



**INSTITUTO TECNOLÓGICO DE CHIHUAHUA**  
**DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**

---

***“HMRDT: HONEYWELL HERRAMIENTA DE  
DIAGNÓSTICO PARA MINNE RACK”***

***TESIS***

**PARA OBTENER EL GRADO DE**

***MAESTRA EN INGENIERÍA MECATRÓNICA***

**PRESENTA:**

***MARIANA LOYA ESTRADA***

**DIRECTOR DE LA TESIS:**

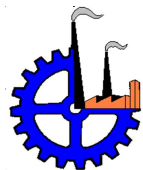
***cDR. ROGELIO BARAY ARANA***



**SEP**  
SECRETARÍA DE  
EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



CHIHUAHUA, CHIH., MÉXICO, JUNIO 2018

“Es importante adquirir el conocimiento de diferentes pensamientos, opiniones y puntos de vista. Si lo haces desde uno solo, te vuelves rígido y tedioso. Si entiendes al resto, serás alguien completo”

## DEDICATORIA

Dedicó esta tesis con todo mi amor y cariño a mis padres, porque ellos siempre estuvieron a mi lado brindándome su apoyo y sus consejos para hacer de mí una mejor persona.

Para Estefanía Loya, mi hermana, por haberme apoyado emocionalmente e incondicionalmente especialmente en estos dos años durante mi periodo de maestría, por sus consejos que nunca olvidaré y todo su amor.,

Para Pavel Medina, gracias por ser parte de mi vida y apoyarme en todos mis propósitos y metas, esto es principalmente para ti.



## **AGRADECIMIENTO**

A mi director de tesis, le agradezco por el tiempo, paciencia y trato que tuvo conmigo a lo largo de mi maestría, espero poder regresarle algún día todo el apoyo que me brindó.

A mis maestros y sinodales que dedicaron tiempo y paciencia para guiarme en el desarrollo de este documento, gracias.

A ICEM por brindarme la información, recursos y apoyo para hacer posible este proyecto, de igual manera a mis compañeros que me ayudaron de una u otra manera para la culminación del mismo.

## RESUMEN

### *HMRDT: HONEYWELL MINNE RACK DIAGNOSTIC TOOL*

Ing. Mariana Loya Estrada

Maestría en Ingeniería Mecatrónica

División de Estudios de Posgrado e Investigación

Instituto Tecnológico de Chihuahua

Chihuahua, Chih., 2018

Director de tesis: cDr. Rogelio Baray Arana

Este proyecto de tesis tiene como objetivo describir el proceso para desarrollo de una herramienta de diagnóstico que simplificará y agilizará el proceso de validación de los equipos denominados *Minne Rack*, los cuales son un equipo de prueba estándar utilizados por la compañía Honeywell, desarrollados y manufacturados por la empresa ICEM (Ingeniería en Control Electrónico y Mecánico, S.A. de C.V.). Esta herramienta será capaz de reconocer las fallas presentes en el equipo de forma inmediata, reduciendo significativamente el tiempo de depuración y validación del equipo, a la vez que se asegura del cumplimiento de los requerimientos del cliente.

En este documento se desarrollan los aspectos referentes al diseño mecánico y al diseño eléctrico-electrónico, así como al desarrollo del hardware y software requerido para la fabricación de la herramienta de diagnóstico HMRDT.

## **ABSTRACT**

### **HMRDT: HONEYWELL MINNE RACK DIAGNOSTIC TOOL**

Eng. Mariana Loya Estrada

Master's Degree in Mechatronics Engineering

Division of Graduate Studies and Research

Technological Institute of Chihuahua

Chihuahua, Chih., 2018

Thesis director: cDr. Rogelio Baray Arana

The purpose of this thesis project is to describe the process for the development of a diagnostic tool that will simplify and expedite the validation process of equipment called Minne Rack, which is a standard test equipment used by the company Honeywell, developed and manufactured by the ICEM company (Ingeniería en Control Electrónico y Mecánico, S.A de C.V). This tool will be able to recognize the faults present in the equipment immediately, significantly reducing the time of debugging and validation of the equipment, while ensuring compliance with customer requirements.

In this document, the aspects related to the mechanical design and electrical-electronic design are developed, as well as the hardware and software development required for the manufacture of the HMRDT diagnostic tool.

# Contenido

<b>LISTA DE FIGURAS .....</b>	<b>XII</b>
<b>LISTA DE TABLAS .....</b>	<b>XIV</b>
<b>CAPÍTULO I. CONCEPTOS GENERALES.....</b>	<b>15</b>
1.1 INTRODUCCIÓN .....	15
1.2 ANTECEDENTES.....	16
1.3 PROPÓSITOS Y EXPECTATIVAS.....	19
<b>CAPÍTULO II. MARCO TEÓRICO.....</b>	<b>22</b>
<b>2.1 SOFTWARE .....</b>	<b>22</b>
2.1.1 EAGLE PROFESSIONAL V9.0.0. ....	22
2.1.2 SOLIDWORKS .....	22
2.1.3 CODEGEAR DELPHI.....	23
2.1.4 CCS COMPILER .....	24
<b>2.2 EQUIPO .....</b>	<b>25</b>
2.2.1 MULTIFUNCIONAL SWITCH/MEASURE 34980A KEYSIGHT .....	25
2.2.2 34952A MODULO MULTIFUNCIONAL E/S .....	26
2.2.3 34921A MULTIPLEXOR DE BAJA FRECUENCIA .....	29
2.2.4 34937A MÓDULOS DE INTERRUPTOR DE PROPÓSITO GENERAL.....	31
<b>2.3 ANÁLISIS DE FALLAS. ....</b>	<b>32</b>
2.3.1 ANÁLISIS DEL MODO DE FALLA Y SUS EFECTOS (FMEA) .....	34
2.3.1.1 RESEÑA HISTÓRICA (FMEA).....	34
2.3.1.2 ANÁLISIS DE ÁRBOL DE FALLA (A.A.F). ....	36
2.3.1.3 EFECTOS DE LA FALLA POTENCIAL .....	36

2.3.1.4 CAUSAS DE FALLAS POTENCIALES.....	37
2.3.1.5 OCURRENCIA.....	37
2.3.1.6 DETECCIÓN.....	37
2.3.1.7 PRN.....	38
2.3.1.8 ACCIÓN (ES) RECOMENDADA (S).....	38
2.3.1.9 RPN RESULTANTE.....	38
2.3.2 DIAGRAMA CAUSA – EFECTO.....	39
2.3.3 POR QUE (5´W).....	39
<b>2.4 DISEÑO CONCEPTUAL.....</b>	<b>40</b>
2.4.1 DESCRIPCIÓN DE LOS REQUERIMIENTOS DE CLIENTE.....	40
<b>2.4.3 LAS ESPECIFICACIONES DE INGENIERÍA.....</b>	<b>42</b>
<b><u>CAPÍTULO III. DESCRIPCIÓN DEL TRABAJO.....</u></b>	<b><u>46</u></b>
<b>3.1 OBJETIVOS.....</b>	<b>46</b>
3.1.1 OBJETIVO GENERAL.....	46
3.1.2 OBJETIVOS ESPECÍFICOS.....	46
<b>3.2 HIPÓTESIS Y VARIABLES.....</b>	<b>46</b>
<b>3.3 JUSTIFICACIÓN.....</b>	<b>48</b>
<b>3.4 EXPLICACIÓN.....</b>	<b>49</b>
3.4.1 EXPLICACIÓN GENERAL.....	49
<b>3.4.2 DIAGRAMAS DE BLOQUES.....</b>	<b>50</b>
3.4.2.1 DIAGRAMA PRINCIPAL DEL SISTEMA.....	51
<b>3.4.3 EXPLICACIÓN TÉCNICAS.....</b>	<b>52</b>
<b><u>CAPÍTULO IV. RESULTADOS.....</u></b>	<b><u>80</u></b>
<b><u>CAPÍTULO V. CONCLUSIONES.....</u></b>	<b><u>89</u></b>

<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>92</b>
---	-----------

<b>ANEXOS .....</b>	<b>95</b>
---------------------	-----------

## **LISTA DE FIGURAS**

ILUSTRACIÓN 1.1 LOGO DE ICEM .....	15
ILUSTRACIÓN 1.2 CURVA DE BAÑERA.....	17
ILUSTRACIÓN 1.3 EQUIPO DE PRUEBA MINNE RACK.....	18
ILUSTRACIÓN 2.1 LOGO EAGLE .....	22
ILUSTRACIÓN 2.2 LOGO SOLIDWORKS.....	22
ILUSTRACIÓN 2.3 LOGO CODEGEAR DELPHI .....	23
ILUSTRACIÓN 2.4 34980 <sup>a</sup> KEYSIGHT.....	26
ILUSTRACIÓN 2.5 DIAGRAMA DE BLOQUES SIMPLIFICADO DEL 34952A .....	28
ILUSTRACIÓN 2.6 DIAGRAMA SIMPLIFICADO 34921A.....	30
ILUSTRACIÓN 2.7 DIAGRAMA SIMPLIFICADO 34937A.....	31
ILUSTRACIÓN 2.8 DIAGRAMA CAUSA-EFECTO .....	39
ILUSTRACIÓN 2.9 PROCESO DE LOS 5 ¿PORQUÉ? .....	39
ILUSTRACIÓN 2.10 DIAGRAMA DE BLOQUES .....	41
ILUSTRACIÓN 2.11 DIAGRAMA DE BLOQUES .....	41
ILUSTRACIÓN 2.12 DIAGRAMA DE BLOQUES .....	42
ILUSTRACIÓN 2.13 DIAGRAMA DE BLOQUES .....	42
ILUSTRACIÓN 3.1 DESCRIPCION DE LAS VARIABLES .....	47
ILUSTRACIÓN 3.2 DIAGRAMA DE BLOQUES .....	48
ILUSTRACIÓN 3.3 SUBSISTEMA MECANICO.....	55
ILUSTRACIÓN 3.4 SUBSISTEMA MECANICO (VISTA FRONTAL).....	55
ILUSTRACIÓN 3.5 SUBSISTEMA MECANICO (VISTA POSTERIOR) .....	56
ILUSTRACIÓN 3.6 SUBSISTEMA MECANICO (VISTA LATERAL ABIERTO) .....	56
ILUSTRACIÓN 3.7 SUBSISTEMA MECANICO (VISTA LATERAL CERRADO) .....	57
ILUSTRACIÓN 3.8 SUBSISTEMA MECANICO (VISTA LATERAL CERRADO) .....	57
ILUSTRACIÓN 3.9 DISEÑO INTEGRADO.....	58
ILUSTRACIÓN 3.10 DISEÑO INTEGRADO.....	59
ILUSTRACIÓN 3.11 DISEÑO INTEGRADO(PARTE FRONTAL).....	59

ILUSTRACIÓN 3.12 DISEÑO INTEGRADO (PARTE POSTERIOR) .....	59
ILUSTRACIÓN 3.13 SUBSISTEMA ELECTRONICO .....	60
ILUSTRACIÓN 3.14 LAYOUT DEL PCB .....	61
ILUSTRACIÓN 3.15 SLOT 1 POWER .....	62
ILUSTRACIÓN 3.16 SLOT FISICO.....	63
ILUSTRACIÓN 3.17 SLOT POWER VISTA INTERNA .....	63
ILUSTRACIÓN 3.18 CONEXIONES DEL BLOQUE DE POWER A PCB .....	64
ILUSTRACIÓN 3.19 CONEXIÓN DE MULTIPLEXORES .....	64
ILUSTRACIÓN 3.20 SLOT 2 RELAYS.....	65
ILUSTRACIÓN 3.21 SLOT 2 Y BLOQUES DE RELAYS .....	66
ILUSTRACIÓN 3.22 SLOT 3 MULTIPLEXORES .....	67
ILUSTRACIÓN 3.23 SLOT 3 MULTIPLEXORES CONEXIONES .....	68
ILUSTRACIÓN 3.24 SLOT 4 USB .....	69
ILUSTRACIÓN 3.25 SLOT 7 INTERLOCKS .....	70
ILUSTRACIÓN 3.26 SLOT 8 Y 9 AIR.....	71
ILUSTRACIÓN 3.27 SLOT 10 COAXIAL.....	72
ILUSTRACIÓN 3.28 SLOT 11 DIO .....	73
ILUSTRACIÓN 3.29 PIC18F2550.....	74
ILUSTRACIÓN 3.30 E/S DE LAS TARJETAS QUE ESTAN INTERCONECTADAS CON EL RACK.....	74
ILUSTRACIÓN 3.31 SECCION DIO PCB .....	75
ILUSTRACIÓN 3.32 DIAGRAMA LOGICO (LOGICA POSITIVA).....	76
ILUSTRACIÓN 3.33 DIAGRAMA SIMPLIFICADO DE CADA PUERTO E/S.....	76
ILUSTRACIÓN 3.34 DIAGRAMA DE BLOQUES .....	77
ILUSTRACIÓN 3.35 INSTRUCCIONES PARA MANUFACTURAR PCB.....	78
ILUSTRACIÓN 3.36 PCB MANUFACTURADA .....	78
ILUSTRACIÓN 4.1 PRUEBA DE VOLTAJES.....	81
ILUSTRACIÓN 4.2 RESULTADO PRUEBA VOLTAJE.....	81
ILUSTRACIÓN 4.3 PRUEBA MULTIPLEXORES .....	82
ILUSTRACIÓN 4.4 RESULTADOS DE LOS MULTIPLEXORES.....	82
ILUSTRACIÓN 4.5 PRUEBA DE RELAYS .....	83
ILUSTRACIÓN 4.6 RESULTADOS RELAYS.....	83
ILUSTRACIÓN 4.7 DESCONEXION -12VDC.....	84
ILUSTRACIÓN 4.8 PRUEBA -12VDC .....	84
ILUSTRACIÓN 4.9 RESULTADOS -12VDC.....	85

ILUSTRACIÓN 4.10 DESCONEXION DE 12VDC .....	85
ILUSTRACIÓN 4.11 PRUEBA DE 12VDC .....	86
ILUSTRACIÓN 4.12 RESULTADO 12VDC .....	86
ILUSTRACIÓN 4.13 GRAFICA DE RESULTADOS .....	88

## LISTA DE TABLAS

TABLA 2.1 REQUERIMIENTOS .....	40
TABLA 2.2 ESPECIFICACIONES DE INGENIERÍA.....	44
TABLA 3.1 FICHA TÉCNICA MINNE RACK .....	52
TABLA 3.2 TABLA DE ESPECIFICACIONES DEL <i>HMRDT</i> . .....	53



## CAPÍTULO I. CONCEPTOS GENERALES

### 1.1 Introducción

El presente documento contiene la descripción de las actividades de investigación y desarrollo tecnológico realizadas para el proyecto de tesis para obtención del grado de Maestría de Ingeniería en Mecatrónica del Departamento de Estudios de Posgrado e Investigación (DEPI) del Tecnológico Nacional de México en Chihuahua, Chih. Este proyecto fue desarrollado en el departamento de diseño de la empresa Ingeniería en Control Electrónico y Mecánico S.A. de C.V. (ICEM) [1], la cual se dedica al diseño y manufactura de equipos de inspección, ensamble y prueba para la industria manufacturera, ubicada en la ciudad de Chihuahua, Chih., México, contando como asesor de tesis al cDr. Rogelio E. Baray Arana.

ICEM es una empresa 100% Chihuahuense, fundada en 1995 bajo el nombre de Luis Antonio Montoya Jaquez, actualmente ubicada en el complejo industrial Chihuahua, dedicada a dar soporte a la industria manufacturera, mediante el desarrollo de proyectos de automatización y control enfocados a mejorar la productividad y calidad de sus procesos.

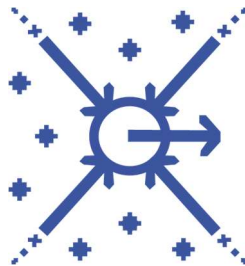


Ilustración 1.1 Logo de ICEM

La presente tesis titulada “*HMRDT: HONEYWELL HERRAMIENTA DE DIAGNOSTICO PARA MINNE RACK*” o por sus siglas en inglés, “*HONEYWELL MINNE RACK DIAGNOSTIC TOOL*”, consiste en el diseño y desarrollo de una *fixtura* de prueba para diagnosticar fallas en los equipos de prueba estándar utilizados por Honeywell denominados *Minne Rack*.

Para el desarrollo de este proyecto se requirió el apoyo de las áreas de diseño mecánico y diseño electrónico de la empresa antes mencionada, así como de las áreas de compras, almacén e integración de equipo; obteniendo así un sistema que integra diversas áreas de conocimiento bajo un enfoque mecatrónico (mecánico/electrónico) y con la aplicación de alta tecnología, enfocados en la satisfacción de las necesidades y expectativas del cliente y el cumplimiento de sus requerimientos.

### **1.2 Antecedentes**

Actualmente el concepto de confiabilidad, es ampliamente conocido en el ambiente de diseño de productos y en la manufactura del producto, sin embargo, su alcance ha evolucionado en la medida que se ha comprendido que no era suficiente lograr una alta disponibilidad de equipos y productos en el mercado, sino también disminuir al mínimo la probabilidad de falla de las máquinas críticas, durante la operación, es decir, lograr conseguir una alta confiabilidad en su operación, para garantizar la eficacia de la operación y eficientar el uso del equipo.

Las consecuencias de una falla pueden ir desde la pérdida económica y de producción, pasando por las horas hombre improductivas de operaciones, hasta la degradación y rotura de las propias máquinas.

## CAPÍTULO I

Una alta disponibilidad no implica necesariamente una alta confiabilidad, pero una alta confiabilidad si implica una buena disponibilidad y seguridad, en la medida que la maquinaria, el proceso o equipos, presentan una baja probabilidad de falla.

Para el caso de la maquinaria pesada, la confiabilidad será el producto de la confiabilidad individual de cada sistema que la compone.

Cuando hay una falla la pieza queda completamente inservible, o a pesar que la maquina funciona, no cumple su función satisfactoriamente, es decir, cuando su funcionamiento es poco confiable debido a las fallas y presenta riesgos.

Las causas de esto se debe a:

- Mal diseño, mala selección del material
- Imperfecciones del material, del proceso y/o de su fabricación.
- Errores en el servicio y en el montaje
- Errores en el control de Calidad, mantenimiento y reparación
- Factores ambientales, sobrecargas.

La curva de la bañera, es una grafica que representa los fallos durante el periodo de vida útil de un sistema o maquina. Se llama así porque tiene la forma de una bañera cortada a lo largo.

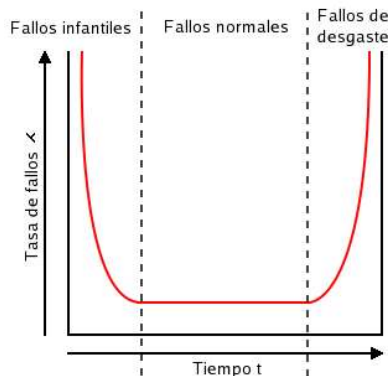


Ilustración 1.2 Curva de bañera

## CAPÍTULO I

Así pues el reconocimiento oportuno de fallas en un equipo de prueba industrial es una actividad fundamental para la optimización y buen funcionamiento del proceso productivo de una empresa tal como lo es Honeywell [2]. Identificar el origen de una falla en un lapso de tiempo prácticamente inmediato, hace que su depuración y eliminación sea más ágil y eficaz, lo cual minimiza el tiempo muerto y maximiza el tiempo de producción.

ICEM ha desarrollado los equipos de prueba Minne Rack para la empresa Honeywell desde el año 2009, produciendo en promedio 5 equipos al año. En este tiempo se han empleado técnicas de debug o depuración de fallas poco eficientes, que representaban una inversión de tiempo significativa para validar al 100% el equipo y asegurar su conformidad con las especificaciones y requerimientos del cliente; cuando esta cantidad de tiempo se invierte en tareas no productivas, puede representar, a la larga, pérdidas de millones de dólares en producción.

Desafortunadamente, aunque se han probado varias técnicas para acelerar el proceso de reconocimiento de fallas, aun no se cuenta con un sistema preciso y más eficiente para el diagnóstico de estos equipos.



Ilustración 1.3 Equipo de prueba Minne Rack

### 1.3 Propósitos y expectativas

La detección de fallas de los equipos denominados *Minne Rack*, desarrollados por ICEM para la empresa *Honeywell*, ha sido hasta la fecha de manera manual, lo que resulta poco práctico y funcional, e impacta de forma negativa en el cumplimiento de los objetivos de entrega de ICEM y los tiempos de producción de *Honeywell*; sin embargo, el desarrollo e implementación del uso de una herramienta de diagnóstico para *debug* y validación de estos equipos, daría solución a este problema, mejorando el desempeño del equipo, minimizando tiempos de entrega y dando mayor confiabilidad a los resultados obtenidos.

La herramienta de diagnóstico que se pretende desarrollar (HMRDT), deberá ser capaz de:

- a) Analizar cada una de las tarjetas instaladas en el equipo,
- b) Medir voltajes y cargas [3],
- c) Detectar fuentes de alimentación e instrumentos [4] y
- d) Estandarizar el proceso de diagnóstico y validación del equipo en un tiempo mínimo, logrando así mejores resultados y un mayor desarrollo del producto.

El sistema HMRDT, combina las fuerzas complementarias de sus componentes y sensores, lo que resulta en una alineación rápida y precisión de navegación óptima, tal y como sucedió en 1990, con un modelo de prueba de desarrollo avanzado para el sistema de navegación integrado de helicópteros, probado en el centro de tecnología avanzada (ATC) de *Honeywell* [5]. Una funcionalidad de aislamiento y reconfiguración de detección de fallos monitoriza la integridad del sensor, identifica los componentes fallidos y reconfigura automáticamente el sistema para integrar de forma óptima los componentes restantes, proporcionando así una degradación mínima del rendimiento en caso de un fallo del sensor. Este tipo de pruebas han dado a estos equipos un alto

grado de confianza en el diseño del sistema de navegación y en el software que implementa los algoritmos de mezcla de sensores en el sistema en tiempo real. [6]

El propósito principal de esta tesis es diseñar e integrar una herramienta de diagnóstico para equipos de prueba *Minne Rack* para la detección de fallas; lo que para un ingeniero puede ser muy complejo de identificar, para un equipo de diagnóstico es una tarea sencilla.

