las computadoras falla. Refiérase al Capítulo 14, “*Setting Up Redundancy*”, del “Lookout Developer’s Manual” para más información.
- Capacidades Web – Con el reporte de Objetos y Cliente Web, usted puede monitorear e incluso controlar los procesos a través de Microsoft Internet Explorer. Usando el control “Lookout WebBrowser

ActiveX”, usted puede acceder a la web desde dentro de un proceso en Lookout. Para mas información refiérase a la sección “Creando una Página Web Cliente en Lookout” del Capítulo 12, en el “Lookout Developer’s Manual”.

2.2 Sistema de monitoreo y comunicación.

2.2.1 El protocolo de comunicación "DDE".

El intercambio dinámico de datos DDE es el protocolo de mensajes de Microsoft usado por aplicaciones parecidas a Microsoft Excel y Lookout para “enlazar” datos en otras aplicaciones. Cuando los datos en una aplicación fuente cambian, este protocolo actualiza dinámicamente todos los datos enlazados (en tiempo real).

Con DDE, usted puede enlazar dinámicamente otras aplicaciones de Windows a Lookout. Si Lookout obtiene datos desde otras aplicaciones, Lookout es el cliente. Pero si otra aplicación obtiene datos desde Lookout, Lookout es el servidor. Lookout puede ser ambos, servidor y cliente, en una

configuración “peer to peer”, refiérase al ejemplo “DDE peer to peer” del Capítulo 11, del “Lookout Developer’s Manual”.

La aplicación es responsable para el establecimiento de un enlace DDE con el servidor. Para establecer un enlace DDE, la aplicación cliente debe de identificar la localidad de los datos deseados. Una dirección de tres-niveles identifica la localidad de los datos: *Service, Topic, Item*.

Service, especifica el nombre de la aplicación servidor a enlazarse con el cliente. Cada aplicación que soporta DDE tiene un único nombre de servicio. Por ejemplo, Lookout es el nombre de servicio de Lookout, y EXCEL es el nombre de servicio de Microsoft Excel.

Topic, es el segundo nivel en una dirección de tres-niveles. Para algunas aplicaciones de servidores como Excel y Lookout, “*Topic*” especifica un archivo en particular. En Lookout “*topic*” es el nombre del archivo del proceso.

Item, identifica el dato específico o valor a ser enlazado entre el servidor y el cliente. Un Lookout item es el nombre del objeto, seguido por un “data member” (ver Capítulo 1, sección “Data

Members”, del “Lookout Developer’s Manual”) específico, si es requerido. Un *item* en un “spreadsheet” (tipo de archivo que contiene cadenas de datos), como la celda B3 en Microsoft Excel, debería ser r3c2.

Lookout puede actuar como cliente DDE, servidor DDE, y como ambos. Por lo tanto hay tres maneras básicas para enlazar Lookout hacia otras aplicaciones usando DDE.

2.2.2 Comunicación del "PLC" con "Kepserver".

KepserverEx es una aplicación de windows de 32 bits que proporciona los medios para extraer datos e información desde un amplio y numeroso rango de dispositivos (por ejemplo los PLC) y sistemas industriales dentro de “aplicaciones cliente” en una PC Windows. KepserverEx cae bajo la categoría de una aplicación Servidor. Es muy común escuchar el término “aplicación cliente /servidor” en uso a través de algunas disciplinas de software y segmentos de negocio. En el mercado industrial, usualmente significa el compartimiento de datos de producción o manufactura entre una variedad de aplicaciones

que van desde programas de interface hombre-maquina e históricos de datos.

Sin contar con el campo de los negocios, las aplicaciones cliente/servidor tienen una cosa en común: un método de estandarización para compartir datos. En el campo industrial, varias tecnologías cliente/servidor han sido desarrolladas en los últimos diez años. Inicialmente algunas de esas tecnologías fueron patentadas. En varios casos esas patentes de arquitecturas cliente/servidor fueron ampliamente usadas pero quedaron sin uso un tercio de esas aplicaciones. Tempranamente en el desarrollo de Windows, una tecnología cliente/servidor genérica llamada DDE o Intercambio Dinámico de Datos. DDE proporcionó una arquitectura básica que permitiría a varias aplicaciones de windows compartir datos, pero había un problema. DDE no fue diseñado para el mercado industrial. Este carecía mucho de la velocidad y peso deseado en un escenario industrial. Sin embargo, esto no fue inconveniente para que DDE llegue a ser una arquitectura dominante, debido a su disponibilidad en más aplicaciones de Windows. A tiempo, las variaciones en el DDE de Microsoft fueron desarrolladas por algunos de los principales vendedores

del mercado. Esas variaciones direccionaron algunas de las capacidades en velocidad y fiabilidad de DDE, pero varias personas en el medio industrial acordaron que un mejor sistema necesitaba ser desarrollado.

Con la ventaja de Sistemas Operativos de 32 bits, y el uso de Ethernet para proporcionar comunicación entre dispositivos, había una necesidad para una rápida y eficaz transferencia de datos entre los programas de las aplicaciones. Es así como OPC vio su nacimiento dentro de la industria.

Los servidores OPC (OLE para Control de Procesos) proporcionan un método de estandarización que permite múltiples aplicaciones industriales para compartir datos de una manera rápida y robusta.

El OPC proporcionado en el paquete de programas de Kepware (donde se encuentra KepserverEx) ha sido diseñado como un programa de dos-partes. El primer componente proporciona todo sobre el OPC y la conectividad DDE como una función de interface del usuario. La segunda parte comprende los “drivers” (software que permite un enlace entre dos dispositivos) de

comunicación. Este diseño de dos-partes permite al usuario agregar múltiples opciones de comunicación para sus aplicaciones SCADA mientras utilizan solo un solo servidor OPC.

Parte del diseño innovador de la tecnología Servidor OPC/DDE de Kepware es la separación del Driver del Protocolo del Hardware de la Tecnología del servidor. Esta separación permite al usuario usar uno o más drivers al mismo tiempo en el mismo servidor.

2.2.3 Comunicación "Kepserver" con "Lookout".

Lookout puede operar como un Cliente OPC para comunicarse con Kepserver, a través del objeto *OPCClient*. Podemos acceder a este objeto desde el menú principal de la siguiente manera:
Menu Object >> Create >> Drivers >> OPCClient.

En este objeto se debe de escoger primero los parámetros para el Servidor OPC (*OPC Server Settings*) tales como el nombre del Servidor instalado en la PC (*KEPware.KEPServerEx.V4* o

National Instruments.LookoutOPCServer); su localización, esto es, si se encuentra en la PC local o en un equipo remoto; la forma como aparecerán las variables del servidor en el objeto; y la forma de comunicación con el servidor OPC, que usualmente debe de ser asincrónica. Luego se escogen los parámetros del grupo OPC (*OPC Group Settings*) tales como el tiempo de actualización del servidor al cliente (*Update Rate*) que esta en milisegundos.

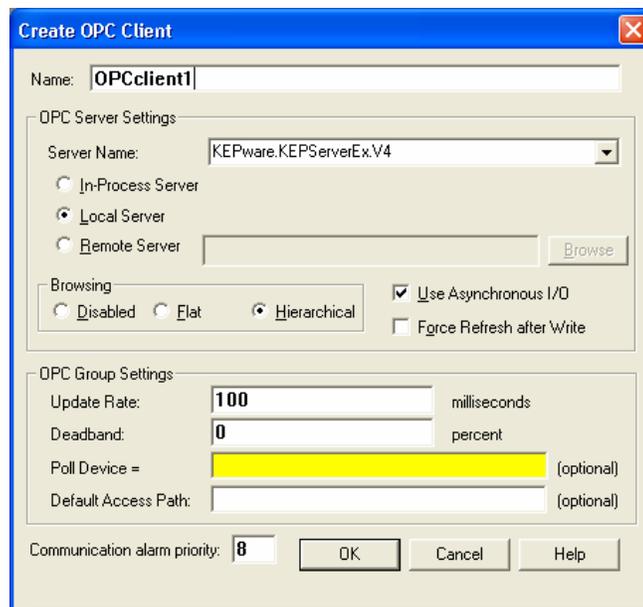


Figura 2.2. Cuadro de diálogo del objeto OPCClient.

Una forma para visualizar los valores del servidor OPC, es crear una expresión. Para insertar una expresión procedemos de la siguiente manera: *Menu Insert >> Expresión*

En la parte izquierda del cuadro de diálogo que aparecerá a continuación se verán los objetos del proceso. Para acceder a las variables del *OPCClient* se expande las carpetas hasta llegar a la que se quiere visualizar.

El siguiente paso es escoger la clase de “display” y su formato que se observará sobre el panel de control del proceso. De este modo se puede crear cualquier expresión con las variables que aparecen en la carpeta el servidor OPC.

2.3 Selección de la aplicación en lookout.

Para crear un proceso en Lookout, se selecciona en el menú *File >> New*. Entonces el cuadro de diálogo *Crear Proceso (Create Process)* aparecerá, junto con el *Explorador de Objetos (Object Explorer)* y la *ventana de alarmas* de Lookout.

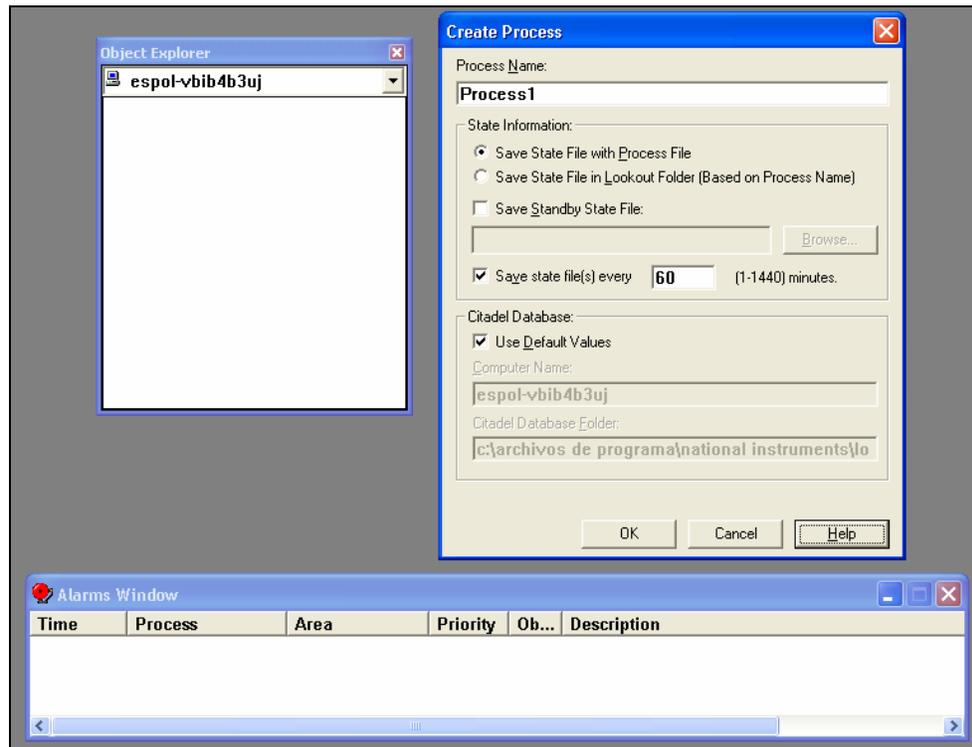


Figura 2.3. Cuadro de diálogo para crear un proceso nuevo en Lookout.

El nombre del proceso (*Process Name*) debe ser asignado respetando las mismas convenciones que tienen los nombres de los objetos en Lookout; esto es, no debe tener más de 32 caracteres de longitud y no debe de llevar espacios, para más información refiérase al Capítulo 5, sección “*Steps to Create an Object*”, del “Lookout Developer’s Manual”.

La sección Información de Estado (*State Information*), es donde se especifica el lugar donde se quiere que Lookout guarde los archivos de

estado (.l4t) para un proceso. Esta información es leída y usada cuando el proceso es abierto, y es actualizada cuando el proceso es cerrado.

La sección Base de Datos Citadel (*Citadel Database*) es donde se selecciona el lugar de la base datos Citadel (ésta es la base datos propia de Lookout) en que Lookout guarda los datos para un proceso. Se puede escoger el destino, diferente a el que está por *default*, que uno desee de la base de datos de cualquier proceso, deseleccionando la casilla "Use Default Values" y cambiando la dirección de la carpeta de la base de datos de Citadel.

Al aceptar los parámetros escogidos para el proceso el cuadro de diálogo se cerrará y aparecerá otro cuadro de diálogo para configurar el Panel de Control donde se desarrollarán los objetos escogidos para la elaboración de la pantalla para el proceso.

También se puede escoger una nueva dirección de la ruta de la base de datos Citadel en el cuadro de diálogo de Opciones de Sistema (*System Options*), cambiándole la dirección (*Default Path*) en la sección *Citadel Database*.

La aplicación desarrollada para nuestro proceso está compuesta por dos *Process*. El uno que opera a manera de cliente, llamado *Process1*, que es donde se encuentran las pantallas del proceso; y el otro que opera como servidor, llamado *Servidor*, donde se encuentra el objeto con el “driver” del PLC (*GE_Series90*).

Para acceder al objeto *GE_Series90*, se procede de la siguiente forma: *Menu Object >> Create >> Drivers >> GE_Series90*. En el cuadro de diálogo de este objeto, se debe de configurar el protocolo de comunicación entre el PLC y Lookout, la velocidad de transmisión de los datos (*Data Rate*), el puerto de comunicación a usarse, etc. Para mas detalles referirse a la ayuda de Lookout (*Menu Help >> Contents*) en la sección *Drivers*.

Los elementos del objeto del *GE_Series90* son enlazados con los objetos de *Process1* por medio de las opciones de conexión con que cuenta Lookout. Refiérase a la sección *Conectando Objetos* del Capítulo 5 del “Lookout Developer’s Manual”.

2.3.1 Configuración de las pantallas.

Para poder diseñar el ambiente de las pantallas, Lookout cuenta con una librería de gráficos, los cuales pueden ser llamados de la siguiente forma: *Menu Insert >> Grafics*.

Estos gráficos residen en la carpeta *Graphics*, la cual se encuentra en la carpeta *Lookout 5.0*, en el directorio donde se instaló el programa. Pero, también se pueden insertar gráficos en Lookout, creados desde otros programas de edición de imágenes. Para usarlos dentro de una aplicación deben ser guardados en la carpeta *Graphics* bajo el formato de mapa de bits (.bmp) o archivos Meta File (.wmf) para que el programa los reconozca.

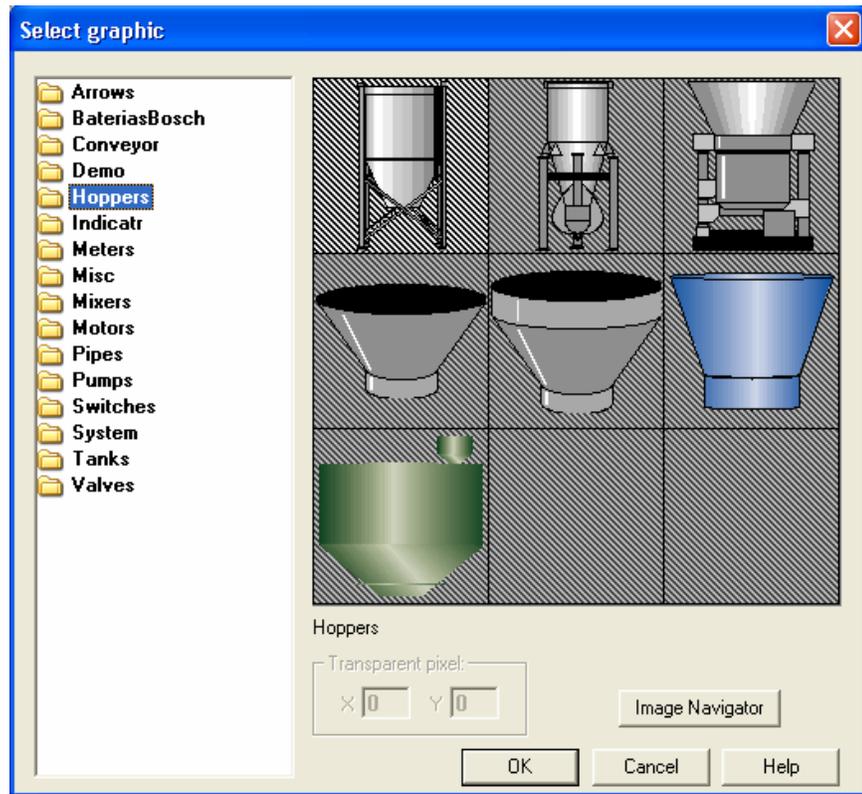


Figura 2.4. Cuadro de diálogo para insertar un gráfico en Lookout.

Además cuenta con los gráficos del *Image Navigator* en el cual existen imágenes de la mayoría de los procesos industriales. Se puede acceder a éste haciendo un click en el botón “Image Navigator” del cuadro de dialogo mencionada anteriormente.

Lookout cuenta con *objetos* para la animación de los procesos representados en las pantallas. Para acceder a cualquiera de

estos objetos realizamos los siguientes pasos: *Menu Object >> Create*; o también dando “click derecho”, dentro del *Object Explorer*, sobre el elemento *Proceso* donde se aplicará dicho objeto, y escogiendo la opción *New Object*.

Para trabajar con el *Object Explorer* tenemos que abrirlo del siguiente modo: *Menu Object >> Object Explorer*. Una vez realizado cualquiera de los pasos anteriores, aparecerá un cuadro de diálogo en el que se mostraran todos los objetos (control, calculation, drivers, etc.) con que cuenta Lookout.

Además, se pueden insertar *expresiones* (matemáticas, lógicas, de texto) que son mostradas en la pantalla o usadas en los objetos del proceso. Para acceder a éstas se procede de la siguiente forma: *Menu Insert >> Expresión*.

También se pueden insertar cuadros de texto a manera de etiquetas para nombrar a partes del proceso por ejemplo. Para insertar un cuadro de texto se procede de la siguiente forma: *Menu Insert >> Text/Plate/Inset*.

Para mas detalles acerca de la creación de objetos en Lookout, referirse al Capítulo 5, *Creating Lookout Processes*, del “*Lookout Developer’s Manual*”.

2.3.2 Edición de las pantallas.

2.3.2.1 Pantalla de la máquina empastadora.

Dentro de nuestro diseño y basándonos en lo que físicamente dispone la empresa, hemos considerado para el desarrollo de esta pantalla la banda transportadora (Banda 1) que lleva las rejillas de plomo de manera vertical, el “Grid Picker” que recoge y coloca las rejillas de manera horizontal en la siguiente banda transportadora (Banda 2), la tolva donde se encuentra la pasta que recubrirá a las rejillas, así como visualizadores que nos indican el estado de los motores MB1 y MB2 de las bandas transportadoras 1 y 2 respectivamente. El motor del “Grid Picker” (MGP) es de velocidad variable y es manejado por un variador de velocidad, cuyo panel de configuración de parámetros es mostrado en esta pantalla.

Para comenzar la edición de una pantalla (panel de control) en Lookout, se debe de trabajar en el Modo Edición (*Edit Mode*). Para habilitar este modo podemos ir al *Menu Edit >> Edit Mode*, o usando el teclado con la combinación *Ctrl + space*.

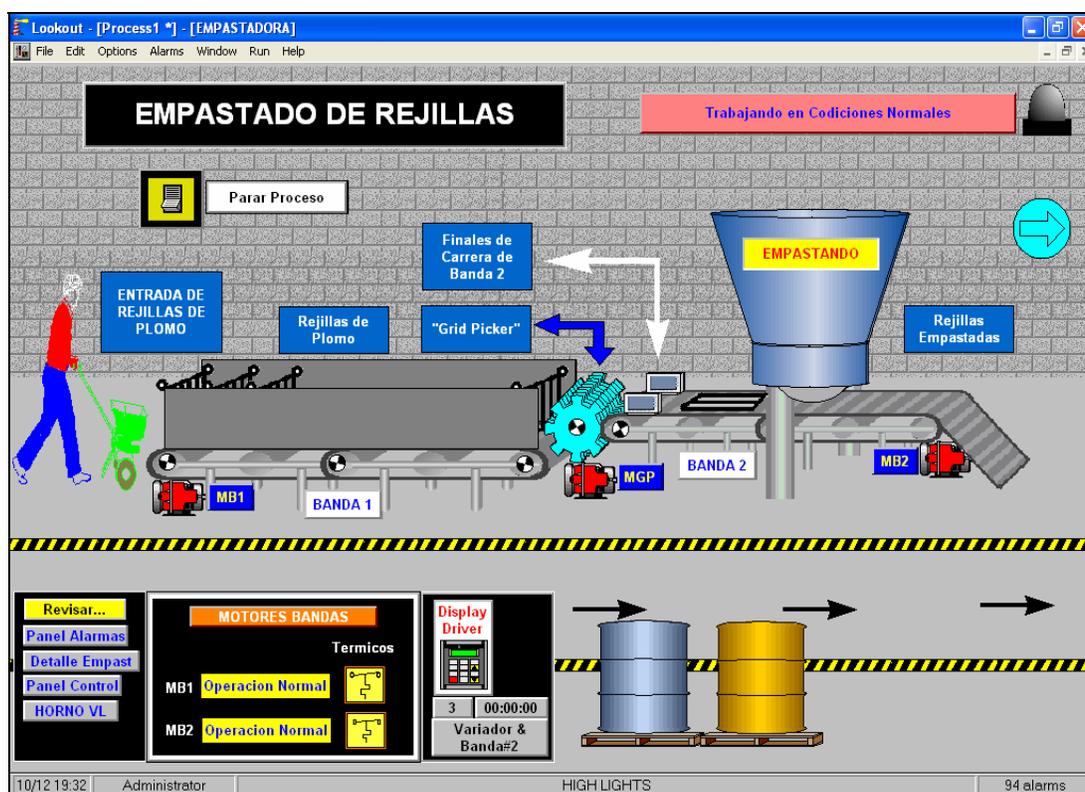


Figura 2.5. Vista de la pantalla de la Máquina Empastadora y sus componentes.

La animación de esta pantalla esta comandada por el objeto "SeqMain", que es un objeto tipo "Sequencer" que se encuentra en la carpeta *Control* de los objetos de Lookout. Para acceder a este objeto lo realizamos de la siguiente forma:

Menu Object >> Create >> Control >> Sequencer, o también dando "click derecho", dentro del *Object Explorer* (Explorador de Objetos), sobre el elemento *Proceso* donde se aplicara dicho objeto, y se procederá del siguiente modo para escoger el objeto: *New Object >> Control >> Sequencer*.

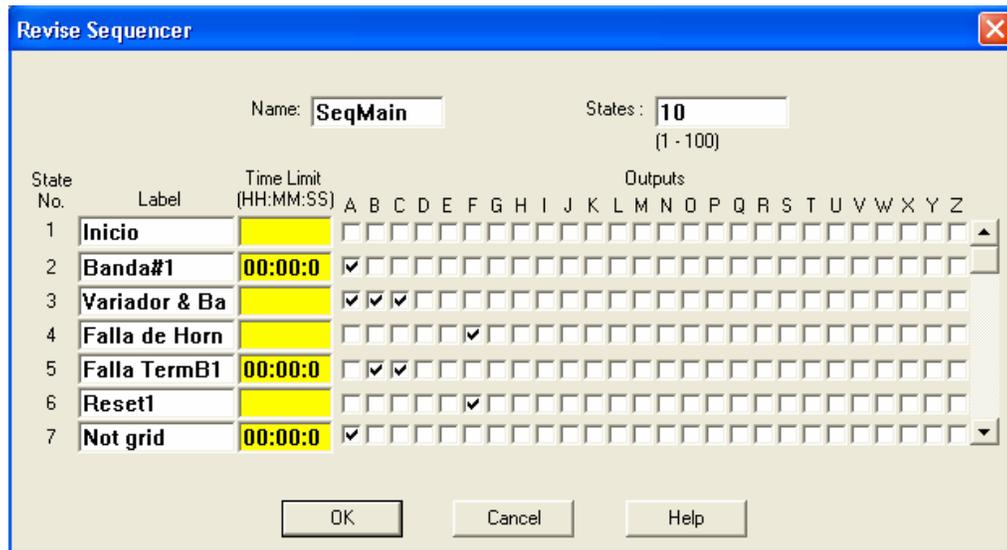


Figura 2.6. Objeto *SeqMain* (*Sequencer*), que maneja la animación de la pantalla de la máquina empastadora.

Este objeto, genera una secuencia de estados (que pueden ser hasta un número no mayor a 100), los cuales están asociados a las salidas propias de este objeto (el número de salidas disponibles es de 26). En el campo “Time Limit” se programará, el tiempo en que permanecerá en el estado actual hasta pasar al siguiente estado, o una condición para que cambie de estado al momento de cumplirse ésta.

Para programar dichas condiciones, lo hacemos a través de la herramienta “*Edit Conexion*”, a la cual podemos acceder haciendo click derecho sobre el objeto *SeqMain* en el *Object Explorer* y escogiendo dicha opción.

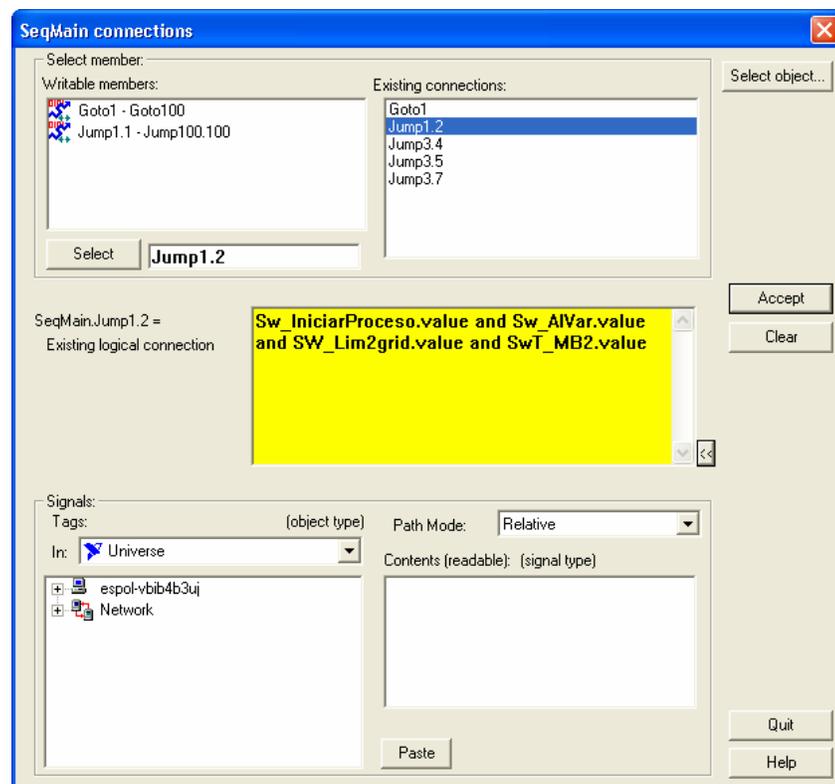


Figura 2.7. Herramienta *Edit Conexion*, para programar los saltos de estado del *Sequencer: SeqMain*.

La programación para el secuenciador *SeqMain* es la siguiente:

SeqMain.Jump1.2 = Sw_IniciarProceso.value and
Sw_AIVar.value and Sw_Lim2grid.value and
SwT_MB2.value

SeqMain.Jump3.4 = \\.\Servidor\GE1.M1 and
not(\\.\Servidor\GE1.Q4)

SeqMain.Jump3.5 = not(SwT_MB1.value)

SeqMain.Jump3.7 = not(Sw_Limtnogrid.value)

SeqMain.Goto1 = not(Sw_IniciarProceso.value and
Sw_AIVar.value and Sw_Lim2grid.value and
SwT_MB2.value) or SeqMain.F

Donde las variables representan lo siguiente:

Sw_IniciarProceso: es un objeto “switch” que me permite iniciar el proceso en la máquina empastadora.

Sw_AIVar: es un objeto “switch” que representa cualquier falla en el variador de velocidad del motor del “Grid Picker”.

Sw_Lim2grid: es un objeto “switch” que representa el final de carrera que detecta 2 rejillas en la banda transportadora 2.

Sw_Limtnogrid: es un objeto “switch” que representa el final de carrera que detecta la no presencia de rejillas en la banda transportadora 2.

SwT_MB1: es un objeto “switch” que representa una falla térmica en el motor de la banda transportadora 1.

SwT_MB2: es un objeto “switch” que representa una falla térmica en el motor de la banda transportadora 2.

Los comandos usados son: el JUMP que es un salto de estado incondicional y el GOTO que es un salto condicional. La programación se la hace de manera lógica; y las instrucciones se ejecutan de derecha a izquierda, es decir, debe cumplirse la condición de lado derecho del signo igual para que se cumpla la instrucción que está al lado izquierdo.

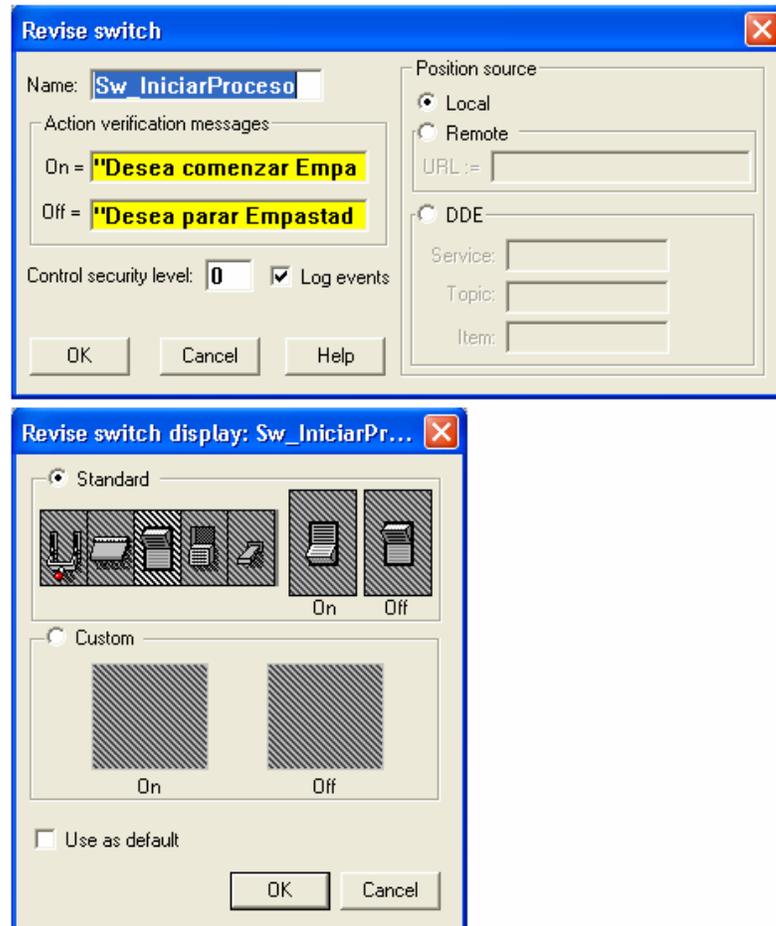


Figura 2.8. Cuadro de diálogo del Objeto *Switch* y su representación gráfica dentro del panel de control.

El proceso empieza al presionar el switch: *Sw_IniciarProceso*, entonces se enciende el motor de la banda transportadora 1 (MB1), para llevar las rejillas de plomo hacia la otra banda. Luego de 5 segundos se encienden los motores del Grid Picker (MGP) y de la

banda transportadora 2 (MB2), para llevar las rejillas hasta la tolva de la máquina empastadora, para que sean empastadas por ésta.

Para simular el movimiento de las rejillas se usa el objeto *Animator*, al cual se accede de la siguiente forma: *Menu Object >> Create >> Display >> Animador*.

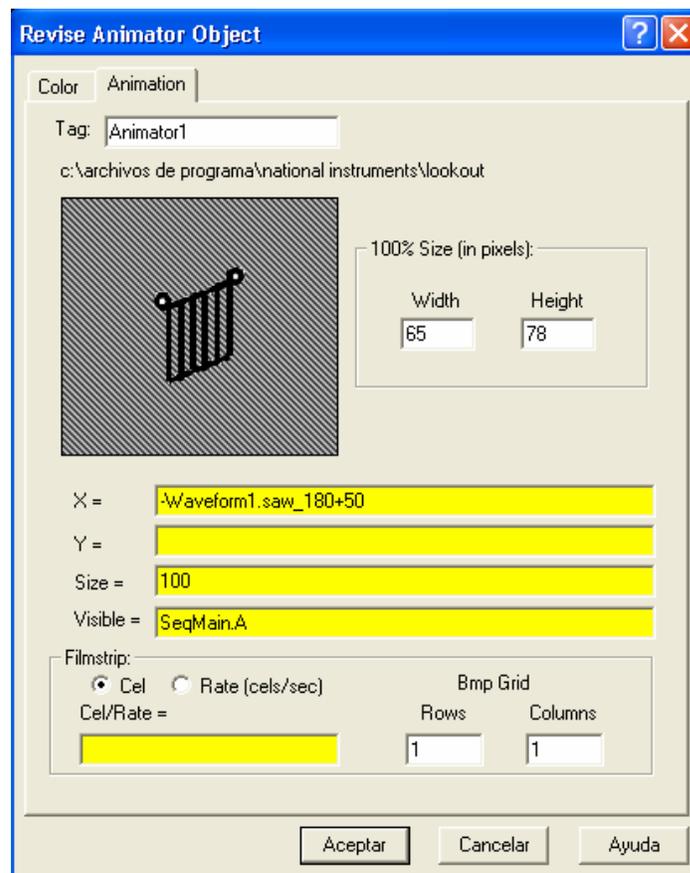


Figura 2.9. Cuadro de diálogo del objeto *Animator*.

Este objeto muestra una imagen, en este caso la rejilla de plomo, la cual es desplazada de forma horizontal gracias a una expresión ($x = -\text{waveform1.saw_180} + 50$), con su tamaño original ($\text{Size} = 100$) y siendo visible solo cuando se cumpla una condición ($\text{Visible} = \text{SeqMain.A}$).

Para crear la expresión del *Animator* se ha hecho uso del elemento *waveform1.saw_180* del objeto *Waveform*. Para acceder a este objeto se procede de la siguiente forma: *Menu Object >> Create >> Calculation >> Waveform*. Este objeto genera señales sinusoidales, cosinusoidales, cuadradas, dientes de sierra, triangulares y aleatorias. Para más detalles de este objeto refiérase a la sección de *Lookout Objects* que se encuentra en la ayuda de Lookout (*Lookout Help*).

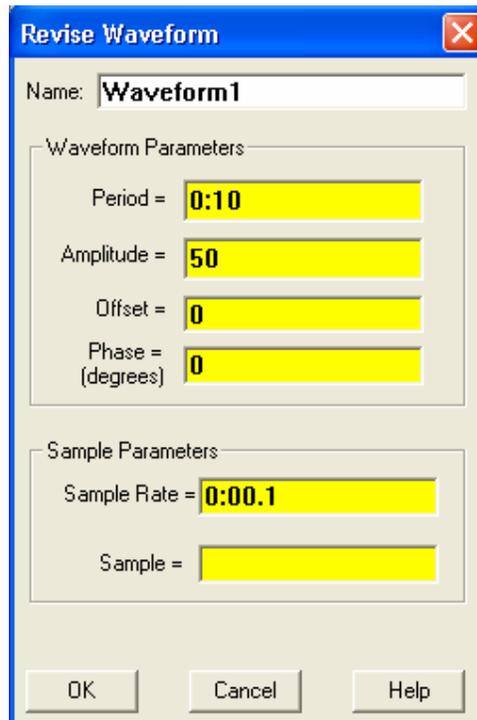


Figura 2.10. Cuadro de diálogo del objeto *Waveform*.

Existen dos finales de carrera en la banda transportadora 2. El primero es para detectar la no presencia de rejillas en la banda y el segundo es para determinar si hay dos rejillas pegadas la una con la otra. Estos trabajan a modo de “microswitch” y envían una señal de alarma si se accionan. Están representados por objetos *switch* normalmente cerrados.