Este problema da origen a muchos otros, como, por ejemplo: identificar en dónde exactamente ocurrió la falla del equipo, qué voltajes y cargas permiten la validación de éste [7], dónde se registraron las fallas, cómo asociar herramientas de medición programables, qué valores de medición son los correctos, etcétera.

El funcionamiento de la herramienta de diagnóstico sugerida, es tan complejo, que para su desarrollo se ha requerido invertir mucho tiempo de investigación en sus diferentes secciones. La herramienta de diagnóstico de *Minne Rack* presenta, en esencia, los siguientes retos [8]:

- a) Medición: el equipo debe medir todos los voltajes y cargas de los instrumentos y recursos instalados tales como: fuentes opcionales y programables, tarjetas de *RELAYS, MUX, DIO, INTERLOCK, COMM*
- b) Datos de los recursos del equipo a probar: Criterios a utilizar para determinar si los datos resultantes son correctos.
- c) Reconocimiento de fallas: El equipo debe reconocer en dónde exactamente ocurrió la falla.
- d) Interfaz gráfica: el equipo debe tener una interfaz gráfica amigable para el usuario en la cual se desplieguen los datos medidos y qué fue lo que falló.
- e) Registro de fallas: El sistema debe guardar un registro de fallas y compilación para usos prácticos del usuario (justificaciones de paro de producción para la planta).

## CAPÍTULO I

El objetivo de la investigación se centra en el problema de sistemas de medición y representación gráfica, es decir, es necesario aplicar un sistema de medición que permita definir datos reales y físicos que podamos comparar con los datos ideales representándolos visualmente. Por lo tanto, el equipo que se pretende desarrollar constará de los siguientes subsistemas:

- Sistema de medición de parámetros eléctricos [3], [9].
- Sistema de electroválvulas para probar las salidas de aire del equipo.
- Sistema mecánico para ensamble en el *Minne Rack*.
- Hardware representado en *PCB* [10] [11] [12] [13].

La estructura del presente documento es la siguiente: En el capítulo II se muestra el marco teórico que incluye trabajos realizados con anterioridad, en el capítulo III se establece la definición del problema, objetivos y metas, en el capítulo IV se muestra como se desarrollo el proyecto en todas sus etapas, en el capitulo V se hace referencia a los resultados obtenidos en la elaboración del proyecto y la implementación de la herramienta de diagnostico en los equipos *Minne Rack*, así como a las conclusiones del trabajo realizado.

## CAPÍTULO II. MARCO TEÓRICO

### 2.1 Software

#### 2.1.1 EAGLE professional v9.0.0.

*EAGLE* (por sus siglas en inglés *Easily Applicable Graphical Layout Editor*) es un programa de diseño de diagramas de circuito eléctrico y *PCBs*, con el auto enrutador más famoso alrededor del mundo de los proyectos electrónicos DiY, debido a que muchas versiones de este programa tienen una licencia Freeware y gran cantidad de bibliotecas de componentes alrededor de la red. [14].



Ilustración 2.1 Logo EAGLE

#### 2.1.2 SolidWorks

*SolidWorks* es un software *CAD* (diseño asistido por computadora) para modelado mecánico en 2D y 3D, desarrollado en la actualidad por *SolidWorks Corp.*, una filial de *Dassault Systèmes, S.A. (Suresnes, Francia)*, para el sistema operativo Microsoft Windows. Su primera versión fue lanzada al mercado en 1995 con el propósito de hacer la tecnología *CAD* más accesible.



Ilustración 2.2 Logo SolidWorks



El programa permite modelar piezas y conjuntos y extraer de ellos tanto planos técnicos como otro tipo de información necesaria para la producción. Es un programa que funciona con base en las nuevas técnicas de modelado con sistemas CAD. El proceso consiste en traspasar la idea mental del diseñador al sistema CAD, "construyendo virtualmente" la pieza o conjunto de piezas. Posteriormente todas las extracciones (planos y ficheros de intercambio) se realizan de manera automatizada [15].

### 2.1.3 CodeGear Delphi

Embarcadero Delphi, antes conocido como *CodeGear Delphi*, *Inprise Delphi* y *Borland Delphi*, es un entorno de desarrollo de software diseñado para la programación de propósito general con énfasis en la programación visual.

En *Delphi* se utiliza como lenguaje de programación una versión moderna de Pascal llamada *Object Pascal*. Es producido comercialmente por la empresa estadounidense *CodeGear* (antes lo desarrollaba Borland), adquirida en mayo de 2008 por *Embarcadero Technologies*, una empresa del grupo Thoma Cressey Bravo, en una suma que ronda los 30 millones de dólares. En sus diferentes variantes, permite producir archivos ejecutables para *Windows*, *MacOS X*, *iOS*, *Android*, *GNU/Linux* y la plataforma *.NET*. [16]



Ilustración 2.3 Logo CodeGear Delphi

### 2.1.4 CCS Compiler

El Portable *C Compiler* o Compilador de C Portable (también conocido como pcc o, algunas veces, como pccm - máquina de compilador de C portable) fue uno de los primeros compiladores para el lenguaje de programación C escrito por Stephen C. Johnson de los Laboratorios *Bell* a mediados de los 70, basado, en parte, en las ideas de un trabajo previo de Alan Snyder en 1973. Fue muy influyente en su tiempo como uno de los primeros compiladores que podía ser fácilmente adaptado con el fin de generar código para diferentes arquitecturas de computadora.

A principios de los 80s, la mayoría de los compiladores de C estaban basados en pcc. El compilador ha tenido una larga vida, siendo incluido desde la versión 4.3BSD-Reno en 1990 hasta que el compilador de C de GNU lo sustituyó en 4.4BSD.

Las claves para el éxito del pcc fueron sus capacidades de portabilidad y diagnóstico:

- El compilador fue diseñado de tal manera que solamente unos pocos ficheros de su código fuente eran dependientes de la máquina.
- Era robusto ante los errores de sintaxis de forma que rehusaba compilar programas inválidos.
- Aislaba el código dependiente de la máquina que era necesario reescribir manualmente en lenguaje ensamblador de la máquina de destino.
- Auto-optimizaba en la primera pasada.

Estas características eran novedosas en los tiempos en los que pcc fue escrito. El primer compilador de C, escrito por Dennis Ritchie usando como método un analizador sintáctico descendente recursivo estaba fuertemente ligado al PDP-11, y dependía de un optimizador -dependiente de la máquina- que mejoraba el código en

una segunda pasada. Por el contrario, el compilador multi-pasada de Johnson tenía en cuenta el número de accesos a memoria en las secuencias de código de cada expresión generando el código con el menor número de accesos a memoria [17].

### **2.2 Equipo**

#### **2.2.1 Multifuncional *switch/measure* 34980A Keysight**

El multifuncional 34980A proporciona una funcionalidad que es fácil de configurar y acelera el tiempo de prueba de un equipo. Maneja un cambio en el sistema arriba de 26.5 GHz y proporciona mediciones básicas y control total o parcial del sistema. También ofrece medidores *DMM*, contadores, totalizador, E/S digitales con capacidades de patrón y salidas analógicas con formas de onda básicas.

El 34980A aloja hasta 8 módulos enchufables para brindar la flexibilidad que se necesita, si se está midiendo la temperatura, el voltaje AC o DC, resistencia, frecuencia, corriente o mediciones personalizadas, el 34980A ofrece la funcionalidad que necesita.

Conmuta en diferentes medidas con una señal de alto rendimiento que conmuta hasta 300V sin necesidad de acondicionamiento de una señal externa. Se puede elegir entre diferentes tipos de interruptores y topologías con rangos de frecuencia desde DC hasta 26.5GHz.

Este sistema ofrece multiplexores de alta densidad para escanear múltiples canales, matrices para conectar múltiples puntos en uno.

Interruptores de tiempo y de propósito general para control simple y necesidades de alta potencia.

El multifuncional tiene conmutaciones de densidad media a alta, aplicaciones como verificación de diseño, pruebas funcionales y adquisición de datos.

Sus señales se cambian al dispositivo de medición correcto sin comprometer la integridad de la señal.

## CAPÍTULO II

Se puede acceder al DMM a través de cualquier módulo de conmutador que se conecta al bus analógico, o directamente desde el conector del mismo, este DMM brinda la flexibilidad de medir 11 tipos de entradas:

- Temperatura con termopares, RTD o termistores
- Tension DC y AC
- Resistencia de 2 y 4 hilos
- Frecuencia y periodo
- Corriente DCy AC

Cada canal de conmutación puede configurarse independientemente para la medición.



Ilustración 2.4 34980A Keysight

### 2.2.2 34952A Modulo Multifuncional E/S

El módulo multifuncional 34952A con *DIO*, *D/A* y totalizadores, combina cuatro puertos de 8 bits de entrada/salida digital, un totalizador de 100KHz y dos de +- 12 salidas analógicas con referencia a tierra.

**E/S Digitales.** Las entradas/salidas digitales (*DIO*) constan de 4 puertos de 8 bits con entradas y salidas compatibles con TTL. Las salidas de drenaje abierto pueden hundirse hasta 400mA. Desde el panel frontal se puede leer datos de una sola entrada

## CAPÍTULO II

de 8 bits a la vez. Se puede configurar operaciones de los puertos de DIO para 8, 16 o 32 bits.

Los canales DIO están conectados mediante resistencias internas de pull-up a 5V cuando están configuradas como entradas.

**Entrada de totalizador.** El totalizador de 32 bits puede contar pulsos a una velocidad de 100 kHz. Se puede configurar para contar con el flanco ascendente o el flanco descendente de la señal de entrada.

Una señal alta TTL aplicada a la terminal *Gate* permite el conteo y una señal baja desactiva el conteo. Una señal baja TTL aplicada a la terminal *Not-gate* habilita el conteo y una señal alta desactiva el conteo. Los totalizadores cuentan solo cuando ambas terminales están habilitadas.

**Salidas Analógicas (DAC).** Las dos salidas analógicas son capaces de emitir voltajes entre +-12 voltios con 16 bits de resolución. Cada canal DAC es capaz de 10mA de máxima corriente. Se utilizan las dos salidas analógicas para generar voltajes de polarización hacia el *DUT* para controlar sus fuentes de alimentación analógicas programables, o según lo establecido en el sistema de control. Las salidas se programan directamente en voltios.

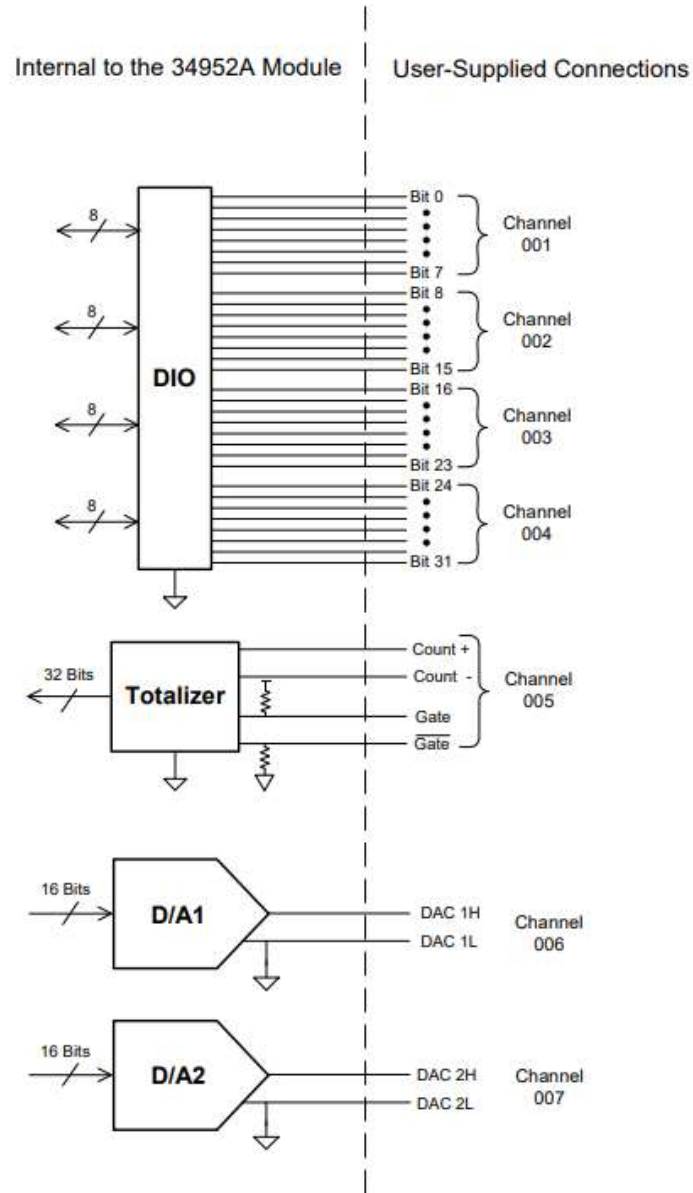


Ilustración 2.5 Diagrama de bloques simplificado del 34952A

### 2.2.3 34921A Multiplexor de baja frecuencia

El multiplexor de 34921A de 40 canales, está dividido en dos bancos con 20 interruptores de armadura de enclavamiento (canales 1-20 y 21-40) en cada uno. Este módulo ofrece cuatro relés con fusibles adicionales (canales 41-44) para realizar mediciones de corriente de AC y DC con el DMM interno sin necesidad de derivaciones externas. Estos canales cuentan con conexiones “*make-before-break*” para garantizar un flujo de corriente continuo al cambiar una corriente de un canal a otro.

Este módulo también contiene nueve relés de bus analógico (canales 911-914, 921-924 y 931) cuatro en cada banco que pueden conectar el banco de relés al sistema de buses analógicos y uno que conecta los relés de corriente a la corriente de entrada del DMM. A través de Abus1 y Abus2 se puede conectar cualquiera de los canales al DMM interno para mediciones de tensión o resistencia.

Usando los comandos del programa o el panel frontal del mainframe (34980A) puede controlar cada uno de los canales individualmente, y así configurarlo en los siguientes modos:

- Dos Muxes independientes de 20 canales y 2 hilos. Esta configuración no requiere el uso de cableado externo ni la conexión a través de los buses analógicos internos.
- Un mux de 20 canales y 4 hilos. Esta configuración no requiere usar cableado externo ni conexión a través de los buses analógicos internos.

Para mediciones de resistencia de 4 cables, el instrumento se empareja automáticamente en el canal “n” en el banco 1 (fuente) con el canal “n+20” en el banco

2 (senzar) para proveer estas conexiones. Este control solo ocurre cuando existen los 4 cables.

- Un mux de 40 canales y 2 cables. Debe utilizar cableado externo o conectarse a través de el bus analógico interno para esta configuración. Por ejemplo, cerrando el bus de los canales analogicos 913 y 923, conecta el banco uno y dos a través del Abus3 o externamente puede conectar COM1 a COM2 para crear esta configuración.

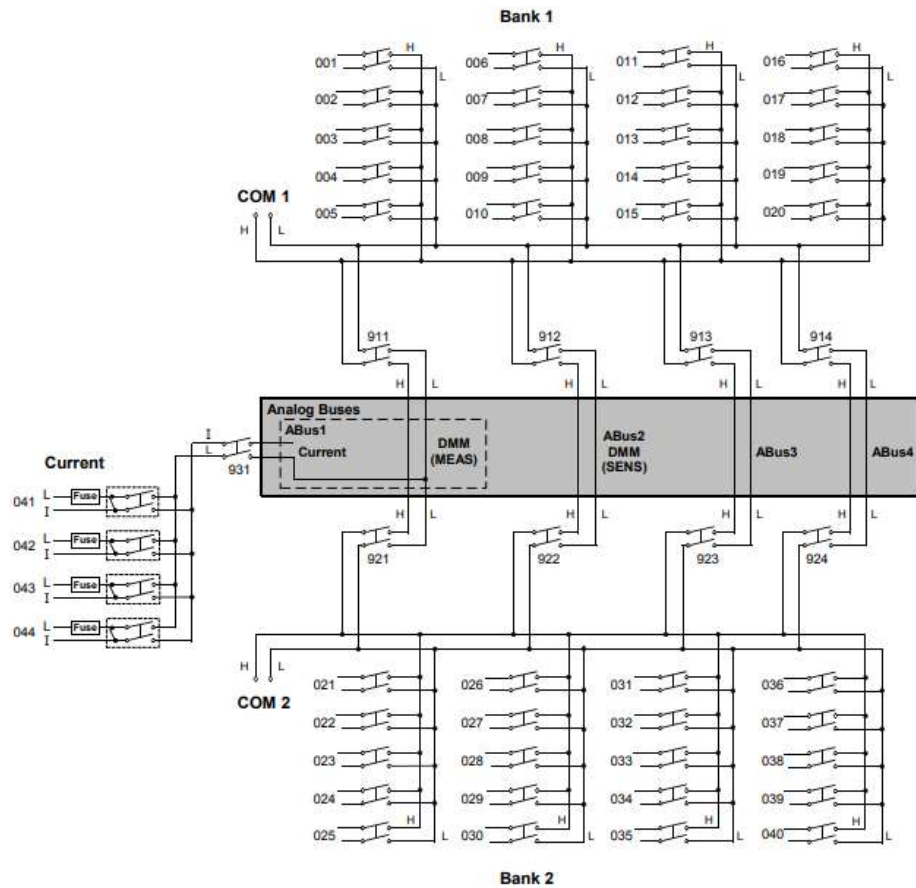


Ilustración 2.6 Diagrama Simplificado 34921A



### 2.2.4 34937A Módulos de interruptor de propósito general

El 34937A proporciona control independiente de 32 relés que incluyen:

- 28 relés de forma C, cada uno clasificado para 1Amp a 60W por canal
- Cuatro relés de forma A (SPST), cada uno clasificado para 5 Amp a 150W por canal.

Un sensor de temperatura en estos módulos dispara interrupciones cuando la alta corriente induce un calor excesivo en los módulos, esta situación de sobrecalentamiento genera un evento SRQ cuando alcanza el umbral de 70° C.

Cargas reactivas (aquellas que incluyen inductancia o capacitancia significativa) puede causar picos de voltaje o picos de corriente durante las operaciones de conmutación.

Los módulos de propósito general están diseñados para conmutar cargas reactivas.

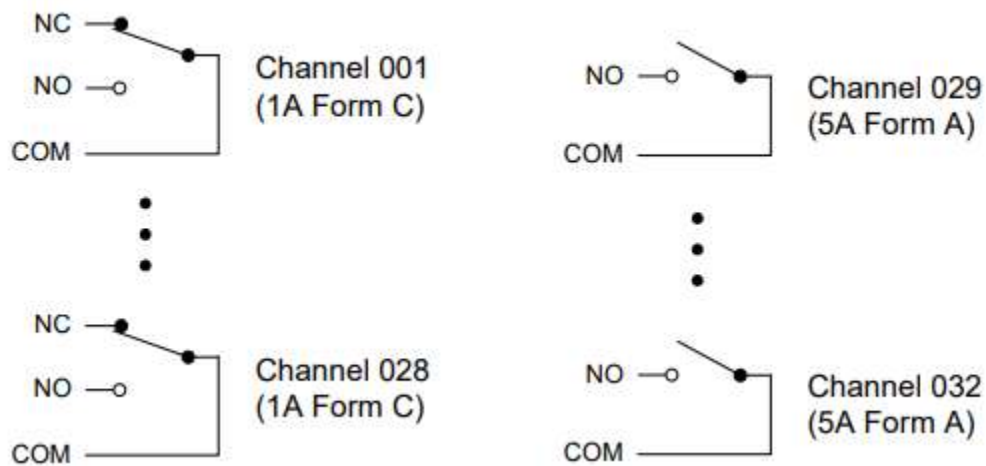


Ilustración 2.7 Diagrama simplificado 34937A

### **2.3 Análisis de fallas.**

El programa de detección analítica de fallas DAF, proporciona las habilidades y destrezas para la solución y prevención de problemas en ambientes productivos, acompañando los esfuerzos de mejoramiento continuo.

Algunos beneficios mas evidentes del programa son:

- Reducción del tiempo de reparación
- Minimización de tiempo de preparación y arranque de quipos
- Disminución de fallas repetitivas
- Aumento en la disponibilidad de equipos
- Reducción de retrabajos y desperdicio
- Reducción en la frecuencia de fallas
- Mejora del mantenimiento preventivo
- Reducción de costos por fallas de calidad
- Mayor eficiencia en el trabajo en equipo

Es indispensable que el departamento de conservación cuente invariablemente con un inventario de conservación, el cual es un listado de los recursos por atender, sean estos equipos, instalaciones o construcciones; y que además se haya establecido el índice ICGM (RIME).

De esta forma, utilizando el código máquina y combinándolo con el principio de Pareto, obtenemos el inventario jerarquizado de conservación (vital, importante y trivial).

## CAPÍTULO II

El análisis de falla es un examen sistemático de la pieza dañada para determinar la causa raíz de la falla, y usar esta información para mejorar la confiabilidad del producto.

El análisis de falla está diseñado para:

- Identificar los modos de falla (la forma de fallar del equipo)
- Identificar el mecanismo de falla (el fenómeno físico involucrado en la falla)
- Determinar la causa raíz (el diseño, defecto, o cargas que llevaron a la falla)
- Recomendar métodos de prevención de la falla

Las causas mas comunes de falla se derivan por (la lista no es exhaustiva):

- Mal uso o abuso de los equipos.
- Errores de montaje
- Errores de fabricación
- Mantenimiento inadecuado
- Errores de diseño
- Material inadecuado
- Tratamientos termicos incorrectos
- Condiciones no previstas de operación
- Inadecuado control o protección ambiental
- Discontinuidades de colada
- Defectos de soldadura
- Defectos de forja

Los tres principios básicos a respetar:

- Localizar el origen de la falla
- No presuponer una causa determinada
- No realizar ensayos destructivos sin un análisis previo cuidadoso

### **2.3.1 Análisis del modo de falla y sus efectos (FMEA)**

El análisis del modo de falla y sus efectos (FMEA por sus siglas en inglés), es un proceso sistemático para la identificación de las fallas potenciales del diseño de un producto para identificar en etapas tempranas las deficiencias del diseño y también para identificar las capacidades de un proceso en la manufactura del producto y evitar que introduzca fallas potenciales en el producto. Las fallas se detectan antes de que estas ocurran, con el propósito de eliminarlas o de minimizar el riesgo asociado a las mismas.

Por lo tanto, el FMEA puede ser considerado como un método analítico estandarizado para detectar y eliminar problemas de forma sistemática y total, cuyos objetivos principales son:

- Reconocer y evaluar los modos de fallas potenciales y las causas asociadas con el diseño y manufactura de un producto.
- Determinar los efectos de las fallas potenciales en el desempeño del sistema y la gravedad de sus efectos.
- Identificar las acciones que podrán eliminar o reducir la oportunidad de que ocurra la falla potencial.
- En el caso del diseño ayuda a identificar las debilidades en los controles del proceso de diseño.
- Analizar la confiabilidad del sistema, por medio de identificar las deficiencias en el diseño del producto desde sus etapa conceptual.
- Realizar la formalización de la documentación el proceso de diseño del producto.

#### **2.3.1.1 Reseña histórica (FMEA)**

La técnica del FMEA fue desarrollada en el ejército de los Estados Unidos por los ingenieros de la *National Agency of Space and Aeronautical* (NASA), y era conocido como el procedimiento militar MIL-P-1629, titulado “Procedimiento para la ejecución

## CAPÍTULO II

de un Modo de Falla sus Efectos y el Análisis de lo crítico de sus efectos y elaborado el 9 de noviembre de 1949; este era empleado como una técnica para evaluar la confiabilidad y para determinar los efectos de las fallas de los equipos y sistemas, en el éxito de la misión y la seguridad del personal o de los equipos.

En 1988 la Organización Internacional para la Estandarización (ISO), publicó la serie de normas ISO 9000 para la gestión y el aseguramiento de la calidad; los requerimientos de esta serie llevaron a muchas organizaciones a desarrollar sistemas de gestión de calidad enfocados hacia las necesidades, requerimientos y expectativas del cliente, entre estos surgió en el área automotriz el QS 9000, este fue desarrollado por la Chrysler Corporation, la Ford Motor Company y la General Motors Corporation en un esfuerzo para estandarizar los sistemas de calidad de los proveedores; de acuerdo con las normas del QS 9000 los proveedores automotrices deben emplear Planeación de la Calidad del Producto Avanzada (APQP), la cual necesariamente debe incluir FMEA de diseño (DFMEA) y de proceso (PFMEA), así como también un plan de control.

Posteriormente, en febrero de 1993 el grupo de acción automotriz industrial (AIAG) y la sociedad Americana para el Control de Calidad (ASQC) registraron las normas FMEA para su implementación en la industria, estas normas son el equivalente en el procedimiento técnico de la Sociedad de Ingenieros Automotrices SAE J- 1739.

Los estándares son presentados en el manual de FMEA aprobado y sustentado por la Chrysler, la Ford y la General Motors; este manual proporciona lineamientos generales para la preparación y ejecución del FMEA.

Actualmente, el FMEA se ha popularizado en todas las empresas automotrices americanas y es utilizado en diversas áreas de una gran variedad de empresas a nivel mundial.

Aunque el método del generalmente ha sido utilizado por las industrias aeroespacial y automotriz, este es aplicable para la detección y bloqueo de las causas de fallas potenciales en productos y procesos de cualquier clase de empresa, ya sea que estos se encuentren en operación o en fase de proyecto; así como también es aplicable para sistemas administrativos y de servicios.

### **2.3.1.2 Análisis de Árbol de Falla (A.A.F).**

Debido a que el DFMEA requiere de identificar la causa raíz de una posible falla potencial, es importante tener una técnica complementaria para determinar las causas que generan esa falla potencial. La técnica del diagrama del árbol de falla es un método que nos permite identificar todas las posibles causas de un modo de falla en un sistema en particular. Además nos proporciona una base para calcular la probabilidad de ocurrencia por cada modo de falla del sistema. Esta técnica es conveniente aplicarla en sistemas que contengan redundancia.

Mediante un A.A.F podemos observar en forma gráfica la relación lógica entre un modo de fallo de un sistema en particular y la causa básica del fallo.

### **2.3.1.3 Efectos de la falla potencial**

Los efectos de la falla potencial, son esos efectos que percibe el cliente final al manipular u operar el producto, y estos efectos tienen una ponderación que se obtiene por medio de tablas que determinan de forma aproximada la gravedad de los efectos.

El primer paso para el análisis de riesgos es cuantificar la severidad de los efectos, éstos son evaluados en una escala del 1 al 10 donde 10 es lo más severo. Estas ponderaciones se basan en la experiencia que cada empresa tiene con sus productos y la manera en que el cliente percibe la calidad y funcionamiento del mismo.

### **2.3.1.4 Causas de fallas potenciales.**

Luego de que los efectos y la severidad han sido listadas, se deben de identificar las causas de los modos de falla.

En el DFMEA las causas de falla son las deficiencias del diseño que producen un modo de falla.

### **2.3.1.5 Ocurrencia.**

Las causas son evaluadas en términos de ocurrencia, ésta se define como la probabilidad de que una causa en particular ocurra y resulte en un modo de falla durante la vida esperada del producto, es decir, representa la remota probabilidad de que el cliente experimente el efecto del modo de falla.

El valor de la ocurrencia se determina por medio de tablas, que asignan un valor por probabilidad de que ocurra una determinada causa que genera una falla potencial, en caso de obtener valores intermedios se asume el superior inmediato, y si se desconociera totalmente la probabilidad de falla se debe asumir una ocurrencia igual a 10. Estas tablas son específicas para cada empresa y para cada producto.

En ocasiones se comete el error de usar las tablas que los manuales de la industria automotriz utilizan, pero en muchas ocasiones estas tablas no reflejan la probabilidad de ocurrencia para un producto electrónico que para un producto mecánico.

### **2.3.1.6 Detección.**

La detección es una evaluación de las probabilidades de que los controles actuales de diseño propuestos. Eliminan la posibilidad de generar un diseño deficiente, en ocasiones se determina que no existen controles de detección para la etapa de diseño, por lo cual en acciones sugeridas siempre se debe de plantear

controles de diseño. En este caso se pondera que tan efectivo es el método para reducir los errores en el diseño.

### **2.3.1.7 PRN**

El número de prioridad de riesgo (PRN por sus siglas en inglés) es el producto matemático de la severidad, la ocurrencia y la detección, es decir:

$$PRN = S * O * D$$

Este valor se emplea para identificar los riesgos más serios para buscar acciones correctivas.

### **2.3.1.8 Acción (es) recomendada (s).**

Cuando los modos de falla han sido ordenados por el NPR, las acciones correctivas deberán dirigirse primero a los problemas y puntos de mayor grado e ítemes críticos. La intención de cualquier acción recomendada es reducir los grados de ocurrencia, severidad y/o detección. Si no se recomienda ninguna acción para una causa específica, se debe indicar así.

Un DFMEA tendrá un valor limitado si no cuenta con acciones correctivas y efectivas. Es la responsabilidad de todas las actividades afectadas el implementar programas de seguimiento efectivos para atender todas las recomendaciones.

### **2.3.1.9 RPN resultante.**

Después de haber identificado la acción correctiva, se estima y registra los grados de ocurrencia, severidad y detección finales. Se calcula el RPN resultante, éste es el producto de los valores de severidad, ocurrencia y detección.

El ingeniero en proceso es responsable de asegurar que todas las acciones recomendadas sean implementadas y monitoreadas adecuadamente. El DFMEA es un documento vivo y deberá reflejar siempre el último nivel de actualización del diseño.



### 2.3.2 Diagrama Causa – Efecto.

El Diagrama de Causa y Efecto es utilizado para identificar las posibles causas de un problema específico. La naturaleza gráfica del Diagrama permite que los grupos organicen grandes cantidades de información sobre el problema y determinar exactamente las posibles causas. Finalmente, aumenta la probabilidad de identificar las causas principales.

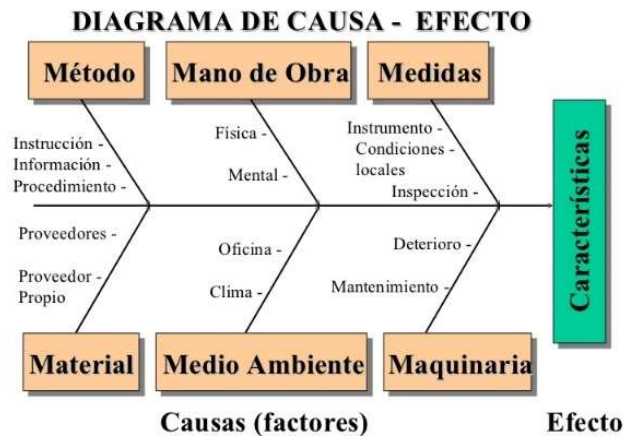


Ilustración 2.8 Diagrama Causa -Efecto

### 2.3.3 Por que (5'W).

La técnica de los 5 Porqué es un método basado en realizar preguntas para explorar las relaciones de causa-efecto que generan un problema en particular. El objetivo final de los 5 Porqué es determinar la causa raíz de un defecto o problema.

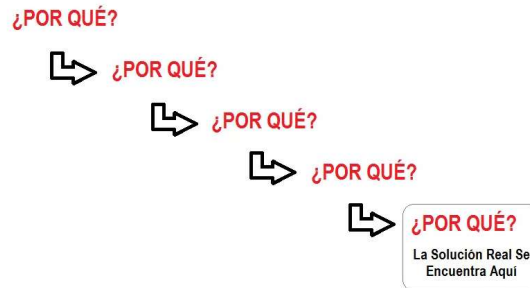


Ilustración 2.9 Proceso de los 5 ¿Por qué?

## 2.4 Diseño Conceptual.

### 2.4.1 Descripción de los requerimientos de cliente.

Para poder conceptualizar cual era el interés particular del cliente se realizaron entrevistas y encuestas, como resultado se obtuvo el desarrollo la siguiente tabla de requerimientos:

Tabla 2.0.1 Requerimientos

Sistema General: HMRDT			
Subsistema: Mecánico (M), Electrónico (E), Software (S)			
Concepto	Descripción	Función	Requerimiento de cliente
M01	Estructura		NA
M02	Cubierta	Conectar con la interface genrad	Equipo liviano y de fácil manejo
M03	Nidos	Sostener el producto	NA
M04	Soporte	Sostener al equipo	NA
E01	Sistema operativo	NA	Sistema operativo del rack
E02	Fuente programable	Alimentacion	Fuente programable del rack
E03	Plataforma de prueba	Realizar un programa ideal para cubrir las funciones del equipo	C++
E04	Voltajes	Tener el voltaje adecuado para el funcionamiento del equipo	24 V, 110 V, 5 V, 12 V
E05	Presión de aire	Tener la presión adecuada para el funcionamiento del equipo	90 PSI
E06	Corriente	Tener la corriente electrica adecuada para el funcionamiento del equipo	
S01	Slots	Slots de conectores para interface genrad	# de parte especificos
S02	PCB	Hardware de prueba para el rack	NA
S03	Interface Genrad	Conexión entre HMRDT y rack	# de parte especificos

## 2.4.2 Diseño Conceptual

### 2.4.2.1 Sistema general

En base a los requerimientos del cliente se clasifican los subsistemas y se desarrolla el diagrama a bloques funcional del sistema completo que nos permite determinar los subsistemas que componen el sistema, tal y como muestra la figura 2.10,

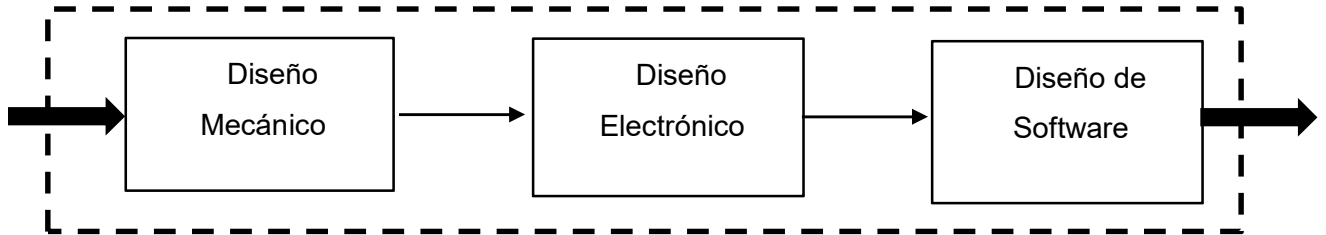


Ilustración 2.10 Diagrama de bloques

### 2.4.2.2 Subsistema mecánico

Cada uno de estos subsistemas se delimitan (línea punteada) para realizar el análisis de cada uno y enonces basados en los requerimientos del cliente ir diseñando cada uno de los elementos que los componen. El subistema mecánico tiene la estructura que muestra el siguiente diagrama a bloques funcional de la figura 2.11.

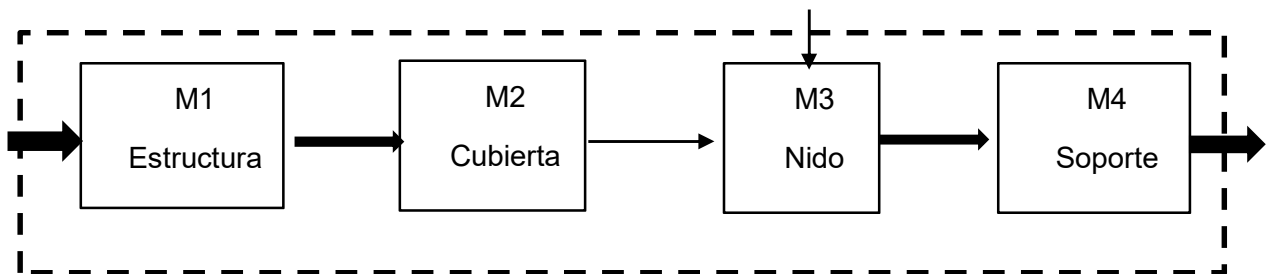


Ilustración 2.11 Diagrama de bloques

### 2.4.2.3 Subsistema Electrónico

Cada uno de los subsistemas contiene los elementos que deberán representar un elemento físico que realiza un afunción en el subistema y el cual debe de ser diseñado para realizar la función a la cual fue destinado Figura 2.12

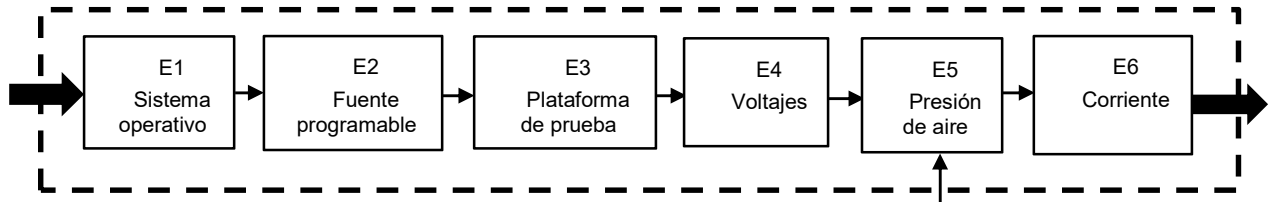


Ilustración 2.12 Diagrama de bloques

### 2.4.2.4 Subsistema de software

El tercer subsistma consiste en el desarrollo y diseño del software que debe de controlar todos y cada uno de los subsistemas de forma que todas las operaciones y funciones a realizar por cada uno de los subsistemas estén coordinadas entres si. La figura 2.13 muestra el diagrama a bloques funcional.

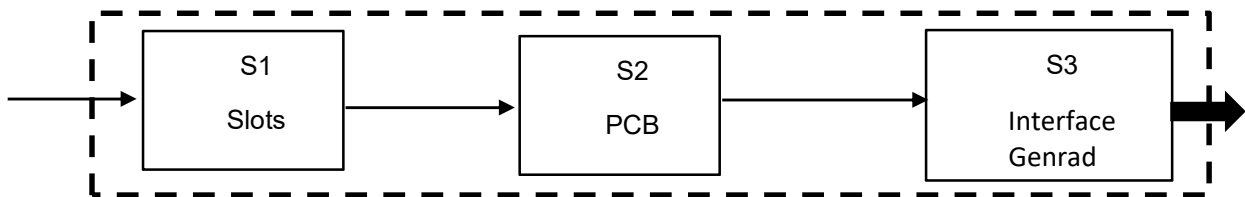


Ilustración 2.13 Diagrama de bloques

### 2.4.3 Las especificaciones de ingeniería

Las especificaciones de ingeniería es un conjunto de atributos o características tangibles e intangibles particulares bajo una forma fácilmente reconocible e identificable, las cuales delimitan el campo de interacción entre las ellas, la definición clara de las especificaciones deberán ser tenidas en cuenta a lo largo de todo el proceso.

Estas especificaciones cubren los requerimientos necesarios para la fabricación del HMRDT, estas deben ser diseñados, fabricados y aprobados en total cumplimiento con los estándares y códigos para el cual va a ser fabricado.

Dentro de estas especificaciones se pueden encontrar de estos tipos y hacen más sencilla la conceptualización del proyecto:

- Características organolépticas, que se determinan en el proceso productivo, a través de controles estandarizados.
- Criterios subjetivos, tales como imágenes, ideas, hábitos que el usuario final emite sobre la herramienta.

Para tener como resultado la conceptualización del diseño se sigue el modelo lineal del proceso de diseño, que consiste en identificar las fases del diseño que se integran de la siguiente manera:

- Fase 1, Diseño conceptual: se buscan conceptos o principios de la solución del problema, para la cual se analiza el problema identificado, se sintetiza una o varias posibles soluciones y se evalúa con respecto a las especificaciones.
- Fase 2, Diseño preliminar: Se avanza en la concretización de una solución al problema, determinando componentes e interacciones con suficiente grado como para poderla evaluar objetivamente.
- Fase 3, Diseño detallado: Se generan todas las especificaciones necesarias para la producción del producto-solución. La elaboración de planos, etapas de fabricación, identificación de componentes, selección de proveedores, etc.

Tabla 2.0.2 Especificaciones de Ingeniería

ESPECIFICACIONES DE INGENIERÍA				
<p><b>Inspección y pruebas:</b> El fabricante debe ejecutar las pruebas funcionales completas del equipo antes de ser entregado. El instrumento debe poseer un certificado de pruebas de fábrica y de calibración, indicando orden de compra, Código (nomenclatura), cuando sea necesario.</p>				
<p><b>Inalatación y puesta en marcha:</b> El Proveedor debe dar asistencia en sitio para la instalación, configuración, pruebas y puesta en servicio del equipo. De ser necesario debe incluirse entrenamiento. El servicio requerido debe ser provisto en un período menor de 24 hrs. (o lo que aplique).</p>				
<p><b>Garantía:</b> El proveedor debe garantizar el correcto funcionamiento del equipo ó instrumento de acuerdo a las hojas de datos anexas a este documento. El equipo debe estar garantizado contra defectos de fabricación y diseño por un período mínimo de 12 meses o mayor, a partir de la fecha de envío.</p>				
<b>SUBSISTEMA: MECÁNICO</b>				
Elemento	Variables	Especificación	Tolerancia	Rango de operación
M1	Estructura	NA	NA	NA
M2	Cubierta	Aluminio		
M3	Nido	NA	NA	NA
M4	Soporte	NA	NA	NA
<b>SUBSISTEMA: ELECTRÓNICO</b>				
Elemento	Variables	Especificación	Tolerancia	Rango de operación
E1	Sistema operativo	W7, Borland	NA	NA
E2	Fuente programable	Keysight NC700C	240VDC/50 A	110VDC/12AMP
E3	Plataforma de prueba	C++	NA	NA
E4	Voltaje	24 V, 110 V, 5 V, 12 V		
E5	Presión de aire	90 PSI	110PSI	90-110 PSI
E6	Corriente	1Amp	1-2Amp	1.3Amp

## CAPÍTULO II

<b>SUBSISTEMA: SOFTWARE</b>				
Elemento	Variables	Especificación	Tolerancia	Rango de operación
S1	Slots	Ver Anexo i	NA	NA
S2	PCB			
S3	Genrad Interface			

El marco teórico desarrollado en la presente tesis está basado en las herramientas de análisis de falla que se utilizaron para determinar cuáles serían las pruebas en las que se debería enfocar nuestra fixtura *HMRDT*. Todas estas herramientas se utilizan para llevar a cabo un diseño satisfactorio que cumpla con los requerimientos del cliente y que a su vez se traduzcan en especificaciones de ingeniería que van a permitir al diseñador desarrollar un producto confiable.

## CAPÍTULO III. DESCRIPCIÓN DEL TRABAJO

### 3.1 Objetivos

#### 3.1.1 Objetivo general

Construir una herramienta de diagnóstico para validación de equipos de prueba que mida voltajes y resistencias de sus instrumentos y tarjetas con base a su funcionamiento, permitiendo así el reconocimiento inmediato de fallas a través de su propia interfaz gráfica.

#### 3.1.2 Objetivos específicos

- Definir cuáles son las características eléctricas/electrónicas de los instrumentos y tarjetas a probar.
- Diseñar el *hardware* electrónico para el instrumento de medición.
- Diseñar el *hardware* mecánico del *fixture*.
- Definir tipos de sistemas de medición programables que se puedan implementar en la interfaz gráfica.
- Implementar una interfaz gráfica para el usuario con base a lo que se va a diagnosticar.
- Implementar una interfaz/diseño mecánico con el cual se ensamblará el *fixture* al Minne Rack.
- Implementar diseño de *PCB's* para la elaboración del sistema completo ya integrado del *fixture*.

### 3.2 Hipótesis y variables

La empresa *Honeywell* es una importante empresa multinacional estadounidense que produce una variedad de productos de consumo, servicios de ingeniería y sistemas aeroespaciales para una amplia variedad de clientes, desde compradores particulares, hasta grandes corporaciones y gobiernos.



Se han detectado como parte del proceso productivo de la empresa, problemas de fallas recurrentes de prueba y validación en los equipos denominados *Minne Racks*, los cuales son equipos de medición y prueba estándar para varios productos de la compañía.