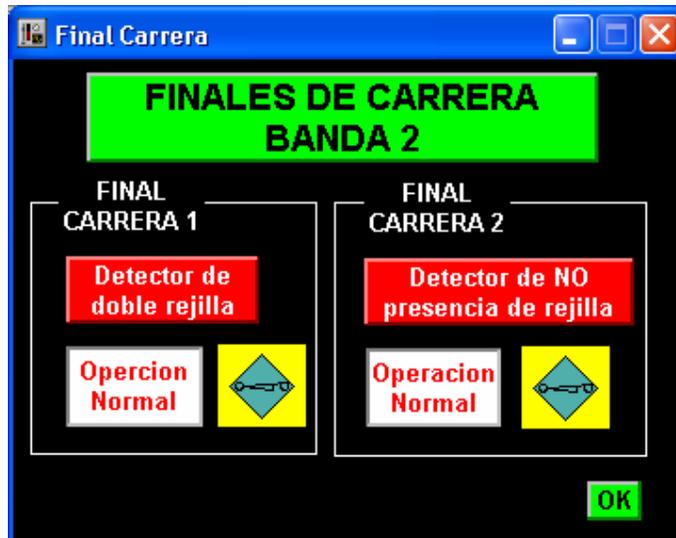


Figura 2.11. Pantalla que representa los finales de carrera en la banda transportadora 2.

El proceso se detiene si existe alguna falla en los elementos de protección de los motores MB1 y MB2, o en el variador de velocidad que controla al motor MGP. Los relés térmicos de los motores son representados y simulados con elementos *switch*.

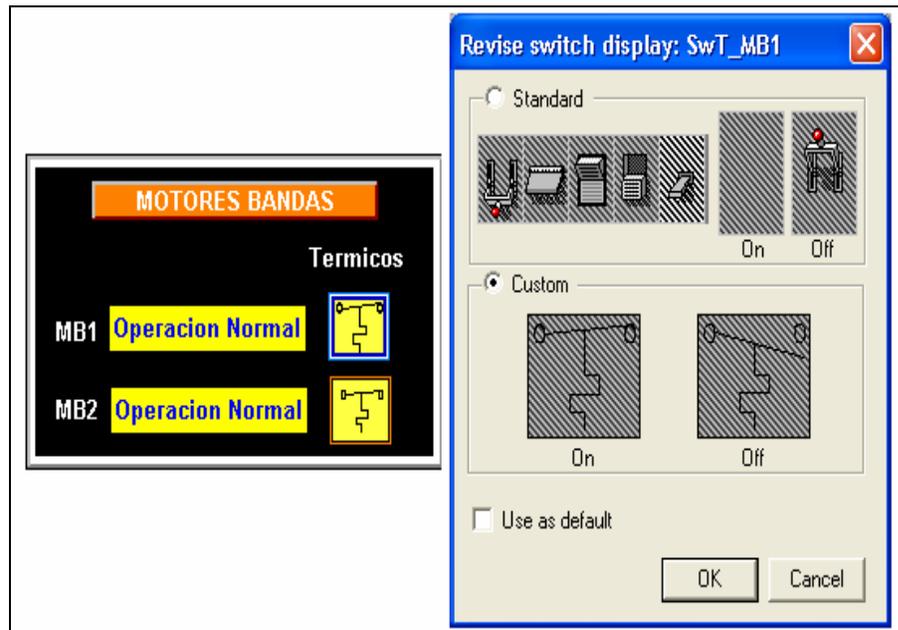


Figura 2.12. Objetos *switch* que representan los relés térmicos de los motores MB1 y MB2.

También puede detenerse por alguna falla en el variador de velocidad (Driver), el cual tiene una ventana que representa su panel de operación.

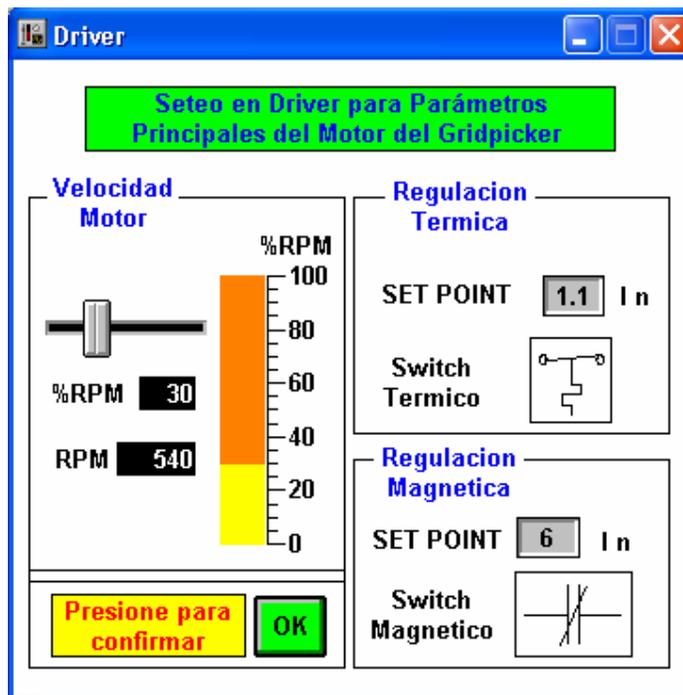


Figura 2.13. Ventana que representa el panel de operación del variador de velocidad con algunos de sus parámetros.

Al ocurrir alguna de estas fallas el proceso vuelve a su estado inicial, que es el estado de reposo, hasta que se solucione el problema y se reinicie el proceso.

Para solucionar el problema se debe de observar donde ocurrió la falla en la Pantalla de Alarmas del proceso, la cual se tratara más adelante.

2.3.2.2 Pantalla del horno.

El horno de secado es el siguiente lugar al que van las rejillas luego de ser empastadas. La pantalla para este proceso consta de una banda transportadora la cual es accionada por el motor MBH; y el horno que tiene su sistema de suministro de combustible, su sistema de ventilación y su sistema de control. El sistema de suministro de combustible es controlado por válvulas solenoides y por un “damper” que regulan el flujo de combustible hacia el horno.

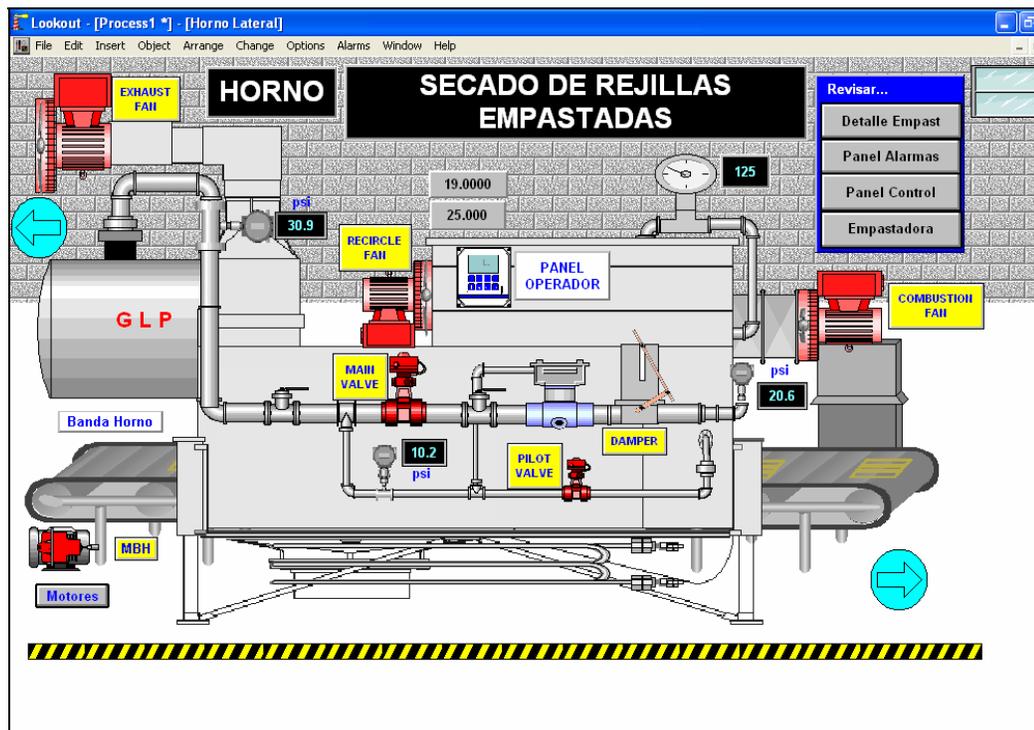


Figura 2. 14. Pantalla del horno de secado de las rejillas de plomo.

El horno tiene un panel de operador desde donde se puede encender éste y comenzar con este proceso de secado. Además en este panel se puede observar la temperatura y la presión internas del horno, así como indicadores de los motores o válvulas encendidas al momento.



Figura 2. 15. Pantalla del panel de operador del horno.

El movimiento de las rejillas en la banda es similar al de las bandas en la pantalla de la máquina empastadora, usando un objeto *Animator* para simular su desplazamiento horizontal.

La animación del motor de la banda, está implementada por medio de una expresión con *display*

gráfico para su visualización. La expresión que insertamos y utilizamos es la siguiente: `\\.\Servidor\GE1.Q4`. Esta condición se cumple si la salida digital %Q4 del servidor (PLC) está activada.

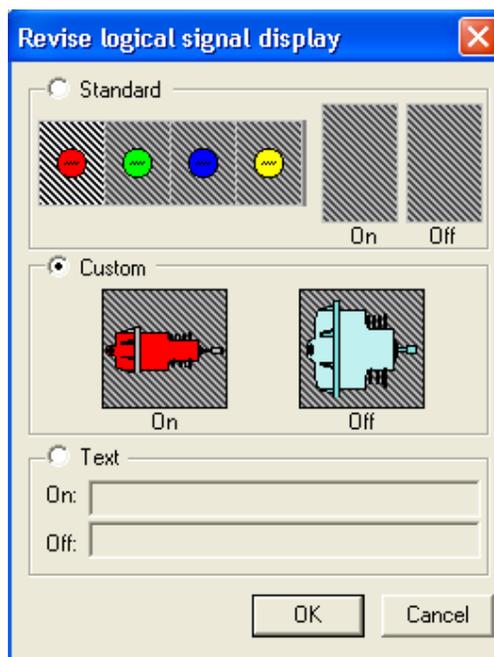


Figura 2. 16. *Display* gráfico para la expresión que activa el motor de la banda transportadora del horno.

Los motores de los ventiladores (Exhaust Fan, Recircle Fan, Combustión Fan) y las válvulas solenoides (Pilot Valve, Main Valve) son animados por medio del objeto *Multistate*. Para acceder a este objeto procedemos del

siguiente modo: *Menu Object >> Create >> Display >> Multistate.*

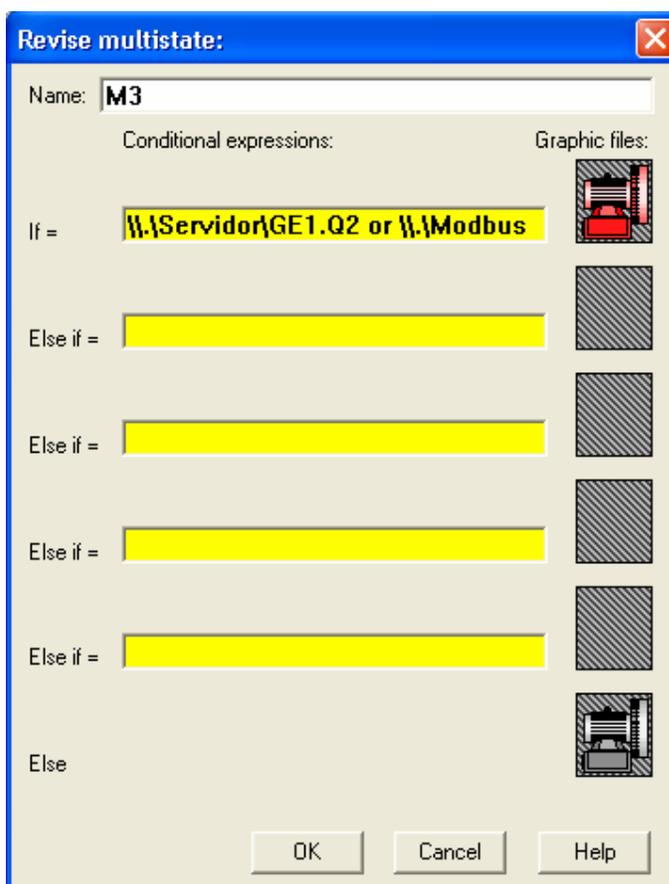


Figura 2. 17. Cuadro de diálogo del objeto *Multistate*.

Este objeto muestra un gráfico, que corresponde a la expresión que sea verdadera en ese momento. Se puede establecer en este objeto 5 posibles condiciones (expresiones), para las cuales tienen asociado un

gráfico correspondiente y un gráfico más en caso de que ninguna condición se cumpla. La condición que tiene prioridad es la primera.

Para los motores de los ventiladores y válvulas solenoides se tiene las siguientes condiciones:

	Condición
Exhaust Fan	\\.\Servidor\GE1.Q2
Combustión Fan	\\.\Servidor\GE1.Q3
Recircle Fan	\\.\Servidor\GE1.Q2
Pilot Valve	\\.\Servidor\GE1.Q6
Main Valve	\\.\Servidor\GE1.Q7

La secuencia de animación para los objetos de esta pantalla obedece a la secuencia lógica del programa que reside en el PLC (ver Anexo A). Las salidas digitales (%Q2, %Q3, %Q4, %Q5, %Q6, %Q7) del PLC se encuentran enlazadas con algunos de los objetos de esta pantalla, de la siguiente forma:

%Q2 activa a los motores de los ventiladores Exhaust Fan y Recircle Fan.

%Q3 activa el motor del ventilador Combustión Fan.

%Q4 activa el motor de la banda transportadora del horno.

%Q6 activa la válvula solenoide Pilot Valve.

%Q7 activa la válvula solenoide Main Valve.

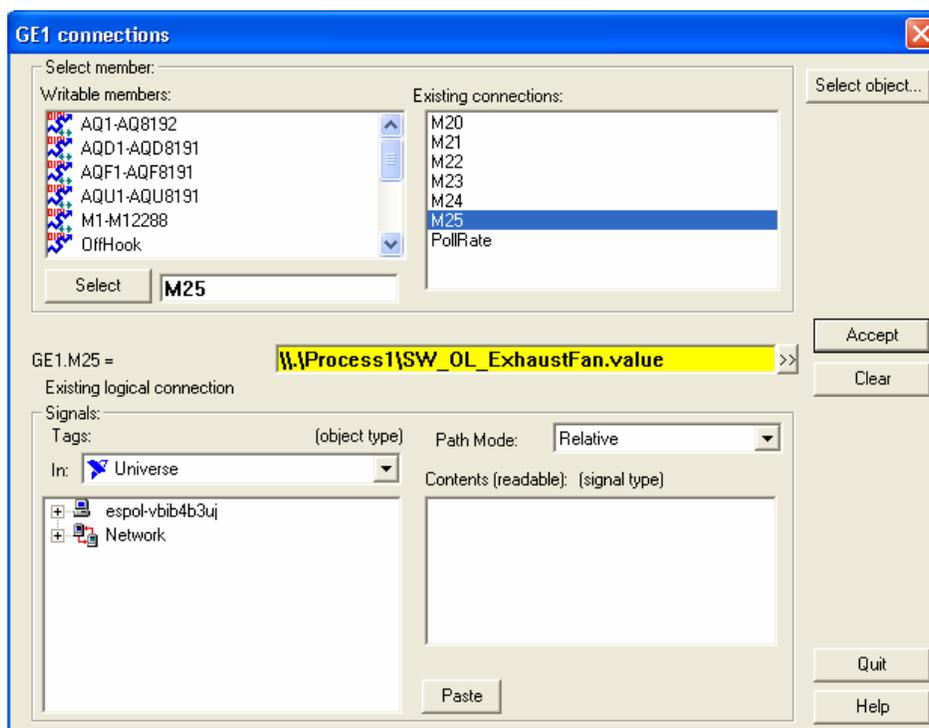


Figura 2. 18. Cuadro de diálogo, para la conexión entre los contactos auxiliares del PLC (objeto GE1) y los objetos de la pantalla del horno.

Los contactos auxiliares (%M) usados en el programa del PLC trabajan a manera de entradas digitales auxiliares y las expresiones que las enlazan a los objetos de la pantalla del horno son las siguientes:

GE1.M20 = \\.\Process1\Exp8

GE1.M21 = \\.\Process1\Pb_HornoON.value

GE1.M22=\\.\Process1\SW_OLMotorbandaHorno.value

GE1.M23 = \\.\Process1\SW_OL_CombustionFan.value

GE1.M24 = \\.\Process1\SW_OL_RecircleFan.value

GE1.M25 = \\.\Process1\SW_OL_ExhaustFan.value

El proceso en el horno comienza al pulsar el objeto Pushbutton: *Pb_HornoOn* en la pantalla del panel operador. Para acceder a este objeto se procede de la siguiente forma: *Menu Object >> Create >> Control >> Pushbutton*. Luego se accionan los demás objetos dependiendo de los valores de las señales de las salidas y entradas del PLC.

El proceso de esta pantalla se detiene si se ocasiona una falla térmica en cualquiera de los motores que

manejan la banda del horno o los ventiladores. Los dispositivos térmicos de los motores están representados por objetos *switch*.

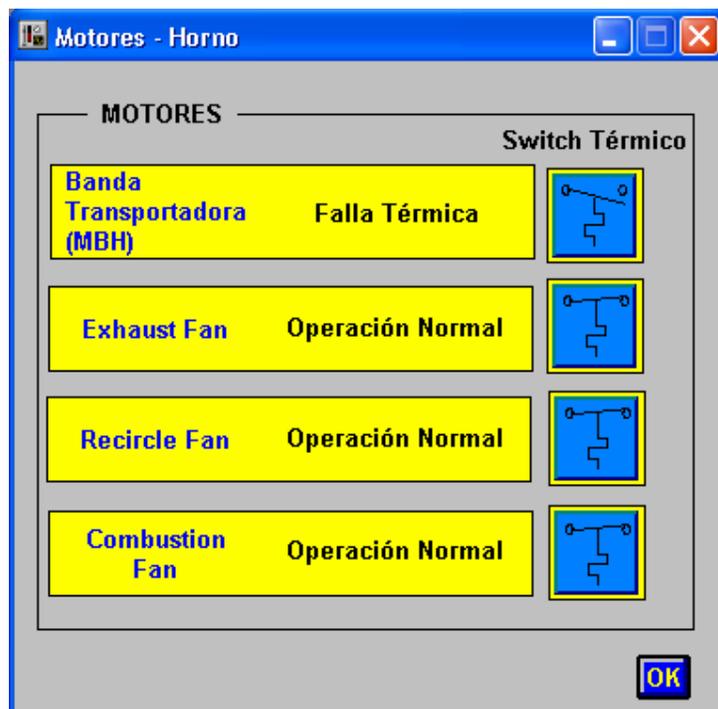


Figura 2. 19. Pantalla que muestra los dispositivos térmicos y el estado de operación de los motores en la pantalla del horno.

También el paso de las rejillas empastadas hacia el horno, o el motor de la banda transportadora, se detiene si la temperatura del horno es menor al 80% de la temperatura establecida dentro de los parámetros PID (ver mas adelante Pantallas de Control) para el control de temperatura. Esto se lo hace con la expresión:

`Pot_T_Horno.value>0.8*Pot_SP_Horno.value.`

Donde `Pot_T_Horno` y `Pot_SP_Horno` son objetos *Pot*, que representan a un potenciómetro. Para insertar este objeto procedemos del siguiente modo: *Menu Object >> Create >> Control >> Pot.*

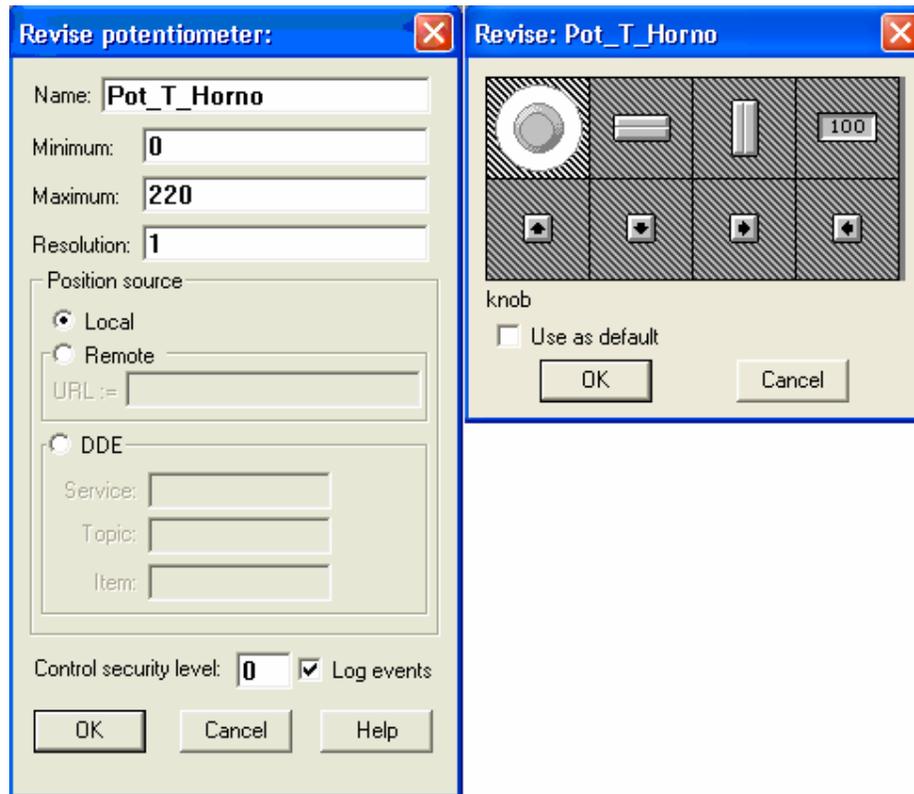


Figura 2. 20. Cuadro de diálogo del objeto *Pot* y su representación gráfica.

Al cumplirse la condición de esta expresión se activa %M20, y ésta hace que la salida %Q4, que está enlazada al motor de la banda transportadora del horno, se desactive.

2.3.2.3 Pantalla de recepción de rejillas.

Las rejillas de plomo luego de ser secadas en el horno son llevadas hacia un cuarto de recepción de rejillas terminadas, por medio de una banda transportadora. También se lleva un control estadístico del número de rejillas y de su peso.



Figura 2. 21. Pantalla del cuarto de recepción de rejillas de plomo terminadas.

Para el movimiento de las rejillas se hizo uso de objetos *Animator* que simulan el desplazamiento de las mismas en la banda transportadora. Estos objetos están configurados de tal manera que solo se hacen visibles si se cumple la condición en su campo *visible* de su cuadro de diálogo.

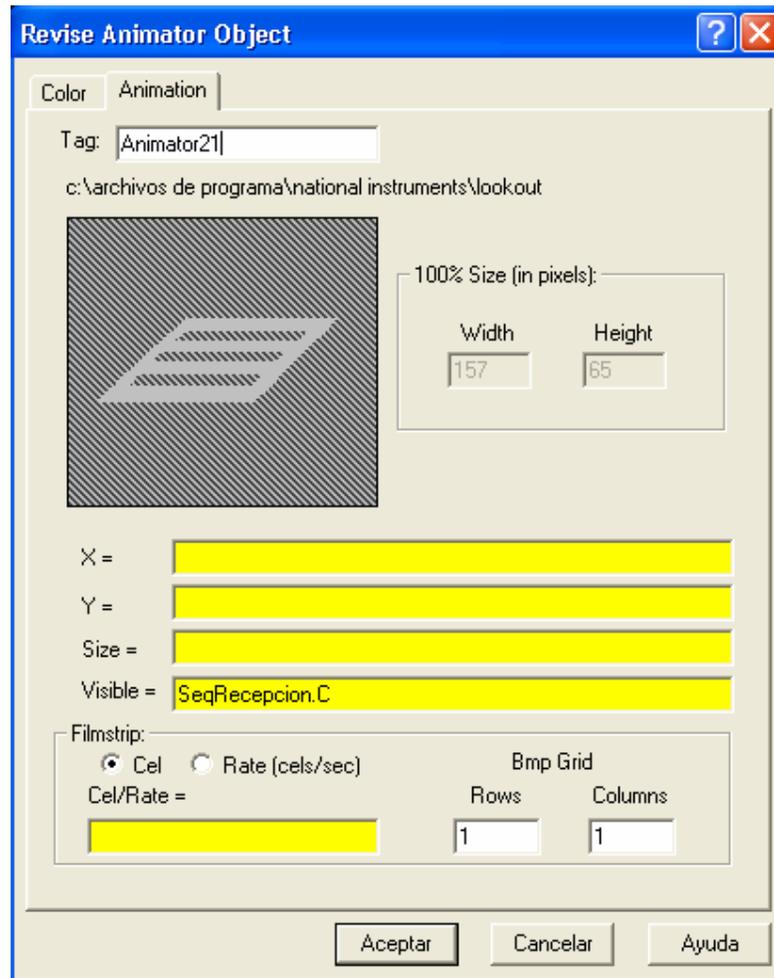


Figura 2. 22. Cuadro de diálogo del objeto *Animator: Animator21* de la pantalla de recepción de rejillas terminadas.

Las condiciones para que sean visibles las rejillas (objetos *Animator*) en la banda transportadora, provienen del objeto *Sequencer: SeqRecepcion*. Estas condiciones corresponden a las salidas del objeto *SeqRecepcion*.

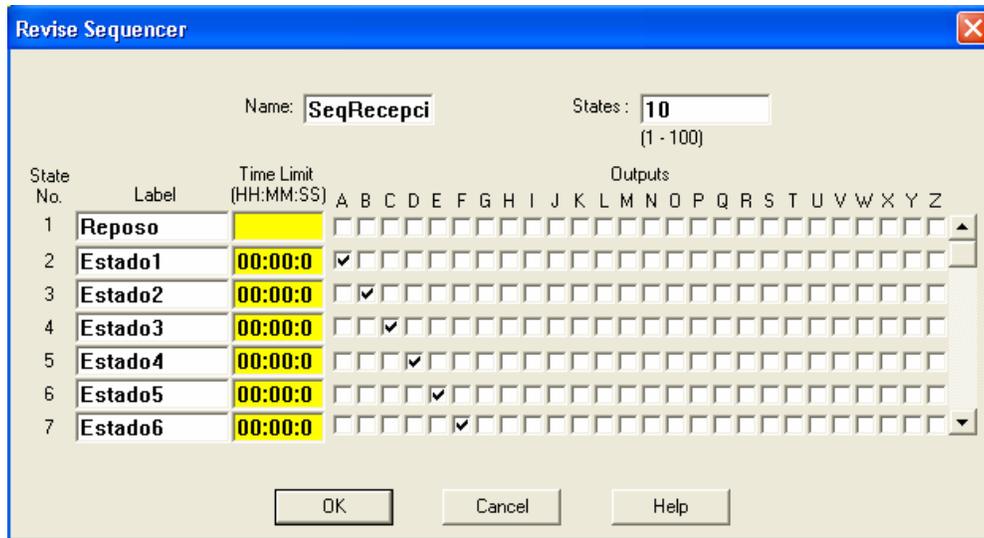


Figura 2. 23. Cuadro de diálogo del objeto *Sequencer: SeqRecepcion*.

La programación para este *Sequencer* es la siguiente:

`SeqRecepcion.Goto1 = not(\\.\Servidor\GE1.Q4)`

`SeqRecepcion.Jump1.2 = \\.\Servidor\GE1.Q4`

`SeqRecepcion.Jump10.1 = SeqRecepcion.I`

Al momento de que las rejillas pasan por una balanza electrónica, se guarda un reporte del peso de éstas y del número de rejillas terminadas. Esto se lo hace por medio del objeto *Spreadsheet*, al cual podemos acceder de la siguiente forma: *Menu Object >> Create >> Logging >> Spreadsheet*. Este objeto guarda en el

disco duro de la PC donde se ejecuta la aplicación en Lookout, el valor de los datos que se escogen en el cuadro de diálogo de este objeto, cuando el valor de un dato cambia o un evento ocurre. Los datos son almacenados en un archivo en Excel el cual tiene formato de texto separado por comas.

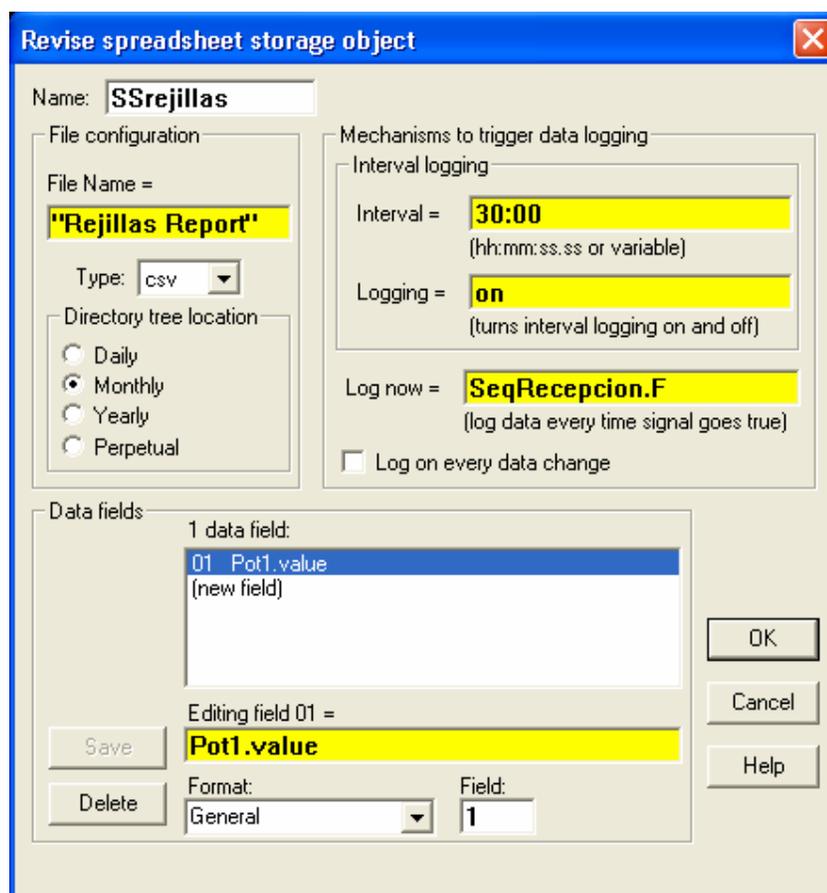


Figura 2. 24. Cuadro de diálogo del objeto *Spreadsheet: SSrejillas*.

El *Spreadsheet: SSrejillas* guarda el peso de las rejillas, el cual está representado por el potenciómetro (objeto *Pot*) *Pot1*. El nombre con que se guarda el archivo es “Rejillas Report” y lo hace en un periodo de cada 30 minutos, y cuando ocurra el evento “*SeqRecepcion.F*”, es decir, cuando se active la salida “F” del objeto *SeqRecepcion*.

El número de rejillas es almacenado con la ayuda del *Spreadsheet: SSnum_rejillas*. El número es incrementado y guardado en un archivo de nombre “Rejillas en el día”, cada vez que se active la salida “F” del objeto *SeqRecepcion*. El conteo se lo hace a través del objeto *Accumulator*, al cual podemos acceder de la siguiente forma: *Menu Object >> Create >> Calculation >> Accumulator*. El intervalo con que se guardan los datos de 30 minutos. Para más información acerca del *Spreadsheet* y *Accumulator*, referirse a la sección *Lookout Objects* de la ayuda de Lookout.

Los archivos generados y guardados son almacenados por Lookout dentro de subdirectorios, especificados por la base de datos Citadel en el cuadro de diálogo de Opciones de Sistema (*System Options*).

2.3.2.4 Pantalla de la tolva de la máquina empastadora.

La tolva de la máquina empastadora es donde se realiza la mezcla para formar la pasta que va a recubrir a las rejillas. Cuenta con aditamentos internos a manera de batidoras para la mezcla uniforme de los componentes de la mezcla.

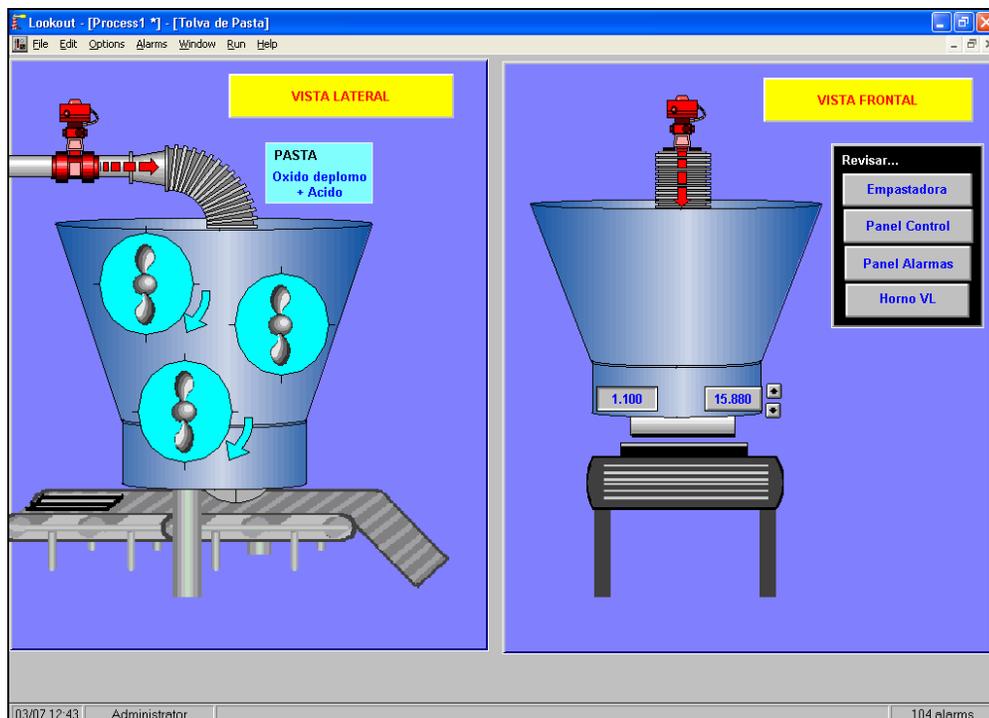


Figura 2. 25. Pantalla de la tolva de la máquina empastadora.

El movimiento de las batidoras es realizado por un objeto *Animator*, para el cual se selecciona en la sección *Filmstrip*, de su cuadro de diálogo, “*Rate (cels/sec)*”. El cual divide al gráfico del objeto en partes iguales (como filas y columnas). Esto hace que las partes del grafico se muestren seguidas, dando la impresión de que se mueven en un mismo sitio. La expresión en “*Cell Rate*” determina el momento en que empieza la animación y la rapidez con que ésta es mostrada.

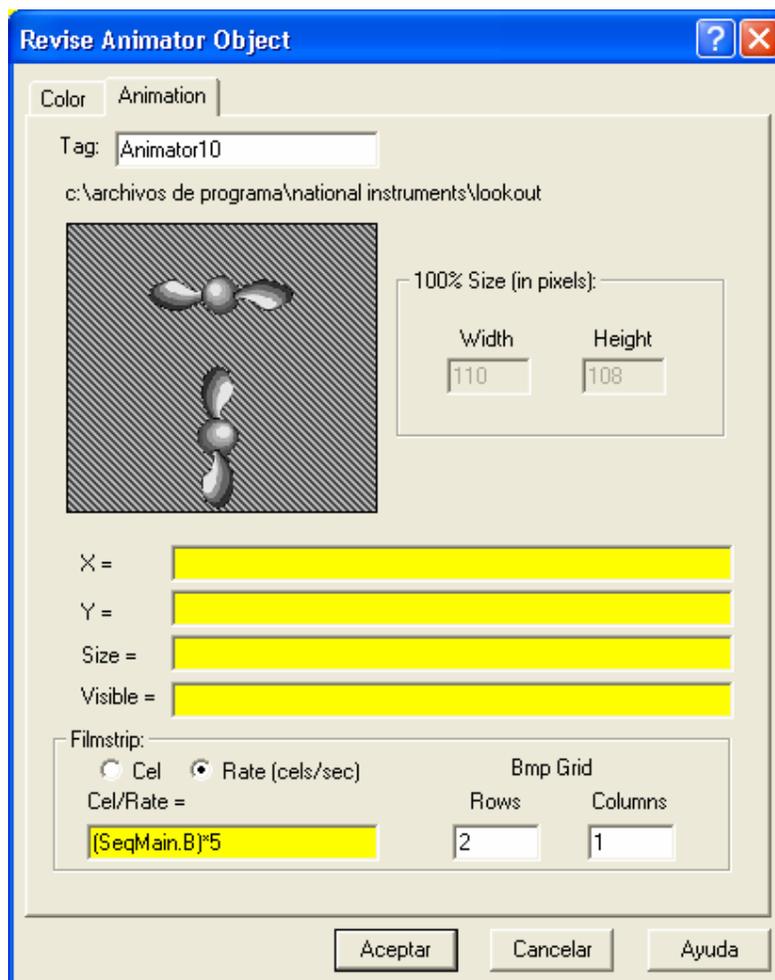


Figura 2. 26. Cuadro de dialogo del objeto *Animator*, para el movimiento de las batidoras de la tolva.