Este problema genera que no se puede medir ni validar ningún producto de la empresa al fallar un Minne Rack, lo que genera problemas de producción, tiempos de entrega y pérdidas de millones de dólares; por tanto, se estima necesario desarrollar un equipo de prueba o herramienta de diagnóstico (Fixture), que verifique cada uno de los Minne Rack's midiendo todas las variables posibles, reflejando los resultados mediante una interfaz gráfica, permitiendo identificar rápidamente el origen de la falla en el equipo.

Implementando esta solución, se detecta rápidamente el lugar de la falla y es más rápido encontrar la solución al problema, permitiendo que haya más tiempo de producción, obteniendo así las siguientes variables:

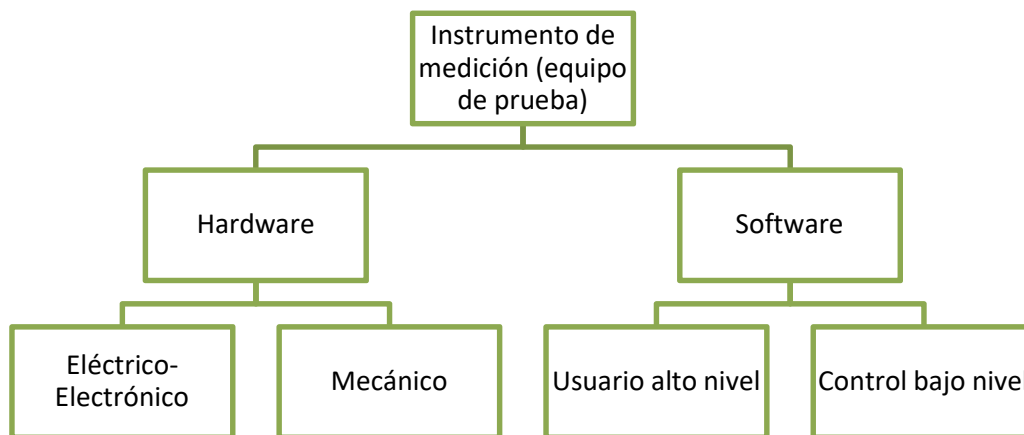


Figura 3.1 Descripción de las variables

Haciendo el análisis de lo que conlleva el equipo, las variables detectadas son:

- Voltaje
- Corriente
- Resistencia
- Fixture (Elementos mecánicos y *hardware*)
- Interfaz gráfica (programación)
- Interfaz de comunicación
- Software (diagnóstico)
- Tiempo de validación y verificación

### **3.3 Justificación**

El desarrollo de una herramienta de diagnóstico nos permitirá analizar un equipo en modo de falla, debido a que será posible medir cada uno de los voltajes, cargas del equipo, tarjetas e instrumentos, etcétera; localizando así la falla de forma inmediata, reduciendo tiempo de análisis y aprovechando el tiempo de producción.

Con esta tecnología se hacen posibles las pruebas de validación, con lo que, el ingeniero a cargo del equipo y/o proyecto puede asegurar que el producto defectuoso no salga en venta al cliente brindando así un mejor servicio, teniendo como beneficio por parte de la empresa la satisfacción del cliente y su confiabilidad.

Entre otros beneficios que se obtienen, a parte de la ganancia en tiempo de producción, es en la reducción de costos en el servicio que los proveedores pueden darles a los equipos, lo que puede cobrar una empresa como proveedor en arreglar esos equipos, lo que puede invertir una sola vez en esta herramienta de diagnóstico.

Como se mencionó anteriormente, el uso e implementación de esta tecnología tiene un gran potencial debido a todos los beneficios que brindará a la industria, además que es un proyecto bastante completo en su ámbito en lo que a la Maestría en Ingeniería en Mecatrónica se refiere.

### 3.4 Explicación

#### 3.4.1 Explicación general

El proceso de desarrollo del proyecto se llevó a cabo por medio de los siguientes pasos:

1. Revisar las diversas especificaciones a diagnosticar de un *Minne Rack*: Se realiza un análisis de cuáles son las principales pruebas que se deben diagnosticar de manera rápida y precisa, tales como: fuentes opcionales y programables, tarjetas de *RELAYS*, *MUX*, *DIO*, *INTERLOCK*, *COMM*. De igual manera se revisan las especificaciones de los requerimientos necesarios para iniciar con el diseño del proyecto.
2. Realizar un análisis de entradas y salidas, véase Figura. 4.1 Diagrama de bloques.
3. Diseñar diagramas eléctricos, electrónicos y neumáticos. Se desarrollan los diagramas con la ayuda del software *EAGLE*, haciendo referencia a los Anexos i al xxii.
4. Diseñar mecánicamente la fixtura de prueba. Con ayuda de un diseñador mecánico se desarrollaron los planos mecánicos necesarios con las especificaciones necesarias para la integración mecánica de la herramienta de prueba.
5. Diseñar *software* de secuencia de operación e interfaz con usuario (secuencias, drivers y herramientas de diagnóstico) véase Tabla 4.2 Código HMRDT.

6. Realizar lista de material a comprar para el proyecto. Se desarrolla desde el inicio del diseño, con la finalidad de contemplar todas las partes que son necesarias y requeridas para la integración.
7. Integrar la fixtura (piezas mecánicas, componentes eléctricos, etc.), con ayuda de un técnico de integración con especialidad electromecánica se realiza la integración de las partes mecánicas y los componentes eléctricos, que por separado solo son piezas y después de ser integrados se convierten en un todo. Anexo xxiv al xxxviii.
8. Realizar la prueba y *debuggeo*, se realizará pruebas de voltaje, conexiones y que no exista ninguna falla en el equipo con la finalidad de que cuando se realicen las pruebas funcionales este en óptimas condiciones.
9. Realizar pruebas funcionales completas simulando las condiciones de producción de planta y hacer las correcciones que se requieran.

### 3.4.2 Diagramas de Bloques

El siguiente diagrama de bloques muestra la interrelación de los subsistemas y la conexión que tiene con el *Minne Rack*, donde se inicia la generación de información que debe ser detectada por el HMRDT.

En base a los siguientes diagramas de bloques se puede observar la representación gráfica de las interrelaciones del proceso tanto de manera general, así como, de manera interna contemplando sus entradas y salidas.

3.4.2.1 Diagrama principal del sistema

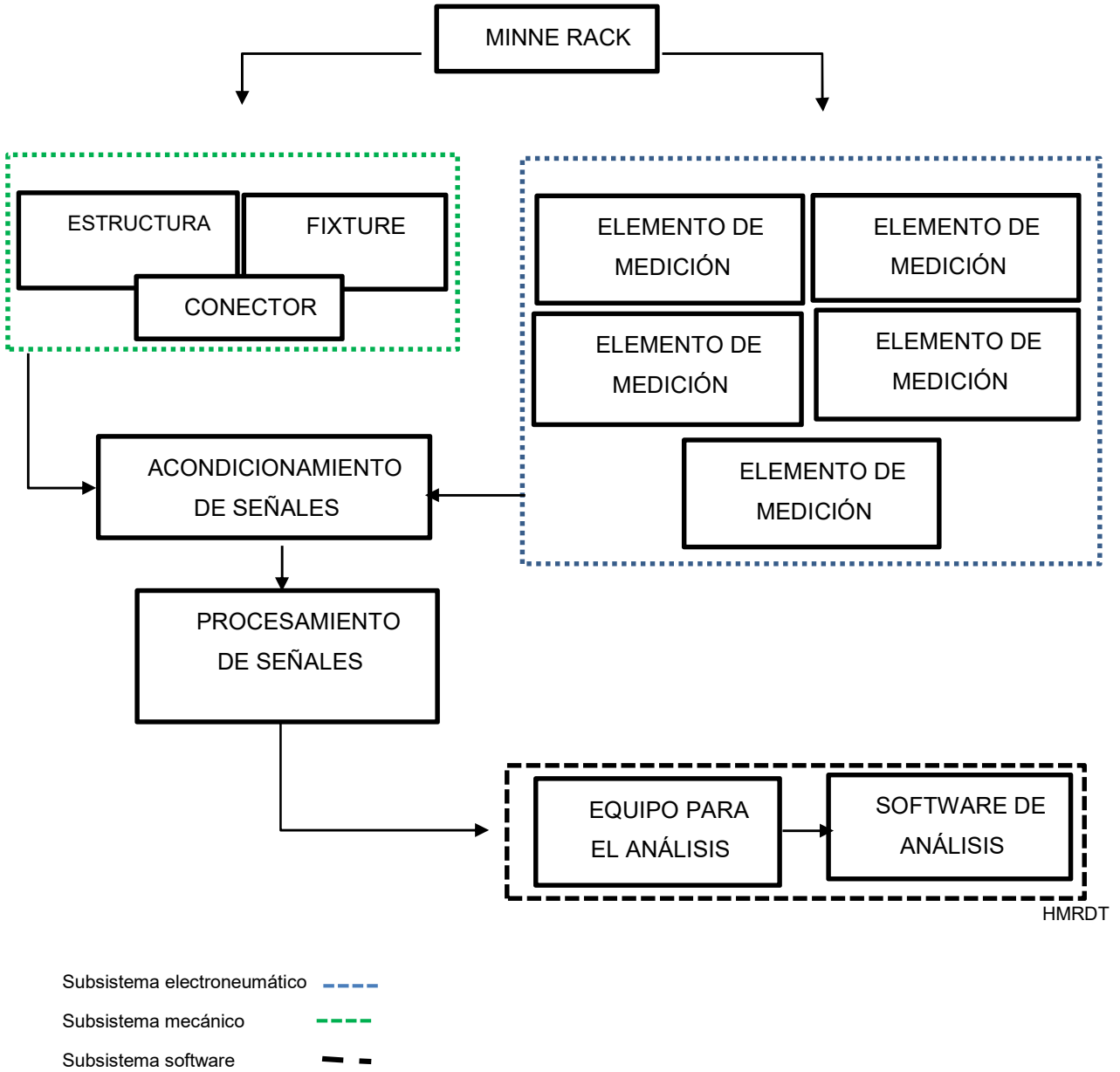


Ilustración 3.2 Diagrama de bloques

### 3.4.3 Explicación técnicas

Inicialmente se realizó una revisión de la ficha técnica del *Minne Rack* Tabla 3.1 Ficha técnica *Minne Rack*, donde dicha ficha arrojó la información y los factores necesarios para determinar los elementos para poder desarrollar el diagrama eléctrico del equipo dejando referencia de lo que se debe de diagnosticar.

Tabla 3.0.1 Ficha técnica *Minne Rack*

 ICEM	FICHA TÉCNICA <b>MINNE RACK</b>	<b>Honeywell</b>
<b>Descripción:</b>	Gabinete estándar, consiste en una plataforma de hardware para prueba eléctrica.	
<b>Modelo:</b>	Minne Rack	
<b>Marca:</b>	ICEM	
<b>Serie:</b>	MR000 00XXXXXX	
<b>Fabricante:</b>	Ingeniería en Control Electrónico y Mecánico, S.A. de C.V.	
<b>ESPECIFICACIONES TÉCNICAS</b>		
<b>Materiales:</b>	Estructura de aluminio extruido de 45x45mm Cubierta de MDF con fórmica antiestática Tapas laterales de ABS Cubierta frontal de policarbonato	
<b>Peso:</b>	200 Kg aprox.	
<b>Dimensiones:</b>	Largo: 104.7cm x Ancho: 81.2cm x Altura: 168.9 cm	
<b>Frecuencia AC:</b>	60 Hz	
<b>Voltaje AC:</b>	120 V	
<b>Fuentes DC:</b>	Omron: (+) 24V 5A Sola: (+)5V 3A, (+)12V 600mA, (-)12V 600mA	
<b>Presion Aire:</b>	90 PSI	
<b>Principales componentes:</b>	Abanicos Paro de emergencia marca Allen-Bradley Switch principal mara Allen-Bradley Interface Genrad	

	Regulador de voltaje Tripp-Lite LCR-2400 Válvulas y accesorios neumáticos marca SMC Monitor LED 18.5” Computadora industrial marca Advantech 34921A Agilent 40Ch armature multiplexer 34937A Agilent 32Ch general purpose switch 34952A Agilent Multifunction module 32bit 2Ch 34980A Agilent Multifunction main frame Fuente AC Kikusui PCR500M
<b>Características especiales:</b>	El equipo es un sistema personalizado para uso exclusivo de Honeywell, consta de una plataforma y gabinete tipo estándar, con instrumentos de medición y control, capaces de adaptarse a las necesidades específicas de prueba para cada modelo de termostato que se desee probar. Entre las principales pruebas realizadas se encuentra: temperatura, corriente, voltaje y comunicación.

Después de que se tiene las especificaciones técnicas del *Minne Rack*, se pueden obtener las siguientes especificaciones con las que debe cumplir el *HMRDT*, para que sea compatible y cumpla con las necesidades de prueba del *Minne Rack*.

Tabla 3.0.2 Tabla de especificaciones del *HMRDT*.

Requerimiento subsistema electroneumático	
Voltaje requerido	24 V, 110 V, 5 V, 12 V
Consumo de corriente	Alimentación por medio de Interface genrad al equipo bajo prueba
Presión de aire	Alimentación por medio de Inetrface genrad al equipo bajo prueba
Requerimiento subsistema mecánico	
Dimensiones generales (AxLxAlto)	Según se indique en el plano de ensamble general
Peso	30 kg
Conexión ethernet	NO
Gabinete de aluminio	SI
Etiquetas de identificación	SI
Acabado de maquinado	Anodizado Natural
Sistema de unidades de tornillería	Milimétrica

Requerimiento subsistema software	
Interface mediante PCB	SI
Genrad Interface Blocks Fixture	SI
HMRDT Software application	SI
Plataforma de prueba	C++
Load PCB for electrical test for the slots:	
Slot 1 Power	SI
Slot 2 Relays	SI
Slot 3 Multiplexor	SI
Slot 4 USBs	SI
Slot 7 Interlocks	SI
Slot 8 Pneumatic	SI
Slot 10 Coaxial	SI
Slot 11 DIO	SI
Slot 12 Comm/MMI	SI

El sistema requiere las siguientes condiciones para arrancar, las cuales se mencionan a continuación:

- a) Alimentación eléctrica de 110VAC
- b) Conexión neumática con presión de aire de 90 PSI
- c) Todas las puertas del equipo bien cerradas. El equipo cuenta con 1 puertas de acceso, con interlock magnético de seguridad. El sistema no permite que se abran las puertas durante la operación, a menos que se presione el paro de emergencia o se deshabilite el interlock.
- d) Paro de emergencia liberado. Para liberar el paro de emergencia, gire el botón en sentido de las manecillas del reloj.



### 3.4.3.1 Subsistema Mecánico

En base a la información anterior se pudo diseñar un *fixture* compatible al *rack*, es decir, de acuerdo al subsistema mecánico, se adaptó una estructura que pudiera conectar los bloques del *Minne Rack* (véase anexo i *Genrad Interface*).

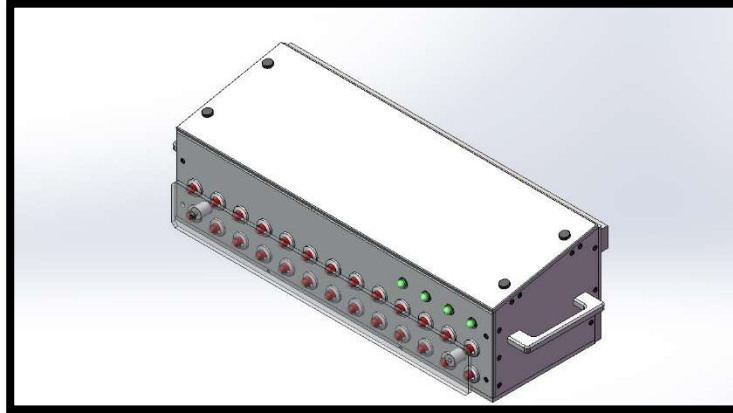


Ilustración 3.3 Subsistema Mecanico

Honeywell maneja un estándar en sus *fixtures* mecánicamente hablando, ya que la plataforma es la misma para todos sus equipos de prueba, el *fixture* está diseñado para encajar en cualquier *Minne Rack*, por lo tanto no se realizó un análisis de fuerzas entre los bloques de la interfaz *genrad*, ya que están fabricados para este propósito, además que es un mismo proveedor el que vende la parte de la interfaz *genrad* por parte del *rack* tanto por la parte del *fixture*.

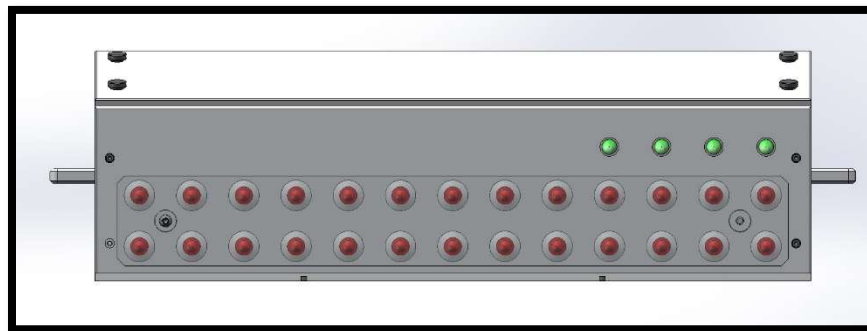


Ilustración 3.4 Subsistema Mecanico (Vista Frontal)

Otra razón más por la cual no se hizo un análisis de este tipo es por que también se tomaron referencias de diseños anteriores debido al estándar manejado ya en la empresa.

Gracias al estándar que maneja Honeywell en sus *fixtures*, se logró implementar el diseño en *Solidworks*.

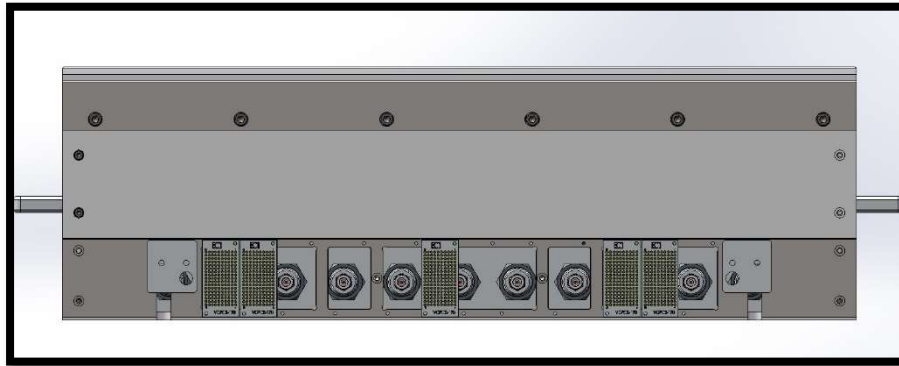


Ilustración 3.5 Subsistema Mecánico (Vista posterior)

Se contempló para el diseño toda la contraparte de la interfaz *genrad*, también en que espacio se podía agregar la *PCB* en la cual estarán todas las conexiones electricas para las pruebas, como se muestra en las figuras del diseño en *3D*.

Se tomaron en cuenta indicadores neumaticos para la prueba de las electroválvulas y lamparas para los voltajes, explicados ambos posteriormente en el subsistema eléctrico.

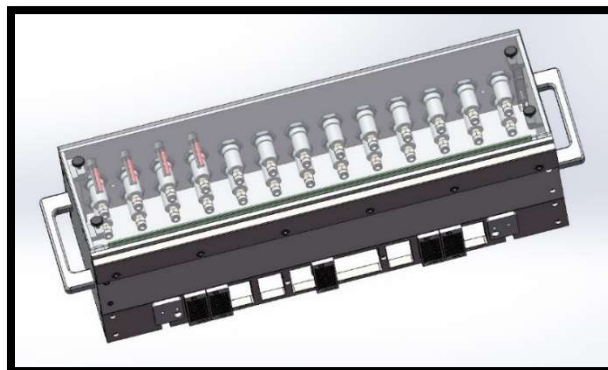


Ilustración 3.6 Subsistema Mecánico (Vista Superior)

Se tomaron en cuenta las agarraderas del equipo para que el usuario pueda cargar el *fixture*, y una guarda de seguridad para que los indicadores neumaticos no se rompan al ser golpeados por algo externo.

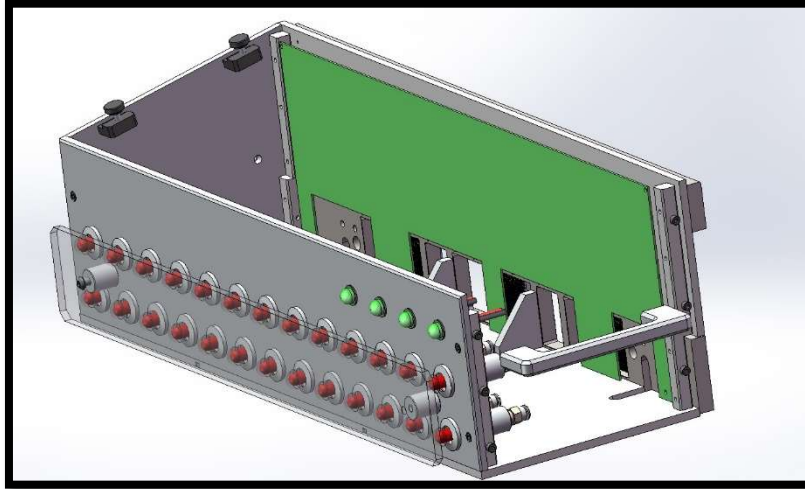


Ilustración 3.7 Subsistema Mecánico (Vista Lateral Abierto)

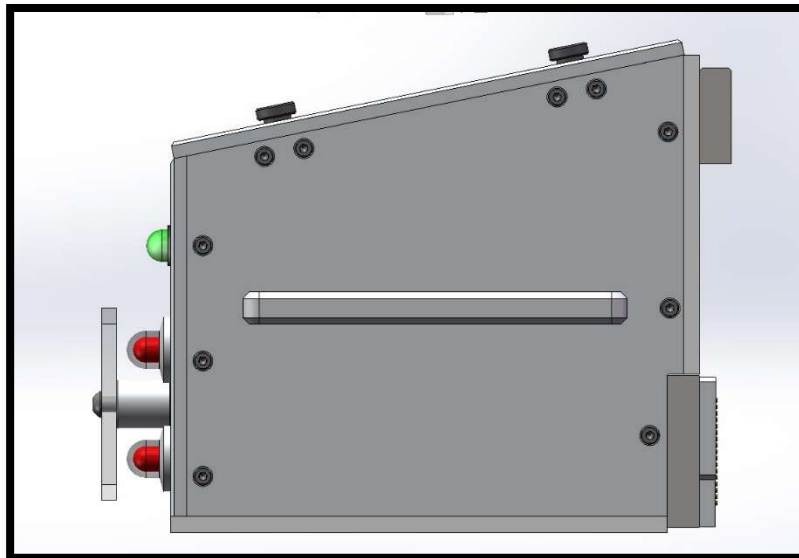


Ilustración 3.8 Subsistema Mecánico (Vista Lateral Cerrado)

## CAPÍTULO III

Después de terminar el diseño mecánico en Solidworks, se realizaron los planos correspondientes (véase *anexo xxix-xxxviii*) de las piezas, se mandaron a maquinar y anodizar en aluminio con un proveedor externo.

Se seleccionó como material el aluminio ya que es un material ligero, y el *fixture* internamente no iba a traer más de 1Kg de peso en sus componentes, así que entre más ligero este el equipo, mejor, dado que es manipulado por varias personas al cargar y descargar y esto a su vez se convierte en un punto de seguridad a favor.

Después de recibir las piezas el siguiente paso es integrar el *fixture* completo para diagnosticar que no se tenga que retrabajar nada, dadas las variaciones que pueda tener cada pieza por el maquinado.

En las siguientes figuras se muestra el resultado del diseño ya integrado:



Ilustración 3.9 Diseño Integrado

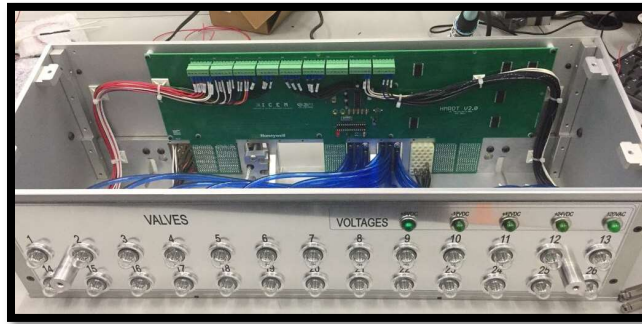


Ilustración 3.10 Diseño Integrado



Ilustración 3.11 Diseño Integrado (Parte Frontal)

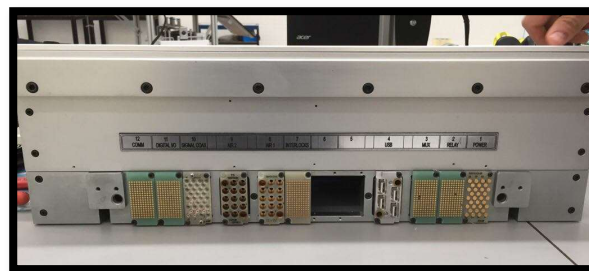


Ilustración 3.12 Diseño Integrado (Parte posterior)

**3.4.3.2 Subsistema Eléctrico / Electrónico.**

Como ya se mencionó anteriormente, todos los instrumentos, fuentes y tarjetas del *Minne Rack*, (que son los elementos a diagnosticar) están conectados en los bloques de la interfaz *genrad* por parte del *rack*, así que para establecer los criterios de diseño del subsistema eléctrico/electrónico, estos se basan en absolutamente todas las señales que el *rack* tiene conectadas a la interfaz *genrad*, ya que el *fixture* por medio de la misma interfaz a través de sus bloques que están conectados en los *slots*, van a tener acceso a dichas señales.

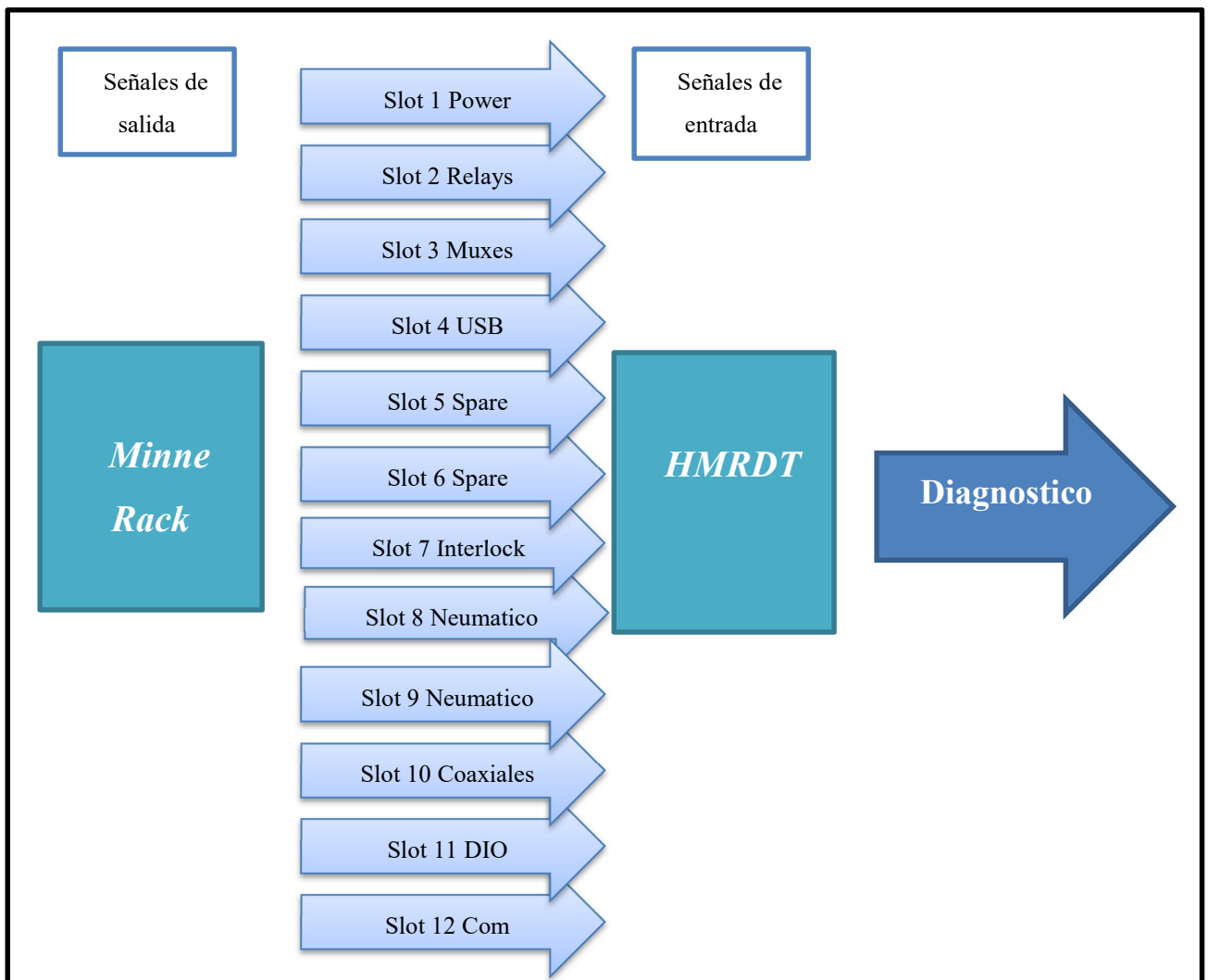


Ilustración 3.13 Subsistema Electrónico

Así pues se separa por bloque cada una de las pruebas electricas que se le van a realizar al *rack*.

Se dispuso a diseñar una tablilla electrónica que se pudiera adaptar a todas las señales de entrada que se tienen del *rack*, es decir, conectar una tablilla a los bloques por parte del *fixture* para poder manejarlas por medio del *software*, el cual se menciona posteriormente.

Dadas las dimensiones del *fixture*, las cuales tenemos por el diseño mecánico, se diseñó el *layout* del *PCB*, para poder adaptar cada uno de los bloques como se muestra en las siguientes figuras:

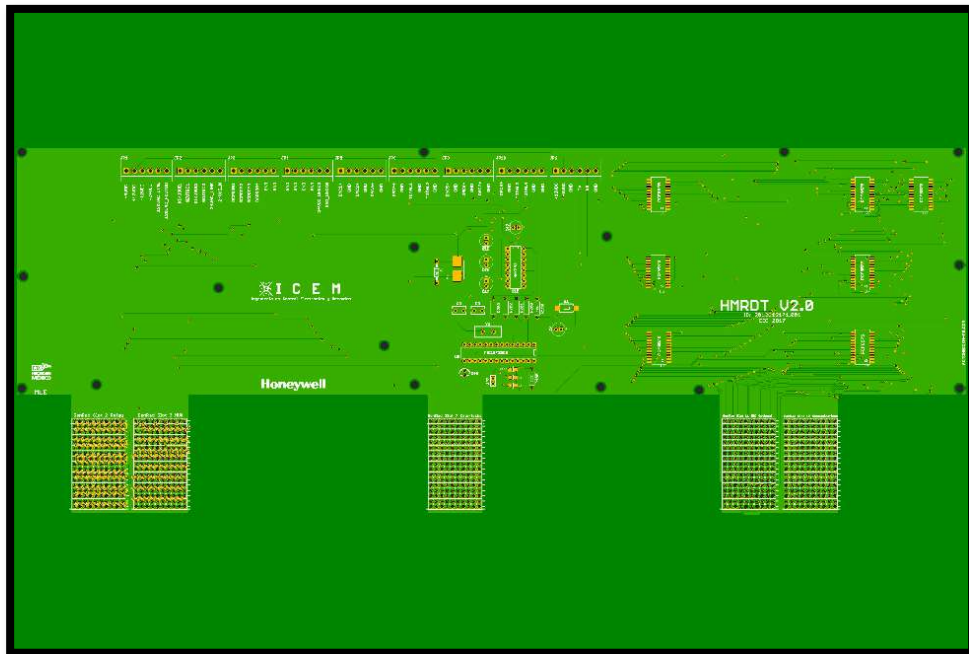


Ilustración 3.14 Layout de la PCB

A continuación se describe cada uno de los bloques, sus características electricas con respecto al diseño del *PCB*, su función, tipo de prueba y como está conectado eléctricamente al *fixture*:

**Slot 1 (Power).** Para esta parte del circuito se tomaron en cuenta los voltajes del *rack* como se muestra en la imagen, el bloque de *power (slot 1)* tiene 32 puntos de contacto en los cuales están conectadas todas las fuentes del rack, ya sean programables, AC/DC, y fuentes fijas.

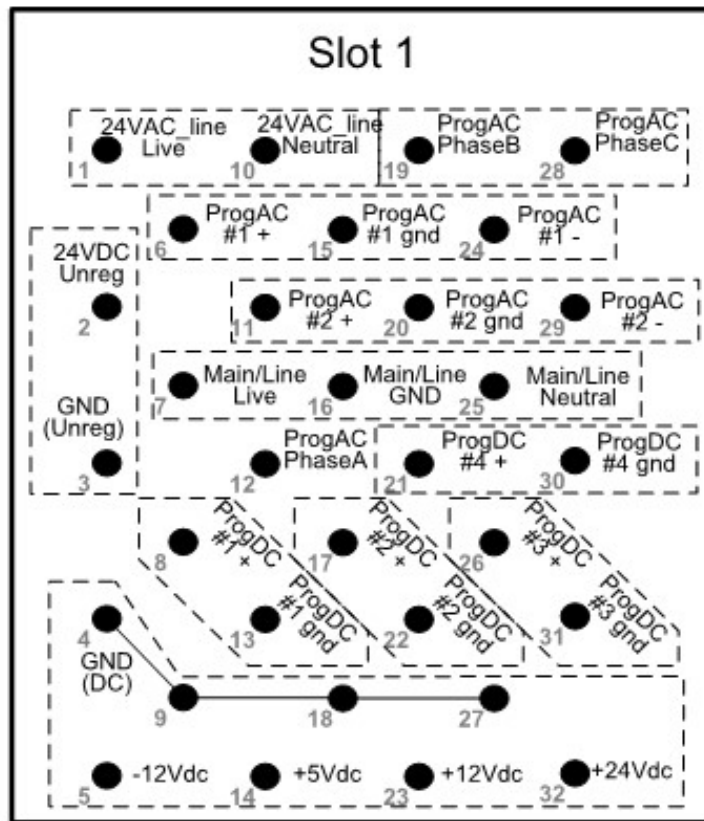


Ilustración 3.15 Slot 1 Power



Así mismo el bloque de *power* por el lado del *fixture* tiene los mismos 32 puntos de contacto. Este bloque soporta 250VAC, está adaptado para que pasen 25Amps sin problema, cada punto de contacto tiene una resistencia de 5mΩ, una capacitancia de 21pf, inductancia de 3nH y su durabilidad es de 25,000 ciclos.[21]

También está adaptado para que se le puedan soldar cables con su respectivo calibre de acuerdo al voltaje que se le va a programar a cada contacto.

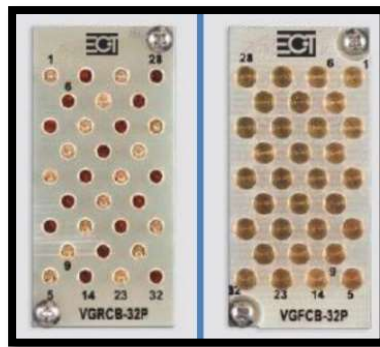


Ilustración 3.16 Slot Físico

Así que lo que se realizó en este caso, fue soldar los 32 puntos como se muestra en la figura 3.17

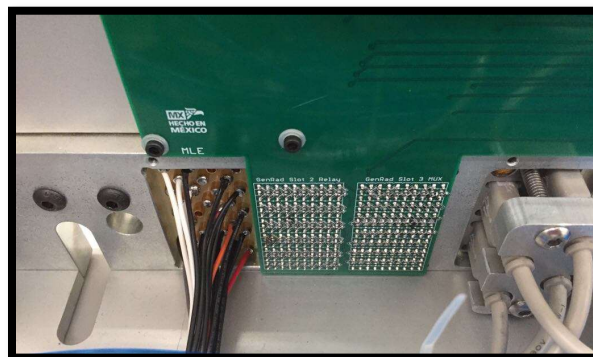


Ilustración 3.17 Slot Power vista interna

## CAPÍTULO III

Una vez que se soldaron los cables, estos fueron conectados a la tablilla por medio de unos conectores, como se muestra en la figura 3.14 para así poder medir estos mismos mediante los multiplexores de la tarjeta 34921A que el *Minne Rack* tiene por estándar.

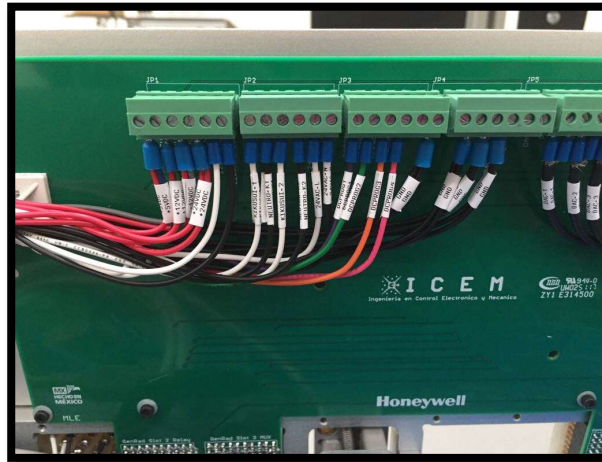


Ilustración 3.18 Conexiones del bloque de Power a PCB

En la siguiente parte del esquemático de la tablilla se puede observar como están conectados los voltajes de los conectores hacia los multiplexores (ver anexo xxi para ver diagrama completo):

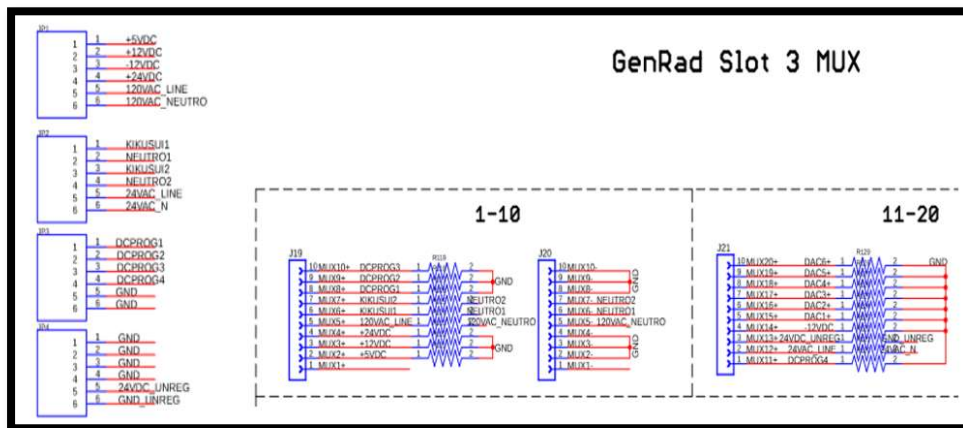


Ilustración 3.19 Conexion de multiplexores

Cuando se inicia la prueba por medio del software, los primeros multiplexores medirán los voltajes que hay en el *rack*, y así podemos descartar la primer prueba.

**Slot 2 (Relays).** En esta sección del circuito de la tablilla, se pusieron a prueba los relays de la tarjeta 34937A la cual está conectada en el *rack* en el *slot* 2. Este *slot* tiene un bloque de 170 contactos, es decir, podemos conectar 170 señales en el, con un máximo de 250VAC y 5 Amps en cada uno. Cada pin de transferencia tiene 10mΩ de resistencia y su durabilidad es de 100,000 ciclos.[22]

Como podemos ver en la figura de la derecha, en el *rack* cada pin ya está asignado a un contacto de cada *relay*, por lo cual tomamos este criterio para realizar el circuito de prueba.

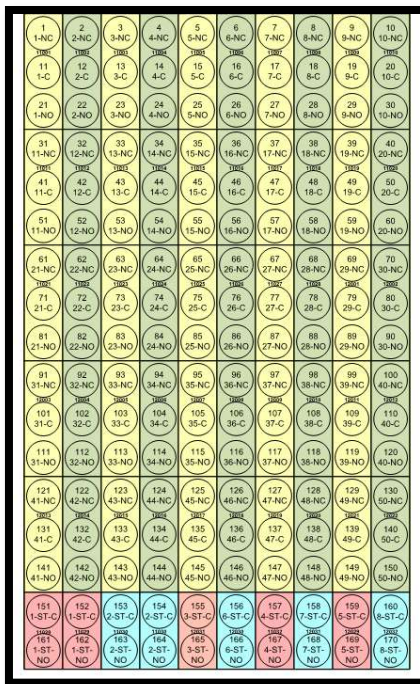


Ilustración 3.20 Slot 2 Relays

Basandonos en la información que el *rack* nos brinda, se dispuso a conectar cada pin de los *NC* (contactos normalmente cerrados) y de los *NO* (contactos normalmente abiertos) del lado del *fixture* a una resistencia en serie de “x” valor, conectando el común de cada relay a el primer multiplexor de la tarjeta 34921A para poder medirlos (*ver anexo xxii*), es decir, dentro de la prueba de *relays* por medio del software vamos a indicar que el multiplexor mida dicha resistencia de cada contacto para saber cual es el estado de cada uno de los *relays*, si el contacto abre o no, ya que en cada uno de ellos nos tiene que dar un valor en especifico.

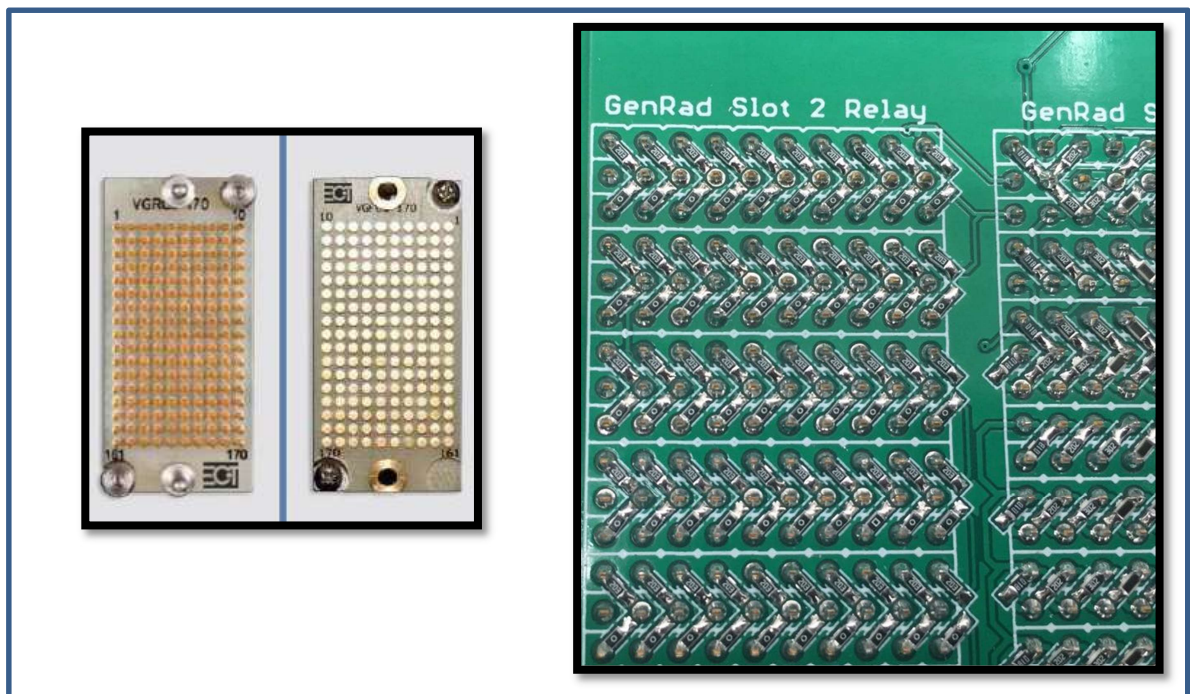


Ilustración 3.21 Slot 2 y bloques de Relays



**Slot 3 (Multiplexores).** El área de los multiplexores tiene asignado ya un orden asi como el area de los *relays*, tienen un orden especifico en el *rack* conectado en el *slot* 3, este utiliza el mismo bloque de 170 contactos con las mismas especificaciones electricas ya mencionadas anteriormente.