En la parte inferior de la tolva existe un rodillo que es con el cual se pega la pasta a la rejilla. Para implementar este rodillo se hace uso de un objeto *Animator*. Este rodillo tiene un movimiento vertical según el espesor de la rejilla a empastar y es

controlado por la salida de un objeto *PID*. Para acceder a este objeto se procede del siguiente modo: *Menu Object >> Create >> Control >> PID*.

The screenshot shows the 'Revise PID Controller' dialog box with the following configuration:

- Name: PID1
- Type: Position
- Process Variable: $[0.06 * (Pot1)] + 4$
- Setpoint: Recipe2.EspesorRej (Min: 8, Max: 20)
- Manual Output: Pot_Man_Emp (Min: 1, Max: 20)
- Sample Pulse: 0
- Reset: Pot_Res_Em
- Gain: Pot_Gain_Em
- Rate: Pot_Rate_Em
- Automatic Enable: Sw_IniciarPro
- Output Time: 0:30:00
- Low Limit Enable: Off
- High Limit Enable: Off
- Add proportional increment:
- Freeze Enable:

Figura 2. 27. Cuadro de diálogo de objeto *PID*.

Para la configuración de este objeto se debe de escribir una expresión para la variable de proceso (*Process Variable*), el valor requerido de la variable (*Setpoint*), una salida manual (*Manual Output*), la constante proporcional (*Gain*), la constante integral (*Reset*), la constante derivativa (*Rate*) y un habilitador (*Automatic*

Enable) para la operación del objeto en modo automático. Los valores para estas expresiones son explicados en las siguientes pantallas.

2.3.2.5 Pantallas de control y de alarmas.

Las siguientes pantallas muestran los valores de las variables del proceso, así como también para controlar el proceso completo.

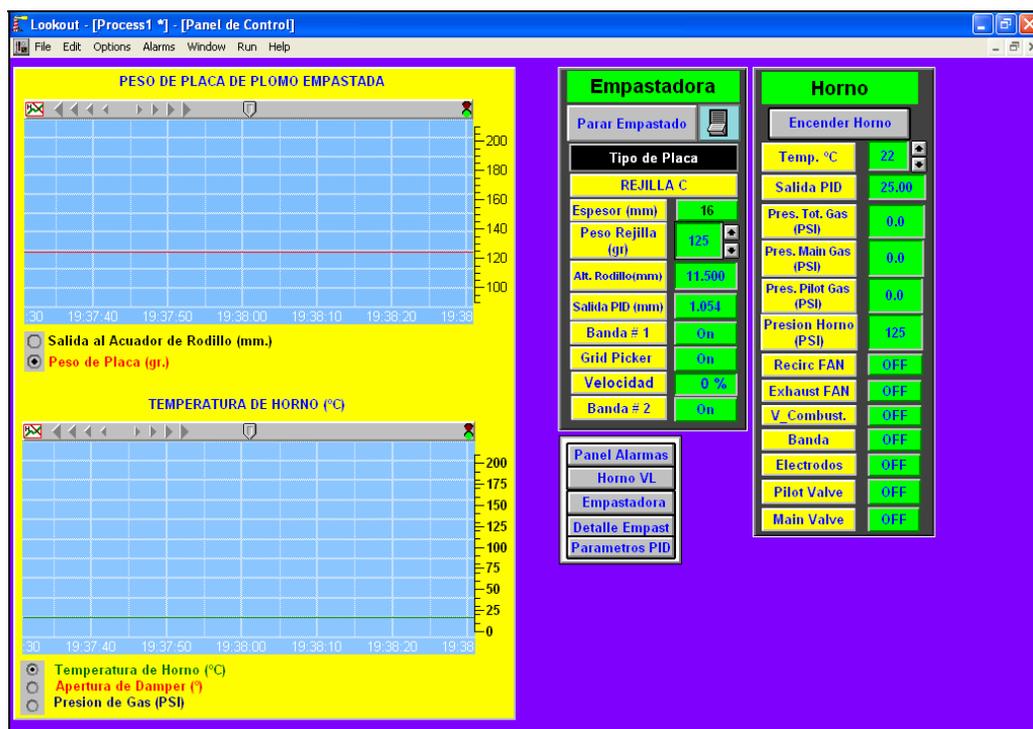


Figura 2. 28. Pantalla del panel de control del proceso.

En esta pantalla se encuentran dos graficadores para las señales de la salida del actuador del rodillo en milímetros, el peso de la rejilla en gramos, la temperatura del horno en °C, la apertura del “damper” en grados y la presión del gas en psi. Estos graficadores son objetos *Hyper Trend*, a los cuales podemos acceder del siguiente modo: *Menu Object >> Create >> Logging >> Hyper Trend*.

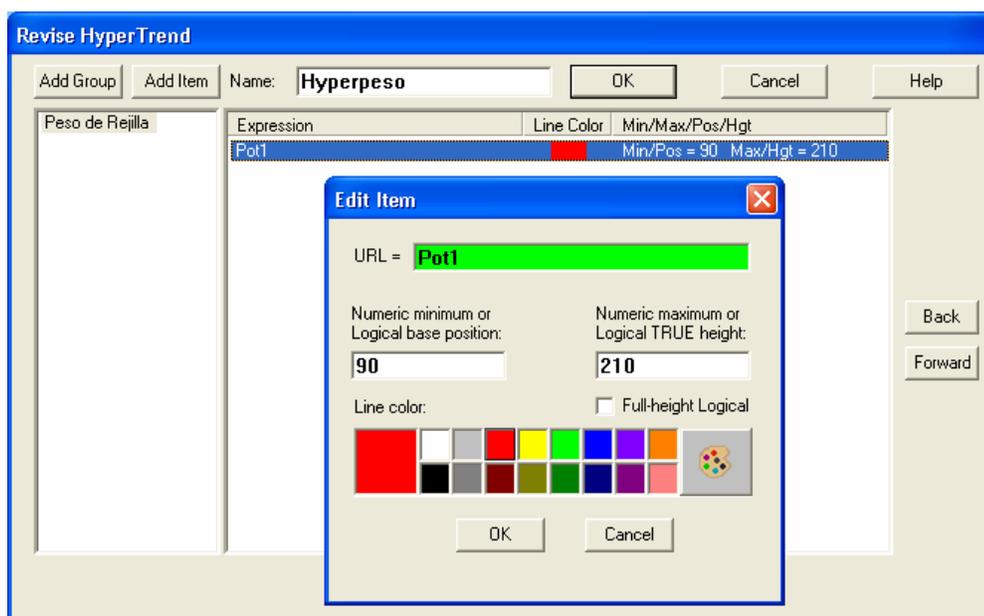


Figura 2. 29. Cuadro de dialogo del objeto *HyperTrend*.

Este objeto grafica los valores de los datos escogidos (*Item*) en tiempo real, desplegando una señal continua. Estos datos pueden ser lógicos o numéricos. Se pueden graficar varios grupos de datos en un mismo objeto *Hyper Trend* agregando grupos (*Add Group*). Para escoger un nuevo dato se agrega un *Item* (*Add Item*) y a continuación aparecerá un cuadro de diálogo para escoger uno de los objetos del proceso. El grafico es presentado en un panel en el cual muestra la hora del proceso y la traza del dato o datos a graficar.

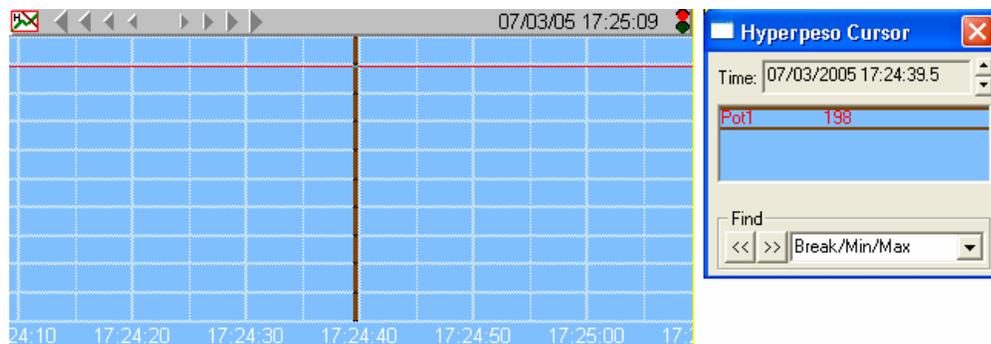


Figura 2. 30. Objeto *Hyper Trend*.

Los valores de los datos históricos mostrados en este objeto son almacenados primero en la base de datos Citadel y luego llevados hasta el panel donde se grafican.

Para escoger una de las señales, mencionadas anteriormente, que se quiere visualizar en el panel del *Hyper Trend*, se hace uso del objeto *RadioButtons*, al cual podemos acceder del siguiente modo: *Menu Object >> Create >> Control >> RadioButtons*.

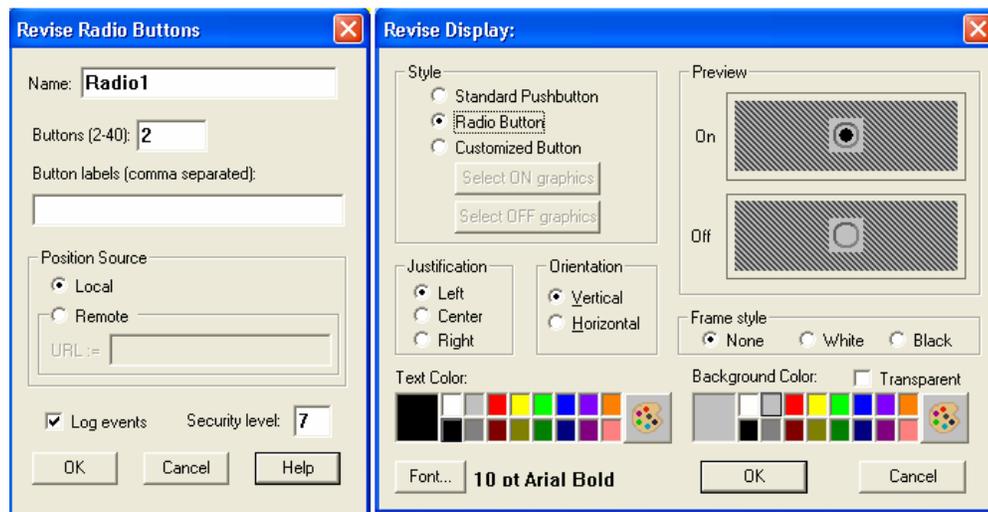


Figura 2. 31. Cuadro de diálogo del objeto *RadioButtons* y su representación gráfica.

Este objeto puede crear de 2 a 40 “botones” en el panel de control. Solo un “botón” a la vez puede ser activado. Los “botones” de este objeto son enlazados con los objetos *Hyper Trend*. Al momento de escoger un botón

hace visible solo un objeto *Hyper Trend*. Los comandos para enlazar ambos objetos (*PushButton* y *Hyper Trend*) son los siguientes:

Hyperpeso.visible = Radio1.2

Hyp_Dam_Pos.visible = Radio2.2

Hyp_Presion.visible = Radio2.3

Hyp_Temp_Horno.visible = Radio2.1

Hyperalt.visible = Radio1.1



Figura 2. 32. Objeto *Hyper Trend* para el peso de la placa (rejilla de plomo).

En esta pantalla también se puede observar los valores de las señales de la maquina empastadora. Como el estado de los motores de las bandas transportadoras, el estado del variador de velocidad, el tipo de rejilla, su espesor y su peso, y la salida del objeto *PID* que controla la altura del rodillo con respecto a la superficie de la banda.

Para escoger el tipo de placa se hace uso del objeto *Recipe*, al cual podemos acceder del siguiente modo:
Menu Object >> Create >> Control >> Recipe.

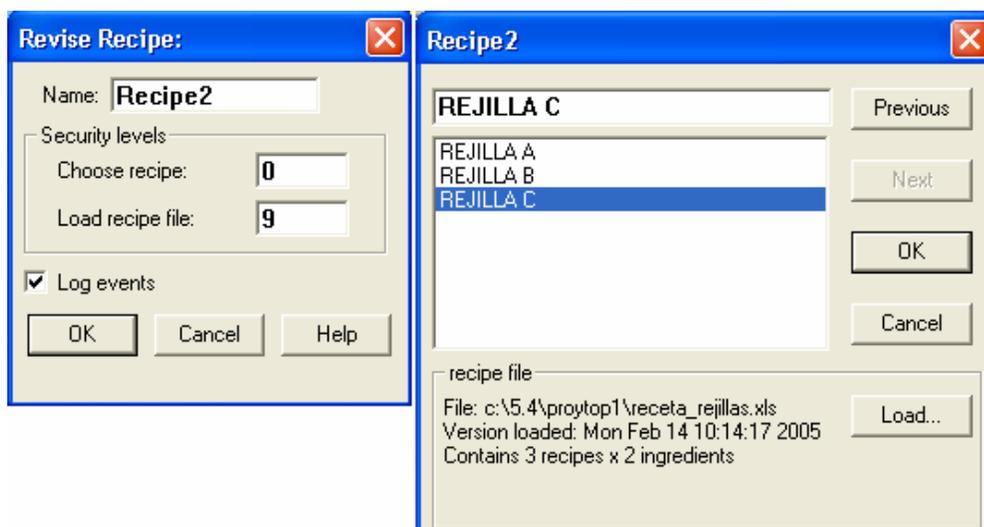


Figura 2. 33. Cuadro de diálogo del objeto *Recipe*.

Este objeto permite importar grandes cadenas de datos, tales como las recetas y sus ingredientes, desde archivos creados en hojas de cálculo de Excel (.xls) hasta Lookout. Se pueden crear hasta mil recetas en una misma hoja de cálculo, y cada receta puede tener hasta 255 ingredientes.

Los nombres de las recetas van en la primera columna y a partir de la segunda fila, los nombres de los ingredientes van en la primera fila a partir de la segunda columna. Una vez creado el objeto *Recipe*, éste permite al usuario escoger entre las recetas existentes en el archivo de Excel, por medio del cuadro de diálogo al hacer click en la receta desde la pantalla de control.

The screenshot shows a Microsoft Excel spreadsheet titled "Cookies.xls". The spreadsheet is organized as follows:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1		Flour	Eggs	Butter	White Sugar	Brown Sugar	Powdered Sugar	Vanilla	Salt	Baking Soda	Cinnamon	Oats	Ginger	Chocolate	Shortening
2	Chocolate Chip	100	50	60	10	8	5	6	3	8	7	0	0	60	43
3	Oatmeal	120	45	50	15	14	3	5	4	7	5	50	0	0	51
4	Ginger Snap	110	40	40	8	11	4	4	5	9	8	0	57	0	68
5															

Figura 2. 34. Archivo de una receta creado en Excel.

Los valores de las señales de los objetos de la pantalla del horno también son mostrados en esta pantalla, tales como el estado de los motores de los ventiladores y de la banda transportadora, el estado de las válvulas solenoides, el valor de la presión de estas, el valor de la presión total dentro del horno, el valor de la temperatura del horno, y el valor de la señal de salida del *PID* que controla dicha temperatura.

Los objetos *PID* son creados, para el control de la temperatura del horno y para el control de la altura del

rodillo de la maquina empastadora. Los parámetros de configuración de estos dos objetos PID, se encuentran en la pantalla Parámetros PID.

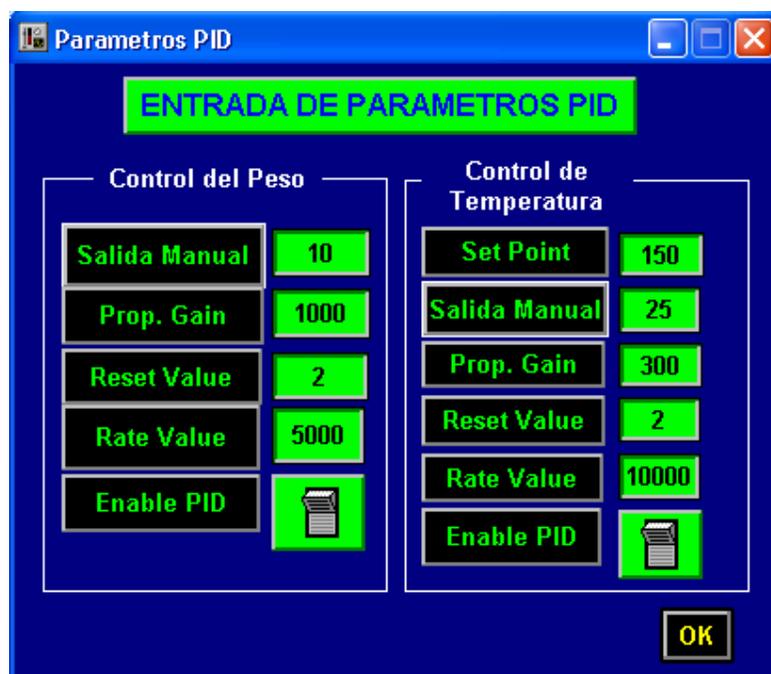


Figura 2. 35. Pantalla de los Parámetros PID.

Para la entrada de los datos de estos parámetros se usan objetos *Pot*, con una representación numérica, que representan los nuevos valores.

La última pantalla a explicarse es la Pantalla de Alarmas, en la cual se visualizan los mensajes de las fallas en los elementos de la maquina empastadora y del horno.

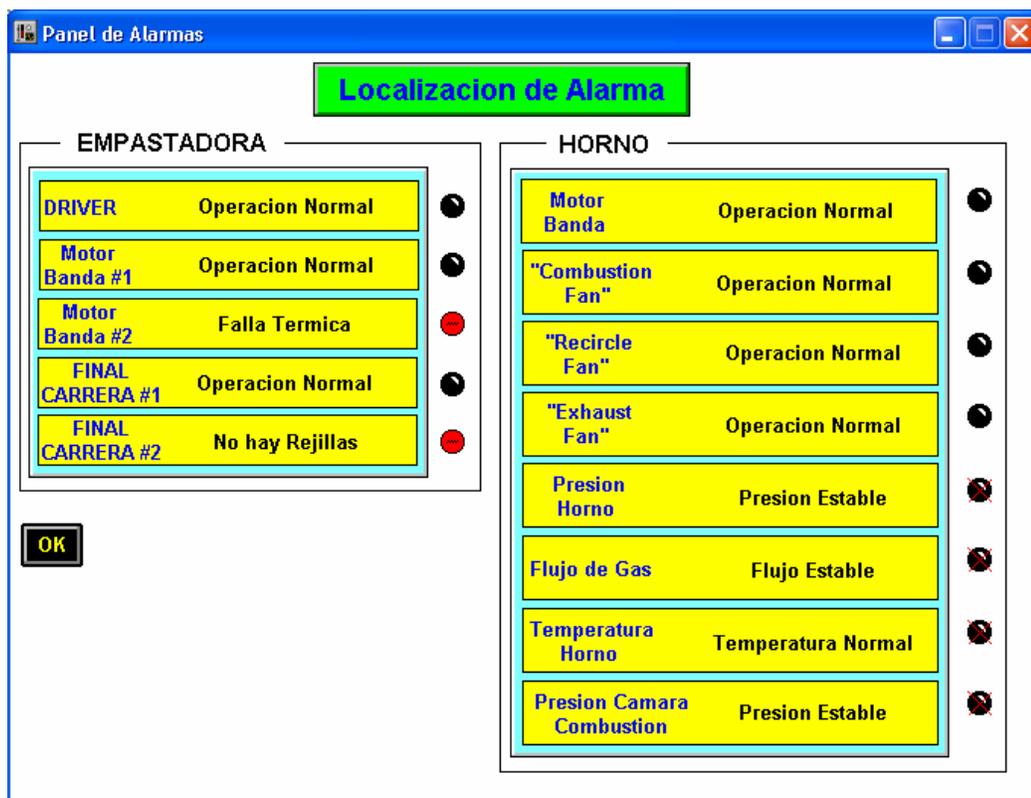


Figura 2. 36. Pantalla de Alarmas del proceso.

Al momento de ocurrir una falla en el proceso se activa una señal de alarma en la pantalla de la máquina empastadora.



Figura 2. 37. Mensaje de alarma en caso de falla de algún elemento del proceso.

Los mensajes de error en la pantalla de alarmas son realizados por expresiones con un display tipo texto.

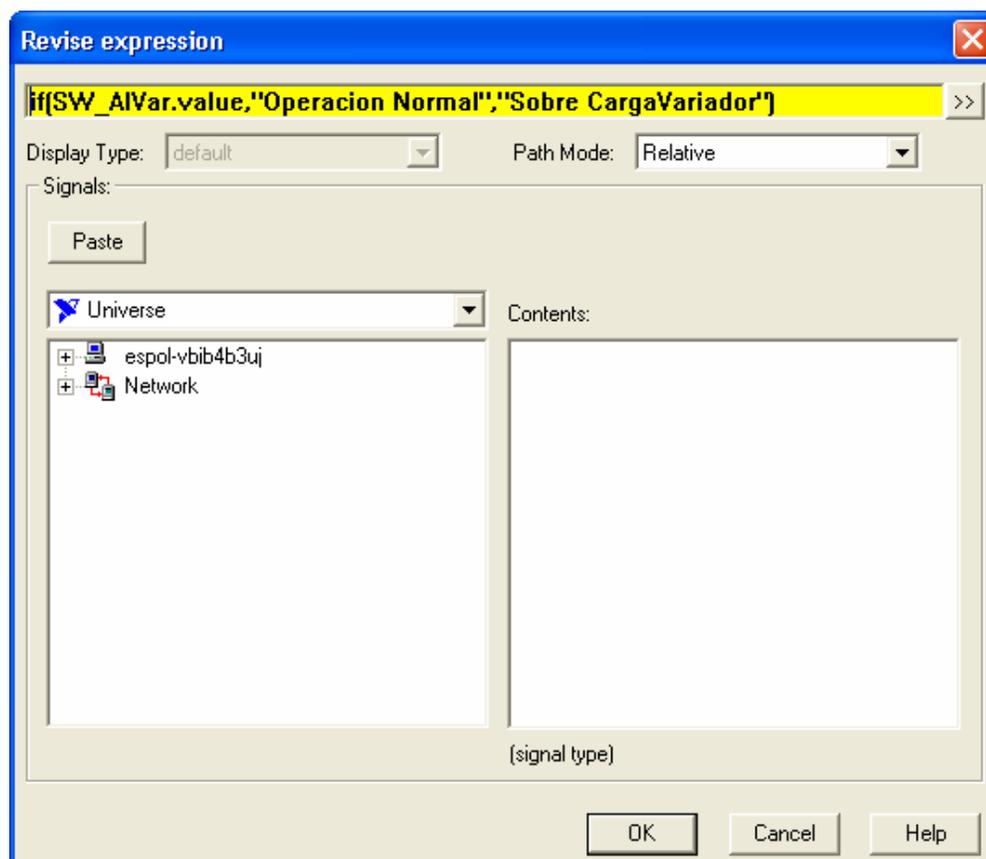


Figura 2. 38. Cuadro de diálogo de la expresión para mostrar un mensaje de falla.

Para lograr que la señal de alarma se muestre de manera intermitente se debe hacer uso de otra expresión: `not(SwT_B2) and Timer1`. Donde *Timer1* es un objeto *Pulse*, al cual podemos acceder de la siguiente manera: *Menu Object >> Create >> Control >> Pulse*.

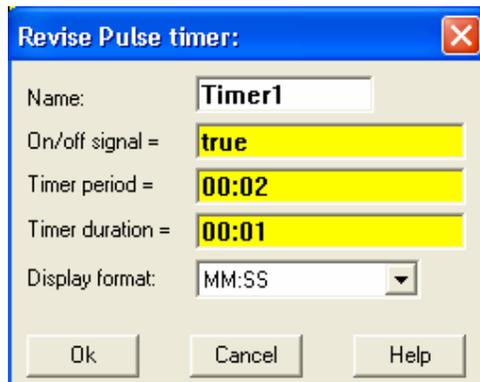


Figura 2. 39. Cuadro de diálogo del objeto *Pulse*.

Este objeto genera una señal tipo pulso (dos posibles estados: 1 y 0), la cual es configurada desde su cuadro de diálogo. Para más información refiérase a la sección *Objetos* de la ayuda de Lookout.

CAPITULO III

3. ANALISIS DE UNA FUENTE DE CONMUTACION

3.1 Introducción.

Una parte importante en el desarrollo de automatizar un proceso, es la fuente de voltaje DC, necesaria para el funcionamiento de diversos componentes electrónicos, específicamente aquellos que necesitan alimentación externa de voltaje, entre estos componentes podemos mencionar:

- Tarjetas de adquisición de datos.
- Sensores que necesiten alimentación externa.
- Módulos de ampliación del PLC.

Estas necesidades, obligó a los diseñadores a desarrollar fuentes DC más compactas, precisas y de mayor eficiencia que las tradicionales fuentes de voltajes DC, es así como nacieron las Fuentes Conmutadas.

Hablar de Fuentes Conmutadas es sinónimo de eficiencia, precisión y estabilidad, como su nombre lo indica, el principio básico de funcionamiento de estas fuentes, es la conmutación en alta frecuencia de interruptores electrónicos controlados (Transistor Bipolar, MOSFET, IGBT, etc.), para de esta forma obtener un voltaje fijo independiente de la cantidad de carga conectada a esta (siempre que no exceda el rango permitido por la fuente).

Por medio de este capítulo se detallará un análisis teórico acerca del diseño de una fuente de conmutación, sus componentes electrónicos y gráficas de salida en los diferentes puntos de la fuente, estas gráficas serán analizadas por medio del simulador de **PSPICE PROFESIONAL** y también por medio del **OSCILOSCOPIO (Tektronix 2220)** que se encuentran en los diferentes laboratorios de la universidad.

3.2 Principio básico de una fuente de conmutación.

El funcionamiento interno de la fuente de conmutación es un poco complejo de entender para una mejor visualización se ha procedido a dividir esta estructura en cuatro grandes etapas que son las siguientes:

- 1.- *Etapas de Rectificación y Filtrado de Entrada.*
- 2.- *Etapas de Conmutación en Alta Frecuencia.*
- 3.- *Etapas de Rectificación y Filtrado de Salida.*
- 4.- *Etapas de Control (Modulación Ancho de Pulso).*

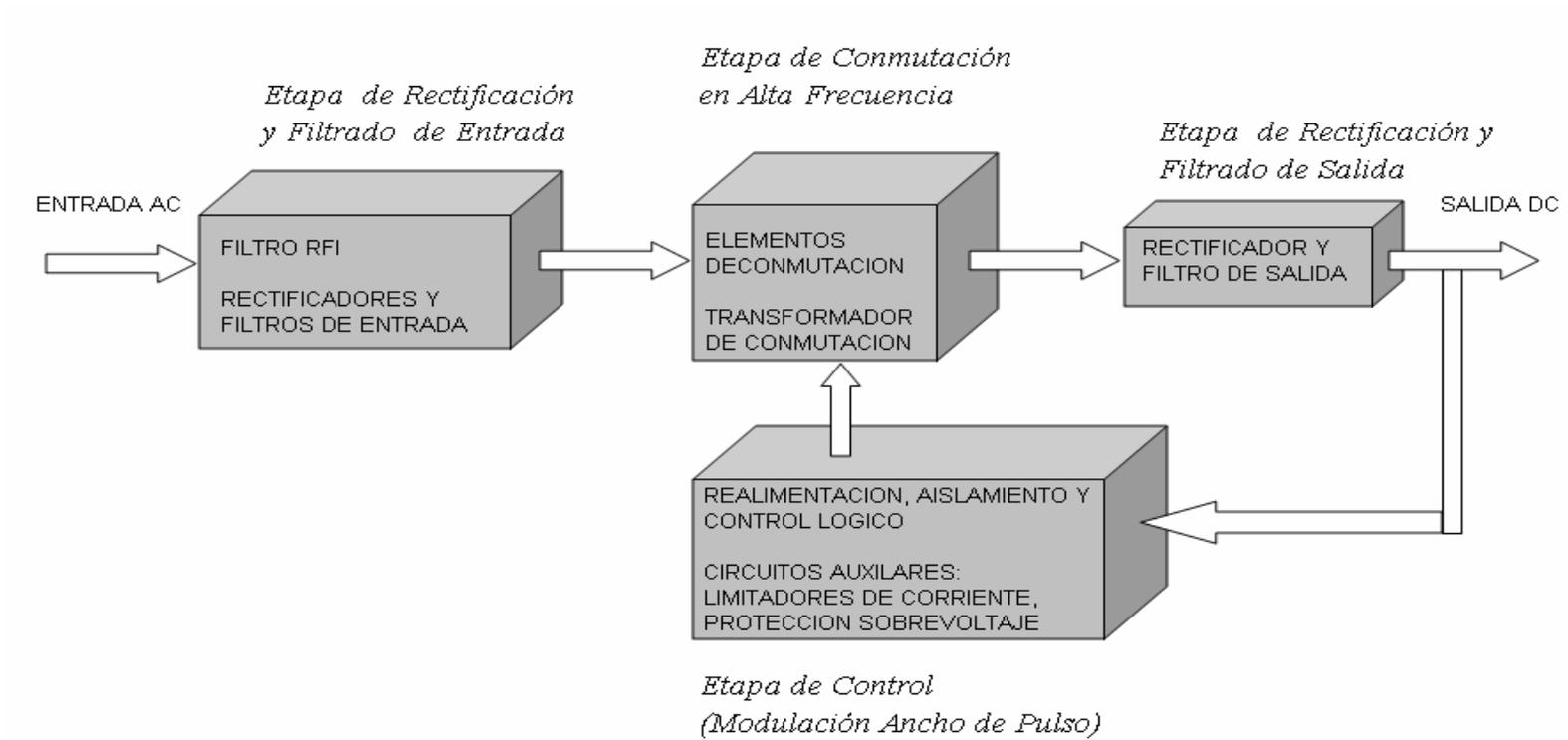


FIGURA 3-1 Diagrama de bloques de las etapas que conforman a una fuente conmutada.

3.3 Diseño, cálculos y análisis de una fuente conmutada.

3.3.1 Etapa de rectificación y filtrado en la entrada.

La función de esta etapa es convertir la señal alterna de una red comercial de baja tensión (120 o 240 V) en una señal continua, esto se logra por medio de un puente rectificador de diodos, que recibe la señal una vez que esta ha pasado por una red LC con el propósito de reducir Interferencias de Radio Frecuencia (RFI, ver figura 3.2), producidos por los componentes de la fuente. Esta red está generalmente formada por dos inductores (L_1 y L_2) en serie a cada una de las líneas AC de entrada, mientras que los capacitores son colocados entre las líneas (llamado C_x) y entre cada línea y tierra (llamados C_y). Además existe una resistencia (R) en paralelo a C_x , esta resistencia permite la descarga de este capacitor, el valor de este resistor está establecido por la norma IEC-380 Sección 8.8 la cual establece que:

“Si el capacitor C_x de una red para reducir las RFI es mayor que $0.1 \mu F$, un resistor de descarga, calculado con la ec. 3.5 es requerido en el circuito”

El funcionamiento de los diodos del rectificador va a depender de la posición del interruptor “Tclose” que es el que habilita la alimentación 120/240 (V). Cuando el interruptor está abierto, se habilita el modo de trabajo 240 (V) y los diodos trabajan como un Rectificador de Onda Completa Tipo

Puente. Si el interruptor está cerrado, el circuito de diodos trabaja como un Rectificador de Onda Completa con Tap Central en el cual solo trabajan D1 y D2 de manera alternada junto con el capacitor C1 como se indica en la figura 3-2.

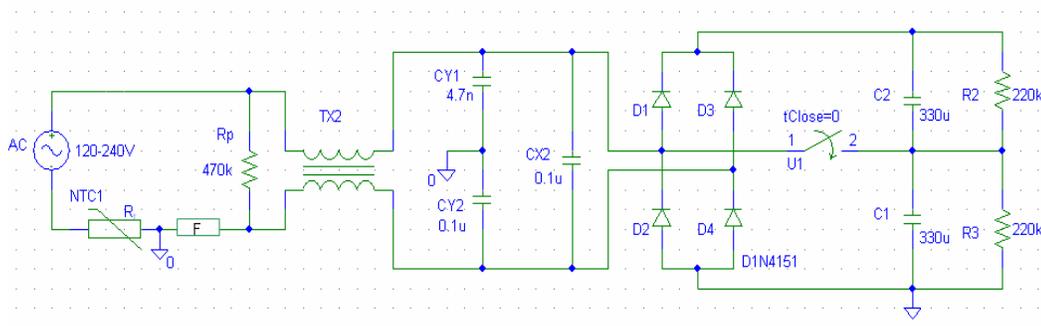


FIGURA 3-2 Fase de rectificación que consta del termistor, la red RFI, el puente rectificador y los capacitores.

Por último un termistor NTC que actúa como elemento de protección de sobre-corriente de la fuente.

Para el cálculo y selección de los elementos de esta etapa, se necesita conocer los resultados obtenidos en la práctica de la potencia de entrada y de salida de la fuente, para ello hemos preparado un resumen de fórmulas para mayor facilidad.

FORMULAS.

$$P_{IN} = \frac{P_{OUT}}{\eta} \dots\dots\dots (3.1)$$

$$C = \frac{I * t}{\Delta V} \dots\dots\dots (3-2)$$

$$V_{dc} = 2 * (Vp) \dots\dots\dots (3.3)$$

$$I = \frac{P_{IN}}{V_{dc}} \dots\dots\dots (3.4)$$

$$R_{DESC} = \frac{t_{DESC}}{2.21 * Cx\epsilon q} \dots\dots\dots (3.5)$$

Los valores permitidos para los elementos de la red RFI se encuentran estandarizados.

Por medio de los datos de placa de la fuente conmutada conocemos que la potencia de salida es de 145(W), lo que nos permite por medio de la fórmula (3-1) encontrar la eficiencia una vez calculada la potencia de entrada.

Para calcular la Potencia de Entrada, necesitamos conocer la corriente de entrada (I_{IN}) la cuál fue encontrada por medio de un amperímetro dando como resultado $I_{IN}=1.79(A)$, además se conoce que el $V_{AC}=120(V)$, por medio de estos datos podemos calcular la Potencia de Entrada:

$$P_{IN} = V_{AC} * I_{IN}$$

$$P_{IN} = 120(V) * 1.79(A)$$

$$P_{IN} = 214.8 (W).$$

Por medio de este cálculo de la Potencia de Entrada procedo a encontrar la eficiencia a través de la fórmula (3.1).

$$\eta = \frac{145(W)}{214.8(W)} = 0.67$$

Para hallar el valor de los condensadores, es necesario calcular la corriente de la fuente así como conocer el voltaje de rizado de la fuente a plena carga y el tiempo de carga del capacitor, los dos últimos valores los obtenemos prácticamente analizando las señales del osciloscopio para esta etapa (Figura A1).

Antes de encontrar la corriente de carga debemos encontrar el valor del Voltaje DC a la salida de esta etapa a través de la fórmula (3-3).

$$V_{dc} = \sqrt{2} * 2 * (120)$$

$$V_{dc} = 339.4 (V)$$

Por medio de este valor del Voltaje DC procedemos a encontrar el valor de la corriente de carga que la calcularemos a través de la fórmula (3-4).

$$I = \frac{214.8(W)}{339.4(V)}$$

$$I = 0.63(A).$$

Una vez encontrada la corriente de carga procedemos a encontrar el valor de la capacitancia equivalente (C) formada por C_1 y C_2 por medio de la fórmula (3-2).

$$C = \frac{0.63 * (8 * 10^{-3})}{30} = 168(\mu F)$$

Donde $C = 168$ (μF) es la capacitancia equivalente de la suma de las capacitancias en serie, entonces $C_1 = C_2 = 336$ (μF), pero debemos aproximarlos a un valor comercial más cercano, el cual es $C_1 = C_2 = 330$ (μF).