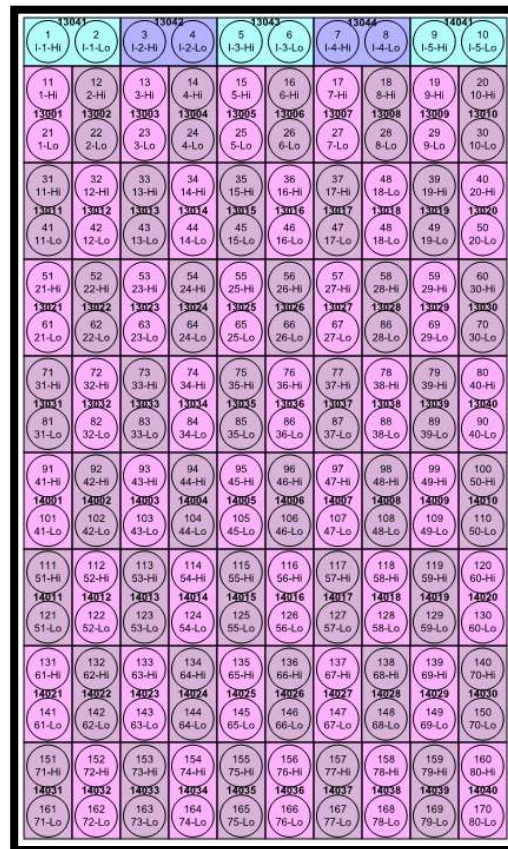


Ilustración 3.22 Slot 3 Multiplexores

Los multiplexores fueron conectados a resistencias en serie para su medición asi como los relevadores, cada una de diferente valor para poder diferenciar a cada uno de ellos, dejando su canal negativo (*Low*) aterrizado.

Un ejemplo de como se conectó en el diagrama (*ver anexo xxii para ver diagrama completo*) es el siguiente:

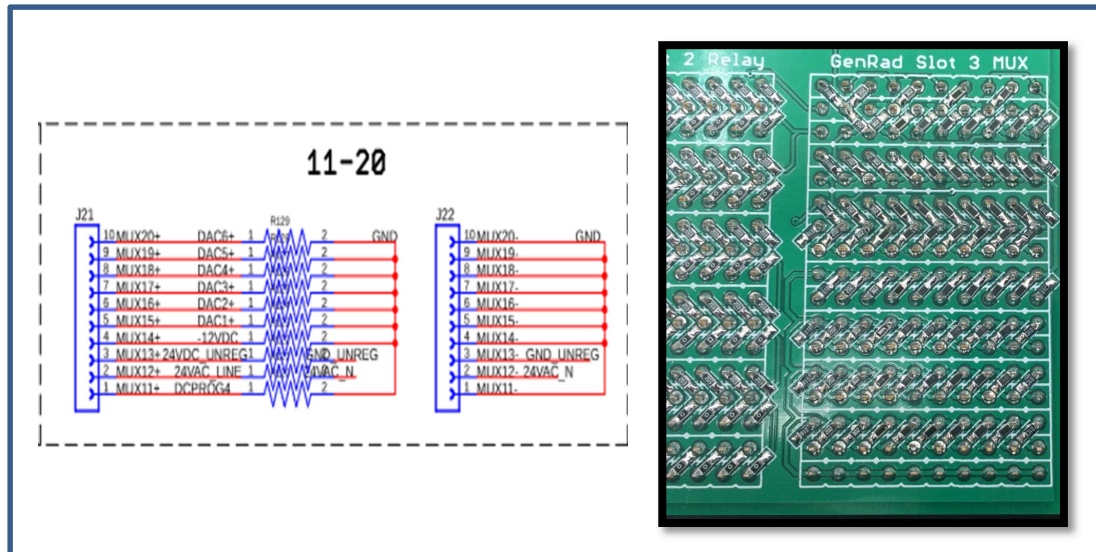


Ilustración 3.23 Slot 3 Multiplexores Conexiones

Los primeros 13 multiplexores fueron reservados para otro tipo de mediciones:

- Multiplexor 1: Prueba todos los relevadores del equipo (*slot 2*).
- Multiplexor 2: Prueba los 5 volts del equipo (*slot 1*).
- Multiplexor 3: Prueba los 12 volts del equipo (*slot 1*).
- Multiplexor 4: Prueba los 24 volts del equipo (*slot 1*).
- Multiplexor 5: Prueba 120VAC (*slot 1*).
- Multiplexor 6: Prueba la primer fuente programable (PCR500M)(*slot 1*).
- Multiplexor 7: Prueba la segunda fuente programable (PCR500M) (*slot 1*).
- Multiplexor 8: Prueba la primer fuente de DC programable (*slot 1*).
- Multiplexor 9: Prueba la segunda fuente de DC programable (*slot 1*).
- Multiplexor 10: Prueba la tercer fuente de DC programable (*slot 1*).
- Multiplexor 11: Prueba la cuarta fuente de DC programable (*slot 1*).
- Multiplexor 12: Prueba 24VAC (*slot 1*).
- Multiplexor 13: Prueba 24VDC UNREG (*slot 1*).

Del Multiplexor 14 en adelante se probarán los multiplexores con el software midiendo la resistencia ya mencionada anteriormente.

**Slot 4 (USB):** El slot 4 prueba los puertos usb de la computadora, los cuales son extensiones USB conectadas directamente a la computadora del *rack*, por medio de un bloque en el cual se adaptan las mismas extensiones USB. En este caso para probar que funcionan los puertos USB, se conectaron 3 memorias con diferente nombre, las cuales debe detectar en el Administrador de dispositivos de la PC.

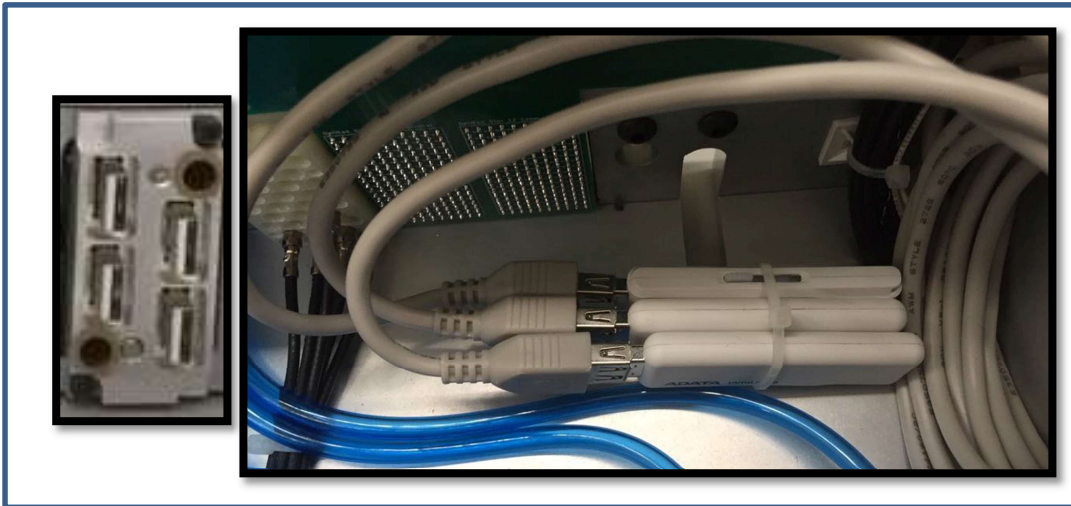


Ilustración 3.24 Slot 4 USB

**Slot 5 y 6 (spare):** Los slots 5 y 6 se encuentran vacíos pero disponibles ya que en los demás *slots* se encuentran todos los recursos del *rack* conectados, esto quiere decir que no hay más recursos que conectar dadas las diferentes versiones de *Minne Rack*.

**Slot 7 (Interlocks):** Los interlocks están ruteados entre si en la tablilla para su prueba (ver anexo vi), es decir, están puenteados entre ellos para activar o desactivar lo que se le conecte, en este caso, las primeras 13 de 26 electroválvulas del *Minne Rack* están conectadas a este bloque, por lo cual si queremos activarlas o desactivarlas, habrá que puentear dichos interlocks.

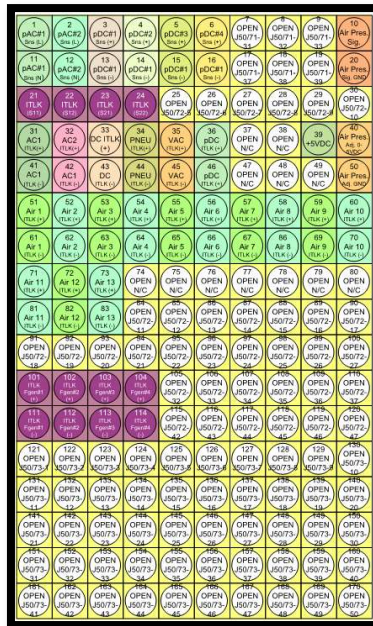


Ilustración 3.25 Slot 7 Interlocks

Este bloque tiene las mismas características que el bloque de los relays y de los multiplexores, tiene 170 posiciones, de las cuales 13 son usadas para las electroválvulas (ver anexo vi) .

**Slot 8 y 9 (Air):** El *Minne Rack* cuenta con dos bloques pneumaticos, originalmente el *Minne Rack* solo tenia uno, el cual como se mencionó anteriormente esta conectado al bloque de *interlocks*. Posteriormente viendo la necesidad de tener



más recursos de este tipo, se agregó un bloque extra, conectándolo a una tarjeta de entradas y salidas 34952A.

Ambos *slots* de aire están conectados a lámparas neumáticas.

Eléctricamente las electroválvulas son controladas por ciertas salidas reservadas de la tarjeta 34952A, las cuales por medio del software podemos activar o desactivar.

Estos bloques tienen la capacidad de resistir 120 psi @ 75 F y tienen una durabilidad de 25,000 ciclos [23] .

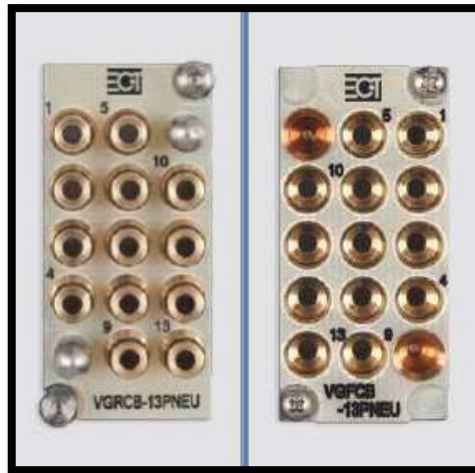


Ilustración 3.26 Slot 8 y 9 Air

A estos bloques se le conectan mangueras de 4mm de diámetro interno.

Para probar todas las electroválvulas se conectaron mangueras, y a estas a su vez a indicadores neumáticos ya vistos en el diseño mecánico, cada vez que se activa una de las electroválvulas, el indicador neumático correspondiente a esa electroválvula se activará.

**Slot 10 (Coaxial):** El Minne Rack cuenta con un bloque para cables coaxiales, los cuales soportan 250VAC, 0.5 Amps, tienen 5mΩ de resistencia en el contacto, cuentan con una impedancia de 50Ω, una capacitancia de 21pf, una inductancia de 3nH, un ancho de banda de 1Ghz y una durabilidad de 25,000 ciclos.[24]

Actualmente solo se tienen 6 cables coaxiales conectados (color amarillo) a una DAC y unos *Totalizer* los cuales también están conectados a los muxes para realizar una medición.

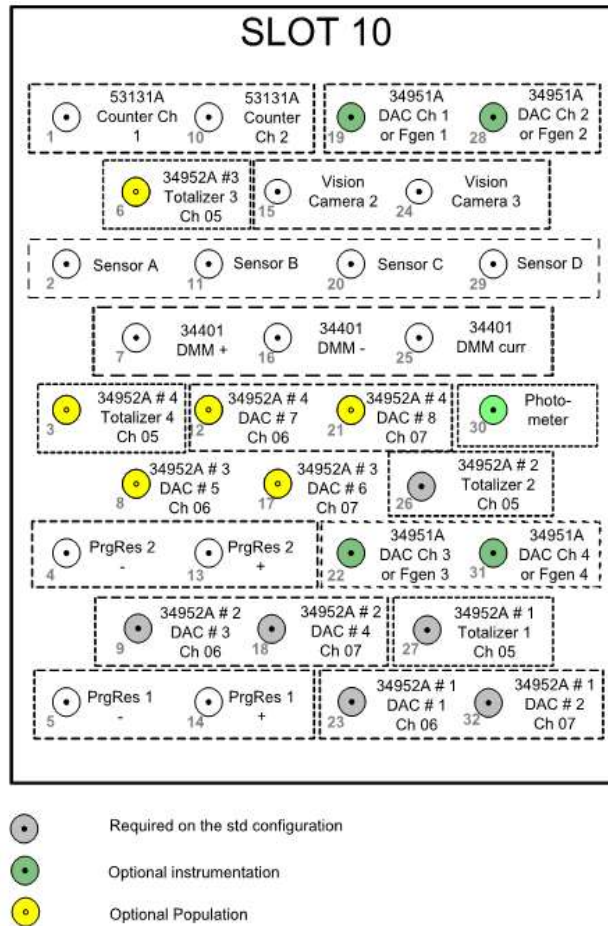


Ilustración 3.27 Slot 10 Coaxial

Por medio del software programamos diversos voltajes a través de sus fuentes programables en cada uno, para poder diferenciarlos y así poder lograr una medición en cada uno.

**Slot 11 (DIO):** Una de las partes más esenciales a probar son las tarjetas de E/S, la tarjeta 34952A que se encuentra conectada en el slot 11.

Este slot tiene un bloque de 170 contactos igual que el slot de relays, multiplexores e interlocks.

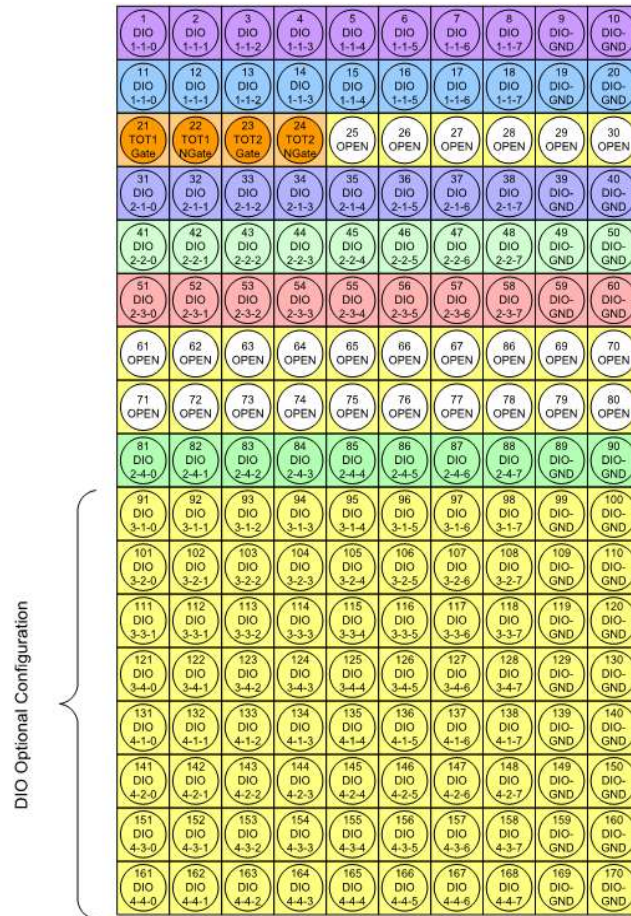


Ilustración 3.28 Slot 11 DIO

El *Minne Rack* por estándar usa una tarjeta de E/S, pero hay una versión de *Minne Rack* donde usa dos, por lo tanto el fixture va a estar con todos los recursos disponibles para probar todas las tarjetas que el rack pueda soportar.

Debido a la estructura y lógica con la que fueron diseñadas esas tarjetas, se diseñó un circuito (ver anexo xxii) para controlar dicha tarjeta.

Se adaptó el PIC18F2550 y el integrado PCF8575 de *Texas Instruments*

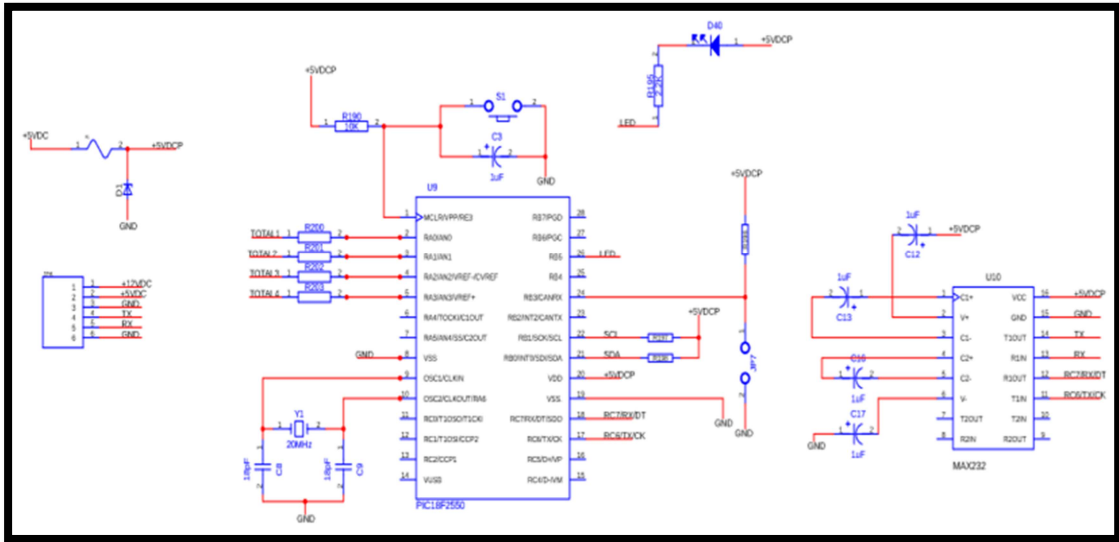


Ilustración 3.29 PIC18F2550 y el integrado PCF8575

Con el PIC18F2550 controlamos todos los integrados PCF8575 a los cuales conectamos todas las E/S de las tarjetas que están interconectadas en el rack.

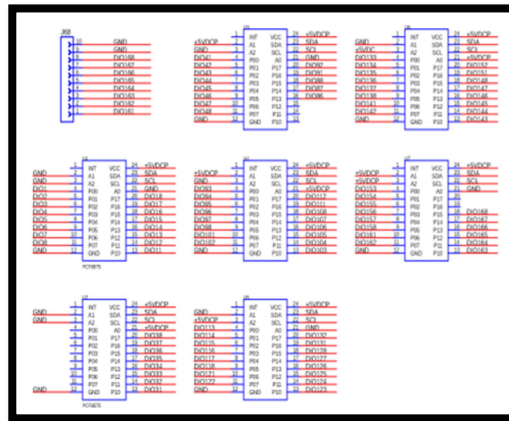


Ilustración 3.30 E/S de las tarjetas que están interconectadas en el rack

Se seleccionó dicho integrado ya que nos proporciona una expansión de E/S remotas de propósito general a través de comunicación I2C y se optó por el PIC18F2550 por que podemos manipular señales mediante I2C.

Esto quiere decir que se usaron varios integrados para controlar y/o mandar datos a todas las E/S posibles en un *Minne Rack*.

Este PIC se comunica a traves de un puerto serial, ya que conectamos un integrado *MAX232* para poder tener una comunicación de ese tipo, este a su vez se conectó a un adaptador de serial a USB, para cubrir el ultimo puerto de USB que faltaba por probar, dados que son 4 los puertos de USB que se prueban y como se mencionó anteriormente, solo se conectaron 3 memorias USB para la prueba.

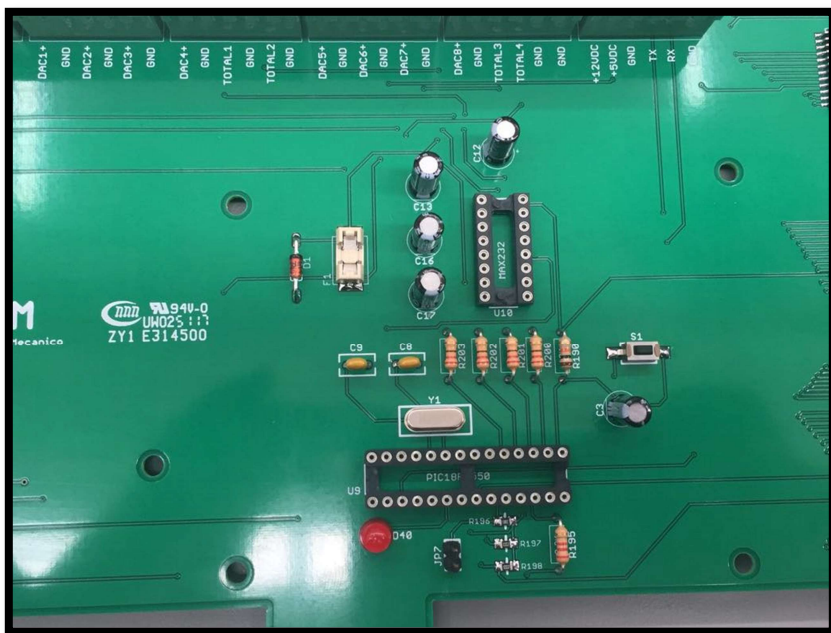


Ilustración 3.31 Seccion DIO PCB

La lógica del programa para utilizar la comunicación I2C se basó en el siguiente diagrama:

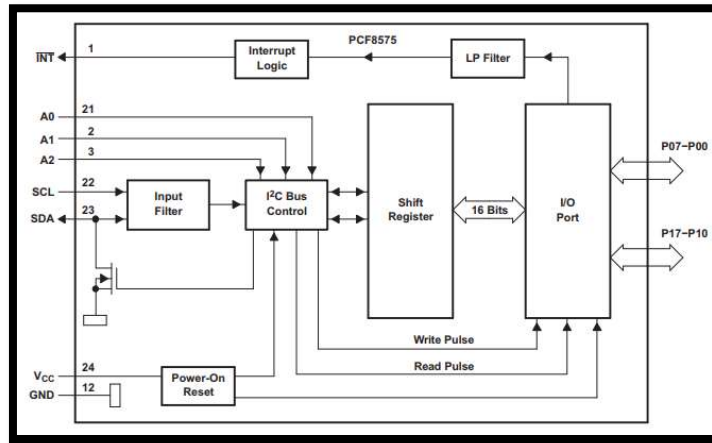


Ilustración 3.32 Diagrama Logico (logica positiva)

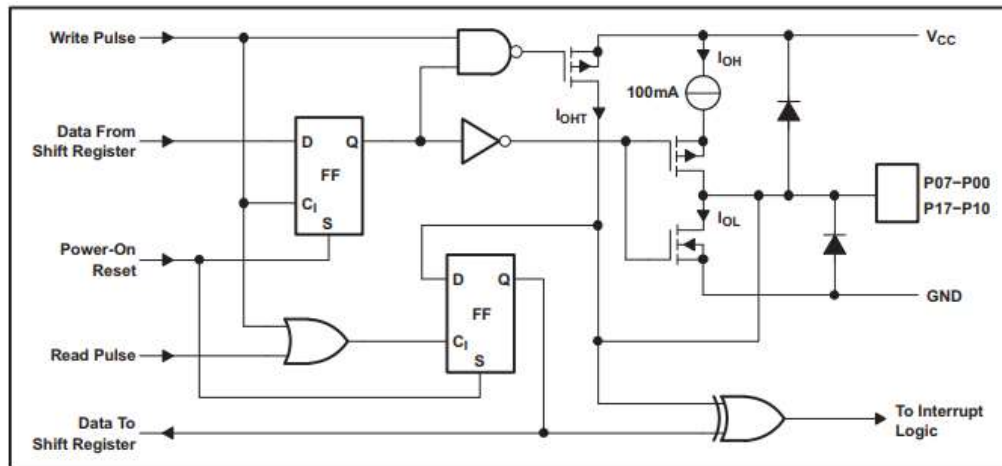


Ilustración 3.33 Diagrama simplificado de cada puerto E/S

Teniendo así el siguiente diagrama de bloques:

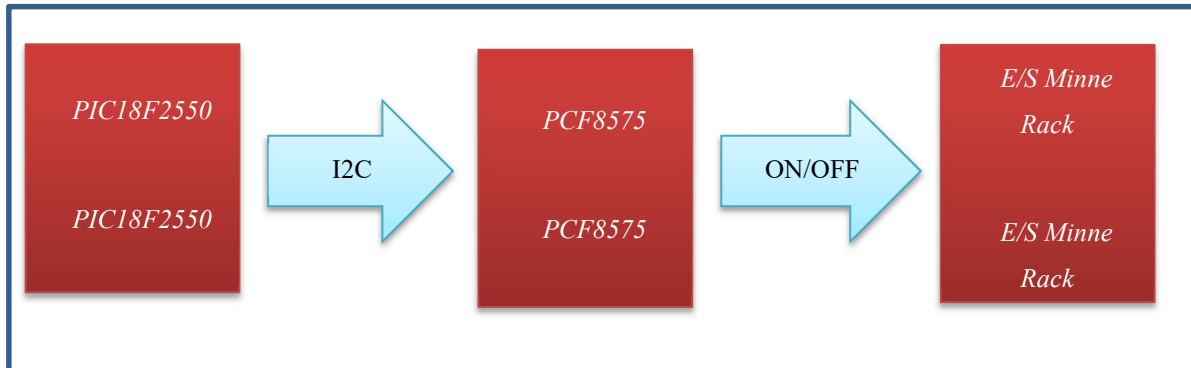


Ilustración 3.34 Diagrama de bloques

Dado que la *PCB* para el *fixture* no hará realmente pruebas a algún producto, solo medirá, es decir, los altos voltajes no los utilizará, estarán ahí pero solo por medición, el ancho de las pistas queda descartado como algo significativo ya que no pasará alta corriente por ninguna de ellas.

Para manufacturar la *PCB* es necesario crear sus archivos *CAD*, en esta ocasión se generaron archivos *Gerber*, los cuales describen cada capa de la tablilla.

Una vez que fue diseñada la tablilla, con las características eléctricas que se describieron de cada bloque de la interfaz, se mandó manufacturar con un proveedor externo con las siguientes especificaciones:



# CAPÍTULO III

Different Design in Panel : 1 e.g.

\* Size (single): Length X Width mm inch<sup>3</sup>→mm

\* Quantity (single): 5 pcs

Layers: 1 Layer **2 Layers** 4 Layers 6 Layers 8 Layers 10 Layers 12 Layers 14 Layers

Material: **Normal FR-4 Board** Aluminum Board Rigid-Flex Board

FR4-TG: **TG 130-140** TG 150-160 TG 170-180

Thickness: 0.4 0.6 0.8 1.0 1.2 **1.6** 2.0 2.4 \* Unit: mm

Min Track/Spacing: 4/4mil 5/5mil **6/6mil** 8/8mil †

Min Hole Size: **0.2mm** 0.25mm 0.3mm † No Drill

Solder Mask: **Green** Red Yellow Blue White Black  
Purple Matte black Matte green None

Silkscreen: **White** Black None

Gold fingers: Yes **No**

Surface Finish: **HASL with lead** HASL lead free Immersion gold OSP Hard Gold Immersion Silver(Ag)  
None

Via Process : **Tenting vias** Plugged vias Vias not covered  
\*For Gerber files, this choice is useless. It will be made according to files as default.

Finished Copper: **1 oz Cu** 2 oz Cu 3 oz Cu 4 oz Cu

Ilustración 3.15 Instrucciones para manufacturar PCB

Obteniendo asi la siguiente tablilla:

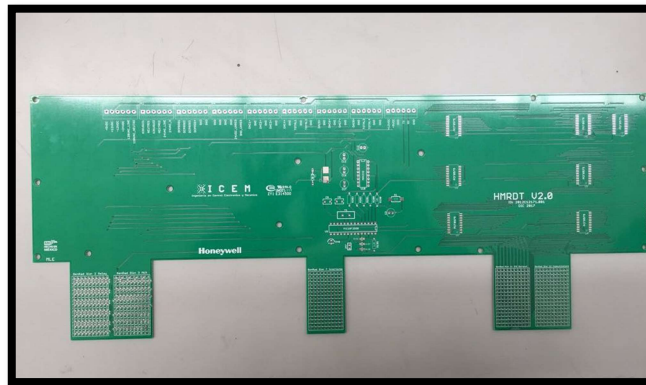


Ilustración 3.36 PCB Manufacturada



### 3.4.3.3 Subsistema de Software

Después de que se trabajó con el diseño mecánico y eléctrico, se generó el código del *fixture del HMRDT*, que se muestra en el anexo xxxix. Código HMRDT

Se crearon funciones para cada prueba y una clase exclusiva para probar las E/S del rack. A continuación se muestra un diagrama de flujo general de la funcionalidad del programa:

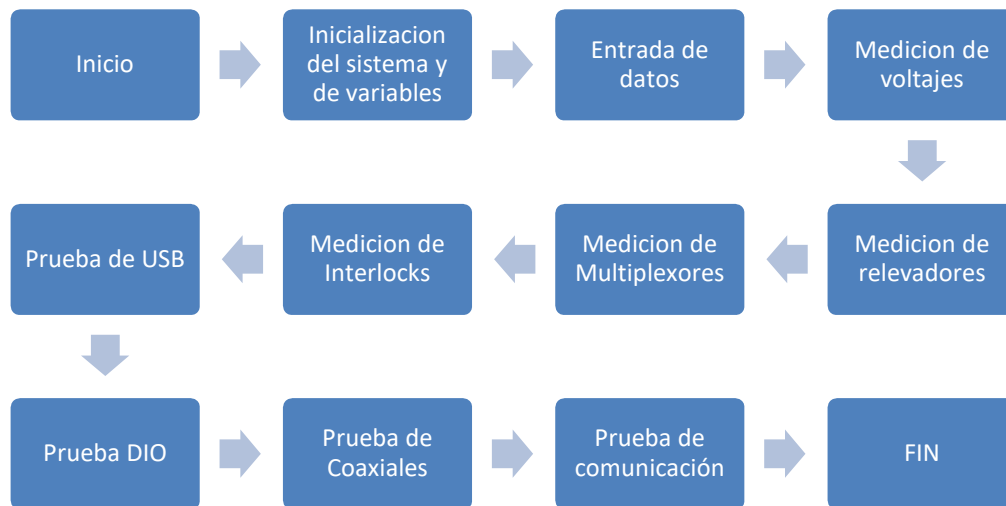


Ilustración 3.37 Diagrama Software

## CAPÍTULO IV. RESULTADOS

Toda falla deja pistas, mismas que permiten encontrar su origen. El diseñador debe conocer muy bien las teorías de las fallas a fin de interpretar adecuadamente estas pistas. El diseño de productos con un alto nivel de confiabilidad implica también el tener los equipos e instrumentos para la detección de fallas, tanto en la etapa de diseño del producto como en la etapa de la manufactura del mismo.

Toda máquina tiene sus niveles normales de ruido, vibración y temperatura. Cuando se observe algún aumento anormal de estos niveles, se tienen los primeros indicios de que hay alguna falla. Los operarios de las máquinas deben ser instruidos para que avisen al detectar estos síntomas que presenta la máquina.

Gracias a este proyecto de tesis se logró la optimización en la prueba de validación de los equipos *Minne Rack*, con la cual se obtendrá una mayor productividad, esto significa un resultado positivo para cualquiera de las dos empresas (Honeywell e ICEM) que quiera automatizar dicha operación.

Para validar el programa y el *fixture*, se enlistaron todas las variables a probar, haciendo que cada variable se midiera con el HMRDT tanto en su estado válido como en estado de error, se implementó que pasaran y fallaran cada una de las mediciones que el *HMRDT* tiene que hacer, así que a continuación se muestran las mediciones hechas cuando un *Minne Rack* estándar no tiene fallas:

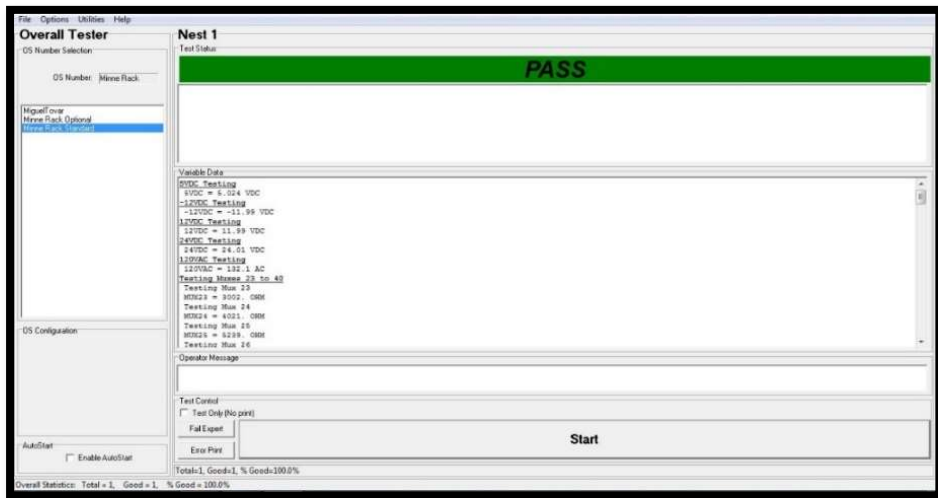


Ilustración 4.1 Prueba de Voltajes

Haciendo *zoom* a la figura anterior obtenemos los resultados de las mediciones de la primer prueba: voltajes.

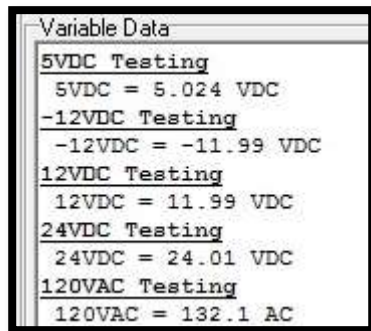


Ilustración 4.2 Resultado Prueba Voltaje

Como podemos observar en la figura, cuando tenemos todas las fuentes conectadas correctamente, se encontrarán voltajes medidos dentro del rango de límites establecidos en el *software* (límites generados por Honeywell), es decir, la fuente de 5 volts no puede estar arriba de 5.1 volt y debajo de 5, puesto que la prueba fallaría.

Después de la prueba de voltajes continuamos con la prueba de los Multiplexores como se muestra en la figura a continuación:

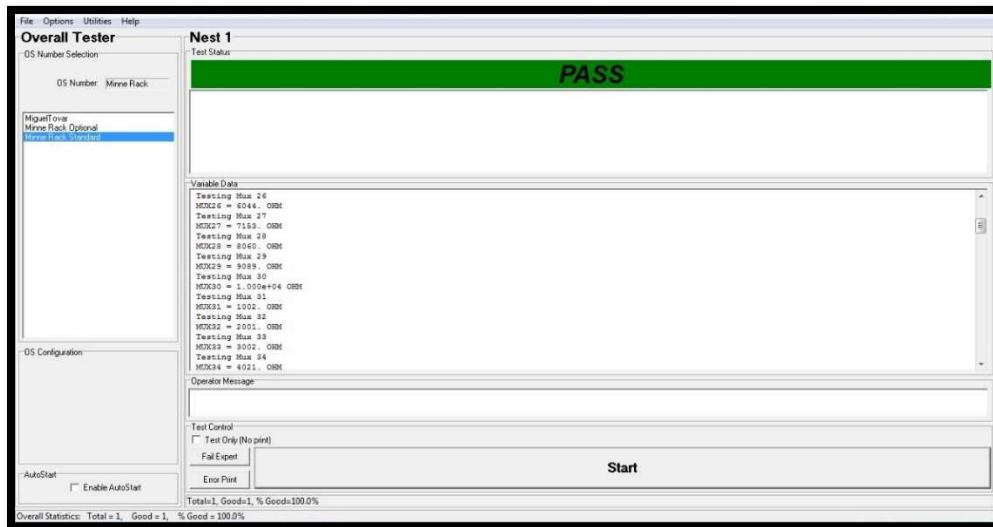


Ilustración 4.3 Prueba Multiplexores

Como ya se mencionó en el capítulo anterior, cada multiplexor fue conectado a una resistencia en serie, con valor de 1K a 10K cada decena, dado que las resistencias tienen una variación de 10%, nos basamos en poner los límites en más o menos 10% del valor de cada resistencia, queda fuera del rango del valor, es que la prueba falló y se necesita checar el multiplexor.

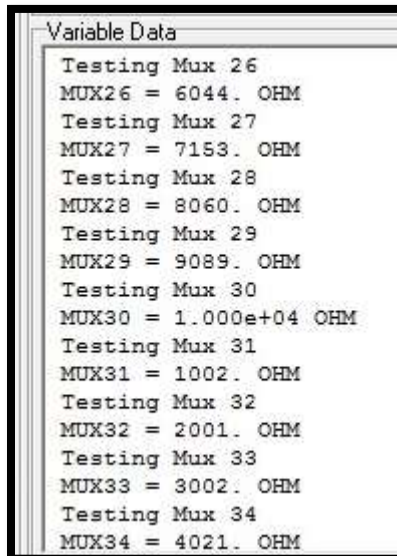


Ilustración 4.4 Resultados de los Multiplexores

La siguiente prueba fue la de los relevadores, se va a medir una resistencia en especifico en cada contacto del NA o NC por medio de un multiplexor, las mediciones son las siguientes:



Ilustración 4.5 Prueba de Relays

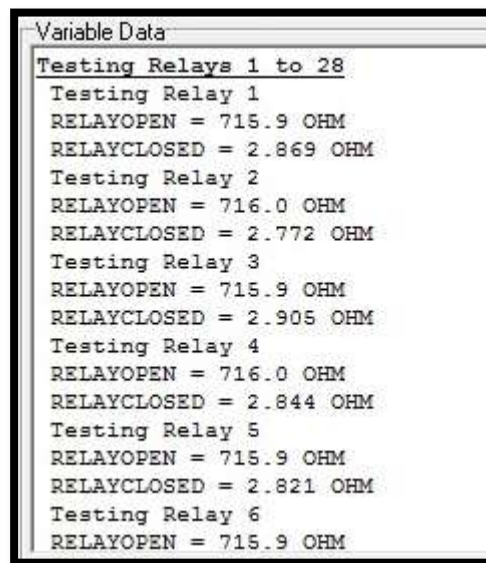


Ilustración 4.6 Resultados Relays

Y así son todas las pruebas consecutivamente. Después de haber probado todo el *Minne Rack*, se pusieron a prueba todas las fallas posibles, así que cuando una

## CAPÍTULO IV

medición está fuera de su límite, el equipo debe realmente detectarlo y mostrar las fallas, como se muestra a continuación:



Ilustración 4.7 Desconexión -12VDC

Se desconectaron los -12VDC del *Minne Rack* y se corrió la prueba, la cual nos muestra los resultados a continuación;

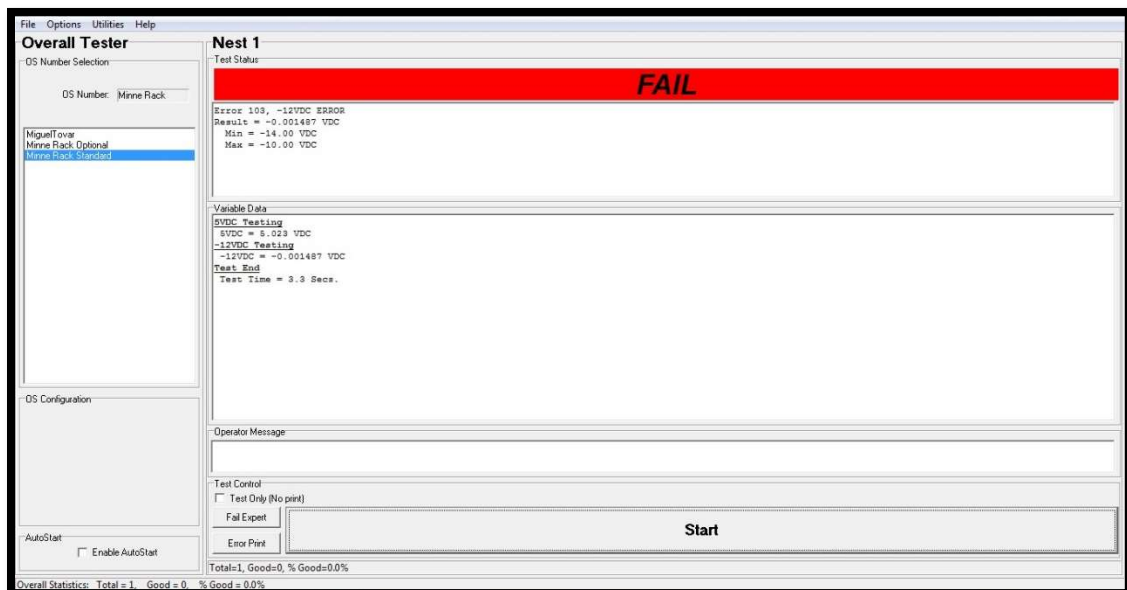


Ilustración 4.8 Prueba -12VDC

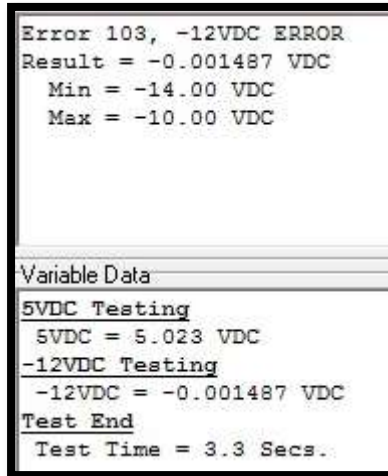


Ilustración 4.9 Resultados -12VDC

El *fixture* intentó medir los -12VDC, los cuales no estaban conectados al *Minne Rack*, y el mismo diagnóstico indica que cuando se están probando los 12VDC, sólo esta leyendo -0.001487VDC, lo cual nos lo indica como error de voltaje en una ventana arriba de la prueba.

Otro ejemplo de una falla es la prueba de los +12VDC, la imagen que se muestra a continuación indica que los +12VDC fueron desconectados del equipo:



Ilustración 4.10 Desconexión de 12VDC

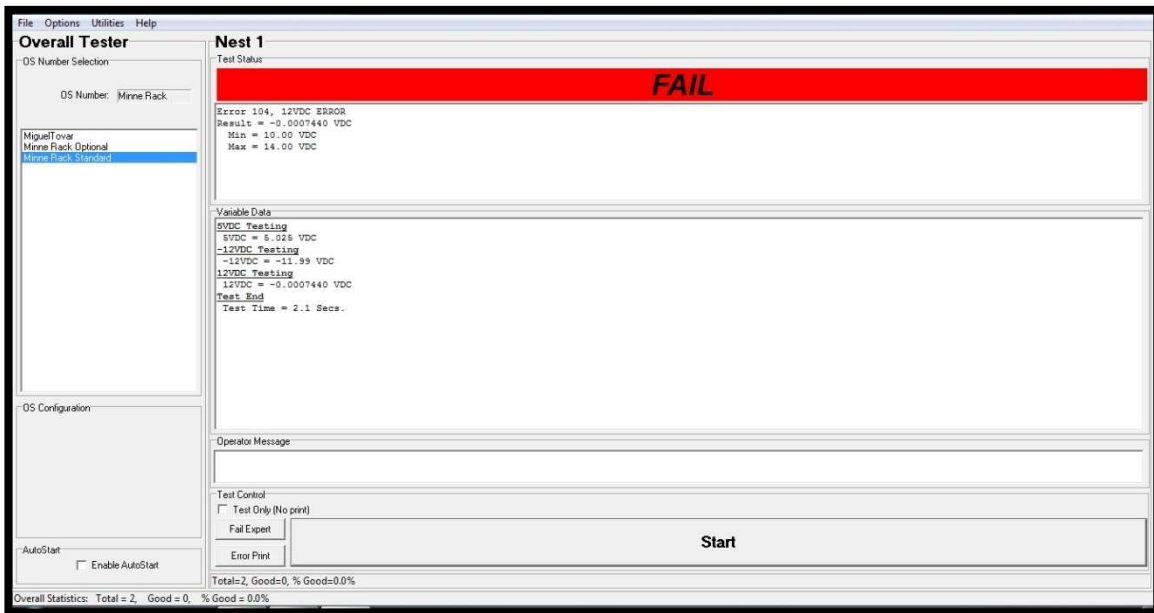


Ilustración 4.11 Prueba de 12VDC

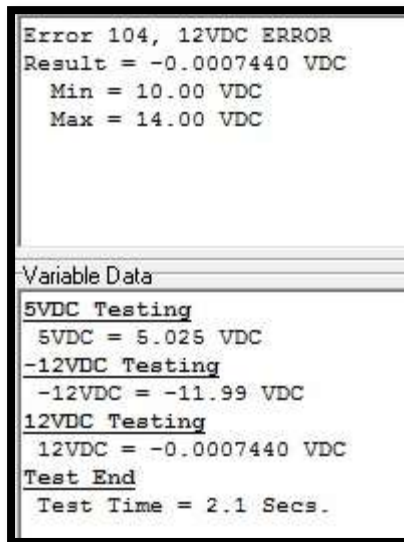


Ilustración 4.12 Resultado 12VDC



## CAPÍTULO IV

Y así se hicieron fallar cada una de las partes del *Minne Rack*, quitando y poniendo tarjetas, o voltajes y/o desconectando instrumentos del mismo, obteniendo todos los resultados deseados.