Los valores de los elementos que componen el filtro RFI de nuestra fuente son los siguientes:

$$L = 3.29 \text{ mH.}$$

$$C_x = 4700 \text{ pF}$$

La elección de los valores de estos dos últimos elementos se la justifica mediante un análisis de la respuesta de frecuencia de la red, esto se lo hace en el Anexo B de esta tesis.

Utilizando la fórmula (3.5), tenemos que el valor de la resistencia de la red de protección contra interferencias RFI es:

$$R = \frac{1}{2.21 * 2 * 0.47(\mu F)} = 481(k\Omega)$$

Por cuestiones comerciales se debe aproximar a un valor de 450 (K Ω).

3.3.2 Etapa de conmutación en alta frecuencia.

Esta etapa consiste en un “Convertidor de Alta Frecuencia”, que se encarga de convertir el voltaje (DC) proveniente de la etapa anterior a una señal cuadrada (AC), cuya frecuencia y ciclo de trabajo depende de la señal de control proveniente de la etapa de control (la cuál analizaremos más adelante).

Existen muchas configuraciones de convertidores, sin embargo nos dedicaremos a un breve análisis del convertidor tipo “Medio Puente” (Fig.3.3), que es el que posee la fuente que es objeto de nuestro análisis.

Las dos principales ventajas de esta configuración son: primero, que trabaja a 120 o 240 (V) sin preocuparnos por los altos voltajes producidos en los transistores, y la segunda razón es que nos entrega una señal simétrica en la salida, lo que evita la saturación del núcleo del transformador conectado a la salida.

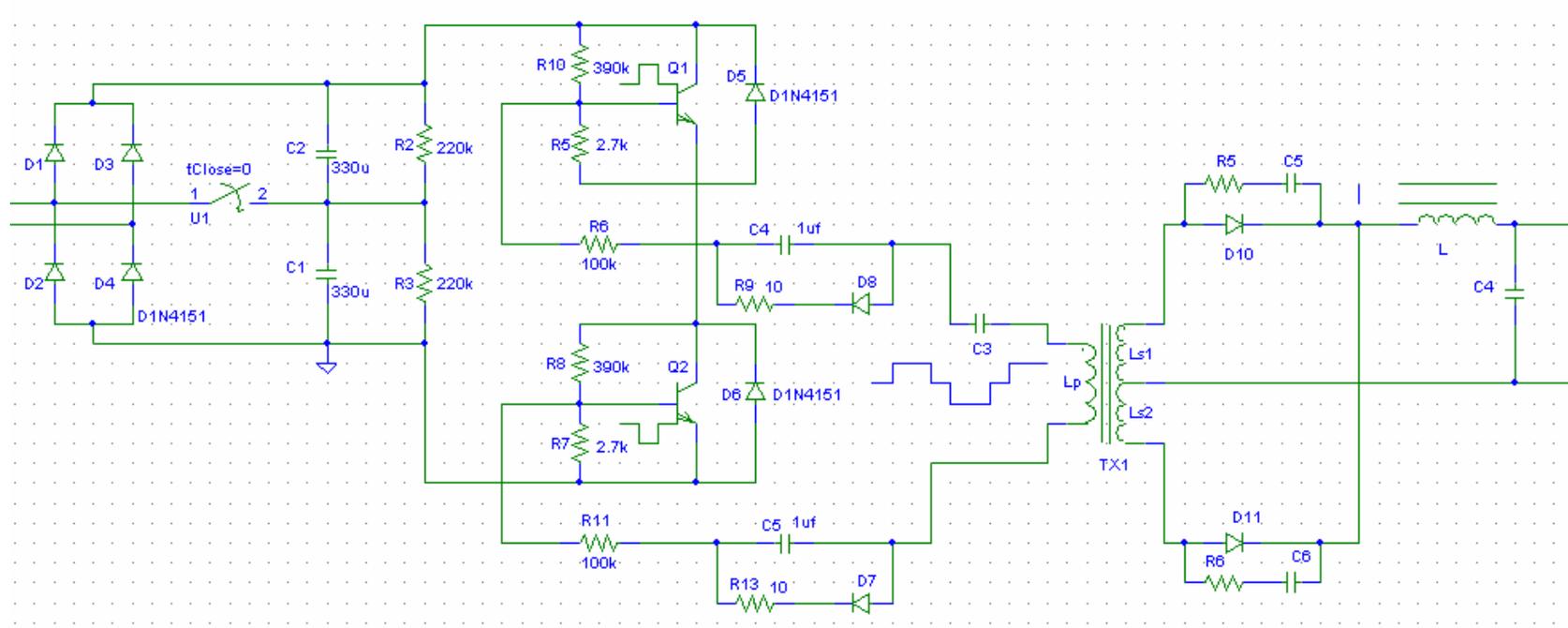


Figura 3.3 Esquema del convertidor tipo “Medio Puente” de nuestra fuente. Notemos que el modo de trabajo de los transistores y el transformador es el mismo independiente de la alimentación utilizada (120 o 240 V)

Nótese que el transformador de la salida tiene un terminal conectado al punto de unión entre C1 y C2, mientras que el otro terminal está conectado en la unión del colector de Q2 y el Emisor de Q1, como esta pareja de transistores trabajan alternadamente (si Q1 esta saturado, Q2 está en corte y viceversa), tenemos que cuando Q1 conduce, en el transformador se genera un voltaje de $V_{in}/2$, mientras que cuando Q2 conduce, el transformador recibe un pulso de voltaje de $-V_{in}/2$. Por lo que a la salida del transformador se produce un Voltaje de pico a pico = V_{dc} .

El objetivo del capacitor C3, en serie con el transformador es para eliminar cualquier asimetría producida por el convertidor, de esta manera se evita la saturación del núcleo ya que el voltaje promedio de la salida es cero.

CALCULOS TEÓRICOS DE LOS ELEMENTOS

La fórmula para encontrar la corriente de carga del capacitor es la siguiente:

$$I_c = \frac{3 * P_{out}}{V_{in}} \dots\dots\dots(3-6)$$

Para hallar el campo de saturación (B sat) debemos utilizar el valor correspondiente a la ferrita usada por nuestra fuente.

Donde se debe cumplir la condición de que $B_{max} \leq B_{sat}$. Reemplazando, los valores en la ecuación (3-6), procedemos a encontrar la corriente de carga del capacitor, se tiene que:

$$I_c = \frac{3 * 145W}{339.4} = 1.28(A)$$

Posteriormente procedemos a realizar el cálculo del capacitor que se encuentra en serie con el transformador principal, este condensador tiene como objetivo balancear el voltaje proveniente del puente y de esa forma evitar la saturación del cobre (3) y de donde mostramos a continuación todas las posibles fórmulas que vamos a utilizar.

Dada las siguientes ecuaciones:

$$C_3 = \frac{1}{4 * \pi^2 * f_R^2 * \left(\frac{N_p}{N_s}\right)^2 * L} \dots\dots\dots (3-7)$$

$$f_R = \frac{f_S}{4} \dots\dots\dots(3-8)$$

$$a = \frac{V_P}{V_S} \dots\dots\dots(3-9)$$

Para calcular el valor del Capacitor se deben utilizar las fórmulas antes mencionadas y en base a ellas seguimos los siguientes pasos:

La frecuencia de conmutación la obtenemos con el osciloscopio, midiendo la frecuencia de salida del IC "KA7500B", esta es $f_s = 30\text{Khz}$ aproximadamente; por medio de la fórmula (3-8) procedemos a encontrar la frecuencia de resonancia.

$$f_R = \frac{30\text{Khz}}{4} = 7.5\text{Khz}.$$

Una vez encontrada la frecuencia de resonancia y conociendo el valor del inductor $L=4.5$ (uH), procedemos a encontrar la relación de vueltas del transformador principal.

Durante los cálculos que se efectuaron en el laboratorio de electrónica de potencia se pudo verificar por medio de la gráfica obtenida a través del osciloscopio que el voltaje del primario del transformador fue de $V_p = 300(\text{V})$ p-p y el voltaje del secundario fue de $V_s = 80(\text{V})$ p-p lo que nos permite poder encontrar su relación de vueltas a través de la fórmula (3-9).

$$a = \frac{300(\text{V})}{80(\text{V})}$$

$$a = 3.75$$

Conociendo todos los datos necesarios para poder encontrar el valor del capacitor procedemos a encontrarlo por medio de la fórmula (3-7).

$$C_3 = \frac{1}{4 * 3.14^2 * (7.5 * 10^3)^2 * (3.75)^2 * 4.5 * 10^{-6}} = 1.58\mu\text{F}$$

El valor del capacitor C3 de nuestra fuente es de 1 μF , 250 V y es el que utilizaremos en adelante para los demas cálculos. Con respecto a la capacidad de carga del Capacitor se obtienen las siguientes ecuaciones:

$$dt = \frac{T}{2} \delta_{\max} \dots\dots\dots(3-10)$$

$$T = \frac{1}{f_s} \dots\dots\dots(3-11)$$

$$V_c = dt \frac{I}{C} \dots\dots\dots(3-12)$$

Simultáneamente procedemos a calcular el Voltaje de Carga del Capacitor en donde tendremos que calcular lo siguiente:

Primero procedemos a calcular el período de conmutación una vez que conocemos la frecuencia de conmutación a través de su respectiva fórmula.

Si $f_s = 30$ (Khz.), reemplazando en la fórmula (3-11) se tiene:

$$T = \frac{1}{30 * 10^3} = 33,3\mu\text{s}$$

Luego procedemos a calcular el intervalo de tiempo de carga usando su respectiva ecuación lo cual debemos conocer su ciclo de trabajo que se lo representa por δ_{\max} .

Si $\delta_{\max} = 0.8$ reemplazando en la ecuación se obtiene que:

$$dt = \frac{33.33 * 10^{-6}}{2} * 0.8 = 13.33\mu\text{s}$$

Ahora, para siguiente cálculo a la corriente de carga se le agrega un 20% al valor nominal de corriente dando como resultado lo siguiente:

$$I = 1.2I_c = 1.2 * 1.28 = 1.54(A)$$

Otro factor a tener en cuenta para elegir este capacitor es el voltaje de carga usando la ecuación (3-12) de donde se obtiene lo siguiente:

$$V_c = \frac{1.54}{1.0 * 10^{-6}} * 13.33 * 10^{-6} = 20.53V$$

Los picos de voltaje son eliminados a través de los diodos de conmutación, estos picos de voltaje pueden ocasionar algún daño en los transistores debido a la energía almacenada en el inductor del transformador.

A través de la densidad de corriente podemos determinar el tamaño del núcleo y su respectivo número de vueltas, este valor de la densidad de corriente está dada por el fabricante de cables y debe de ser < 1000 cm/amp y en la práctica siempre se usa mínimo 200 o elegimos 500 cm/amp.

$$A_e.A_c = \frac{(0.68 * P_{out} * D) * 10^3}{f * B_{MAX}} \dots\dots\dots(3-13)$$

$$I_p = \frac{3 * P_{out}}{V_{in}} \dots\dots\dots(3-14)$$

Por medio del catálogo del fabricante del núcleo de hierro, busco un núcleo cuya $A_e A_c$ sea por lo menos el 50% más que el $A_e A_c$ calculado.

Reemplazando los valores en la fórmula respectiva tenemos que:

Cálculo de $A_e A_c$ usando la ecuación 3-13.

Para calcular el valor de $A_e A_c$ se debe conocer los valores de $D = 500 \text{ cm/A}$; $f = 20 \text{ KHz}$; $P_{out} = 145 \text{ W}$, al ser reemplazados en la siguiente ecuación (3.13) se tiene que:

$$A_e A_c = \frac{(0.68 * 145 \text{ W} * 500 \text{ cm/A}) * 10^3}{(20 * 10^3) * B_{\max}}$$

$$A_e A_c = \frac{2465}{B_{\max}}$$

Para poder dimensionar el tamaño del alambre a utilizar y el número de vueltas se debe emplear la siguiente fórmula:

$$CM = I_p * D \dots\dots\dots(3-15)$$

Reemplazando los valores de I_p y de D , tenemos:

$$I_p = 1.28 \text{ (A)}$$

$$D = 500 \text{ cm/Amp}$$

$$CM = 1.28 * 500$$

$$CM = 640.83 \text{ cm.}$$

Con este valor de CM el cual se revisó en su respectiva tabla 3-1 que se encuentra en el anexo B, esta tabla pertenece a los conductores el cuál fue ubicado el cable # 21 AWG.

A través de la siguiente fórmula se procede a calcular el número de vueltas del transformador:

$$N_p = \frac{V_p * 10^8}{4 * f * B_{MAX} * A_e} \dots\dots\dots(3-16)$$

Para encontrar el valor teórico del voltaje del primario V_p , reemplazamos en la siguiente formula.

$$V_p = (100 * \sqrt{2}) - 30 = 111.42V$$

Donde 100 es el valor mínimo de voltaje permitido para el trabajo de la fuente.

Procedemos a revisar el catálogo del fabricante en donde encontraremos que el N_p calculado sea menor que el 30% del máximo del área. Si se cumple, entonces están bien dimensionados el núcleo y la bobina del transformador.

El 30% del máximo área es porque es aproximadamente el área que ocupa la bobina del primario.

Como ya conocemos el # de vueltas, encontramos B_{max}

Parte importante en esta etapa son los disipadores de calor instalados en los transistores de conmutación, el análisis de estos elementos se lo realiza en el Anexo C de este documento.

3.3.3 Análisis de la etapa de rectificación y filtro de salida.

El diseño interno de esta fuente consta de una red Snnuber que sirve para eliminar los picos de voltaje que se producen debido a la inductancia del transformador de alta frecuencia, la fase de salida es múltiple por lo que el voltaje es derivado y filtrado desde el secundario del transformador de alta frecuencia. Los elementos constitutivos de esta fase se muestran en la figura 3.4

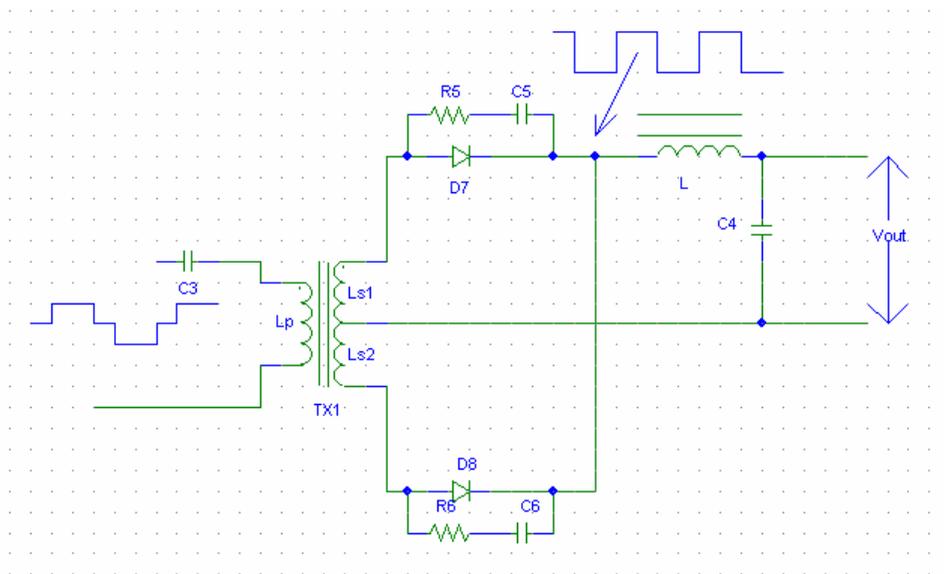


Figura 3.4. Circuito de Rectificación y Filtrado para una una fuente de switcheo tipo “Medio Puente” donde se incluye también la Red Snnuber.

CALCULOS TEÓRICOS

Los elementos que se utilizan en esta fase deben tener la capacidad de soportar corriente máxima de salida, uno de ellos deben ser los diodos que para su estudio pueden ser de tres tipos:

- 1.- Diodos de Alta eficiencia y rápida recuperación
- 2.- Diodos de Alta eficiencia y muy rápida recuperación.
- 3.- Diodos Schottky

La red snubber es un circuito RC en paralelo al puente de diodos que sigue al transformador. La selección de sus componentes se la detalla a continuación:

C_s de 0.01 uF a 0.1 uF

$$R_s = \frac{\sqrt{L_T / C_J}}{a} \dots\dots\dots(3-17)$$

$$P_R = \frac{1}{2} C_s \left(\frac{V_{in}}{a} \right)^2 f \dots\dots\dots(3-18)$$

Además este circuito consta de una red LC que tiene dos funciones: la primera para almacenar energía durante la conmutación y así proporcionar corriente continua en el caso del inductor y la segunda filtrar aún más el voltaje de salida en el caso del capacitor. Las fórmulas son las siguientes:

FORMULAS PARA EL CALCULO DEL TRANSFORMADOR

INDUCTOR.

$$V_L = L \frac{di}{dt} \dots\dots\dots(3-19)$$

$$t_{OFF} = \frac{1}{2} \left[\frac{1 - (E_{OUT}/E_{IN})}{f} \right] \dots\dots\dots(3-20)$$

$$L = \frac{E_{OUT} * t_{OFF}}{0.25 * I_{OUT}} \dots\dots\dots(3-21)$$

NUCLEO

$$A_e A_c = \frac{5.067 * 10^8 (LI_{OUT} D^2)}{KB_{MAX}} \dots\dots\dots(3-22)$$

$$l_g = \frac{(0.4 * \pi * LI_{OUT}^2) 10^8}{A_e B_{MAX}^2} \dots\dots\dots(3-23)$$

$$N = \frac{B_{MAX} l_g}{0.4 * \pi * I_{OUT}} \dots\dots\dots(3-24)$$

CAPACITOR

$$C_{OUT} = \frac{\Delta I_{OUT}}{8f\Delta V_{OUT}} \dots\dots\dots(3-25)$$

$$ESR_{max} = \frac{\Delta V_{OUT}}{\Delta I_{COUT}} \dots\dots\dots(3-26)$$

A continuación hallaremos el valor del periodo del convertidor.

$$T = \frac{1}{f}$$

Asumimos un tiempo muerto de 5us por cada medio ciclo.

El tiempo total de conducción de cada ciclo seria:

$$\frac{50 - 2 * 5}{2} = 20(us)$$

el ciclo de trabajo por cada diodo es $\frac{20}{50} = 0.4$

El pico de corriente máximo (I_{FM}) por diodo esta dado por:

$$I_{FM} = \hat{\alpha}_{DR} * I_{out} \dots\dots\dots(3-27)$$

Para la salida de ± 5 V de la fuente, se tiene en $I_{OUT} = 18(A)$
donde utilizando la fórmula (3-27), tenemos:

$$I_{FM} = (18) * (0.4) = 7.2(A)$$

Por medio de este cálculo el diodo seleccionado para esta fuente tiene que ser de 8 A .

Para la salida de ± 12 V de la fuente, se tiene como $I_{OUT} = 4.2(A)$
en donde utilizando la formula (3-27), tenemos:

$$I_{FM} = (0.4) * (4.2) = 1.68(A)$$

Por medio de este cálculo el diodo seleccionado para esta fuente tiene que ser de 2 A .

A continuación analizaremos el Filtro LC, para ello se debe calcular el valor del inductor, en donde se deberá seguir los siguientes pasos para su respectivo cálculo.

$$L = \frac{(E_{in} - E_{out}) * \Delta t}{\Delta I_L} \dots\dots\dots(3-28)$$

Para un converter "Medio Puente", tenemos que $E_{in} = 2 * E_{out}$ para Vin máximo.

En términos del voltaje secundario

$$t_{OFF} = \frac{1}{2} \left[\frac{1 - (E_{out} / E_{in})}{f} \right] \dots\dots\dots(3-29)$$

Por lo tanto :

$$t_{OFF} = \frac{1}{2} \left[\frac{1 - (0.5)}{20 \times 10^3} \right] \qquad E_{OUT} = 2E_{in}$$

$$t_{OFF} = 12 \mu\text{seg}$$

Para las 4 fuentes, el valor de L lo encontramos mediante:

$$L = \frac{(E_{in} - E_{out}) * \Delta t}{\Delta I_L} \dots\dots\dots(3-30)$$

Para la fuente de +5v, se trabaja con una corriente de 18A dando como resultado un valor de inductor de L=13.33mH.

Para la fuente de -5v, se trabaja con una corriente de 0.3A dando como resultado un valor de inductor de L=800mH.

Para la fuente de +12v, se trabaja con una corriente de 4.2A dando como resultado un valor de inductor de L=133.9mH.

Para la fuente de -12v, se trabaja con una corriente de 0.3A dando como resultado un valor de inductor de L=1.92mH.

Para la fuente de +5v este valor representa $L * I_{OUT}^2 = 4.46 (A^2 * mH)$.

Este valor es el utilizado porque es el mayor de las cuatro fuentes.

El valor del número del núcleo es de 55548, este valor está entre las curvas de impermeabilidad del libro guía y son los siguientes u=26 y u=60.

Intentaremos primero con el valor de u=60 para hallar el número de vueltas con la siguiente ecuación.

$$N = 1000 \sqrt{\frac{L}{L^{1000}}} \dots\dots\dots(3-31)$$

Donde :

$$L = (\text{mH})$$

$$L^{1000} = (\text{mH})/1000 \text{ vueltas.}$$

Para los siguientes cálculos tomamos como referencia del libro guía el valor de $L_{1000} = 61 \text{ mh}/1000 \text{ vueltas}$ para poder encontrar los respectivos números de vueltas en cada una de las fuentes.

Con la fuente de +5v nos da como resultado un valor de numero de vueltas de:

$$N = 1000 * \sqrt{\frac{13.33^{-3}}{61}}$$

$$N = 14.8 \text{ vueltas}$$

Este valor se lo tiene que incrementar en un 20% del número de vueltas encontrado, dando como resultado $N = 18 \text{ vueltas}$.

Para la fuente de -5v se obtuvo una relación de vueltas de

$$N = 1000 * \sqrt{\frac{0.8}{61}} \text{ dando como resultado } 114 \text{ vueltas, con un}$$

incremento del 20% se obtuvo una relación de 137 vueltas.

Para la fuente de +12v se obtuvo una relación de vueltas de 40, mientras que para la fuente de -12v su relación fue de 150 vueltas.

3.3.4 Etapa de control (Modulación de ancho de pulso-PWM).

El Controlador, se encarga de comandar la oscilación del Convertidor de Alta Frecuencia, consiste de un oscilador de frecuencia fija, una tensión de referencia, un comparador de tensión y un modulador de ancho de pulso (PWM). El modulador recibe el pulso del oscilador y modifica su ciclo de trabajo según la señal del comparador, el cual presenta la tensión continua de salida del tercer bloque con la tensión de referencia.

Cabe aclarar que el ciclo de trabajo es la relación entre el estado de encendido y el estado de apagado de una onda cuadrada.

También cabe resaltar que en la mayoría de los circuitos de fuentes conmutadas se encuentran el primer y el tercer bloque de elementos y configuraciones invariables, en cambio el cuarto y el segundo tendrán diferentes tipos de configuraciones. A veces el cuarto bloque está construido con circuitos integrados y en otras ocasiones está implementado con circuitos totalmente transistorizados.

EL KA7500B es un modulador por ancho de pulso de frecuencia fija (PWM). La modulación de los pulsos de salida se logra comparando la señal de diente de sierra creada por el oscilador interno con cualquiera de las dos entradas de control. La etapa de salida está habilitada durante la porción de tiempo que el voltaje del diente de sierra es mayor a las señales de control. Conforme las señales de control aumentan, el periodo de tiempo en que el diente de sierra es mayor decrece, por lo que el tiempo que dura el pulso de salida disminuye.

Las señales de control provienen de dos fuentes: el circuito de control de tiempo muerto (tiempo de apagado) y del circuito amplificador de error. La entrada de control de tiempo muerto es comparada directamente en el comparador de control de tiempo muerto. Este comparador tiene un voltaje de OFFSET fijo de 110 mV, esto provee un tiempo muerto mínimo de 3 %. EL comparador PWM compara la señal de control creada por los amplificadores de error, la función de estos es la de monitorear el voltaje de salida y proveer una ganancia suficiente de modo que, sus milivoltios de variación resulten en una señal de control de suficiente amplitud para que la modulación se realice. Los amplificadores de error también se usan para monitorear la corriente de salida y proveer una limitación de corriente de carga.

Un oscilador lineal interno programable por dos componentes externos, RT y CT en donde para encontrar la frecuencia del oscilador utilizamos la siguiente fórmula:

$$f_{osc} = \frac{1.1}{R_T * C_T} \dots\dots\dots (3-32)$$

La modulación de ancho de pulso de salida está en contraste de la señal de onda cerrado positivo a través de condensador CT, sea de dos señales de control. Las puertas NOR, conducen transistores de salida Q1 y Q2, la cual son habilitadas sólo cuando en la línea de entrada del *flip-flop* llega la señal del *clock* en donde está de condición bajo. Esto ocurre sólo durante esa porción de tiempo cuando el voltaje cerrado es mayor que la señal de control. Por consiguiente, un incremento en el control – la amplitud de la señal causa una disminución lineal correspondiente a la anchura de pulso de salida.

Las señales de control son entradas externas, el control de tiempo muerto (DTC), las entradas de error del amplificador (los pines 1, 2, 15, 16), o la entrada de retroalimentación (pin 3). El tiempo total del comparador de control de tiempo offset 120 efectivo de entrada mV que limita la salida mínima de tiempo muerto para aproximadamente el primer 4 % del diente de sierra. Esto resultaría en un ciclo arancelario máximo de 96 % con la salida de modo el control (pin 13) puso en tierra, y que 48 % con él se conectó a la línea de referencia. El tiempo muerto adicional puede ser impuesto en la salida colocando el tiempo muerto

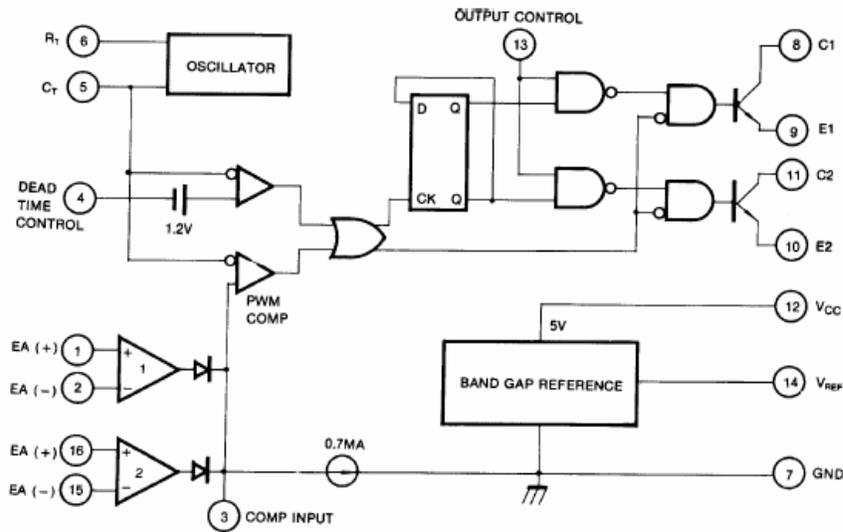


Figura 3.5 *Circuito Interno del CI KA75000B, encargado del control del lazo de Realimentación para el Control del Ancho de Pulso.*

El comparador del modulador de ancho de pulso provee de una manera para los amplificadores de error puedan ajustar la anchura de pulso de salida del por ciento máximo en el tiempo establecido por el tiempo muerto, la entrada de control de tiempo, como el voltaje en el pin de retroalimentación (FEED BACK) tenga una diferencia de 0.5 para 3.5 V. Ambos amplificadores de error tienen un rango común de entrada de modo de 0.3 V para $(V_{CC} - 2 V)$, y puede estar acostumbrado a tener voltaje de salida del suministro de fuerza y corriente. Las salidas del amplificador

de error son alto valor Con esta configuración, el amplificador que demanda salida mínima a tiempo, domina el control del lazo.

Cuando el condensador CT es descargado, un pulso positivo es generado en la salida del fracaso del tiempo muerto, que cronometre el pulso e inhibe los transistores de salida, Q1 y Q2. Con el control de modo de salida conectado para la línea de referencia, el fracaso de pulso dirige los pulsos modulados para cada uno de los dos transistores de salida alternativamente por la operación push pull. La frecuencia de salida es igual para el oscilador.

Cuando las corrientes de salida son requeridas para una simple operación, Q1 y Q2 pueden estar conectados en paralelo, y el pin de control de modo de salida debe estar en tierra. La frecuencia de salida ahora sea igual al del oscilador.

3.3.5 Análisis de error y comparación entre los datos obtenidos por el simulador y los datos reales de la fuente.

Análisis y comparación del cálculo de error.

$$\%error = \frac{|Valor_Teorico - Valor_Practico|}{Valor_Teorico} \dots\dots(3-33)$$

Para el respectivo cálculo de error del valor del capacitor de entrada de la fuente, procedemos a utilizar la fórmula (3-33):

$$\%error_capacitor = \frac{|336 - 330|}{336} * 100 = 1.78\%$$

Para el cálculo de error del valor del Voltaje de rizado pico a pico de la fuente,

$$\%error_Vpp = \frac{|30 - 25|}{30} = 16\%$$

CAPÍTULO IV

4. DISEÑO E IMPLEMENTACIÓN DE UNA TARJETA DE ADQUISICIÓN DE DATOS BASADA EN EL MICROCONTROLADOR 16F877.

4.1 Descripción del Sistema.

4.1.1 Introducción.

En este capítulo describiremos las características técnicas de la tarjeta de adquisición diseñada, la misma que consta de tres entradas digitales, tres entradas analógicas de 0 a 5 Vdc, cuatro salidas digitales a relé y una salida analógica de 0 a 10 Vdc y está diseñada para ser la interfaz para la comunicación entre una PC y el variador de velocidad, instalado en el motor del “Grid Picker”, en la banda #1 de la máquina empastadora.

Lo que se desea es que este módulo sea capaz de realizar la adquisición de los datos analógicos y digitales generados por un variador de velocidad cualquiera, enviarlos a la computadora a

través del puerto serial para que sean procesados, visualizados y luego establecer algún tipo de control en las salidas, que serán las que actuarán en las entradas del variador.

4.1.2 Diagrama de Bloques del Sistema de Adquisición.

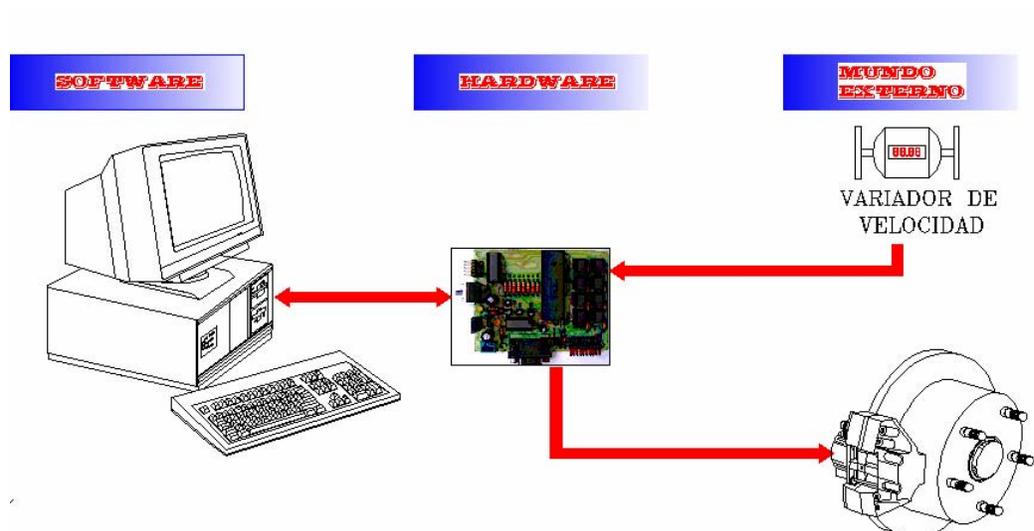


FIGURA 4.1 Diagrama de bloques del sistema

La figura 4.1 nos muestra el diagrama de bloques en el que se muestran las diferentes etapas de las que está constituido

nuestro proyecto, la misma que consta de cuatro bloques principales que se detallan a continuación:

Nuestro primer bloque es el del software que está desarrollado bajo la plataforma del LabVIEW 7.1 – refiérase al CAPITULO 5 - el objetivo de este bloque es que sea capaz de comunicarse con el Hardware (la tarjeta de adquisición), para transferir y recibir datos, además de tomar ciertas decisiones sobre el comportamiento del hardware.

El segundo bloque está constituido por el hardware necesario para que haya comunicación de la computadora con los bloques siguientes, en otras palabras el hardware es el enlace físico necesario para poder realizar la comunicación entre el primer y el tercer bloque. Para que nuestra tarjeta se comunique con el mundo externo tiene tres entradas digitales y tres entradas analógicas además de cuatro salidas digitales y una salida analógica.

Nuestro tercer bloque está compuesto por un variador de velocidad, que en nuestro caso es el mundo externo, las

características del variador de velocidad actualmente instalado en la planta se encuentran detalladas en el Anexo C-1, aunque por el diseño nuestra tarjeta puede acoplarse sin ningún problema a la mayor parte de los variadores existentes en el mercado

Finalmente el cuarto bloque esta constituido por el elemento que es controlado por el variador, es un motor de 3 Hp. Trifásico, para mayores detalles refiérase al Anexo C-2. A este motor consideramos necesario el acople de un sistema de realimentación hacia la tarjeta que nos permita visualizar la velocidad del mismo en tiempo real.

4.2 Diseño del Hardware.

El objetivo de este diseño es hacer una tarjeta de adquisición de datos basada en un microcontrolador de tres entradas analógicas, tres entradas digitales, tres salidas digitales y una salida analógica.

4.2.1 Características Técnicas de la Tarjeta.

4.2.1.1 Entradas

Consta de tres entradas digitales que corresponden al puerto E, ubicados en los pines 8, 9 y 10, está configurado para que cumpla con las funciones de RE0, RE1 Y RE2 respectivamente del PIC 16F877A, el voltaje de polarización de las entradas digitales es de 5 Vdc, podría ser diferente ya que estas entradas están aisladas por opto-acopladores, también consta de tres entradas analógicas que corresponden al puerto A, ubicados en los pines 2 al 4 que están configurados para que cumplan las funciones RA0, RA1 Y RA2 respectivamente, con un rango de trabajo de 0 a 5 Vdc.

4.2.1.2 Salidas.

Las salidas digitales constan de cuatro relés de 12 Vdc, que se encuentran ubicadas en el puerto B, configurados como salidas digitales, desde RB0 a RB3, los mismos que se encuentran ubicados desde los pines 33 al 36 del PIC correspondiente, las salidas digitales se encuentran aisladas del relé por medio del

integrado ULN2003AN -refiérase al anexo D1.1-. Este aislamiento es necesario ya que las corrientes que manejan las salidas del PIC son muy bajas para polarizar los relés, 10A a 120Vac y 6A 28Vdc. La salida analógica se la obtiene a través de un conversor digital analógico TLC7628CN -refiérase al anexo D1.2-. Este integrado convierte la información enviada en forma digital desde el puerto D del PIC el mismo que se encuentra configurado como salida digital. El puerto D está ubicado en los pines del 19 al 22 y del 27 al 30, es decir que nuestra salida analógica tiene 8 bits de resolución. Para obtener un voltaje de salida de 0 a 10 Vdc fue necesario utilizar el integrado TL084 -refiérase al anexo D1.3.-, configurado como acoplador de impedancia y como amplificador con ganancia 2.

4.2.1.3 Alimentación.

La alimentación de la tarjeta es de 120 Vac. En el interior de la tarjeta se han construido tres fuentes de +12, -12 y +5 Vdc, para obtener estos voltajes se

trabaja con los reguladores 7812, 7912 y 7805 respectivamente. La fuente de -12 Vdc fue necesaria para polarizar el integrado TL084. La fuente de +12 Vdc fue necesaria para polarizar a los integrados y a los relés mientras que la de +5 Vdc se la utilizó para polarizar el PIC.

4.2.2 Selección del Microcontrolador.

Para seleccionar el microcontrolador a usar se debió considerar los siguientes aspectos: cantidad de entradas y salidas, recursos internos que debería tener el micro, accesibilidad y precios. La capacidad mínima que debía tener el PIC era de 15 entradas y salidas digitales, 3 canales analógicos, capacidad de comunicación serial y conversores analógicos/digitales para el tratamiento de los datos analógicos. En el mercado actualmente los microcontroladores más usados son los de la familia **MICROCHIP**, por esta razón el integrado que usamos es el PIC16F877A –refiérase al anexo D1.4-.

4.2.3 Diagrama de Bloques de la Tarjeta de Adquisición.

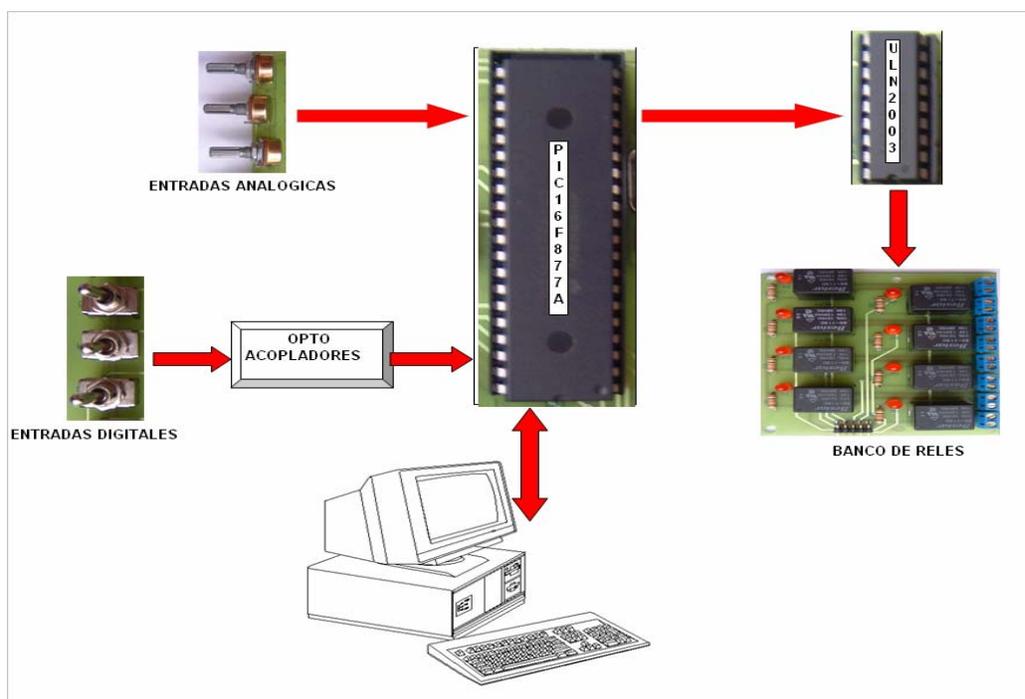


FIGURA 4.2 Diagrama de bloques de la tarjeta

La figura 4.2 nos muestra como está dividida la tarjeta de adquisición. El bloque central está constituido por el PIC 16F877A. El bloque de las entradas digitales está representado por tres botoneras, el bloque de opto-acopladores está constituido por tres acopladores tipo 4N25 y uno 4432 –refiérase al anexo D1.5-. El bloque de las entradas analógicas está

representado por tres potenciómetros que varían de 0 a 5 Vdc, se encuentran conectado al bloque central a través del OPA4342 que es un OPAM configurado en modo acoplador de impedancia o ganancia unitaria –refiérase al anexo D1.6-. El bloque de comunicación entre el computador y el PIC se logra mediante el USART incluido en el PIC y el integrado MAX232 –refiérase al anexo D1.7- que se encarga de convertir las señales TTL a niveles de voltaje establecidos por el protocolo RS232, (para más información acerca del RS232 refiérase al capítulo 5).

El bloque de acoplamiento de las salidas digitales lo constituye el integrado ULN2003 que contiene 8 salidas tipo DARLINGTON, para manejar el bloque de salida compuesto por 4 relés con las características mencionadas anteriormente. El bloque del convertidor digital analógico está compuesto por TLC6528, este bloque se encarga de convertir la información digital proveniente del bloque central en una señal analógica, esta señal analógica pasa por el bloque conformado por TL084 que es el que se encarga de convertir la señal analógica a niveles de voltajes deseados en el bloque de salida analógica.

4.2.4 Diseño de la tarjeta en PROTEL.

Para el diseño de la placa se consideró realizar el revelado de las pistas en una sola capa, aprovechando todos los espacios de tal manera que se puedan reducir costos al máximo. El diseño final de las pistas fue el que se muestra a continuación,

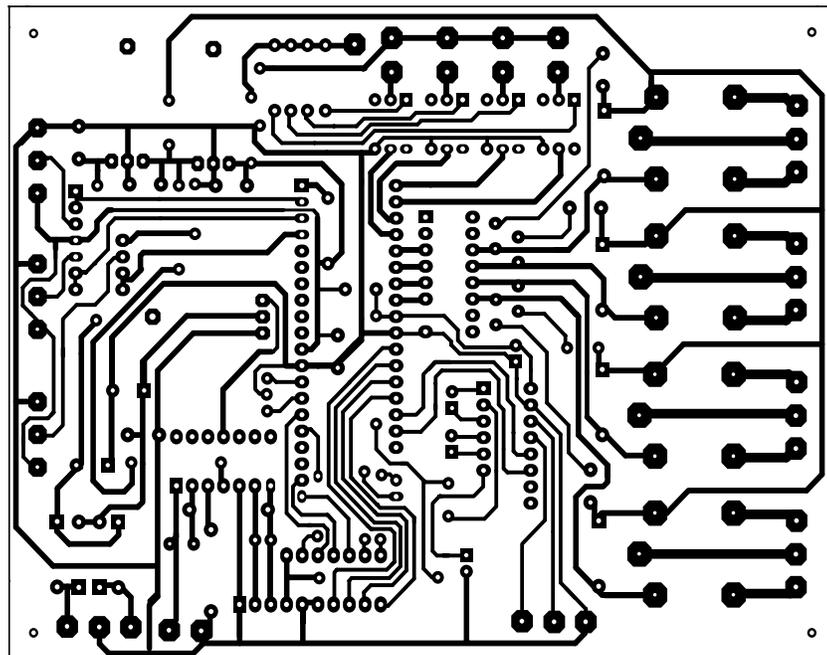


FIGURA 4.3 Circuito Impreso de las pistas de la Tarjeta.

Todas las líneas de estos puertos son multifuncionales, es decir realizan diversas funciones según estén programadas. Sin embargo, todos ellos tienen la capacidad de trabajar como líneas de E/S digitales, a continuación se mencionarán los puertos utilizados del PIC, la forma en que debe estar configurado cada uno de los puertos y para que se los utilizó:

PUERTO A:

Sólo dispone de 6 líneas, denominados RA0-RA5. Son bidireccionales y su sentido queda configurado, según la programación de los bits del registro TRISA. Para nuestro diseño se utilizó las líneas **RA0, RA1, RA2**, como **Entradas Analógicas**.

Por lo tanto el registro TRISA debe de ser configurado de tal manera que los pines mencionados anteriormente estén configurados como entradas, es decir si los bits 0, 1, 2 del registro TRISA se ponen a 1, la línea 0-1-2 (RA0, RA1 y RA2) del puerto A funcionan como entrada por lo que se puede guardar el valor h 'FF' en TRISA.

PUERTO E:

Ocupa la dirección 09h y sólo la tienen los PIC16F877A con 40 patitas. Dispone de 3 patitas multifunción, que se las configuran como entrada o salida, según el valor de los bits de menos peso del registro TRISE, que está ubicado en la dirección 89h. Para nuestro diseño se utilizaron las líneas **RE0, RE1 y RE2** de los pines 8-9-10 del microcontrolador como **entradas digitales**

Las líneas mencionadas trabajarán como entradas digitales que necesitamos para la programación de la tarjeta de adquisición de datos por lo que deberá estar configurado como entrada el registro TRISE con el valor de h'FF'.

PUERTO B:

Dispone de 8 líneas bidireccionales cuya función se elige mediante la programación del registro TRISB, igual que sucedía en el puerto A con TRISA, quedando configuradas como entradas las líneas RB<3:0> y RB<7:6>, mientras que RB<5:4> como salidas. Para

nuestro diseño se utilizó las líneas **RB0, RB1, RB2, RB3**, como **salidas digitales**.

Las líneas mencionadas trabajaran como salidas digitales que necesitamos para la programación de nuestro sistema por lo que deberá estar configurado como salida el registro TRISB con el valor de h'F0'.

PUERTO D:

Este puerto de 8 líneas bidireccionales sólo lo tienen los PIC16F877A de 40 patitas. Ocupa la dirección 08h, mientras que su registro de configuración TRISD ocupa la dirección 88h. Todas las patitas disponen en su entrada de un Trigger Schmitt. Para nuestro sistema se utilizó el puerto D con los pines 19-22 y del 27-30 para obtener una **salida analógica**, la cuál es obtenida a través de un conversor digital/analógico enviada en forma digital desde el puerto D, el mismo que se encuentra configurado como salida digital.

PUERTO C:

Consta de 8 líneas bidireccionales cuyo sentido se configura mediante el registro TRISC. Para nuestro

sistema hemos configurado la línea RC7/RX como entrada, mientras que RC6/TX como salida al igual que RC5, para el control de DCA se utilizó las líneas RC0,RC1 y RC2. Todo este sistema fue configurado para obtener una comunicación serial en donde necesitamos que el pin Tx que será para la transmisión de los datos esté configurado como salida y el pin Rx que será utilizado para la recepción de los datos por el puerto serie esté configurado como entrada por lo que el registro TRISC deberá guardar el valor de h'A8.

4.2.5.2 Módulo de comunicación serial.

Los PIC16F877A contienen un módulo MSSP con dos puertas para comunicación serie <síncrona>, o sea, con señal de reloj, Además, también disponen de un modulo USART capaz de soportar la comunicación serie síncrona y asíncrona. El USART, llamado SCI (Serial Comunicación Interface), puede funcionar como un sistema de comunicación full duplex o bidireccional asíncrono, adaptándose a multitud de periféricos y dispositivos que transfieren información de esta forma, tales como el monitor CRT o el ordenador

PC. También puede trabajar en modo síncrono unidireccional o half duplex para soportar periféricos como memorias, conversores, etc. En resumen, el USART puede trabajar de tres maneras:

1.- ASINCRONA (Full duplex, bidireccional).

2.- SINCRONA – MAESTRO (Half duplex, unidireccional).

3.-SINCRONA – ESCLAVO (Half duplex, unidireccional).

Las transferencias de información se realizan sobre dos líneas TX. (Transmisión) y RX (Recepción), saliendo y entrando los bits por dichas líneas al ritmo de una frecuencia controlada internamente por el USART. En el modo síncrono, la comunicación se realiza sobre dos líneas, la DT que traslada en los dos sentidos los bits a la frecuencia de los impulsos de reloj que salen por la línea CK desde el maestro. En ambos modos las líneas de comunicación son las dos de más peso de la puerta C: RC6/TX/CK y RC7/RX/DT.

Los cuatro bloques que configuran la arquitectura del USART, en modo asíncrono son:

- 1.- Circuito de muestreo.
- 2.- Generador de Baudios.
- 3.- Transmisor asíncrono.
- 4.- Receptor asíncrono.

4.2.5.3 Generador de Baudios.

En el protocolo asíncrono RS-232-C, la frecuencia en baudios (bits por segundo) a la que se realiza la transferencia se debe efectuar a un valor normalizado: 330, 600, 1.200, 2.400, 4.800, 9.600, 19.200, 38.400, etc. Para generar esta frecuencia, el USART dispone de un Generador de Frecuencias en Baudios, BRG, cuyo valor es controlado por el contenido grabado en el registro SPBRG.

Además del valor X cargado en el registro SPBRG, la frecuencia en baudios del generador depende del bit BRGH del registro TXSTA *2*.

En el caso de que BRGH sea 0 se trabaja en baja velocidad y si BRGH=1 se trabaja en alta velocidad.

Según este bit se obtendrá el valor de una constante K

necesaria en la determinación de la frecuencia de funcionamiento, expresada en la ecuación 4.1.

$$\text{Frecuencia en Baudios} = \frac{\text{Fosc}}{(K*(X+1))}$$

Donde **X** es el valor cargado en el registro SPBRG.

Si **BRGH=0**, baja velocidad y **K=64** modo asíncrono

Si **BRGH=1**, alta velocidad y **K=16** modo asíncrono

En nuestro diseño para lograr una perfecta comunicación serial, lo primero que se hizo fue establecer la velocidad de transmisión, para comunicarnos con nuestra tarjeta, como segundo paso se procedió a configurar todos los parámetros en el USART del microprocesador. Como ya se explicó anteriormente la frecuencia de los baudios depende del bit BRGH del registro TXSTA<2>, con el que se puede elegir la velocidad alta (1) o baja (0) en el modo asíncrono.

El valor con el que se debe cargar el registro SPBRG esta dado por la ecuación 4.2.

$$X = \left[\frac{F_{osc}}{(K, \text{Frecuencia Baudios})} \right] - 1$$

Para nuestro sistema se requiere calcular el valor que va hacer cargado en el registro SPBRG, en el cuál está funcionando con una frecuencia de oscilación de 4 MHz, trabajando con el USART en modo asíncrono y baja velocidad, con una frecuencia de 9600 baudios, lo cuál lo remplazamos en la ecuación 4.2 en donde tenemos que :

Para alta velocidad $K = 16$

$$x = \left(\frac{4MHz}{16 \cdot 9600} \right) - 1 = 25.04$$

Para baja velocidad $K = 64$

$$x = \left(\frac{4MHz}{64 \cdot 9600} \right) - 1 = 5.51$$

Si se carga el registro SPBRG con 25 la frecuencia real de trabajo será:

$$\text{Frecuencia en Baudios} = \frac{4000000}{16 \cdot (25+1)}$$

$$\text{Frecuencia en Baudios} = 9615 \text{ baudios}$$

Por lo tanto el error que se origina cargando 25 en el SPBRG será:

$$\text{ERROR} = \frac{(9615-9600)}{9600}$$

$$\text{ERROR} = 0,16\%$$

Analizando los dos valores de BRGH escogemos el valor que esté más próximo al entero, por consiguiente el valor que deberá estar guardado en el registro SPBRG al momento de configurar el USART deberá ser 25.

Existen dos registros más que son los del estado de los datos recibidos y el estado de los datos transmitidos. A continuación explicaremos como se transmite un dato por el USART.

4.2.5.4 Transmisor Asíncrono.

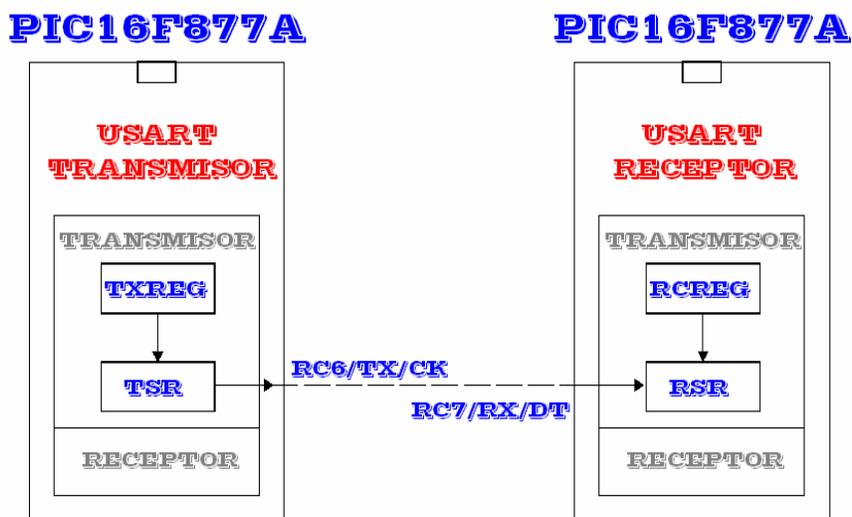


FIGURA 4.5 Comunicación entre el USART transmisor y otro receptor.

En la figura 4.5, se muestra un esquema simplificado de la comunicación entre USART transmisor y otro receptor, contenidos en sus receptivos microcontroladores. El dato que se desea transmitir por el USART transmisor de la figura 4.5 se deposita en el registro de desplazamientos TSR, que va sacando los bits secuencialmente y a la frecuencia establecida. Además, antes de los bits del dato de información incluye un bit de **INICIO** y después de sacar todos los

bits añade un bit de **PARADA**. El USART receptor recibe, uno a uno, los bits, elimina los dos de control y los de información una vez que han llenado el registro del desplazamiento RSR los traslada automáticamente al registro RCREG, donde quedan disponibles para su posterior procesamiento. En nuestro diseño la comunicación de transmisión es de 8 bits, 1 bit de paro y sin bit de paridad, la configuración de las funciones de cada bit del registro TXSTA para el control de la sección transmisora del USART se los detalla a continuación en la tabla 4.1:

BIT 7	CSRC=0	Bit de selección del reloj. Modo asíncrono
BIT 6	TX9=0	Selecciona Transmisión de 8 bits Habilita el bit 9 de transmisión.
BIT 5	TXEN=1	Activa Transmisión. Transmisión Activada.
BIT 4	SYNC=0	Bit de selección del modo del USART. Modo asíncrono
BIT 3		No Implementado.
BIT 2	BRGH=1	Bit de selección de la velocidad de baudios. Alta Velocidad-Modo asíncrono
BIT 1	TRMT=0	Bit de estado del registro de desplazamiento de transmisión TSR no vacío.
BIT 0	TX9D=0	Bit 9 del dato a transmitir Puede ser el bit de Paridad.

TABLA 4.1: Funciones TXSTA: Dirección (98h)

Conociendo cada una de las funciones de cada bit del registro TXSTA podemos comprender mejor el proceso de transmisión y de comunicación para el control de la sección transmisora del USART. Con los datos mencionados anteriormente en el registro TXSTA se deberá guardar el valor de h'A4' o b'10100100'.

4.2.5.5 Receptor Asíncrono.

Los datos se reciben en serie, bit a bit, por la patita RC7/RX/DT y se van introduciendo secuencialmente en el registro desplazamiento RSR.

Funciona a una frecuencia 16 veces más rápida que la de trabajo. Cuando el dato consta de 9 bits hay que programar el bit RX9=1 y el noveno bit de información se colocará en el bit RX9D del registro RCSTA. A continuación detallamos las funciones de cada bit del registro RCSTA en la tabla 4.2.

BIT 7	SPEN=1	Habilitación del puerto serie. Se configuran RC7/RX/DT y RC6/TX/CK.
BIT 6	RX9=0	Selecciona Recepción de 8 bits Habilita el bit 9 de Recepción.
BIT 5	SREN=0	Configura la recepción sencilla. Modo asíncrono
BIT 4	CREN=1	Configura la recepción continua. Modo asíncrono
BIT 3	ADDEN=0	Detección de dirección.
BIT 2	FERR=0	Bit de error de trama. No hay error de trama.
BIT 1	OERR=0	Bit de error de sobrepasamiento. No hay error de sobrepasamiento.
BIT 0	RX9D=0	Bit 9 del dato Recibido. Puede ser el bit de Paridad.

TABLA 4.2: Funciones RCSTA: Dirección (18h)

Conociendo las funciones de cada bit del registro RCSTA se debe guardar el valor h'90' o b'10010000' con lo cual el USART del microcontrolador queda totalmente listo para iniciar la transmisión o recepción de los datos por el puerto serie; lo único que quedaría es escribir el dato que queremos enviar en el registro TXREG o leer el dato en el registro RCREG si lo que queremos es recibir datos por el puerto serie.

4.2.5.6 Convertidor analógico-digital.

Los microcontroladores PIC16F877A poseen un conversor A/D de 10 bits de resolución y 5 canales de entrada en los modelos con 28 patitas (PIC16F873/6) y 8 canales en los que tienen 40 patitas (PIC16F874/7).

La resolución que tiene cada bit procedente de la conversión tiene un valor que está función de la tensión de referencia V_{ref} , de acuerdo con la fórmula siguiente:

La tensión de referencia determina los límites máximos y mínimos de la tensión analógica que se puede convertir. El voltaje diferencial mínimo es de 2V.

El conversor A/D es el único dispositivo que puede funcionar en modo reposo (SLEEP), para ello el reloj del conversor deberá conectarse al oscilador RC interno.

REGISTRO DE TRABAJO

El funcionamiento del conversor A/D requiere la manipulación de cuatro registros:

1. ADRESH: parte alta del resultado de la conversión.
2. ADRESL: parte baja del resultado de la conversión.
3. ADCON0: registro de control 0.
4. ADCON1: registro de control 1.

En la pareja de registros ADRESH: ADRESL, se deposita el resultado de la conversión, que al estar compuesta por 10 bits, solo son significativas 10 de los bits de dicha pareja.

El registro ADCON0 controla la operación del C A/D, mientras que el ADCON1 sirve para configurar las patitas de la puerta A como entradas analógicas o E/S digitales:

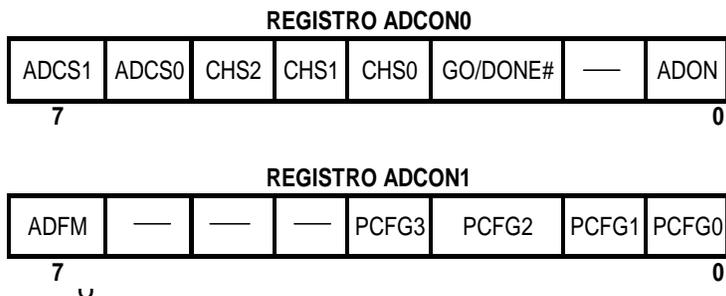


TABLA 4.3: Registros ADCON0-ADCON1.

Bits $ADCON0\langle 7:6 \rangle$ sirven para seleccionar la frecuencia de reloj que se emplea en la conversión, con la siguiente tabla 4.4 mostraremos las diferentes frecuencias de asignación:

ADCS1:ADCS0	FRECUENCIA
0	$F_{osc}/2$
1	$F_{osc}/8$
10	$F_{osc}/32$
11	Frc (Procede del oscilador interno).

TABLA 4.4: Frecuencias de conversión.

Se designa como TAD el tiempo que dura la conversión de cada bit y en el caso de trabajar con valores digitales de 10 bits, se requiere un tiempo mínimo de $12 \times TAD$. El valor de TDA se selecciona por software mediante estos bits (ADCS1:ADCS0) y en los

PIC16F877A nunca debe ser menor de 1.6 microsegundos.

En nuestro diseño escogemos una frecuencia igual a $F_{osc}/8$ y teniendo un reloj de 4 MHz tenemos que el período de conversión es de $2\mu s$ que es mayor al mínimo período requerido de $1.6\mu s$ establecido en la hoja de especificaciones del microcontrolador.

ADCS1:ADCS0	TDA
0	2.Tosc
1	8.Tosc
10	32.Tosc
11	Oscilador RC interno en el C A/D.

TABLA 4.5: Valores de TDA según los bits de los registros ADCS1-ADCS0.

En la tabla mostrada se presentan los valores que toma TDA según los bits ADCS1:ADCS0 y la configuración de trabajo del microcontrolador.

Los bits CHS2-0 seleccionan el canal por el que se introduce la señal analógica a convertir.

El bit GO/DONE# es el bit de estado de la conversión. Poniéndolo a 1 se inicia la conversión y mientras este en 1 está realizando dicha operación. Cuando GO/DONE# pasa a 0 confirma el final de la conversión y la puesta del resultado en la pareja de registros ADRESH: L.

El bit ADON sirve para activar el C A/D poniéndolo a 1 y para inhibir su funcionamiento poniéndolo a 0.

El bit de menos peso (ADFM) del registro ADCON1 selecciona el formato del resultado de la conversión. Si vale 1, el resultado está justificado en el registro ADRESH, que tiene sus 6 bits de más peso a 0; mientras que si vale 0 la justificación se realiza sobre el registro ADRESL, que tiene sus 6 bits de menos peso a 0.

Los restantes cuatro bits (PCFG3-0) de ADCON1 se usan para configurar las patitas de los canales de entrada al convertidor como analógicos o como E/S digitales, para nuestro diseño empleamos tres entradas

analógicas y tres entradas digitales, explicadas anteriormente.

4.3 Diseño del Software para el PIC.

4.3.1 Introducción.

Una de las partes más complicadas e interesante del desarrollo de la tarjeta, fue realizar el programa que iba hacer cargado en el PIC, el principal problema a resolver era establecer un protocolo de comunicación entre el software desarrollado en LabVIEW (refiérase al capítulo 5), y la tarjeta de adquisición, ya que si bien es cierto la información iba hacer transmitida a través del puerto serie en código ASCII, era necesario diferenciar los tipos de señales.

Para las salidas digitales se implementó el siguiente código:
Cada vez que se envíe una letra ente el 1-8, desde la PC, el

PIC activará o desactivará una de las cuatro salidas digitales, como se muestra en la tabla 4.6.

Envío desde PC	Respuesta del PIC
1	Enciende Salida # 1
2	Enciende Salida # 2
3	Enciende Salida # 3
4	Enciende Salida # 4
5	Apaga Salida # 1
6	Apaga Salida # 2
7	Apaga Salida # 3
8	Apaga Salida # 4

TABLA 4.6 Comandos que recibe el PIC para actuar en las salidas digitales.

Tal como se muestra en la tabla 4.6 el funcionamiento es de la siguiente manera: Cuando la PC transmita un “1”, el PIC pondrá en “1” el pin programado como salida digital #1 (RB0), de la misma forma cuando la PC transmita un “5”, el PIC pondrá en “0” el pin programado como la salida digital #1 (RB0). De igual forma se trabaja la salida analógica como se especifica en la tabla 4.7, mostrada a continuación:

Envío desde PC	Respuesta del PIC
l	Decrementa Salida Analógica.
m	Incrementa Salida Analógica
f	Conserva el valor anterior.

Tabla 4.7 Comandos que recibe el PIC para actuar en la salida analógica.

Cada vez que el PIC recibe una “l” o “m” desde la PC, el PIC decrementa o incrementa la variable F_DIGITAL, luego envía este valor al puerto D para su conversión, después transmite este valor a la PC, para que sea visualizado. Si el valor recibido por el PIC es una f, el valor de F_DIGITAL no cambia.

El procesamiento de las entradas analógicas de la tarjeta es como se muestra en la tabla 4.8, que está detallada a continuación:

Envío desde PC	Respuesta del PIC
0	Envío del canal analógico 0
9	Envío del canal analógico 1
C	Envío del canal analógico 2

Tabla 4.8 Comandos que recibe el PIC para actuar en las entradas analógicas.

Cuando el PIC recibe uno de los valores de la tabla anterior, inicia la conversión del canal respectivo, luego transmite el valor del canal, hacia la PC, para una mayor facilidad de este proceso se trabajó con 8 bits de resolución de los canales analógicos.

4.3.2 Programación en MPLAB.

El programa en MPLAB que fue cargado en el PIC, está dividido en las siguientes rutinas:

1. El Programa Principal.
2. Servicios de Interrupción.
3. La conversión de los valores Analógicos a Digital.
4. Codificación e identificación de los valores recibidos desde la PC.
5. El envío de los datos a través del puerto serie a la PC.
6. Las rutinas de retardo.

En el anexo D1.5 se encuentra detallado todo el programa en lenguaje ASM con sus respectivos comentarios.

4.4 Implementación del sistema.

4.4.1 Análisis del sistema con el software de simulación Proteus.

De acuerdo a nuestro diseño se procedió a realizar la simulación mediante el software del programa de PROTEUS.

La ventaja de realizar esta simulación es que nos permite visualizar el desarrollo de nuestra tarjeta, para ir observando todas las posibles variantes para poder realizar una buena comunicación serial con el PIC.

Para la simulación del hardware se procedió a armar el circuito mostrado en la figura 4.6.

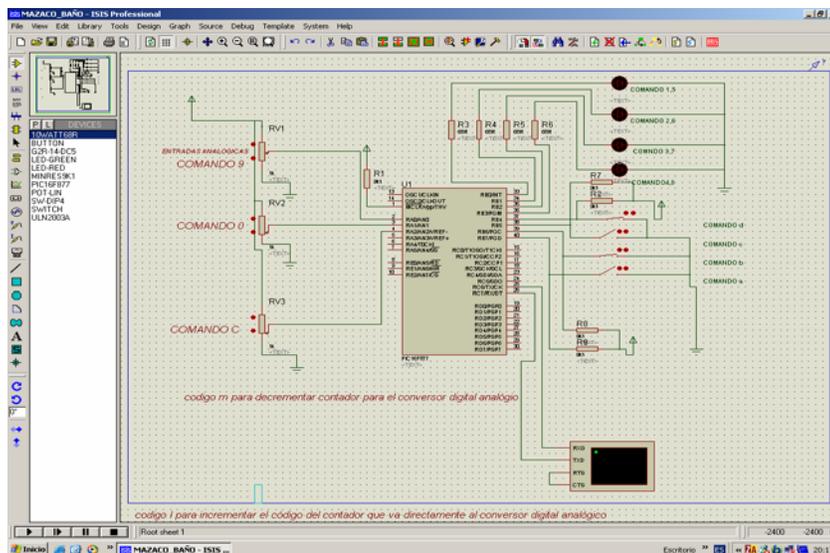


FIGURA 4.6 Esquemático de la tarjeta de adquisición de datos desarrollado mediante el software de Proteus.

4.4.2 Fotos de la Tarjeta de Adquisición de Datos.

Las siguientes fotos nos muestran todos los detalles de nuestra tarjeta de adquisición de datos, como por ejemplo sus cables de conexión, sus diferentes entradas y salidas digitales, analógicas y así como sus salidas a relé.

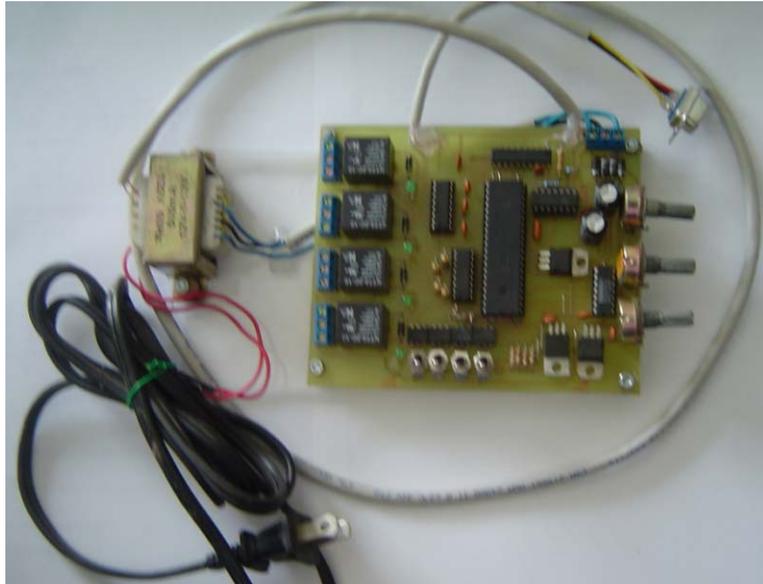


FIGURA 4.7 Vista superior de la tarjeta de adquisición.

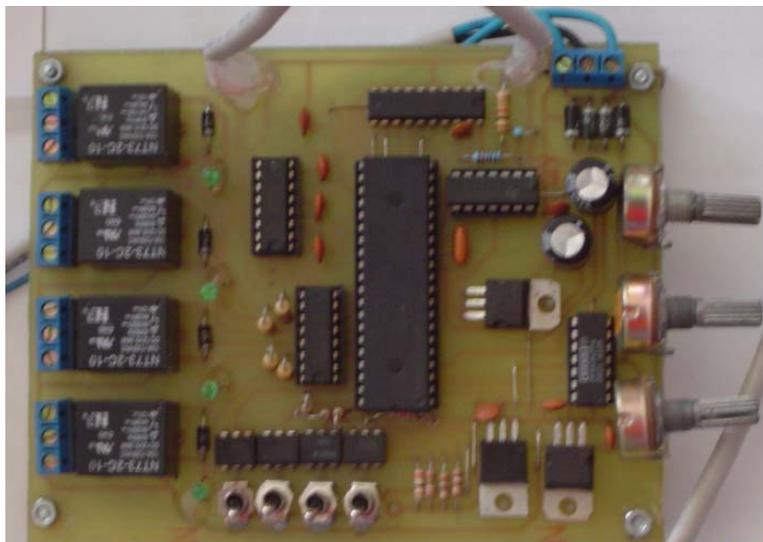


FIGURA 4.8 Fotografía general de las entradas y salidas digitales y analógicas.

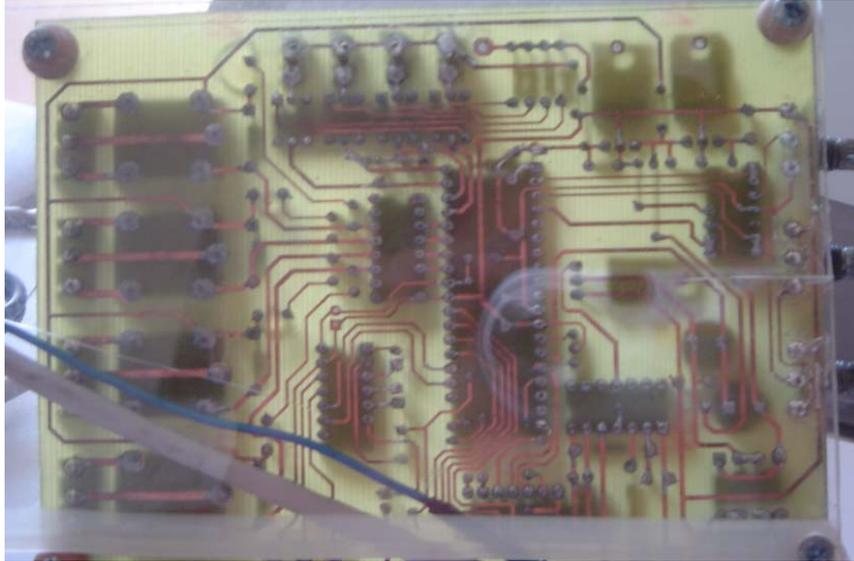


FIGURA 4.9 Fotografía general de las pistas de la tarjeta de adquisición de datos.

4.4.3 Costos.

Lista de los materiales utilizados y sus respectivos precios en el mercado local.

Enero 12 del 2006
 Proyecto: Tarjeta de Adquisición de Datos.

ITEM	DESCRIPCION	UN	CANT	P/UN	SUBTOTAL
LISTA DE MATERIALES PARA ARMAR LA TARJETA					
1	Resistenciasde 10k	U	6	0,05	0,30
2	Resistenciasde 1k	U	5	0,04	0,20
3	Resistenciasde 20k	U	2	0,12	0,24
4	Regulador LM 7812	U	1	0,75	0,75
5	Regulador LM 7805	U	1	0,75	0,75
6	Regulador LM 7912	U	1	0,85	0,85
7	CI 390	U	4	0,55	2,20
8	CI 40N25	U	4	0,70	2,80
9	CI ULN2003	U	1	1,50	1,50
10	CI MAX232	U	1	3,00	3,00
11	Capacitores 0,1uf	U	10	0,15	1,50
12	Capacitores 1000uf	U	2	0,50	1,00
13	Capacitores 10pf	U	1	0,30	0,30
14	4007	U	4	0,80	3,20
15	Diodo Emisor de luz	U	4	0,10	0,40
16	Relay 12 Vd-12A	U	4	1,50	6,00
17	Cristal de Cuarzo 4MHz.	U	1	0,80	0,80
18	CI TL084	U	1	1,50	1,50
19	CI TLC7528	U	1	1,60	1,60
20	PIC16F877A	U	1	9,00	9,00
21	1N5397	U	4	1,20	4,80
22	OPA4332	U	1	0,90	0,90
23	Potenciómetro de 10k	U	3	0,40	1,20
24	Cable de Comunicación	U	1	1,50	1,50
25	Borneras	U	3	0,20	0,60
26	Interruptores	U	4	0,30	1,20
27	Adaptadores 12Vdc fijo	U	2	4,25	8,50
28	Circuito Impreso	Gb	1	20,00	20,00
29	Zócalo de 40 pines	U	1	0,25	0,25
30	Zócalo de 18 pines	U	1	0,10	0,10

VALOR DEL PRESUPUESTO	USD 76,94
IVA 12%	USD 9,23
VALOR TOTAL DEL PRESUPUESTO	USD 86,17

Atentamente

 José Luis PARRALES Santistevan.

Tabla 4.9 Costos de los materiales utilizados en la tarjeta de adquisición de datos.

El valor total del proyecto como podemos observar en la tabla 4.9 ascendió a \$86.17, el diseño de esta tarjeta está hecho con el mejor material, una excelente placa lo que garantiza un mejor funcionamiento y una durabilidad muy buena al proyecto.

CAPITULO V

5. DISEÑO DEL PROGRAMA DE MONITOREO Y CONTROL DE UN VARIADOR DE VELOCIDAD A TRAVES DE LABVIEW 7.1 STUDENT EDITION

5.1 Características y herramientas de labview

LabVIEW es un programa de instrumentación virtual con un ambiente de desarrollo gráfico (utiliza un lenguaje de programación gráfico) utilizado en ciencias e ingeniería que incluye herramientas funcionales para simulación, adquisición de datos, análisis de mediciones y presentación y almacenamiento de datos. Usando LabVIEW usted puede crear sus propios instrumentos virtuales (VIs) y con esto obtener soluciones fáciles y eficientes a problemas comunes en el área industrial e investigativa, los instrumentos virtuales emulan el trabajo de un instrumento real.

Se habla de instrumento virtual cuando se emplea la computadora para realizar funciones de un instrumento clásico pudiendo agregarle al instrumento otras funciones a voluntad, a los programas desarrollados en LabVIEW se los conoce como instrumentos virtuales

Las principales características del ambiente de trabajo y de las aplicaciones desarrolladas en LabVIEW son las siguientes:

- Trabajo en dos paneles.- El desarrollo de un VI se realiza en 2 pantallas de trabajo llamadas Front Panel y Block Diagram íntimamente relacionadas; la primera es la pantalla de trabajo y son visibles los controles (representan las entradas) y los indicadores (representan las salidas), la segunda es la pantalla de programación y es donde se realiza la conexión y programación de los controles e indicadores. Para mayor información refiérase al Capítulo 2 del manual “Learning with LabVIEW 7 Express”
- Trabajos con SubVIs.- los SubVIs son instrumentos virtuales que se pueden usar como íconos dentro de otros instrumentos virtuales, utilizando SubVIs podemos hacer instrumentos virtuales complejos de una manera fácil y comprensible. Refiérase al Capítulo 2 del manual “Learning with LabVIEW 7 Express” para mas información acerca de cómo trabajar con SubVIs.
- Programación con estructuras y lazos.- Los lazos FOR y WHILE, la estructura CASE y las estructuras de secuencia FLAT y STACKED nos permiten gobernar el flujo de ejecución de un VI. Otras opciones que nos da LabVIEW son: “FORMULE NODE” que nos permite

implementar opciones matemáticas complejas. Además podemos ejecutar programas del tipo m-file usando el ícono MATLAB NODE". Para mas información acerca de las estructuras y nodos refiérase al Capítulo 5 del manual "Learning with LabVIEW 7 Express".

- Agrupación de Datos.- A través de los íconos "ARRAY" y "CLUSTER" LabVIEW nos permite agrupar datos ya sean del mismo tipo (arrays) o datos de diferentes tipos. Con LabVIEW podemos manipular la información agrupada en cualquiera de estas dos formas. Refiérase al Capítulo 6 del manual "Learning with LabVIEW 7 Express" para mas información.
- Visualización de Información.- Con LabVIEW podemos mostrar el estado de las variables con las que estamos trabajando -analógicas o digitales- en tiempo real, además de establecer un historial gráfico del comportamiento de las mismas a través del tiempo, esto lo podemos hacer mediante el uso de las opciones "CHART" y "GRAPHS". Para mas información refiérase al Capítulo 7 del manual "Learning with LabVIEW 7 Express".
- Adquisición de Datos.- LabVIEW nos permite utilizar varios drivers comunes tales como GPIB y RS232 para adquirir o transferir información al exterior, para esto LabVIEW cuenta con el "INSTREUMENT I/O ASSISTANT" que es una herramienta que nos permite configurar los drivers de comunicación antes mencionados y

otros. Otra forma de adquisición de datos con LabVIEW es utilizando las tarjetas de adquisición de “National Instruments”, una herramienta poderosa para la configuración de estas tarjetas es el “DAQ Assistant”. Para obtener mas información refiérase a los Capítulos 8 y 10 del manual “Learning with LabVIEW 7 Express” para mas información.

- Paneles Frontales Remotos.- Una nueva herramienta de LabVIEW 7.1 es la habilidad de tener un VI controlado y monitoreado remotamente desde un Web Browser. Esta tecnología le permite desarrollar un VI en una computadora y en pocos pasos ser capaz de ver y correr este VI desde otra maquina usando un Web Browser tal como Internet Explorer o Netscape.

5.2 Creación, Edición y Programación de instrumentos virtuales en LabVIEW 7.1.

5.2.1 Creación de Instrumentos Virtuales

Para la creación de un VI. Debemos abrir LabVIEW 7.1 Express y dar clic en crear un nuevo instrumento virtual como se muestra en la figura 5.1

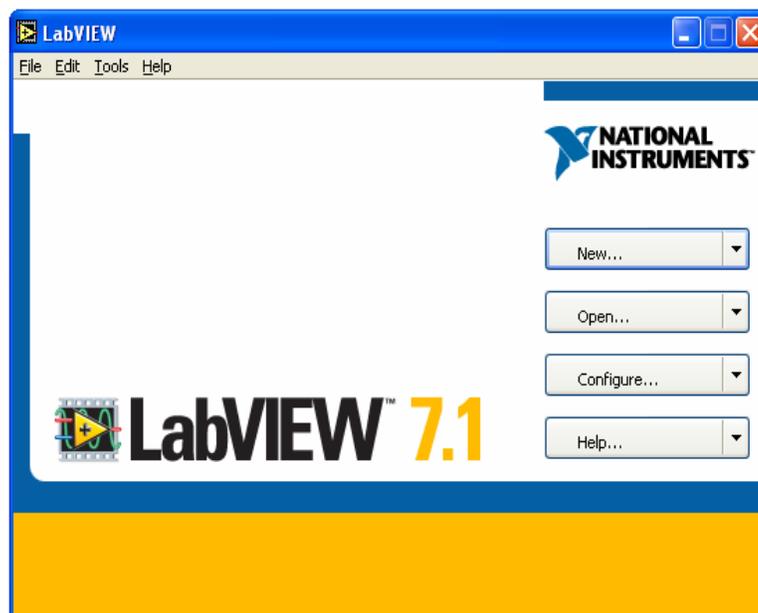


FIGURA 5.1 Pantalla Inicial de LabVIEW 7.1

Si lo que deseamos es crear un nuevo VI, debemos hacer clic en New y luego elegir la opción Blank VI.

Una vez realizado el paso anterior, aparecerán 2 paneles: Panel Frontal y Diagrama de Bloques que se puede ver en la figura 5.2.

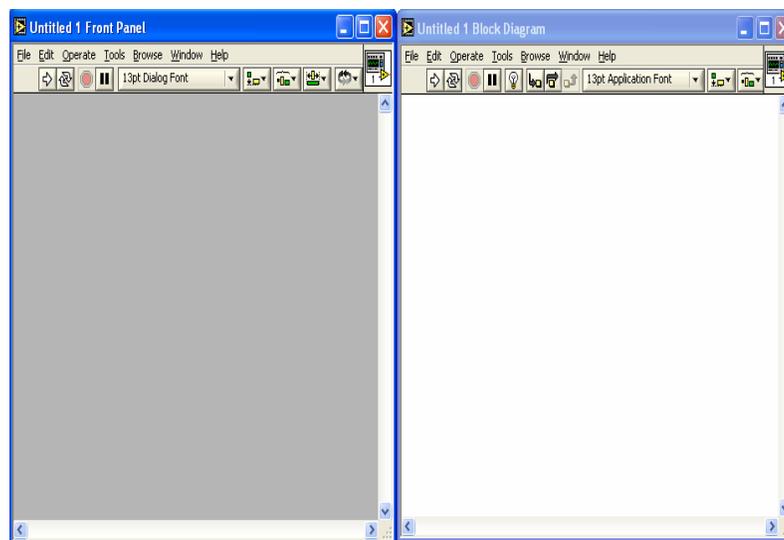


FIGURA 5.2 Panel Frontal y Diagrama de Bloques.

El panel frontal es la interfase del usuario con el VI. El panel frontal se debe construir con controles e indicadores, que son las entradas y salidas que interactúan con las terminales del VI, respectivamente. Los controles son switches, botoneras, marcadores y otros componentes de entradas. Los indicadores son los gráficos, luces y otros dispositivos.

En la figura 5.3, los controles simulan instrumentos de entradas de equipos y suministra datos al diagrama de bloques del VI. Los indicadores simulan salidas de instrumentos y suministra datos que el diagrama de bloques adquiere o genera.

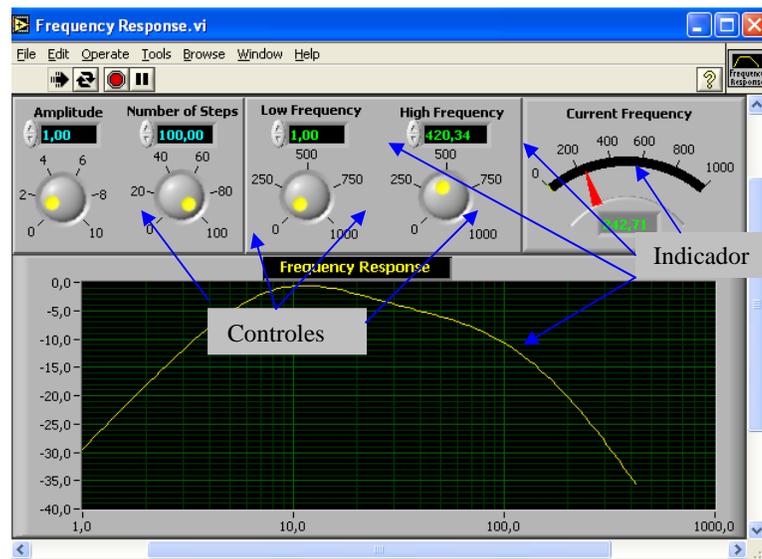


FIGURA 5.3 Panel Frontal de una respuesta de Frecuencia.

El panel frontal también contiene una barra de herramientas, y sus funciones.

El diagrama de bloque contiene el código fuente grafico. Los objetos del panel frontal aparecen como terminales en el diagrama de bloque. Adicionalmente, el diagrama de bloque contiene funciones y estructuras incorporadas en las bibliotecas de LabVIEW VI. Los cables conectan cada uno de los nodos en

el diagrama de bloques, incluyendo controles e indicadores, funciones y estructuras, como se puede apreciar en la figura 5.4.

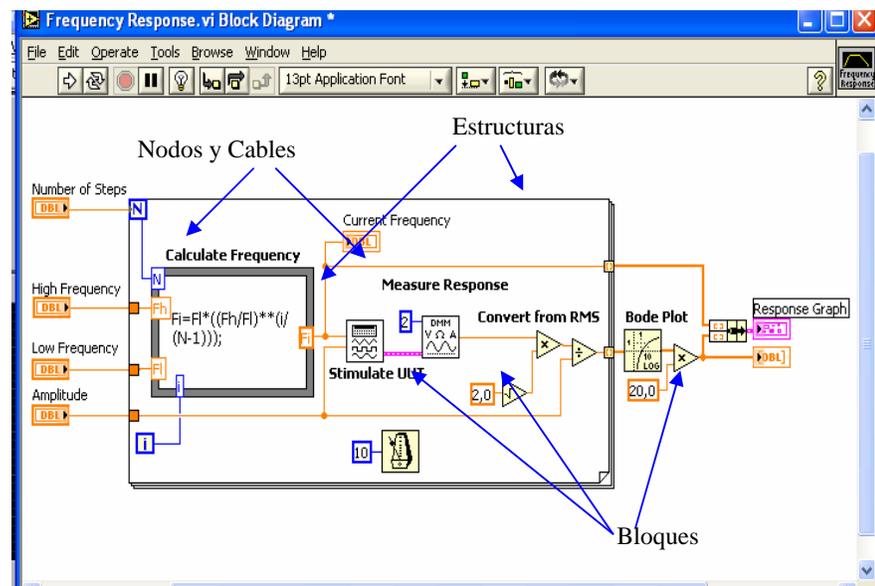


FIGURA 5.4 Diagrama de Bloques para el Panel Frontal de la figura 5.3.

En la creación del objeto virtual en el panel frontal, se escoge el objeto de la barra de herramientas que se observa en la figura 5.5

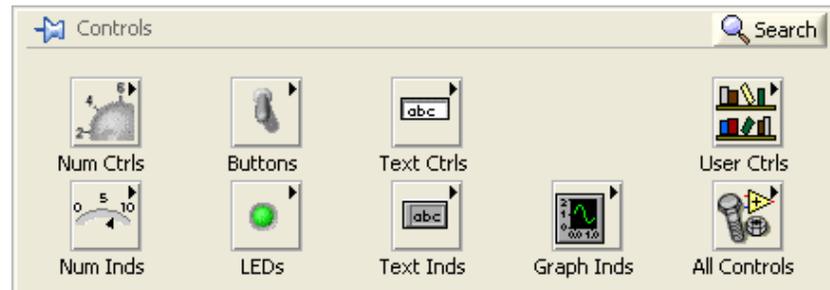


FIGURA 5.5 Barra de herramientas.

En la figura 5.6 se muestra cuando el objeto es una terminal y es creado en el diagrama de bloques. Estas terminales le dan acceso a los objetos del panel frontal del código creado mediante el diagrama de bloque.

Cada terminal contiene información útil referente al objeto al cual corresponde en el panel frontal. Por ejemplo, el color y los símbolos proporcionan el tipo de dato. Números de punto flotante y de doble-precisión, son representados con terminales anaranjadas y las letras DBL. Las terminales booleanas son verdes y son representadas por las letras TF (Verdad y Falso).

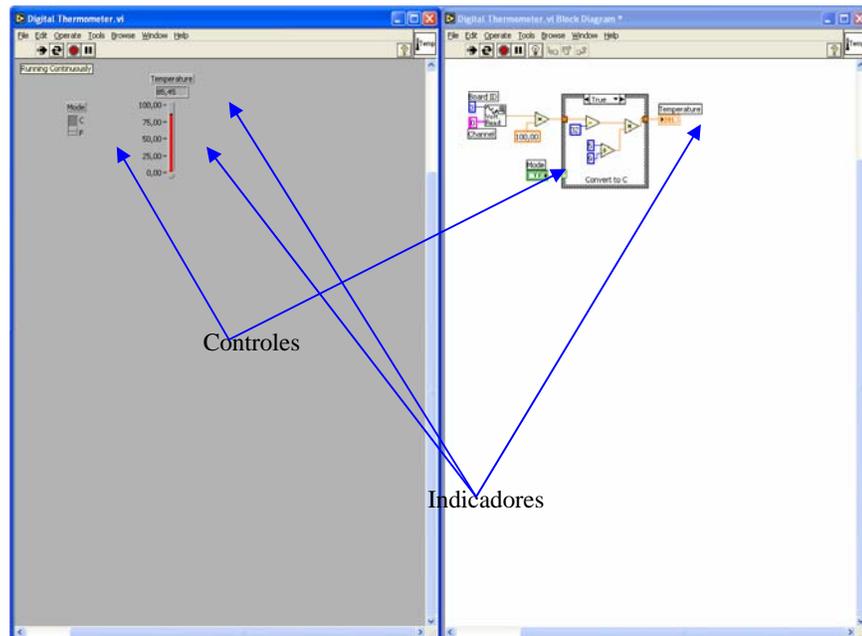


FIGURA 5.6 Representación de controles e indicadores en ambas pantallas.

En general, las terminales anaranjadas deben unirse (cablearse) con las terminales anaranjadas, verdes con verdes, y así sucesivamente. Esta no es una regla que no se pueda romper; por ejemplo LabVIEW permitirá al usuario conectar una terminal azul (valor entero) a una terminal anaranjada (valor fraccional). Pero en la mayoría de casos, busque mejor una igualdad en colores.

Los controles tienen una flecha en el lado derecho y tienen un borde grueso. Los indicadores tienen una flecha en el lado

izquierdo y un borde fino. Cada cable debe tener una fuente o control, y cada cable puede tener varios destinos o indicadores.

5.2.2 Edición de Instrumentos Virtuales

En la edición de instrumentos virtuales, LabVIEW tiene herramientas y menú para cambiar las propiedades por ejemplo el color, escala, texto, posición, alineación, tamaño, etc. Para cambiar el color del objeto y también el del panel frontal es de la siguiente manera:

En la figura 5.7 se muestra el submenú de Windows de la barra de herramientas del panel frontal, habilitar la opción de mostrar paletas de herramientas, escoger el color de preferencia y señalar con el pincel sobre el objeto a cambiar de color.

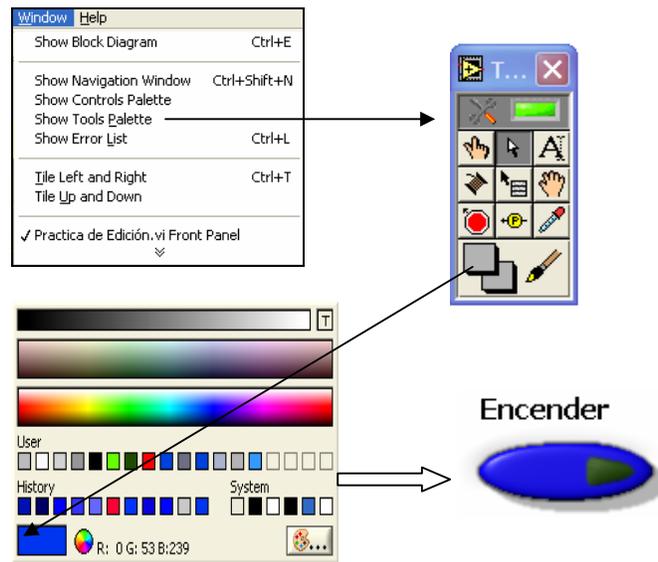


FIGURA 5.7 Cambio de color de un objeto en LabVIEW.

En el caso de insertar un texto, se señala la letra A de la paleta de herramientas y se escribe el texto del objeto, tal como se muestra en la figura 5.8.

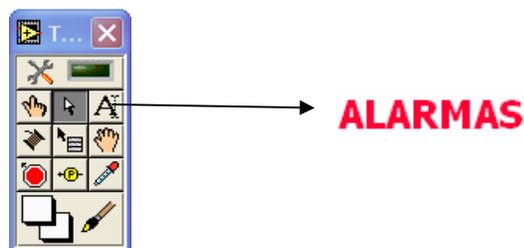


FIGURA 5.8 Editar un texto para un objeto en LabVIEW.

En el cambio de escala del objeto de control, se da un clic derecho con el mouse al objeto, y este muestra un menú con las

propiedades del mismo, y la escala de preferencia. La figura 5.9 muestra el procedimiento para cambiar la escala del objeto.

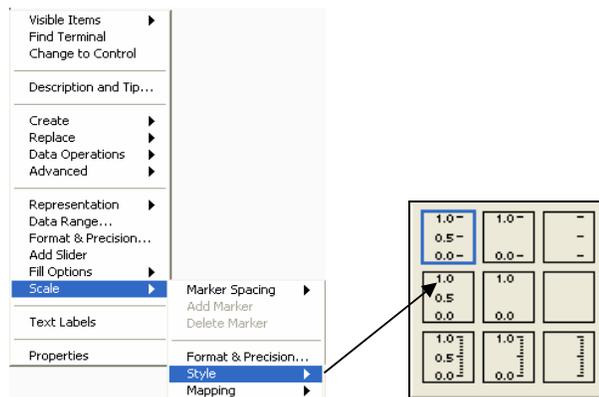


FIGURA 5.9 Cambio de Escala para un objeto en LabVIEW.

5.2.3 Programación de los objetos virtuales.

LabVIEW utiliza un lenguaje gráfico de programación para programar primero, se deben conocer los tipos de datos, la ejecución del flujo de datos, conocer los ciclos Mientras (While) y Para (For) y también estructuras de caso.

LabVIEW tiene una variada gama de tipos de datos. Cuando se pone un control o indicador en el panel frontal, LabView pone un terminal correspondiente a este control o indicador en el

diagrama en bloque. Este terminal desaparece solo cuando se borra el control o indicador. Los enlaces o alambres son los caminos de los datos entre los terminales fuente y los terminales destino. No se permite enlazar dos terminales fuentes ni dos terminales destinos, y si se permite enlazar un terminal fuente a varios terminales destino.

En La figura 5.10 se muestra la representación de los tipos de datos en LabVIEW.

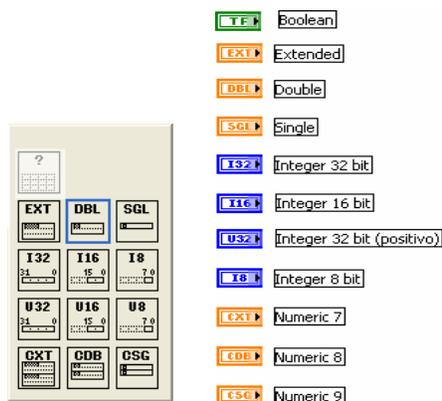


FIGURA 5.10 Tipos de datos en LabVIEW.

LabVIEW sigue un modelo de flujo de datos para ejecutar los VIs. Un nodo del diagrama de bloque se ejecuta cuando todas sus entradas están disponibles. Cuando un nodo completa la ejecución, suministra datos a sus terminales de salida y pasa los datos de salida al siguiente nodo en la trayectoria del flujo de datos.

En LabVIEW existen los ciclos Mientras (While) y Para (For) están localizados en la paleta "Functions Structures". El ciclo Para difiere del ciclo Mientras en que el ciclo Para se ejecuta una serie de veces predeterminada. El ciclo Mientras se deja de ejecutar solo si el valor en la condición terminal existe.

Ciclo Mientras.-Similar al ciclo Haga (Do) o al ciclo Repita-hasta (Repeat-Until) en lenguajes de programación basados en texto, un ciclo Mientras, ejecuta un sub diagrama hasta que la condición sea cumplida.

El ciclo Mientras ejecuta el sub diagrama hasta que la terminal dependiente, recibe un valor Booleano específico. El comportamiento y la apariencia de la terminal dependiente es Continue if True (continúe si es Verdadero), mostrado a la derecha de la figura 5.11, en el diagrama de bloques. La terminal de iteración (una terminal de salida), mostrada a la izquierda, contiene el número de iteraciones completas. El conteo de iteraciones siempre empieza en cero. Durante la primera iteración, la terminal de iteración regresa a cero.

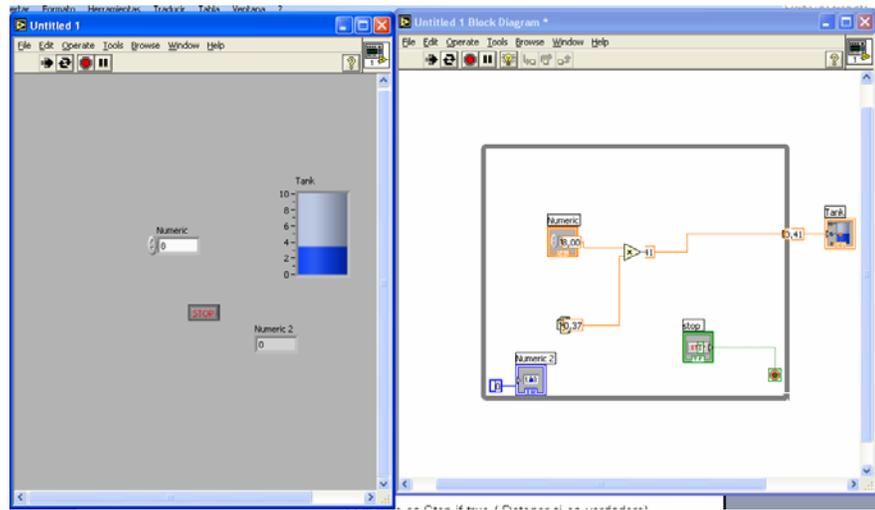


FIGURA 5.11 Ciclo Mientras (While) continua si es verdadero.

Si el ciclo es Stop if true (Detener si es verdadero) tiene la misma configuración del ciclo anterior, la única diferencia, es que el ciclo se detiene si la acción booleana es verdadero.

Ciclo For (Para).- Un ciclo For (Para), mostrado en la figura 5.12, ejecuta una serie varias veces. El valor en la terminal de conteo (una terminal de entrada) representada por la N, indica cuantas veces repetir el sub diagrama y para el caso de la figura 5.12 es de 99 (N-1). La terminal de iteración (una entrada de salida), contiene el número de iteraciones completas. El conteo de iteraciones siempre empieza en cero. Durante la primera iteración, la terminal de iteración regresa a cero.

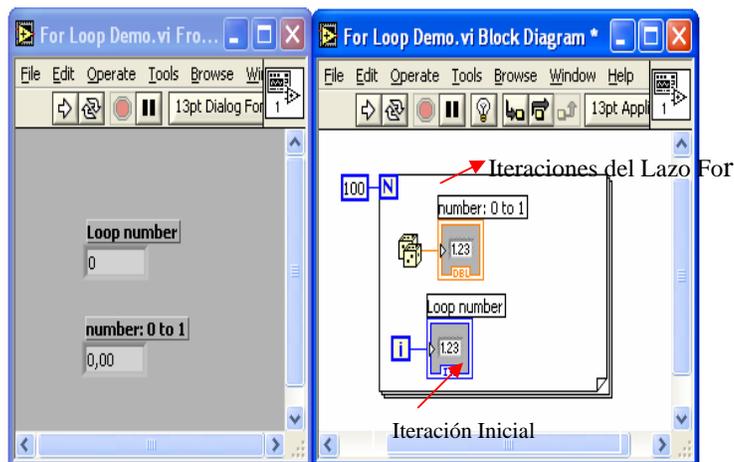


FIGURA 5.12 Ciclo For (Para).

5.3 Descripción del sistema utilizado para la comunicación entre LabVIEW y la tarjeta de adquisición diseñada.

5.3.1 Protocolos de comunicación de LabVIEW.

LabVIEW se comunica y es capaz de controlar instrumentos externos (como osciloscopios o multímetros) usando GPIB (General Purpose Interfase Bus), RS-232 (es un bus de interfase serial estándar), VXI (VME extensions for Instrumentation) entre otros hardware estándar. Para nuestros propósitos hemos hecho uso del puerto de comunicación serial (RS-232).

5.3.1.1 El Puerto de Comunicación RS 232.

Es una forma popular de comunicación entre computadoras o entre una computadora y un periférico. Cuando se lo utiliza con LabVIEW el puerto de comunicación solamente se lo puede conectar a un dispositivo (Diferente a cuando lo utilizamos en una red industrial en la que puede soportar varios dispositivos). Usando el puerto de comunicación serial usted puede enviar y recibir datos sin necesidad de tener instalado un hardware especial. La comunicación serial utiliza un solo canal para enviar datos (un bit a la vez) desde un Terminal (llamado transmisor) hasta otro (llamado receptor). Este método trabaja bien siempre que la velocidad de transferencia sea relativamente baja y cortas distancias.

Hay diferentes protocolos de comunicación serial, el mas conocido es el RS-232, desarrollado por la Asociación de Industrias Electrónicas (EIA) para especificar la interfase serial entre equipos conocidos como Data Terminal Equipment (DTE) -tales como módems o ploters- y otros conocidos como Data

Communications Equipment –tales como computadores y terminales-. El protocolo RS-232 incluyen las características de la señal y el voltaje, las características del conector e instrucciones para conexión Terminal-módem.

El conector para la comunicación por medio del protocolo RS-232 tiene dos presentaciones: el conector de 25 pines y el conector de 9 pines. El conector de 9 pines tiene dos líneas de datos (TxD y RxD) y 5 líneas adicionales para el manejo de la información (denotadas como RTS, CTS, DSR, DCD y DTR como se muestra en la figura 5.13)

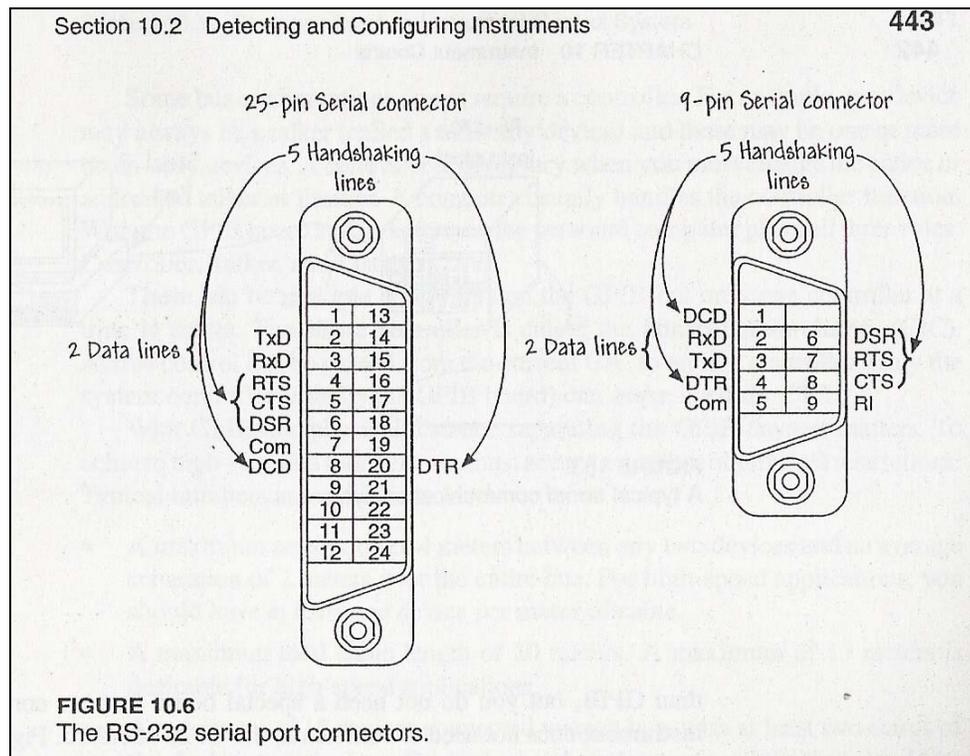


Figura 5.13. Conectores usados para la comunicación serial RS-232.

Para una comunicación serial se requiere especificar cuatro parámetros básicos:

- La velocidad de transmisión (baud rate).
- El número de bits de datos por carácter (data bits).
- El tipo de bit de paridad (parity bit) y
- El número de bits de parada (stop bits).

Cada carácter transmitido es empaquetado en una trama compuesta por un bit de inicio (start bit) seguido

por los bits de datos, el bit de paridad opcional y el o los bits de parada.

5.3.1.2 Instrument Drivers.

Un “Instrument Driver” es un VI que controla un instrumento en particular. Estos VI eliminan la necesidad de utilizar complicados comandos de bajo nivel para en el desarrollo del VI. Con un “Instrument Driver” nosotros podemos interactuar desde el front panel de nuestra aplicación con el hardware que nosotros hayamos construido.

LabVIEW nos ofrece varios VIs que pueden ser usados en la comunicación con el hardware estos VIs pueden ser agrupados en las siguientes categorías:

- Funciones Standard VISA I/O
- Funciones GPIB
- Puerto de comunicación serial.

El VI utilizado en nuestro caso es el VISA (Virtual Instrument Software Architecture), EL visa es un VI para controlar instrumentos GPIB, Serial o VXI y realizar las llamadas apropiadas dependiendo del tipo de instrumento, en nuestro caso el VISA controla al

puerto de comunicación serial. VISA es una aplicación de alto nivel que contiene códigos en bajo nivel para controlar nuestro hardware. Por medio de VISA podemos abrir, leer o escribir en el puerto de comunicación seleccionado. En la figura 5.14 se muestra el VI de inicialización del VISA el cual nos permite establecer los parámetros más importantes de una comunicación serial. (Para más información refiérase a la sección 10.4 del manual “Learning with LabVIEW 7 Express”).

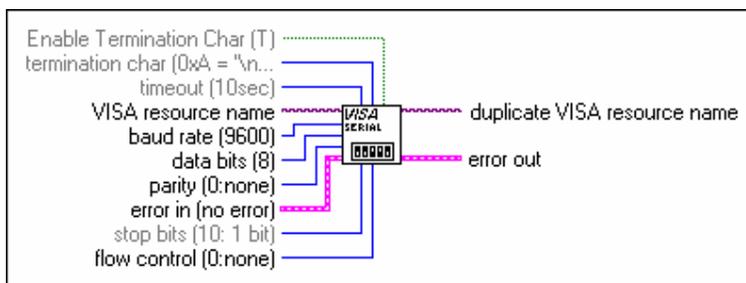


Figura 5.14 VI de inicialización del VISA.

Otros VI importantes son el VISA Read, que nos permite leer el puerto; el VISA Write, que nos permite escribir en el puerto y el VISA Close, que es utilizado para cerrar el puerto con el que trabajemos.

5.3.2 Diseño del Control de la Tarjeta de Adquisición en LabVIEW.

En la figura 5.15 se muestra el panel de control diseñado con el objeto de monitorear y controlar el variador de velocidad del motor “Grid Picker” descrito en los capítulos anteriores.

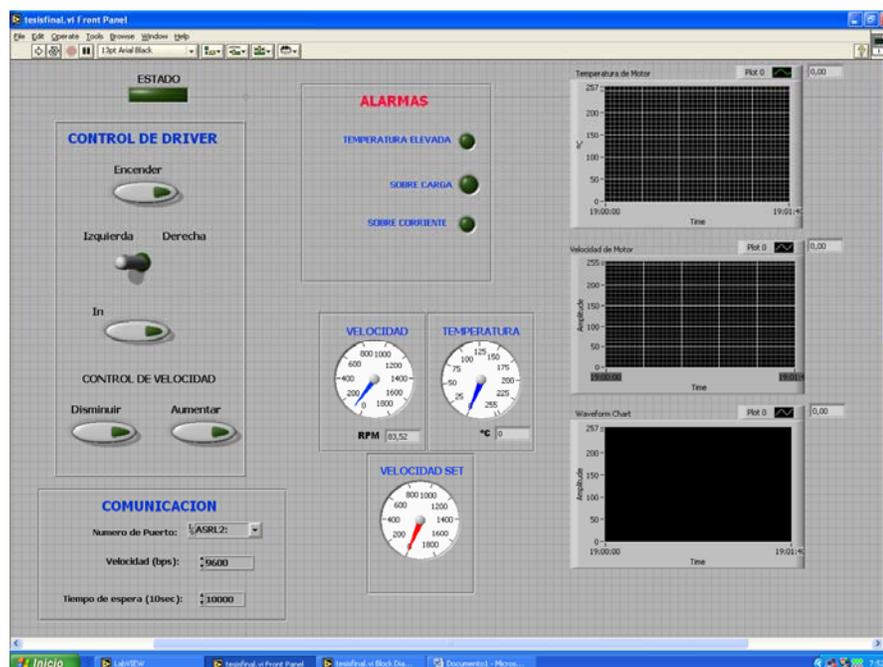


FIGURA 5.15 Panel de Control diseñado para el control de un variador de velocidad.

El Indicador “Estado”, nos indica si el sistema está habilitado o no desde la tarjeta de adquisición. Este habilitador es la Entrada Digital número 0, la programación realizada en el Diagrama de Bloques de LabVIEW pregunta por esta entrada inmediatamente después de abrir el puerto. Como vemos en las figuras 5.16 y 5.17 y según el protocolo desarrollado para la comunicación entre la tarjeta de adquisición y la PC, para saber el estado de la entrada digital número 0 debemos enviar el carácter “a”.

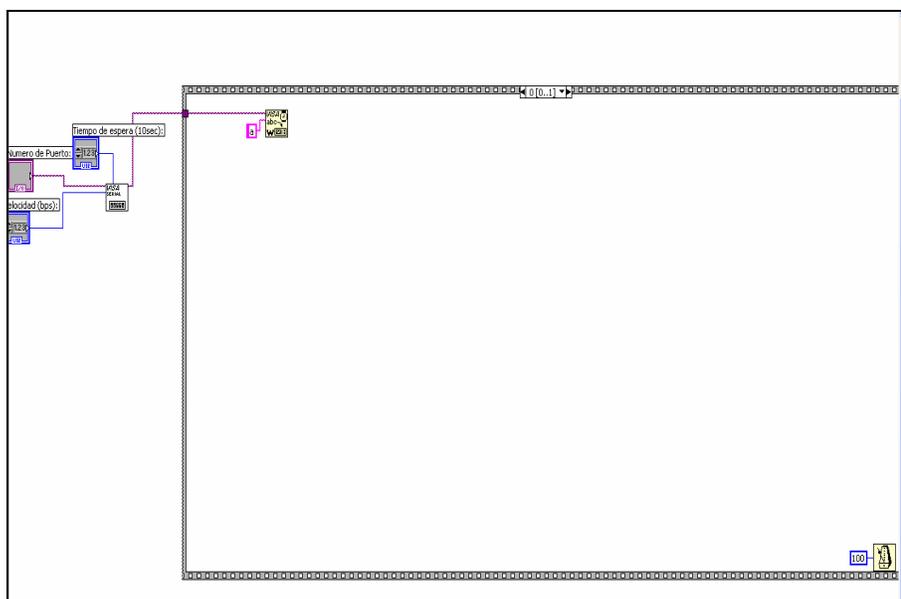


Figura 5.16. Diagrama de Bloques con el primer paso del programa diseñado.

Para realizar eso utilizamos la estructura “Stacked Sequence”, en la trama 0 (Figura 5.16) se escribe en COM 1 el caracter “a”, luego en la trama 1 (Figura 5.17), leemos el puerto y revisamos el contenido de este.

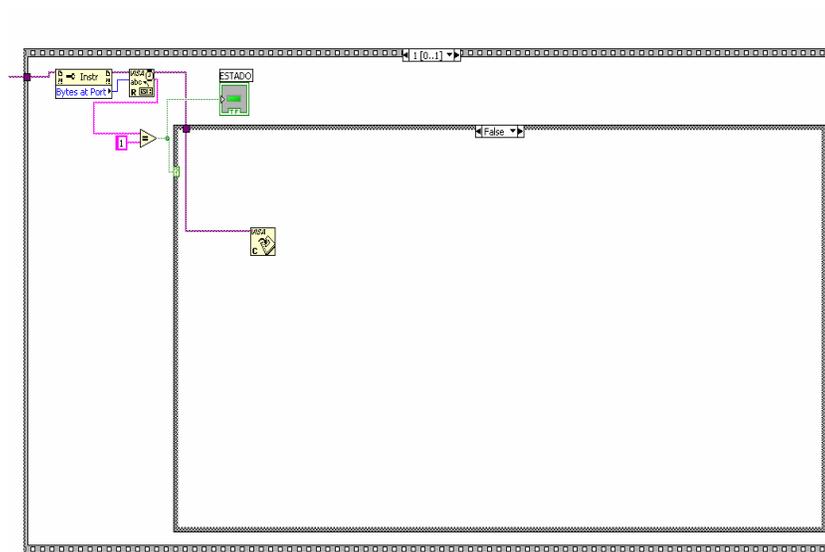


FIGURA 5.17 Diagrama de Bloque Trama 1 de la estructura principal y la ejecución por FALSE de la estructura CASE.

Para el siguiente paso utilizamos una estructura “CASE”, si el carácter leído desde COM 1 es “1” (Figura 5.18), se ejecuta la opción por TRUE de la estructura y el resto del proceso está habilitado; si el carácter leído desde COM 1 es “1” (Figura 5.17), se ejecuta la opción por FALSE de la estructura que cierra el puerto y continua en espera del “1”.

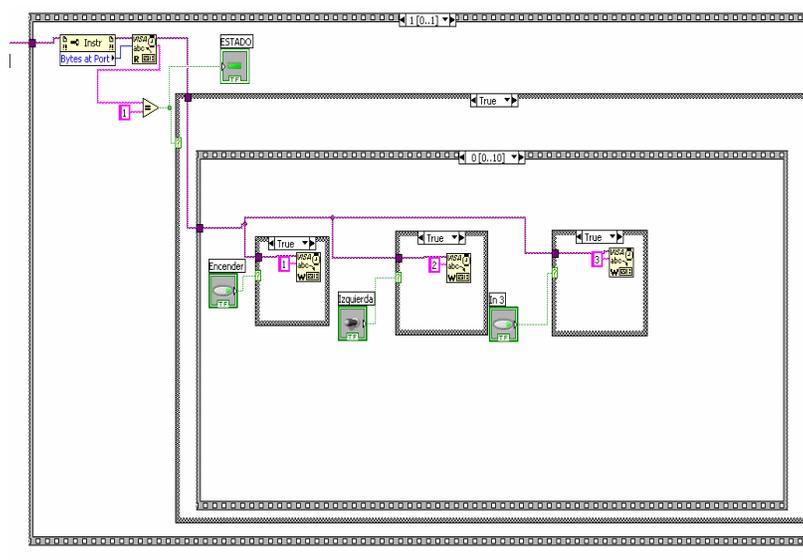


FIGURA 5.18 Diagrama de Bloque Trama 1 de la estructura principal y la ejecución por TRUE de la estructura CASE.

Mientras la entrada digital 0 esté habilitada, se ejecuta las instrucciones contenidas en la condición TRUE de la estructura CASE. La siguiente estructura interior es un “Stacked Sequence” de 11 framas, la frama 0 se muestra en la figura 5.18 y muestra los tres controles que manejan las salidas digitales de la tarjeta de adquisición utilizadas para el control del driver de velocidad. En la figura 5.19 se muestra la parte del panel de control de las salidas digitales y del diagrama de bloques correspondiente al manejo de las salidas digitales. En esta trama escribimos en el COM 1 el código descrito en el capítulo 4 de esta tesis de acuerdo al estado de cada uno de los controles.

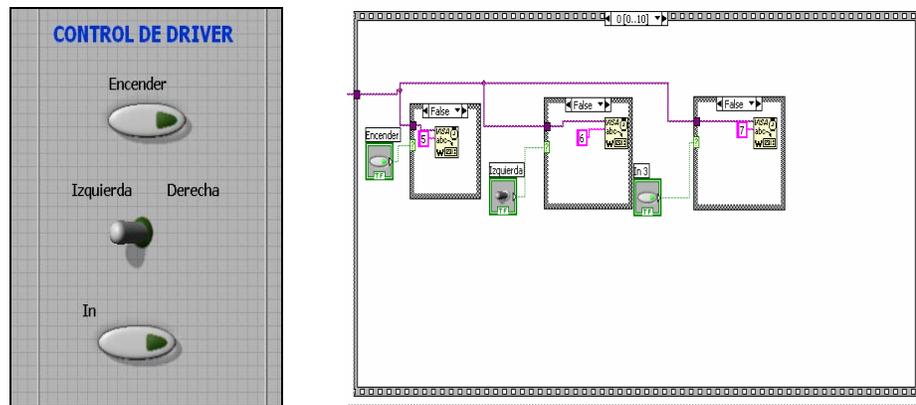


FIGURA 5.19 Diseño en LabVIEW para el manejo de las salidas digitales de la tarjeta de adquisición.

La frama número 1 escribe en el puerto el caracter “C” y genera un retardo de 50 ms. Como se describió en el Capitulo 4 cuando la tarjeta recibe el caracter “C” espera un tiempo y luego envía por el COM 1 el contenido del canal analógico número 2 que simula la entrada desde un sensor de temperatura LM 35 instalado en la carcasa del motor .

En la frama 2 se realiza el procedimiento de lectura del canal analógico número 2, como se muestra en la figura 5.20.

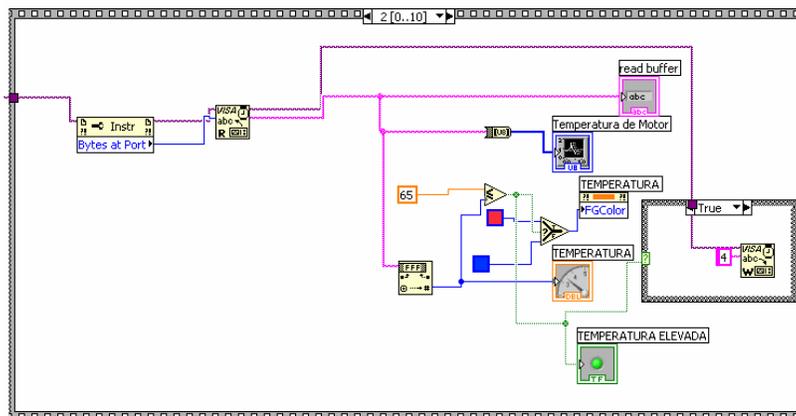


FIGURA 5.20 Frama # 2 correspondiente a la adquisición y visualización del canal analógico # 2.

En la figura anterior realizamos el procedimiento para visualizar el canal 2 mediante el gauge “TEMPERATURA” así como graficar mediante el Waveform Chart “Temperatura de Motor”. También realizamos una comparación para encender la alarma “TEMPERATURA ELEVADA” en caso de que el valor de temperatura exceda los 65° C, con esta condición se activa también la salida digital numero 4 de la tarjeta de adquisición, esto se hace con la estructura CASE mostrada en la parte derecha de la figura 5.20 y en la figura 5.21.

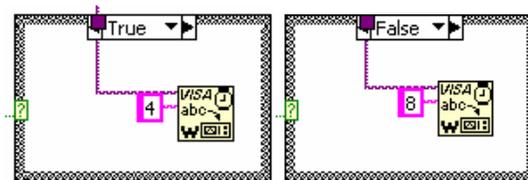


FIGURA 5.21 Estructura CASE para el manejo de la salida digital número 4.

La frama número 3 escribe en el puerto el caracter “0” y genera un retardo de 50 ms. Como se describió en el Capítulo 4 cuando la tarjeta recibe el caracter “0” espera un tiempo y luego envía por el COM 1 el contenido del canal analógico número 1 que simula la entrada desde un sensor para la velocidad del motor.

En la frama 4 se procede a leer el contenido del COM 1 correspondiente a la entrada analógica 1, como vemos en la figura 5.22 la programación es similar a la realizada en la frama 2.

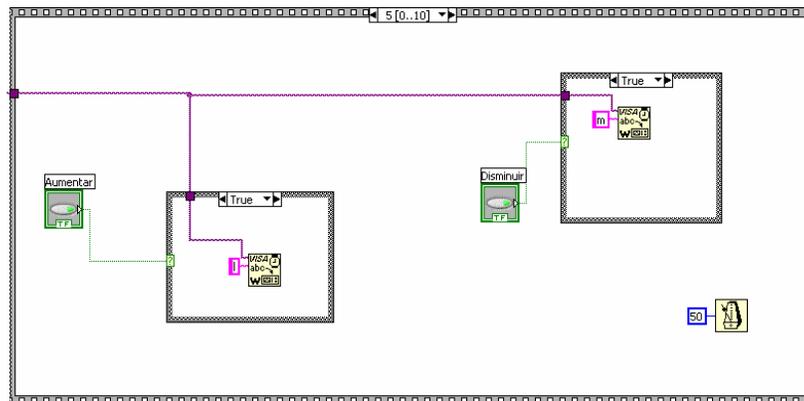


FIGURA 5.23 Frama 5 realizada para el manejo de la salida analógica, se muestran los ciclos tanto para Aumentar o Disminuir su Valor

En la figura 5.24 se muestra la frama número 6, esta frama contiene la programación que se realizó para mostrar el valor que tiene almacenado el microcontrolador, al cual le asignamos el nombre “VELOCIDAD SET”, con este canal intentamos, este valor es manejado desde los controles “AUMENTAR” y “DISMINUIR” explicados anteriormente.

En esta frama también se realiza una pequeña programación para que el color del gauge “VELOCIDAD SET” cambie a rojo una vez que el valor seteado sea mayor a 1750 RPM.

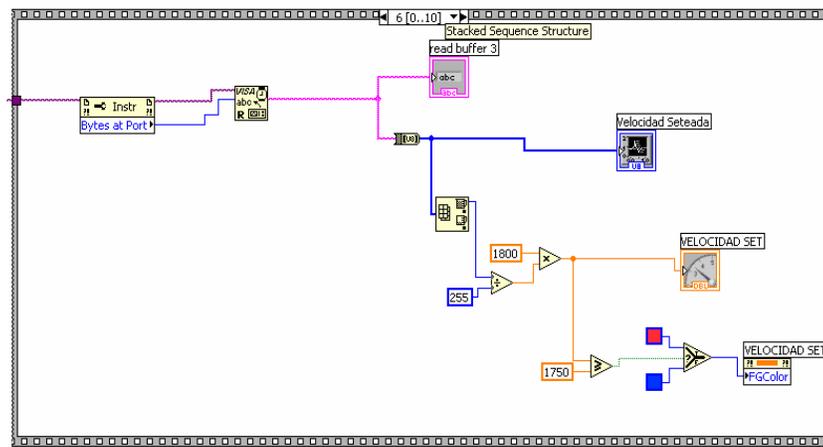


FIGURA 5.24 Frama 6 realizada para mostrar la salida analógica en el panel de control.

El control de las entradas digitales 2 y 3 se las realiza siguiendo el protocolo explicado en el capítulo cuatro, primero escribimos el carácter “b” en la frama 7 (figura 5.25) y luego de un retardo leemos en la frama 8, (mostrada en la figura 5.25) si el valor que me envió la tarjeta es un “1” o un “0”, dependiendo de el carácter enviado procedemos a encender o apagar el indicador “SOBRE CARGA” ubicado en el cuadro ALARMAS del panel de control (Figura 5.15).

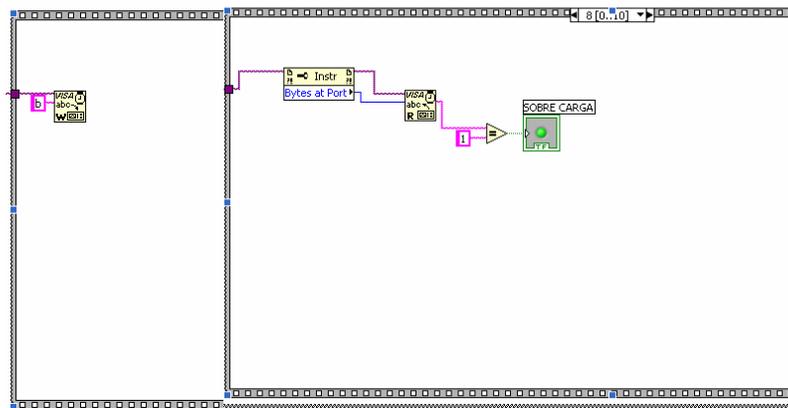


FIGURA 5.25 Framas 7 y 8 utilizadas para leer el valor de la entrada digital #2.

Finalmente las framas 9 y 10 nos ayudan a visualizar la entrada digital número 3 estas framas poseen una programación similar a las framas 7 y 8 con la diferencia que el carácter enviado desde LabVIEW es el “c” y el indicador sobre el que se actúa es “SOBRE CORRIENTE” ubicado en el cuadro ALARMA del panel de control.

Conclusiones y Recomendaciones

En base a los trabajos realizados en nuestra tesis, tales como: recopilación de información, uso de los softwares de simulación y adquisición de datos, diseño e implementación de una tarjeta electrónica, análisis de una fuente de conmutación; podemos concluir lo siguiente.

- El SCADA LOOKOUT 5.0 es un software de gran flexibilidad, ya que permite realizar una comunicación muy rápida y precisa con diversos dispositivos de control, como PLCs de diversas marcas, gracias a la gran cantidad de drivers que posee para su interface de comunicación serial. Es muy ventajoso para el usuario trabajar con este tipo de programa ya que los datos del proceso son manejados y administrados de una manera sencilla, debido a la tecnología OPC y al protocolo de comunicación DDE, para su utilización con otros programas que operan bajo Windows, tales como Excel y SQL Server, que permiten llevar un registro histórico de todos los datos monitoreados. Otra ventaja es que no tiene limitaciones graficas, ya que se pueden importar imágenes para la animación de los objetos, desde otras aplicaciones bajo formato wmf o bmp.

- En el presente trabajo se han desarrollado dos prototipos de dispositivos, el primero fue la adquisición de datos y el otro fue el control, basados en la programación de LabView 7.1, en ambas experiencias se utilizó un buen sistema de comunicación, las mismas que nos dejan otras motivaciones para futuro poder realizar una Instrumentación Virtual Remota, poder también hacer uso de la adquisición de datos Vía Internet y lo que es mejor su aplicación en redes industriales.

A pesar que Internet no fue diseñada para tareas de adquisición de datos y control, su uso se ha extendido en este campo, debido a lo atractivo de su software y su hardware ampliamente soportados y relativamente económicos, así como la eliminación de las barreras de distancias entre los dispositivos y los puntos de monitoreo.

Las aplicaciones realizadas en nuestro proyecto comprenden las entradas y salidas analógicas y digitales, convertidor analógico digital (ADC). Todo esto fue importante ya que profundizamos más nuestros conocimientos en el procesamiento y envío de señales analógicas o digitales a los diferentes actuadores que conocemos a nivel industrial.

Finalmente el estudio de nuestro diseño puede ser aplicado en los diversos campos de la automatización e investigación que incluyan señales digitales para visualizar y controlar en respuestas a cambios con otros parámetros como válvulas, solenoides, relés, etc.

- La utilización de un lenguaje de programación grafica como LabVIEW 7.1 para el desarrollo de interfaces para el monitoreo y control de procesos es una gran alternativa debido a la gran cantidad de herramientas que posee, como controladores e indicadores usados para señales digitales y analogicas, ademas de incluir varios SubVIs que facilitan el desarrollo de aplicaciones complicadas, ademas posee instrucciones para interactuar con otros programas como EXCEL, crear reportes Web y el proceso de comunicación serial es mas confiable y sencillo que cuando se utiliza un software de programación de alto nivel. La desventaja de LabVIEW es que lamentablemente no nos permite comunicarnos con PLCs de la forma que se lo hace con un sistema SCADA y ademas los drivers de comunicación serial hay que comprarlos. En definitiva una aplicación el LabVIEW esta limitada al uso de tarjetas de adquisición.

- Por su característica de baja regulación de voltaje, su tamaño y peso reducido a causa de la ausencia de transformadores reductores en su circuito de entrada, las fuentes de conmutación son la mejor opcion para alimentar equipos electronicos, sin embargo en el analisis realizado en el Capitulo 3 pudimos comprobar que su diseño puede llegar a ser complejota que esto depende de los requerimientos de la carga que se va a alimentar.

INDICE DE FIGURAS**CAPITULO 1**

- FIGURA 1.1 Arquitectura interna de una batería de 12 voltios.
- FIGURA 1.2 Diagrama de bloques del proceso de fabricación de baterías plomo- ácido.
- FIGURA 1.3 Rejillas de plomo para fabricación de baterías.
- FIGURA 1.4 Vista frontal de las rejillas una vez empastadas.
- FIGURA 1.5 Vista de las placas una vez que son colocadas en el sobre.
- FIGURA 1.6 Vista interna de las celdas de una batería.
- FIGURA 1.7 Batería lista. Producto terminado.

CAPITULO 2

- FIGURA 2.1. Pantalla inicial del “Default Panel” que aparece al iniciar Lookout.

- FIGURA 2.2 Cuadro de diálogo del objeto OPCClient.
- FIGURA 2.3 Cuadro de diálogo para crear un proceso nuevo en Lookout.
- FIGURA 2.4 Cuadro de diálogo para insertar un gráfico en Lookout.
- FIGURA 2.5 Vista de la pantalla de la Máquina Empastadora y sus componentes.
- FIGURA 2.6 Objeto SeqMain (Sequencer), que maneja la animación de la pantalla de la máquina empastadora.
- FIGURA 2.7 Herramienta Edit Conection, para programar los saltos de estado del Sequencer: SeqMain.
- FIGURA 2.8 Cuadro de diálogo del Objeto Switch y su representación gráfica dentro del panel de control.
- FIGURA 2.9 Cuadro de diálogo del objeto Animator.
- FIGURA 2.10 Cuadro de diálogo del objeto Waveform.

- FIGURA 2.11 Pantalla que representa los finales de carrera en la banda transportadora 2.
- FIGURA 2.12 Objetos switch que representan los relés térmicos de los motores MB1 y MB2.
- FIGURA 2.13 Ventana que representa el panel de operación del variador de velocidad con algunos de sus parámetros.
- FIGURA 2.14 Pantalla del horno de secado de las rejillas de plomo.
- FIGURA 2.15 Pantalla del panel de operador del horno.
- FIGURA 2.16 Display gráfico para la expresión que activa el motor de la banda transportadora del horno.
- FIGURA 2.17 Cuadro de diálogo del objeto Multistate.
- FIGURA 2.18 Cuadro de diálogo, para la conexión entre los contactos auxiliares del PLC (objeto GE1) y los objetos de la pantalla del horno.

- FIGURA 2.19 Pantalla que muestra los dispositivos térmicos y el estado de operación de los motores en la pantalla del horno.
- FIGURA 2.20 Cuadro de diálogo del objeto Pot y su representación gráfica.
- FIGURA 2.21 Pantalla del cuarto de recepción de rejillas de plomo terminadas.
- FIGURA 2.22 Cuadro de diálogo del objeto Animator: Animator21 de la pantalla de recepción de rejillas terminadas.
- FIGURA 2.23 Cuadro de diálogo del objeto Sequencer: SeqRecepcion.
- FIGURA 2.24 Cuadro de diálogo del objeto Spreadsheet: SSrejillas.
- FIGURA 2.25 Pantalla de la tolva de la máquina empastadora.
- FIGURA 2.26 Cuadro de dialogo del objeto Animator, para el movimiento de las batidoras de la tolva.
- FIGURA 2.27 Cuadro de diálogo de objeto PID.

- FIGURA 2.28 Pantalla del panel de control del proceso.
- FIGURA 2.29 Cuadro de dialogo del objeto HyperTrend.
- FIGURA 2.30 Objeto Hyper Trend.
- FIGURA 2.31 Cuadro de diálogo del objeto RadioButtons y su representación gráfica.
- FIGURA 2.32 Objeto Hyper Trend para el peso de la placa (rejilla de plomo).
- FIGURA 2.33 Cuadro de diálogo del objeto Recipe.
- FIGURA 2.34 Archivo de una receta creado en Excel.
- FIGURA 2.35 Pantalla de los Parámetros PID.
- FIGURA 2.36 Pantalla de Alarmas del proceso.

FIGURA 2.37 Mensaje de alarma en caso de falla de algún elemento del proceso.

FIGURA 2.38 Cuadro de diálogo de la expresión para mostrar un mensaje de falla.

FIGURA 2.39 Cuadro de diálogo del objeto Pulse.

CAPITULO 3

FIGURA 3.1 Diagrama de bloques de una fuente conmutada de computadora

FIGURA 3.2 Etapa de rectificación que consta de la red EMI/RFI, el puente rectificador, los capacitores y el termistor.

FIGURA 3.3 Diagrama Esquemático de la Etapa del Convertidor de Alta Frecuencia

FIGURA 3.4 Arquitectura Interna del CI PWM (KA7500B)

FIGURA 3.5 Diagrama esquemático de la Etapa de Rectificación y Filtro de Salida.

CAPITULO 4

- FIGURA 4.1 Diagrama de bloques del sistema
- FIGURA 4.2 Diagrama de bloques de la tarjeta.
- FIGURA 4.3 Circuito impreso de las pistas de la tarjeta.
- FIGURA 4.4 Circuito impreso de los elementos de la tarjeta.
- FIGURA 4.5 Comunicación entre USART Transmisor y otro Receptor.
- FIGURA 4.6 Esquemático de la tarjeta de adquisición de datos desarrollado mediante el software de Proteus.
- FIGURA 4.7 Vista superior de la tarjeta de adquisición.
- FIGURA 4.8 Fotografía general de las entradas, salidas digitales y analógicas.
- FIGURA 4.9 Fotografía general de las pistas de la tarjeta de adquisición de datos.

CAPITULO 5

- FIGURA 5.1 Pantalla Inicial de LabVIEW 7.1
- FIGURA 5.2 Panel Frontal y Diagrama de Bloques.
- FIGURA 5.3 Panel Frontal de una respuesta de Frecuencia.
- FIGURA 5.4 Diagrama de Bloques para el Panel Frontal de la figura 5.3
- FIGURA 5.5 Barra de herramientas.
- FIGURA 5.6 Representación de controles e indicadores en ambas pantallas
- FIGURA 5.7 Cambio de color de un objeto en LabVIEW.
- FIGURA 5.8 Editar un texto para un objeto en LabVIEW.
- FIGURA 5.9 Cambio de Escala para un objeto en LabVIEW
- FIGURA 5.10 Tipos de datos en LabVIEW

- FIGURA 5.11 Ciclo Mientras (While) continua si es verdadero.
- FIGURA 5.12 Ciclo For (Para).
- FIGURA 5.13 Conectores usados para la comunicación serial RS-232.
- FIGURA 5.14 VI de inicialización del VISA.
- FIGURA 5.15 Panel de Control diseñado para el control de un variador de velocidad.
- FIGURA 5.16 Diagrama de Bloques con el primer paso del programa diseñado.
- FIGURA 5.17 Diagrama de Bloque Trama 1 de la estructura principal y la ejecución por FALSE de la estructura CASE.
- FIGURA 5.18 Diagrama de Bloque Trama 1 de la estructura principal y la ejecución por TRUE de la estructura CASE.

- FIGURA 5.19 Diseño en LabVIEW para el manejo de las salidas digitales de la tarjeta de adquisición.
- FIGURA 5.20 Frama # 2 correspondiente a la adquisición y visualización del canal analógico # 2..
- FIGURA 5.21 Estructura CASE para el manejo de la salida digital número 4.
- FIGURA 5.22 Frama 4 realizada para la adquisición y visualización del canal analógico 1, correspondiente a la velocidad del motor.
- FIGURA 5.23 Frama 5 realizada para el manejo de la salida analógica, se muestran los ciclos tanto para Aumentar o Disminuir su valor.
- FIGURA 5.24 Frama 6 realizada para mostrar la salida analógica en el panel de control.
- FIGURA 5.25 Framas 7 y 8 utilizadas para leer el valor de la entrada digital #2.

INDICE DE TABLAS**CAPITULO 1**

TABLA 1.1 Cuadro comparativo de los usos y ventajas de las clases de baterías.

TABLA 1.2 Tabla comparativa del numero de ciclos y %PD entre dos baterías.

TABLA 1.3 Variación de la energía almacenada de una batería con respecto a la temperatura de operación.

CAPITULO 4

TABLA 4.1 Funciones del registro TXSTA.

TABLA 4.2 Funciones del registro RCSTA.

TABLA 4.3 Registros ADCON0-ADCON1.

TABLA 4.4 Frecuencias de Conversión.

- TABLA 4.5 Valores de TDA según los bits de los registros ADCS1-ADCS0.
- TABLA 4.6 Comandos que recibe el PIC para actuar en las salidas digitales.
- TABLA 4.7 Comandos que recibe el PIC para actuar en la salida analógica.
- TABLA 4.8 Comandos que recibe el PIC para actuar en las entradas analógicas.
- TABLA 4.9 Costos de los materiales utilizados en la tarjeta de adquisición de datos.

INDICE DE ANEXOS

ANEXOS A

- 1.1 Criterio para la selección del disipador de calor.
- 1.2 Simbología.
- 1.3 Análisis de la respuesta de frecuencia de la red de filtrado RFI.
- 1.4 Resultados obtenidos de las mediciones realizadas a plena carga de la fuente conmutación con el osciloscopio Tectronic 2230.
- 1.5 Datasheets de los principales integrados de la fuente de conmutación.

ANEXOS B

- 1.1 Datasheets de los principales integrados utilizados en la tarjeta de adquisición de datos, descrita en el Capítulo 4.

ABREVIATURAS

CAPITULO 1

PD	Profundidad de Descarga de una batería.
VRLA	Valve-Regulated Lead-Acid. (Baterías selladas reguladas por válvula).
AGM	Absorbed Glass Mat. (Baterías selladas con lámina de fibra de vidrio absorbente).
PRV	Válvula de Alivio de Presión
FVs	Sistema de Baterías Solares.
BCP	Baterías de Ciclo Profundo.
Ah	Amperios-Hora.
Kgs	Kilogramos.
MGP	Motor del Grid Picker.

MB1	Motor de la banda transportadora 1.
MB2	Motor de la banda transportadora 2.
MBH	Motor de la banda transportadora del Horno.
GLP	Gas Licuado de Petróleo.
MIJA	Motor de Inducción Jaula de Ardilla.
η_b	Eficiencia Total de una Batería.
η_c	Eficiencia de Carga de una Batería.
η_d	Eficiencia de Descarga de una Batería.

CAPITULO 2

DDE	Dynamic Data Exchange (Intercambio Dinámico de Datos).
-----	--

Ethernet	Es una tecnología estandarizada para redes de área local, ampliamente usada en el campo industrial, especificado en el código estándar IEEE 802.3.
ASCII	American Standard Code for Information Interchange (Código Americano Estándar para Intercambio de Información).
ODBC	Open Data Base Connectivity, es una aplicación de interface de programación (API) estándar, para tener acceso a bases de datos.
OLE	Object Linking and Embedding, es un protocolo de Microsoft que permite que las aplicaciones puedan enviar datos a otras aplicaciones.
OPC	OLE for Process Control (OLE para Control de Procesos).
PLC	Programmable Logic Controller (Controlador Lógico Programable).

SCADA Supervisory Control And Data Adquisition (Supervisión, Control y Adquisición de Datos).

TCP/IP Transmission Control Protocol (Protocolo de Control de Transmisión), es un método para enviar datos entre computadoras. Usado con IP, el Internet Protocol (Protocolo de Internet).

CAPITULO 3

F.C. Fuente de Conmutación.

HMI Human-Machine Interface (Interface Hombre-Máquina).

IGBT Transistor Bipolar de Compuerta Aislada.

LC Inductor-Capacitor.

MOSFET Transistor Efecto de Campo.

PWM Modulación Ancho de Pulso.

RC	Resistor-Capacitor.
RFI	Interferencia por Radio Frecuencia.
Bmax	Campo Magnético Máximo.
Bsat	Campo Magnético de Saturación

CAPITULO 4

DAC Digital Analogic Converter (Convertidor Digital Analógico).

PIC	Programador Integrable controlable.
USART	Universal Synchronous Asynchronous Receiver Transmitter.
BCD	código binario decimal.
μ s	Microsegundos
ms	milisegundos

Bd baudios.

Hz Hercios.

CAPITULO 5

DTE Data Terminal Equipment.

EIA Asociación de Industrias Electrónicas

GPIB General Purpose Interface Bus.

LabVIEW Laboratory Virtual Instrument Engineering Workbench

VI Virtual Instrument.

VISA Virtual Instrument Standard Application.

VXI VME extensions for Instrumentation.

GLOSARIO

WMF	(Windows Metafile) Es un tipo de archivo de imágenes de tipo vectorial, diseñados para trabajar de manera compatible con los softwares de Microsoft.
BMP	(mapa de bits) Es un formato desarrollado para aplicaciones Windows® y, por tanto, muy utilizado. Las imágenes de mapa de bits se denominan así porque el formato crea un mapa de los pixels, línea a línea. El formato BMP no sufre pérdidas y, por tanto, resulta adecuado para guardar imágenes que se desea manipular o ampliar posteriormente. Prácticamente todos los programas de edición de imágenes reconocen el formato BMP.
PSI	Unidad de presión en libras por pulgadas cuadradas, que equivale a 0,068 atmósferas.
PLC	(CONTROLADOR LÓGICO PROGRAMABLE) Los controladores lógicos programables o PLC (Programmable Logic Controller en sus siglas en inglés) son dispositivos electrónicos muy usados en automatización industrial.
SCADA	(Supervisión, Control and Data Acquisition) Sistema de control, supervisión y adquisición de datos en tiempo real que permiten la adquisición de datos (para control o supervisión) desde múltiples puntos y con un destino.
Bd	(baudios) Unidad utilizada para medir la velocidad de transmisión de datos de un módem o una conexión serie entre dos equipos, o entre un equipo y un dispositivo externo. Indica el número de veces por segundo que puede cambiar la señal que se transmite. Suele coincidir con el número de bits por segundo que en teoría es capaz de transmitir/recibir, ya que la modulación de la señal normalmente se realiza a nivel de bit.

UPS	Sistema de alimentación ininterrumpida.
CHART	Tipo de indicador gráfico en LabVIEW 7.1 que permite observar la variación de una señal en el tiempo.
LVM	(LabVIEW Measurement File) Tipo de archivo que genera LabVIEW para almacenar los datos que recibe el VI Expres "Write LabVIEW Measurement File".

BIBLIOGRAFIA

- a) Direcciones electrónicas sobre las baterías de plomo – ácido.
www.ecosolar.giga.com.ar/bateria.htm
www.automecanico.com/auto2001/Bateria.html
www.aca.org.ar/servicios/baterias/index.htm
- b) J. NATIONAL INSTRUMENT.
Lookout Developers Manual, United States, November 2001.
- c) CHRYSSIS GEORGE.
High frequency switching power supplies, United States, 1976.
- d) JOSE M. ANGULO USATEGUI E IGNACIO ANGULO MARTINEZ -
UNIVERSIDAD DE DEUSTO (CAMPUS VICTORIA).
Microcontroladores PIC, Diseño Práctico de aplicaciones, Primera parte
PIC 16F84 PBasic y ensamblador, España, 2003.
- e) JOSE M. ANGULO USATEGUI, SUSANA ROMERO YESA E IGNACIO
ANGULO MARTINEZ - UNIVERSIDAD DE DEUSTO (CAMPUS
VICTORIA).

Microcontroladores PIC, Diseño Práctico de aplicaciones, Segunda parte
PIC 16F84 PBasic y ensamblador, España, 2003.

- f) ROBERT H. BISHOP-UNIVERSITY OF TEXAS AT AUSTIN.
Learning with Labview7 Express, United States, 2004.

ANEXO A1.1

CRITERIO PARA SELECCIÓN
DEL DISIPADOR DE CALOR

Selección del Disipador de Calor

La mayoría de fabricantes de semiconductores proporcionan los datos suficientes para poder calcular el disipador que necesitamos. Necesitamos como punto de partida, la temperatura máxima que puede alcanzar la unión del transistor. Esta temperatura no se deberá alcanzar en ningún caso, para no destruir el componente. Normalmente el fabricante proporciona el "operating temperature range" por ejemplo, -65 to 200 °C indica que la temperatura máxima es de 200°C. Nosotros podemos tomar un coeficiente de seguridad $k = 0.5$ para un diseño normal con temperatura moderada.

$k = 0.6$ para economizar en tamaño de disipador.

$k = 0.7$ cuando el disipador permanezca en posición vertical y en el exterior (mejora de convección).

Con el coeficiente k , y tomando la temperatura máxima de funcionamiento como T_j , tenemos la expresión:

$$T = k T_j - T_a = w (R_{jc} + R_{cd} + R_{da})$$

Donde w representa la potencia en watios (calor) que disipará el componente.

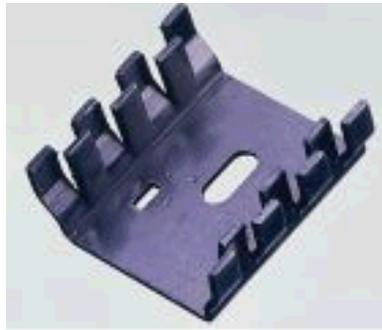
Si no disponemos de estos datos, podemos tomar como $T_j = 135\text{ }^\circ\text{C}$ para transistores de silicio, y $T_j = 90^\circ\text{C}$ para transistores de germanio.

El flujo de calor, desde la unión PN hasta el ambiente tiene que atravesar varios medios, cada uno con diferente resistencia térmica.

- Resistencia unión-cápsula (R_{jc}). Viene dado en manuales y tablas, y depende de la construcción de la cápsula. El tipo TO-3 disipa gran cantidad de calor.
- Resistencia cápsula-disipador (R_{cd}). Depende del encapsulado y del aislamiento, si lo hay, entre el

componente y el disipador. El aislante puede ser mica, pasta de silicona y otros medios. Cada uno presenta diferente resistencia térmica.

- Resistencia disipador-ambiente (R_{da}). Este es el que tratamos de calcular.



radiador para TO-220



radiador para TO-3

Fig. 1. Algunos tipos de disipadores

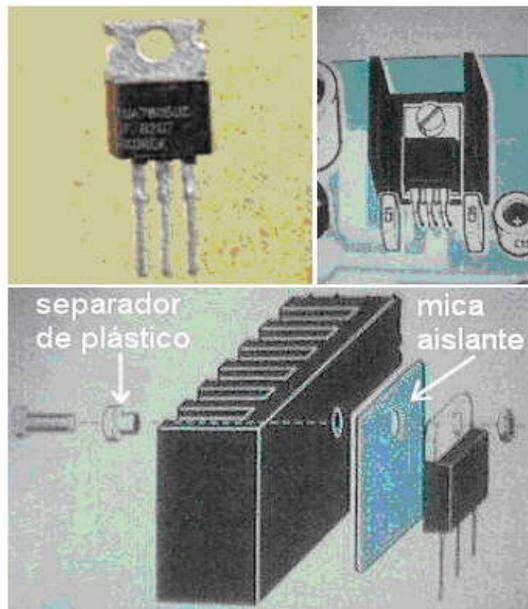


Fig. 2. Cápsula TO-220 simbología que se utiliza para transistores de menos potencia, para reguladores de tensión en Fuentes de alimentación y para transistores y triacs de baja potencia.

Generalmente necesitan un radiador de aluminio, aunque a veces no es necesario, si la potencia que va a disipar es reducida. Abajo vemos la forma de colocarle el radiador y el tornillo de sujeción. Se suele colocar una mica aislante entre el transistor y el radiador, así como un separador de plástico para el tornillo, ya que la parte metálica está conectada al terminal central y a veces no interesa que entre en contacto eléctrico con el radiador.

Cálculo del Disipador de Calor

DATOS:

- Tipo de encapsulado: TO-220 ;
- PD=5 watts ;
- RJC = 3.0 [°C/Watt] ;
- TJ máx. = 150°C ;
- TA máx. = 50°C

Encontrar: El disipador apropiado para mantener la juntura del semiconductor debajo de 150°C en convención natural.

SOLUCIÓN: Utilizando la siguiente ecuación

$$P_D = \frac{T_j - T_A}{R_{\theta JC} + R_{\theta CS} + R_{\theta SA}}$$

luego:

$$R_{\theta SA} = \frac{T_j - T_A}{P_D} - (R_{\theta JC} + R_{\theta CS})$$

Consideramos que el dispositivo está montado con grasa siliconada y sin aislamiento eléctrica.

La resistencia térmica desde la carcasa a la superficie de montaje (R_{CH}) para el encapsulado TO220 puede obtenerse de la siguiente figura:

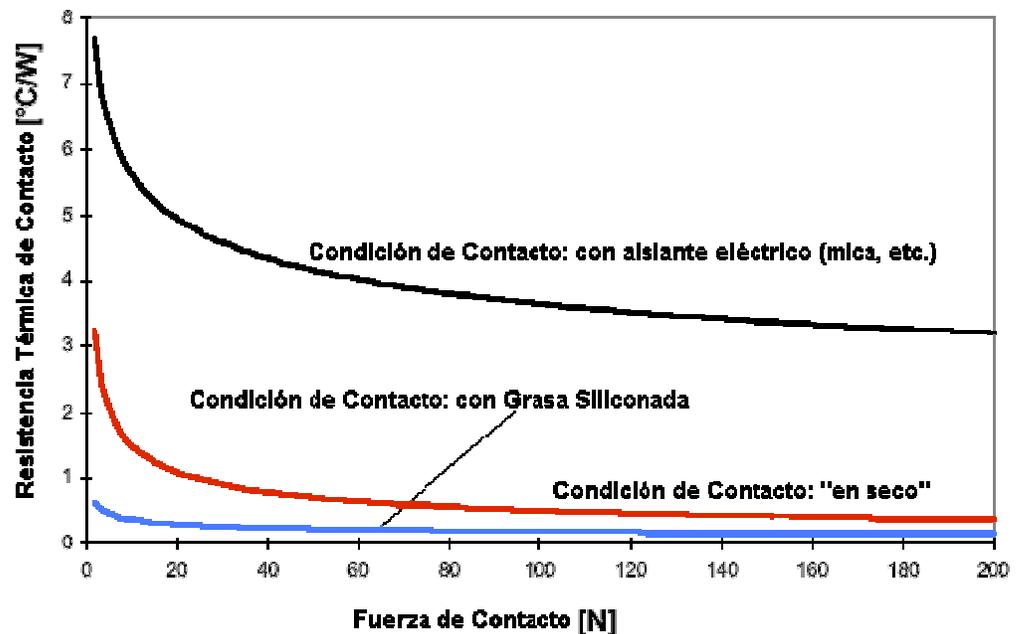


Fig. 3. Curva de Resistencia Térmica vs. Fuerza de contacto en función de la condición de contacto

De la cual se ve que:

$R_{CH} = 0.5[^\circ\text{C}/\text{W}]$ (a: 0.678Nm (Newton-metro) de torque del tornillo de sujeción al disipador), luego será como máximo permitido:

$$R_{\theta SA} = \frac{150^\circ\text{C} - 50^\circ\text{C}}{5\text{W}} - 3.5$$

$$R_{\theta SA} = 16.5^\circ\text{C}/\text{W}$$

Criterios de Selección de los Disipadores

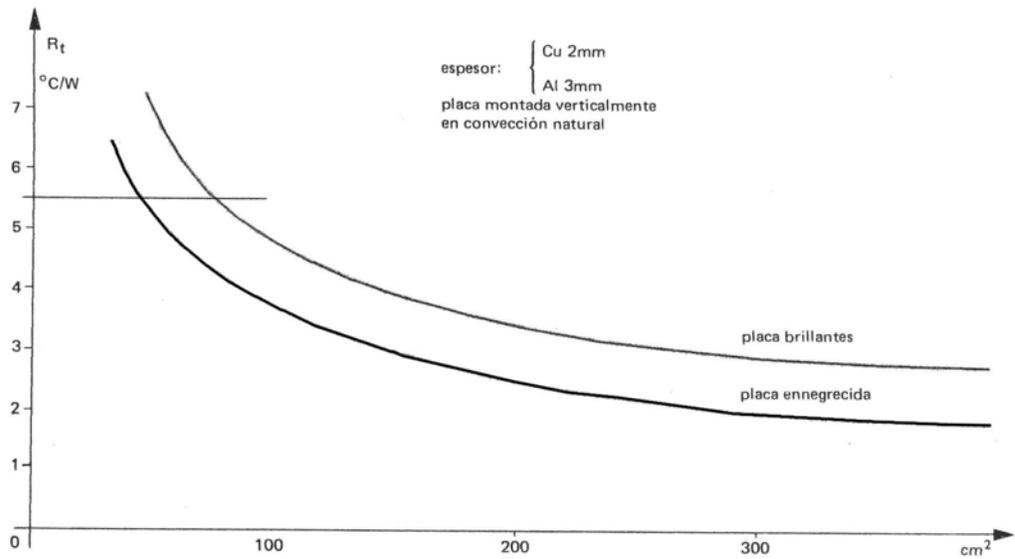


Fig. 4. Espesor del metal disipador.

Este gráfico puede ser utilizado, considerando el espesor del metal del disipador, el mismo que se encuentra especificado en la figura anterior, y seleccionando si el disipador es brillante o ennegrecido. Con estos criterios podemos realizar una selección adecuada de nuestro disipador, encontrando un valor aproximado del coeficiente de disipación calculado.

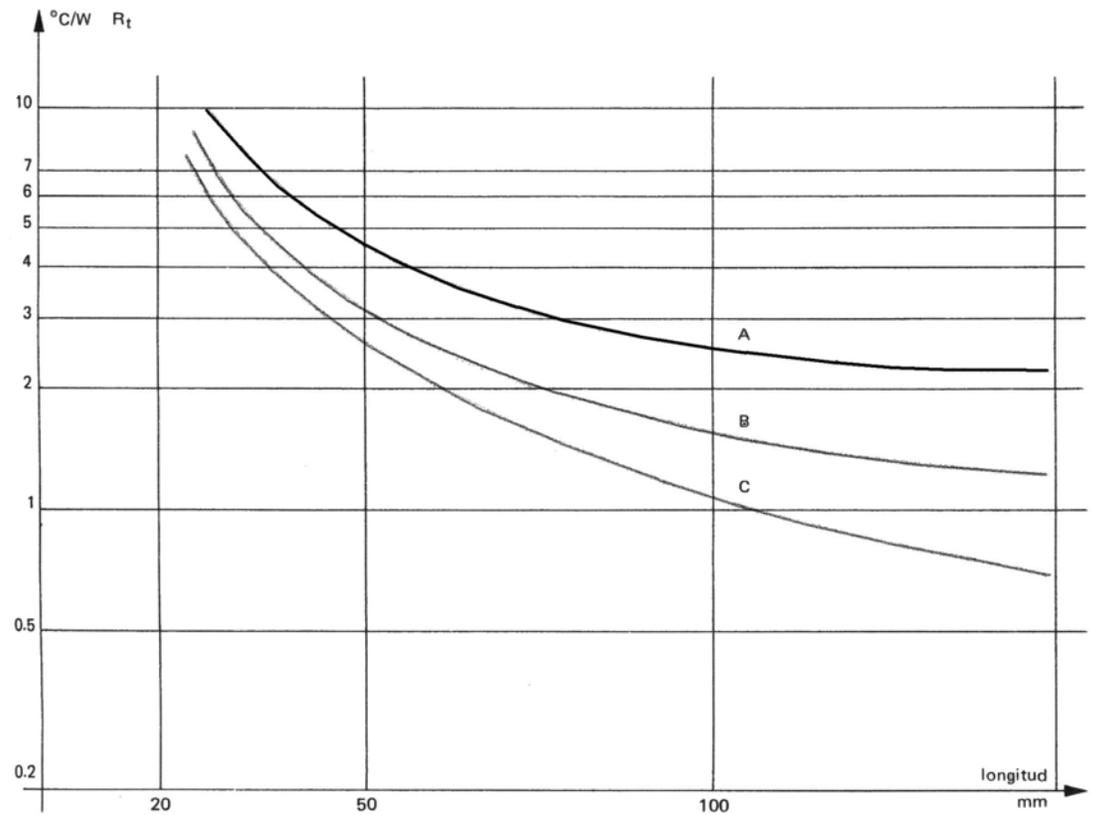
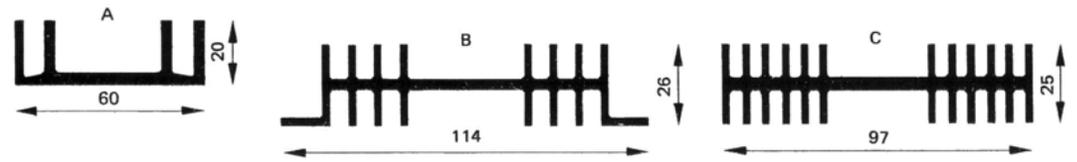


Fig. 5. Coeficiente de disipación.

El gráfico anterior nos muestra el coeficiente de disipación de un disipador dependiendo de la longitud y del número de

aletas del disipador, de manera que se han clasificado en tres categorías A, B, C.

Partimos de la expresión:

$$T = T_j - T_a = P_{ot} * (R_{jc} + R_{cd} + R_{da})$$

Tenemos que calcular el valor de disipador que necesitamos, R_{da} . Despejamos y ponemos un $k = 0.7$ porque vamos a poner el disipador en el exterior y vertical.

$$R_{da} = [(k T_j - T_a) / w] - R_{jc} - R_{cd}$$

$$[(0.7 * 125^{\circ}\text{C} - 25^{\circ}\text{C}) / 7 \text{ W}] - 5^{\circ}\text{C/w} - 1.4^{\circ}\text{C/w} = 2.53^{\circ}\text{C/w}$$

Buscamos en catálogo y encontramos el radiador siguiente:

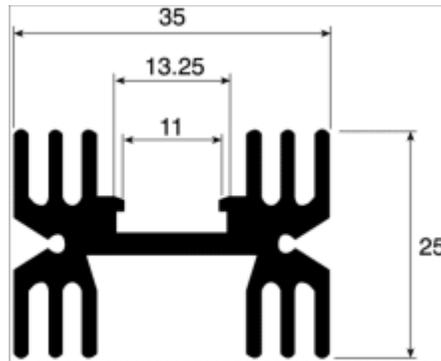


Fig. 6 Dimensiones del disipador

Tiene una $R = 5 \text{ }^\circ\text{C/w}$, es suficiente, en cuenta que ya hemos tomado un coeficiente k de seguridad igual a 0.7 y nos aseguramos de sobra. Con este radiador, podemos calcular la temperatura que alcanzará el mismo cuando el 7805 disipa 7 W de una forma muy sencilla:

$$T_d - T_a = R_{da} * P_{ot}$$

$$T_d = R_{da} * P_{ot} + T_a = 5 \text{ }^\circ\text{C/w} * 7 \text{ W} + 25 \text{ }^\circ\text{C} = 60 \text{ }^\circ\text{C}$$

La elección del coeficiente k es arbitraria por nuestra parte. Podemos perfectamente elegir $k=1$ pero nos arriesgamos mucho. Es preferible en este caso subir la temperatura ambiente de diseño a 30 ó 35 grados. Tener en cuenta que si el dispositivo está en una caja, la temperatura fácilmente sube a 40 grados y más.

Obsérvese cómo influye en el regulador de tensión la diferencia de tensión en entrada y salida para la disipación de potencia.

ANEXO A1.2

SIMBOLOGIA

SIMBOLOGIA

P_{IN} Potencia de Entrada.

P_{OUT} Potencia de Salida.

η Eficiencia.

t Tiempo en el que debe cargarse el capacitor del
rectificador de entrada, en ms.

ΔV Voltaje de rizado pico a pico.

I Corriente de Carga en la Entrada.

V_p Voltaje pico de la fuente de alimentación.

V_{dc} Voltaje DC a la salida de esta etapa.

C Capacitancia Equivalente de Entrada.

I_c Corriente de carga del capacitor

V_{in} Voltaje pico – pico máximo a la salida
del convertidor.

f_R Frecuencia de Resonancia.

f_s Frecuencia de conmutación.

a Relación de vueltas del transformador principal.

L Inductancia del transformador referida al primario.

C_3 Capacitancia en serie con la salida del converter.

V_p Voltaje en el primario del transformador.

V_s Voltaje en el secundario del transformador.

RResistencia de descarga de los capacitores
Cx que forman parte de la red RFI

t_{DESC} Tiempo de descarga de Cx, igual a 1 seg.
Según la norma IEC 380 Sec 8.8

C_{xeq} Suma de los capacitores Cx existentes
en la red de supresión de RFI

C_s Capacitancia de Red Snnuber

R_s Resistencia de Red Snnuber

a Relación de vueltas del primario (N_p) con
el secundario (N_s)

L_T Inductancia del transformador

C_J Capacitancia de juntura del Schkottky (pf)

P_R Potencia de la resistencia R_S

I Corriente promedio primaria (A)

C Capacitancia, (uf)

dt Intervalo de tiempo de carga, (us)

T Periodo de conmutación, (us)

δ_{max} Ciclo de trabajo

fs Frecuencia de conmutación, Khz

P_{out} Potencia de trabajo del transformador, (W)

B_{max} Densidad de flujo pico de operación, (G)

f Frecuencia, (Hz)

D Densidad de corriente por alambre, (A/m^2)

A_e Área efectiva del núcleo, (cm^2)

A_c Área para el enrollamiento del bobinado, (cm^2)

CM Tamaño del alambre, c.m (Circula mil)

I_p Corriente máxima que circula por el primario del transformador, A

D Densidad de corriente, c.m/A

N_p Número de vueltas del primario

V_p Voltaje de Rizado, V

f Frecuencia de oscilación, Khz

B_{max} Densidad de flujo pico de operación, G

E_{OUT} Voltaje de salida de la fuente

E_{IN} Voltaje de conmutación ($2 * E_{OUT}$)

N Número de vueltas

I_{COUT} $0.25 * I_{OUT}$

K 0.4 toroides y 0.8 bobinas

D Diámetro del cable

$t_{OFF} =$ máximo tiempo muerto

A_e Area efectiva del núcleo

f frecuencia de conmutación

l_g Longitud de la abertura del núcleo

ΔV_{OUT} Voltaje de pico a pico

ESR_{MAX} Resistencia del capacitor

E_{in} Voltaje de Entrada.

E_{out} Voltaje de Salida.

Δt Máximo tiempo muerto.

ANEXO A1.3

Análisis de la Respuesta de
Frecuencia de la Red de Filtrado RFI

El objetivo de este análisis es verificar que los valores de los elementos de la red de filtrado RFI, son los adecuados para reducir el ruido producido por los elementos de conmutación de la fuente. Como expusimos en el capítulo 4, la frecuencia de conmutación del convertidor de la fuente es aproximadamente 30 KHz. El objetivo de la red de filtrado RFI es atenuar las señales producidas por la fuente de frecuencias cercanas o mayores a este valor. Es por esto que llevaremos el circuito inicial a su equivalente en dominio de frecuencia, como lo muestran las figuras 1 y 2 de este anexo.

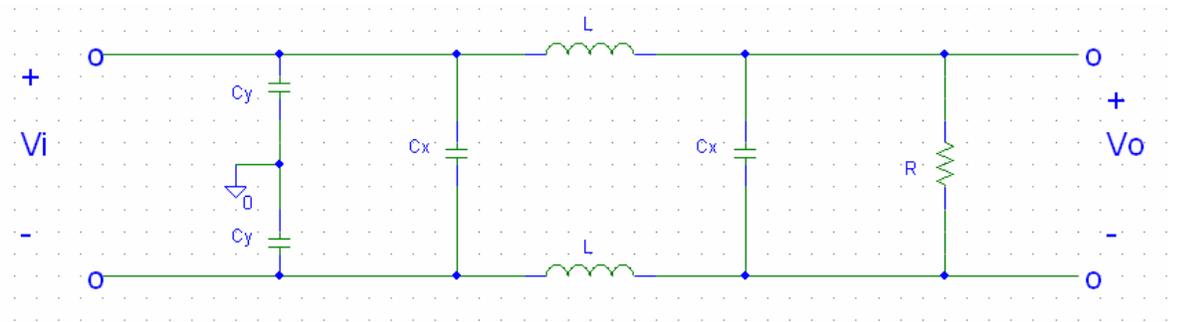


Fig.1. – Red de Filtrado RFI, existente en la fuente objeto de nuestro análisis.

Los valores de los elementos activos de la red, una vez aplicada la transformada de Laplace (en dominio de frecuencias) son los que se muestran en la siguiente figura:

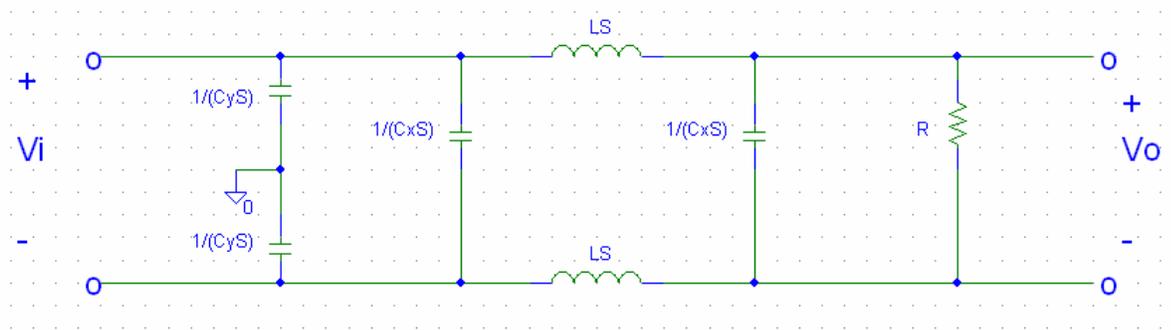


Fig. 2. – Circuito equivalente en dominio de frecuencia de la red de filtrado RFI.

Donde R es el valor de la resistencia equivalente del circuito.

Los inductores que forman parte de la red, comparten el mismo núcleo, pero el sentido del arrollamiento de las bobinas es cruzado (el sentido en que circula la corriente es contraria la una bobina de la otra), esto hace que las inductancias mutuas de las bobinas se anulen.

Luego de reducir la red, el circuito equivalente es el siguiente:

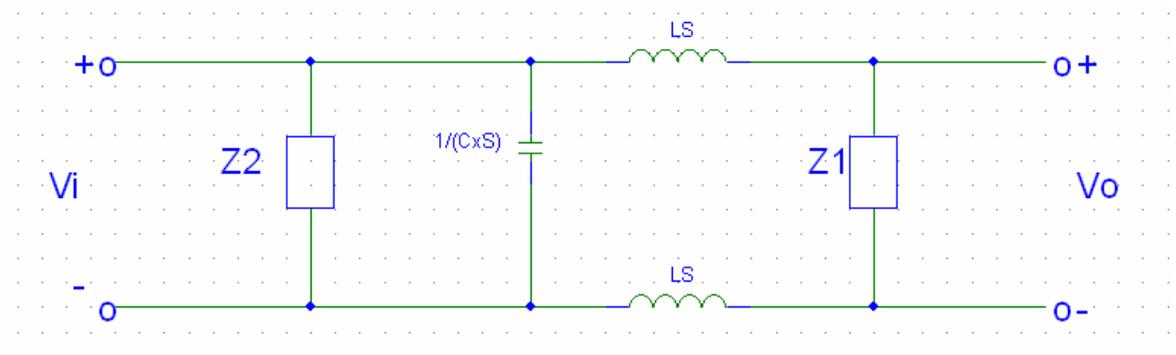


Fig. 3. – Red equivalente de la figura 2

Donde:

$$Z_1 = \frac{R}{1 + RCxS}$$

$$Z_2 = \frac{2}{CyS}$$

Esta red es posible reducirla aun más, la red equivalente final es la siguiente:

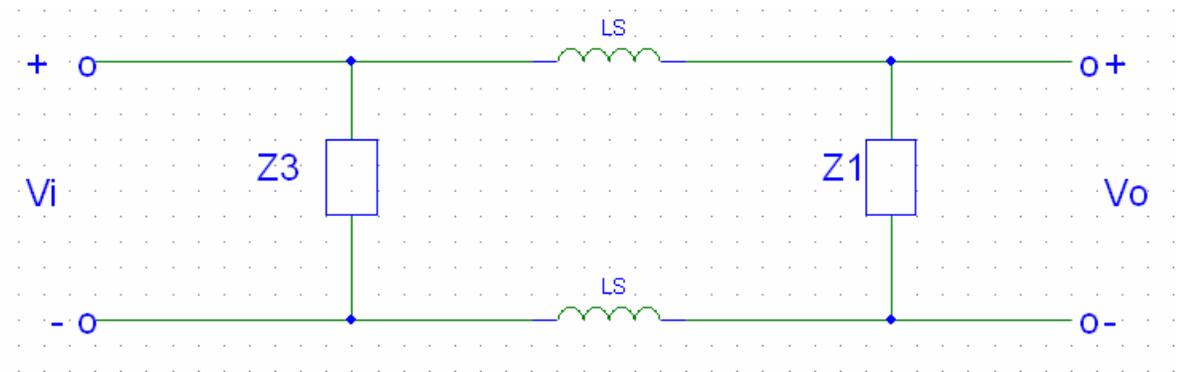


Fig. 4. – Equivalente final de la Red de Filtrado RFI

Donde:

$$Z_3 = \frac{2}{S(2Cx + Cy)}$$

Con esta red es posible encontrar la respuesta de frecuencia del circuito y de esta forma encontrar la o las frecuencias de corte del mismo. Para esto debemos encontrar la relación entre V_o y V_i .

$$V_o = V_i \frac{Z_1}{LS + Z_1 + LS}$$

Reemplazando Z1:

$$\frac{V_o}{V_i} = \frac{R}{2LS + R + 2S^2LRC_x} = G(s)$$

Este valor representa la ganancia de circuito, la misma que es de segundo orden y con dos polos complejos, la forma general de una ecuación de este tipo es:

$$G(s) = \frac{K}{(S / \omega_n)^2 + \frac{2\varepsilon S}{\omega_n} + 1}$$

Por lo tanto:

$$K = 1$$

$$\omega_n^2 = \frac{1}{2LC_x}$$

$$\frac{2\varepsilon}{\omega_n} = \frac{2L}{R}$$

Donde ω_n es el valor de la frecuencia de corte, que es el valor en el cual la ganancia se atenúa a razón de $-40 \frac{dB}{Decada}$.

Los valores de los elementos que forman parte de esta red son los siguientes:

$$L = 3.3mH$$

$$4700 = pF$$

El valor de R lo calculamos con el voltaje y la corriente de entrada, el voltaje al que hacemos los cálculos es 120 Vac y la corriente es la que medimos cuando la fuente trabaja a plena carga.

$$R = 64.32\Omega$$

Con estos valores hallamos que la frecuencia de corte para nuestro

circuito es $\omega_n = 180 \times 10^3 \frac{rad}{seg}$; llevando este resultado a Hz con la

siguiente ecuación:

$$\omega_n = 2\pi f$$

$$f = 28.64KHz.$$

Para señales con frecuencias mayores a este valor, su magnitud se atenuará a razón de $-40 \frac{dB}{Decada}$.

Con esto certificamos que la red de filtrado RFI reduce todos los ruidos generados por los elementos de conmutación de la fuente.

ANEXO A1.4

Resultados Obtenidos de las
Mediciones Realizadas a Plena Carga
de la fuente de conmutación con el
Osciloscopio TECTRONIC 2230

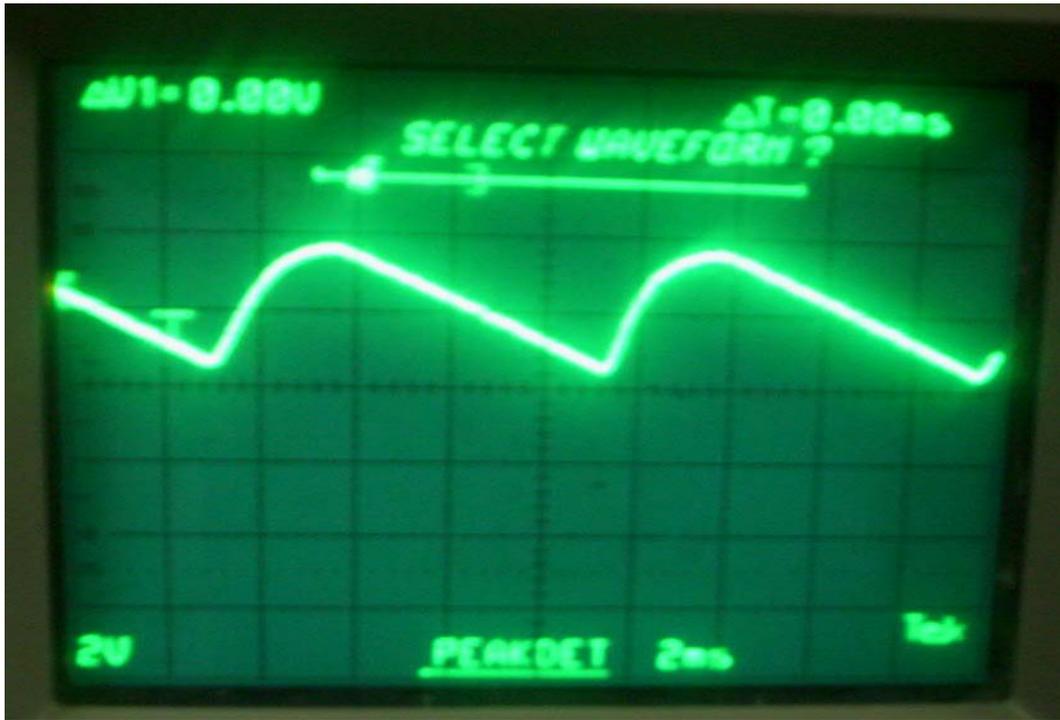


Figura A1: Voltaje de Rizado a la salida de la Etapa de Rectificación y Filtrado de Entrada.



Figura A2: *Voltaje Triangular Generado por el CI “KA7500B”, utilizado para comparar internamente el valor de entrada de la realimentación.*

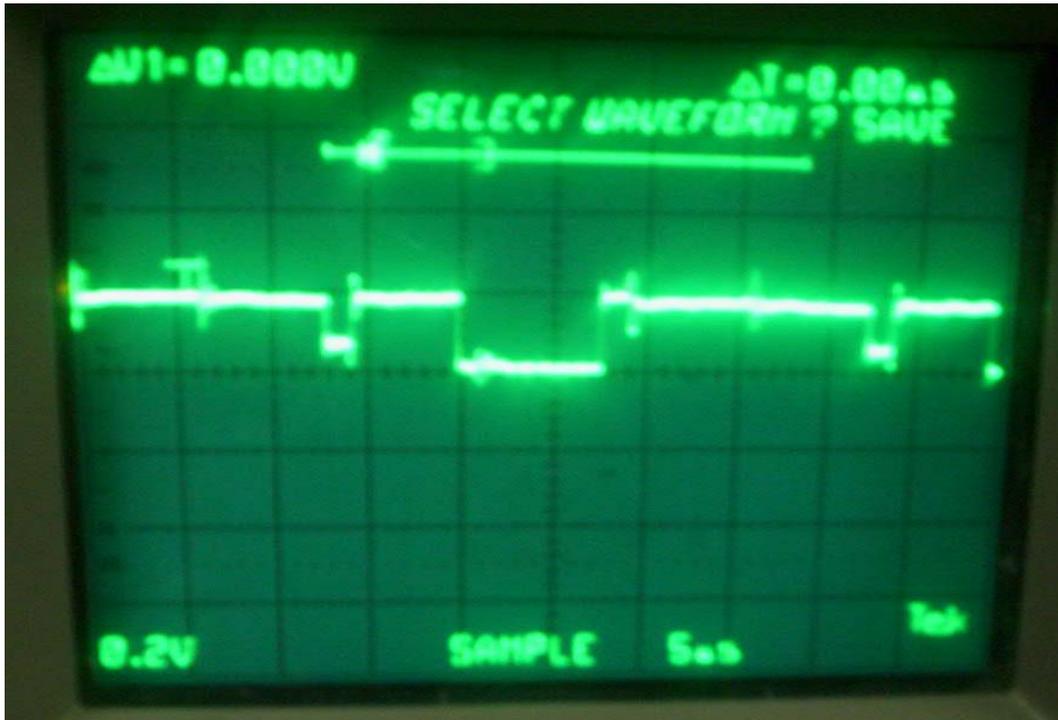


Figura A3: Salida hacia la base del Transistor $Q1$ desde el CI “KA7500B”. Esta Onda es exactamente igual para $Q2$ pero desfasada 180° .

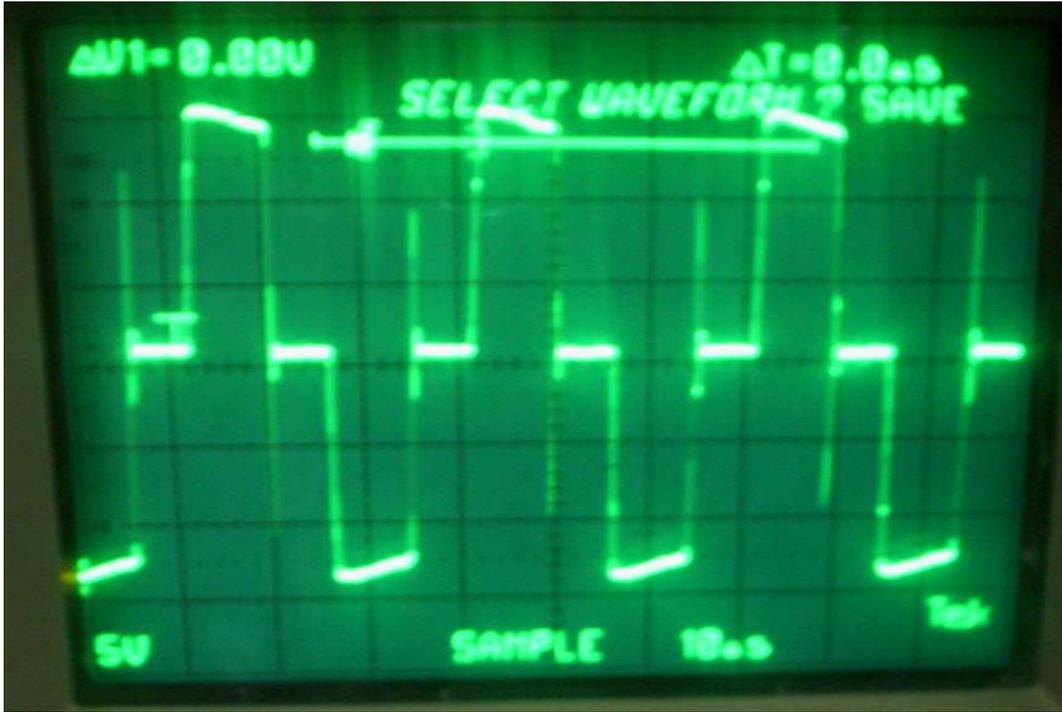


Figura A4: Salida hacia del convertidor “Medio Puente” aplicada al transformador de aislamiento T1.

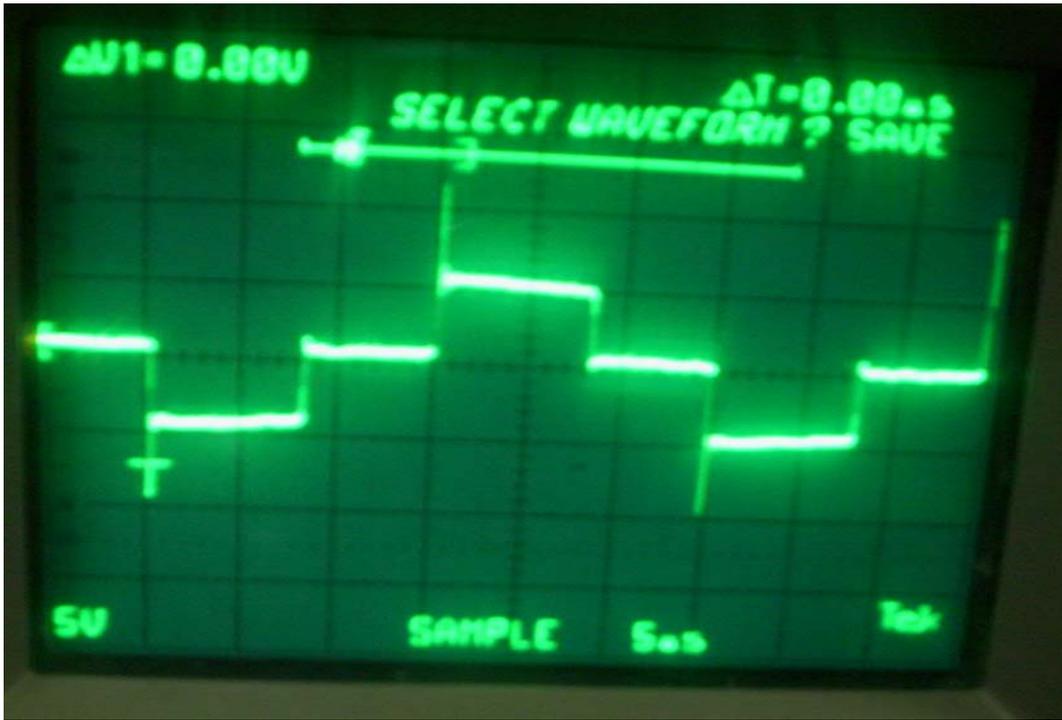


Figura A5. Voltaje medido en el Secundario del Transformador T1.

ANEXOS A1.5

DATASHEETS DE LOS
PRINCIPALES INTEGRADOS
DE LA FUENTE DE
CONMUTACION, DESCRITA
EN EL CAPITULO 3.

ANEXO B1.1

INTEGRADO ULN2003AN

ANEXO B1.2

INTEGRADO TLC7628CN

ANEXO B1.3

INTEGRADO TL084

ANEXO B1.4

INTEGRADO PIC16F877A

ANEXO B1.5

ACOPLADORES 4N25-4432

ANEXO B1.6

OPAM OPA4342

ANEXO B1.7

INTEGRADO MAX232

ANEXO B1.8

LABVIEW 7.1