Antes de realizar el proyecto de tesis, se hizo un estimado de tiempo en el que ICEM tarda en validar un *Minne Rack*, ahora que tenemos las herramientas y el *fixture* HMRDT podemos hacer una comparación entre ambos análisis.

Tabla 4.1 Comparativa HMRDT vs MANUAL

Acción	Tiempo en validación con HMRDT	Tiempo en validación manual
Prueba de voltajes	5s	30min
Prueba de relés	10s	2 hrs
Prueba de Multiplexores	10s	2 hrs
Prueba de E/S	15s	3 hrs
Prueba de comunicación	5 s	2 hrs
Prueba de USB	15s	15s
Prueba de Interlock	3s	30min
Prueba Coaxiales	5s	30min
Prueba pneumática	60 s	40min
	128s	~11hrs 10min 15s

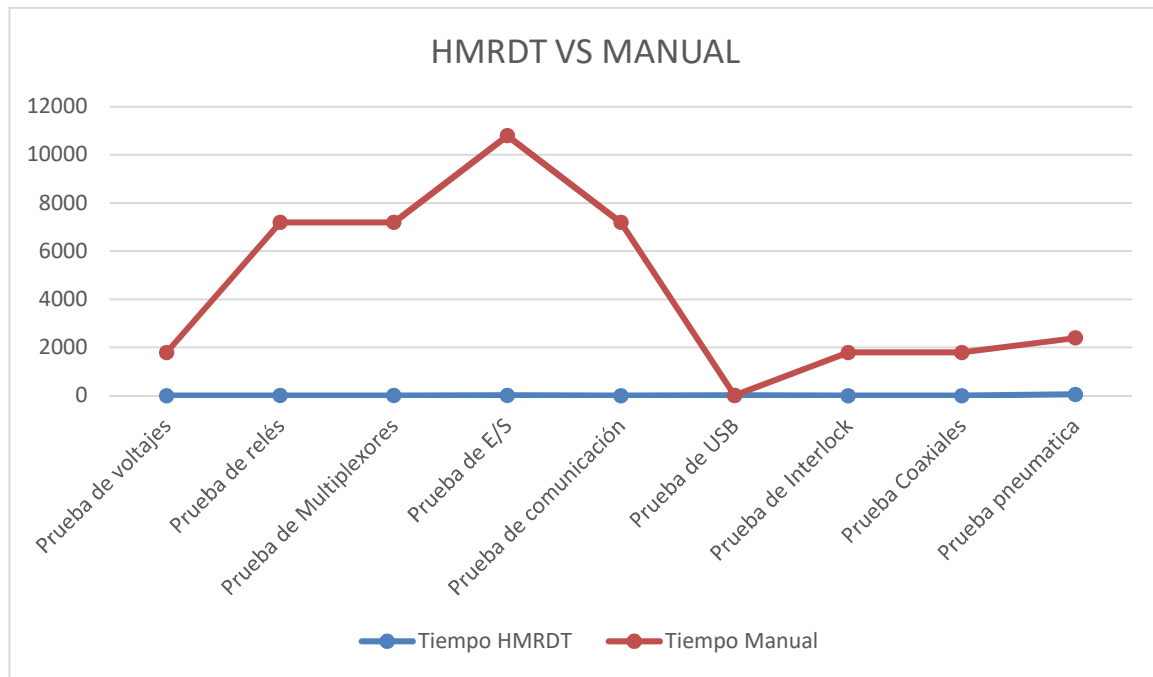


Ilustración 4.13 Grafica de resultados

Los indicadores de análisis de rentabilidad, el tiempo de la prueba del equipo automatizada vs la prueba manual, reflejan una disminución en tiempo y costos, permitiendo obtener mayores ganancias para ambas empresas, ICEM por su parte aceleró el proceso de validación, haciendo la entrega de los equipos de forma más rápida. Lo cual cumple con los objetivos planteados para el presente trabajo.

En cuanto a Honeywell, una vez instalados los equipos en piso y haberlos puesto en operación, estos no han producido fallas y paros de producción que afecten los tiempos y capacidades de producción de la empresa, puesto que gracias a los resultados de este proyecto de tesis, la empresa Honeywell se vió interesada en comprar el proyecto, el cual se encuentra actualmente en la planta de CMO Chihuahua, y CMO Minnesota.

## CAPÍTULO V. CONCLUSIONES

Al diseñar una máquina o equipo se debe tener un profundo conocimiento de la forma en que funciona cada elemento y componentes del sistema y la forma en que estos elementos pueden fallar.

Es importante considerar los requerimientos del cliente, utilizando las herramientas de ingeniería, traducir esos requerimientos en especificaciones de ingeniería para realizar el diseño correcto de los diferentes elementos del sistema completo.

Realizar en forma sistematizada el diseño de un producto utilizando las diversas herramientas de ingeniería, conduce a mejores diseños, libres de fallas potenciales que pueden provocar efectos que van desde inconformidades ligeras, hasta daños severos en los usuarios finales del producto.

En el caso del diseño y desarrollo de equipos de prueba y de instrumentación industrial es de suma importancia considerar todas las posibles fallas que pueden tener los equipos para solucionarlas antes de que estos vayan a producción, un buen proceso de validación para cada uno de ellos asegura varios puntos importantes, por ejemplo una buena satisfacción del cliente, poco tiempo de soporte en planta, que a la larga es un ahorro para la empresa, y trabajo asegurado con el mismo cliente.

Así pues la implementación de la herramienta HMRDT en el proceso de *debuggeo* impactó directamente en la eficacia de realizar las pruebas que anteriormente se llevaban a cabo de manera manual, con dicha implementación se redujo el tiempo de validación del equipo y de sus componentes tales como: identificación de instrumentos, detección de fuentes de alimentación, medición de voltajes y cargas, así mismo se pudo obtener el análisis de las tarjetas en el equipo.

Lo que se puede concluir que al implementar el uso de la herramienta se logra una estandarización al proceso de diagnóstico al equipo de prueba *Minne Rack*.

**Líneas futuras:** A pesar de los buenos resultados obtenidos, cualquier proyecto tiene un área de oportunidad para mejorar. A continuación se describe una de las posibles líneas de actuación que se podrían seguir en el futuro:

El *Minne Rack* tiene la posibilidad de comprarse en conjunto con un sistema de visión llamado *Vision Box*, este equipo contiene una caja que cubre toda la interfaz *genrad* y los *fixtures* que pueden ser conectados en el *minne rack*, por dentro tiene lámparas, cámaras, y unas puertas en las cuales el usuario abre o cierra con algunos botones.

Con el mismo HMRDT podemos adaptar el código para probar todos los sensores, botones y sistema de visión que contenga el *vision box*, por lo cual se puede considerar como un proyecto a futuro para mejorar las herramientas del HMRDT.

**Factores de inhibición:** Se tuvieron algunos problemas con los que se contaron al desarrollar y diseñar el HMRDT, específicamente fueron en la parte eléctrica, ya que al principio se había contemplado hacer varias tablillas modulares que conectarán a una principal. El problema que se tuvo fue que al compilar el código, algunas mediciones no tenían sentido, dado que el sistema modular de *PCB's* estaba conectado entre si por cable plano, el cual metía muchas interferencias al hacer las pruebas.

Otro de los problemas al hacer una segunda revisión de la tablilla fue que se hizo de varias capas, está tablilla al ser de varias capas su costo de manufactura era bastante elevado, por lo que se decidió hacer una tablilla de sólo dos capas.

**Conocimientos adquiridos:** Al comenzar este proyecto se tuvo un campo amplio de investigación y aprendizaje en cuanto a equipos de prueba se refiere.

Se aprendió a analizar las conexiones de varios sistemas de instrumentación, desde analizar cada parte del equipo principal, hasta utilizar todas y cada una de las señales de salida del mismo.

Se aprendió a economizar el proyecto en la medida de lo posible, también a mejorar el tipo de programación, así como a desarrollar un tipo de automatización estándar para la prueba y validación de todos los equipos, ya que es importante demostrar que ICEM como empresa integradora cuente con la capacidad de desarrollar tecnología para la prueba y validación de los equipos del cliente.

**Impacto:** Gracias al programa de maestría se pudo realizar este proyecto para mejorar el proceso de validación de *Minne Racks*, adelantando así el tiempo de entrega de los equipos, teniendo un gran impacto en la satisfacción del cliente.

Dada a la buena satisfacción del cliente, este proyecto tuvo la oportunidad de presentarse ante Honeywell para ver la posibilidad de que ellos pudieran comprar el proyecto. Hoy en día, se han vendido tres HMRDT con un valor de \$18,000.00 dls, uno se encuentra en Honeywell Minnesota y los otros dos en Honeywell CMO Chihuahua, teniendo un impacto considerable económica y académicamente.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Google Maps, «Google Maps,» Google, 08 Febrero 2005. [En línea]. Available: <https://www.google.com.mx/maps/place/ICEM+Ingenieria+en+Control+Electronico+y+Mecanico+S.A.+de+C.V./@28.7163607,-106.1286777,17z/data=!4m5!3m4!1s0x86ea4398abd02665:0x1f401cec3337c172!8m2!3d28.7166313!4d-106.1266981>. [Último acceso: 15 06 2018].
- [2] A. K. y D. S. , «Precise measurement of voltage flicker in electric supply networks by means of data acquisition system,» de *Instrumentation and measurement technology conference, 1999.IMTC/99. Proceedings of the 16th IEEE*, 1999.
- [3] R. S. G. A. I. L. y M. B. , «Comparison between Agilent and National Instruments functional test systems,» de *Intelligent Systems and Informatics (SISY),2010 8th International Symposium on*, 2010.
- [4] J. Z. C. W. y H. R. , «Test program for Honeywell/DND helicopter integrated navigation system (HINS),» de *Position Location and Navigation Symposium, 1990.Recrod. The 1990's- A decade of excellence in the navigation sciences. IEEE PLANS '90., IEEE*, 1990.
- [5] L.Brogan, «Reducing costs through the use of special purpose automatic test equipment (SPATE),» de *AUTOTESTCON '94. IEEE Systems Readiness Technology Conference. 'Cost Effective Support Into the Next Century', Conference Prodeedings.*, 1994.
- [6] R. W. K. T. A. S. y J. K. , «Automated validation test generation,» de *Digital Avionics Systems Conference, 1994. 13th DASC., AIAA/IEEE*, 2002.

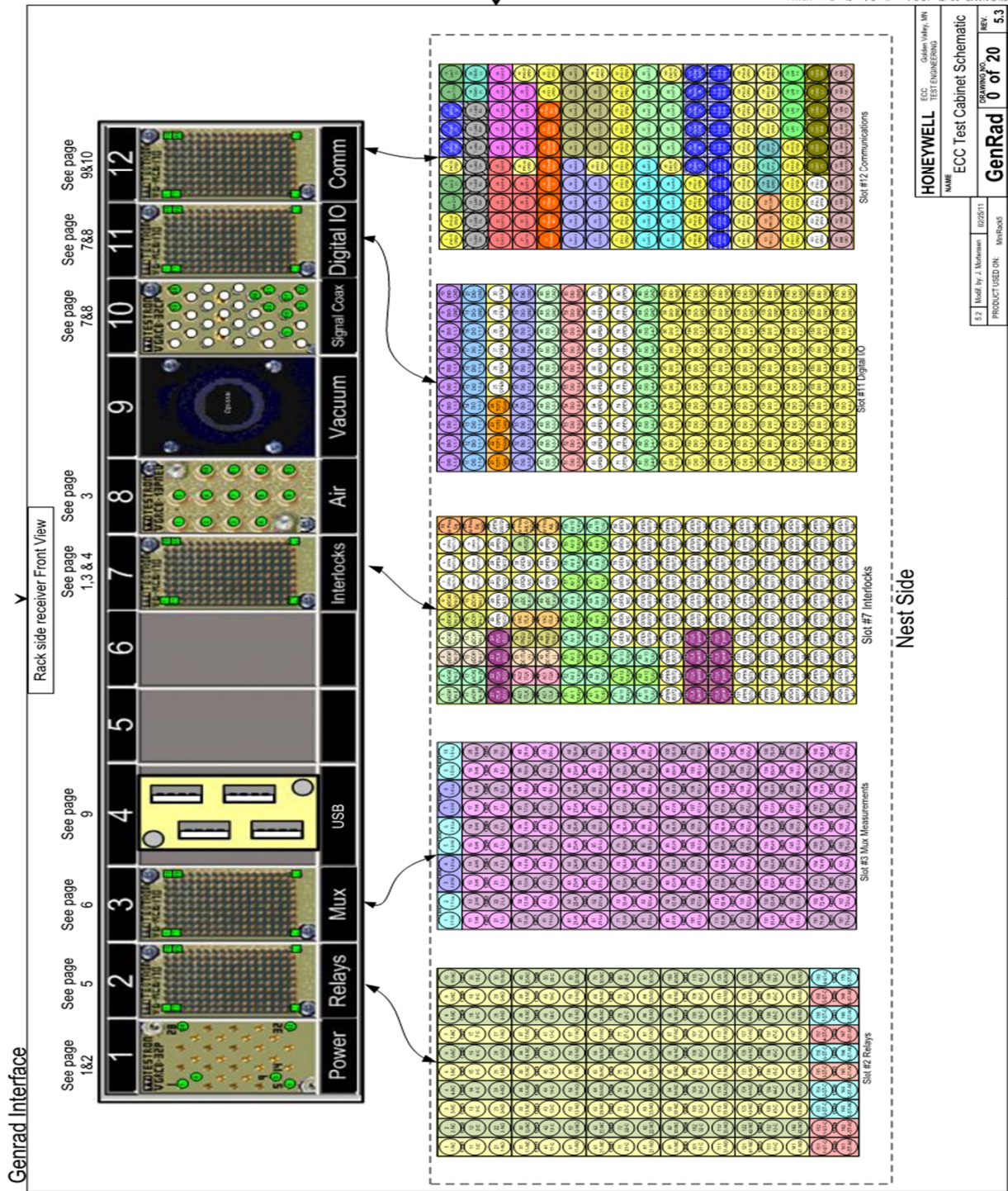
- [7] T. L. S. W. y M. S. , «Using advanced tools to automate the design, generation and execution of formal qualification testing,» de *AUTOTESTCON '96, Test Technology and Commercialization. Conference Record*, 1996.
- [8] M. M. Rahman, S. M. A. A. K. M. y S. E. Reza, «Design and construction of a high-precision electrical parameters monitoring system,» de *Informatics, Electronics and Vision (ICIEV), 2016 5th International Conference on*, 2016.
- [9] S. C. G. H. J. K. S. O. Y. L. C. K. y T. L. , «Development of PCB design guide and PCB deformation simulation tool for slim PCB quality and reliability,» de *Electronic Components and Technology Conference (ECTC), 2013 IEEE 63rd*, 2013.
- [10] C. Hayes, «Circuits by design [Electronics PCB Design],» *Engineering & Technology, IEEE Xplore*, pp. 80-83, 2013.
- [11] Z. K. Baker y J. S. Monson, «In-situ FPGA Debug Driven by On-Board Microcontroller,» de *Field Programmable Custom Computing Machines, 2009. FCCM '09. 17th IEEE Symposium on*, 2009.
- [12] M. M. C. B. T. W. y W. G. , «Embedded systems verification with FPGA-enhanced in-circuit emulator,» de *System Synthesis, 2000. Proceedings. The 13th International Symposium on*, 2002.
- [13] Autodesk, «Wikipedia,» Autodesk, 30 12 2016. [En línea]. Available: <https://es.wikipedia.org/wiki/EAGLE>. [Último acceso: 16 06 2018].
- [14] Autodesk, «Wikipedia,» Autodesk, 1 Noviembre 1995. [En línea]. Available: <https://es.wikipedia.org/wiki/SolidWorks>. [Último acceso: 16 Junio 2018].
- [15] E. Delphi, «Embarcadero,» 02 Julio 2014. [En línea]. Available: [https://es.wikipedia.org/wiki/Embarcadero\\_Delphi](https://es.wikipedia.org/wiki/Embarcadero_Delphi). [Último acceso: 16 06 2018].
- [16] «Wikipedia,» CCS Compiler, [En línea]. Available: <http://www.ccsinfo.com/content.php?page=compilers>.

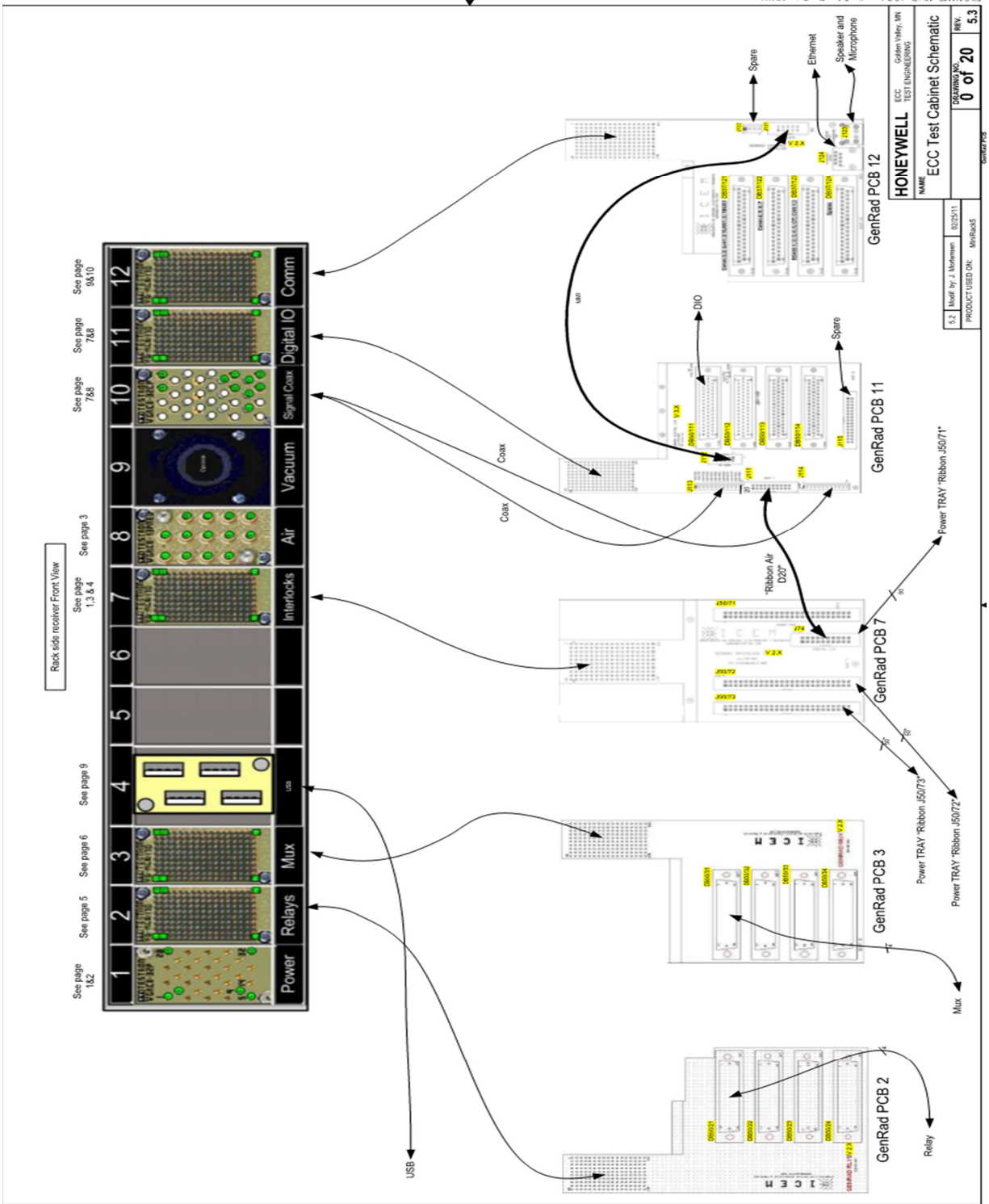
- [17] «Honeywell,» 2016. [En línea]. Available: <http://www.honeywell.com/who-we-are/our-history>.
- [18] E. Charles, «An Introduction to Reliability and Maintainability Engineering,» Boston, Editorial Mc. Graw-Hill, 1997.
- [19] I. G, «Handbook of reliability engineering and management,» New York, Mc. Graw-Hill, 1996.
- [20] A. Kelly, «Maintenance,» England, Butterworth Heinemann, 1994.
- [21] E. Charles. [En línea]. Available: <https://ect-cpg.com/wp-content/uploads/2018/06/VG-170F.pdf>.
- [22] «Everette Charles,» [En línea]. Available: <https://ect-cpg.com/wp-content/uploads/2018/07/VG-13PNEU.pdf>.
- [23] E. Charles. [En línea]. Available: <https://ect-cpg.com/wp-content/uploads/2018/06/VG-30CPF.pdf>.



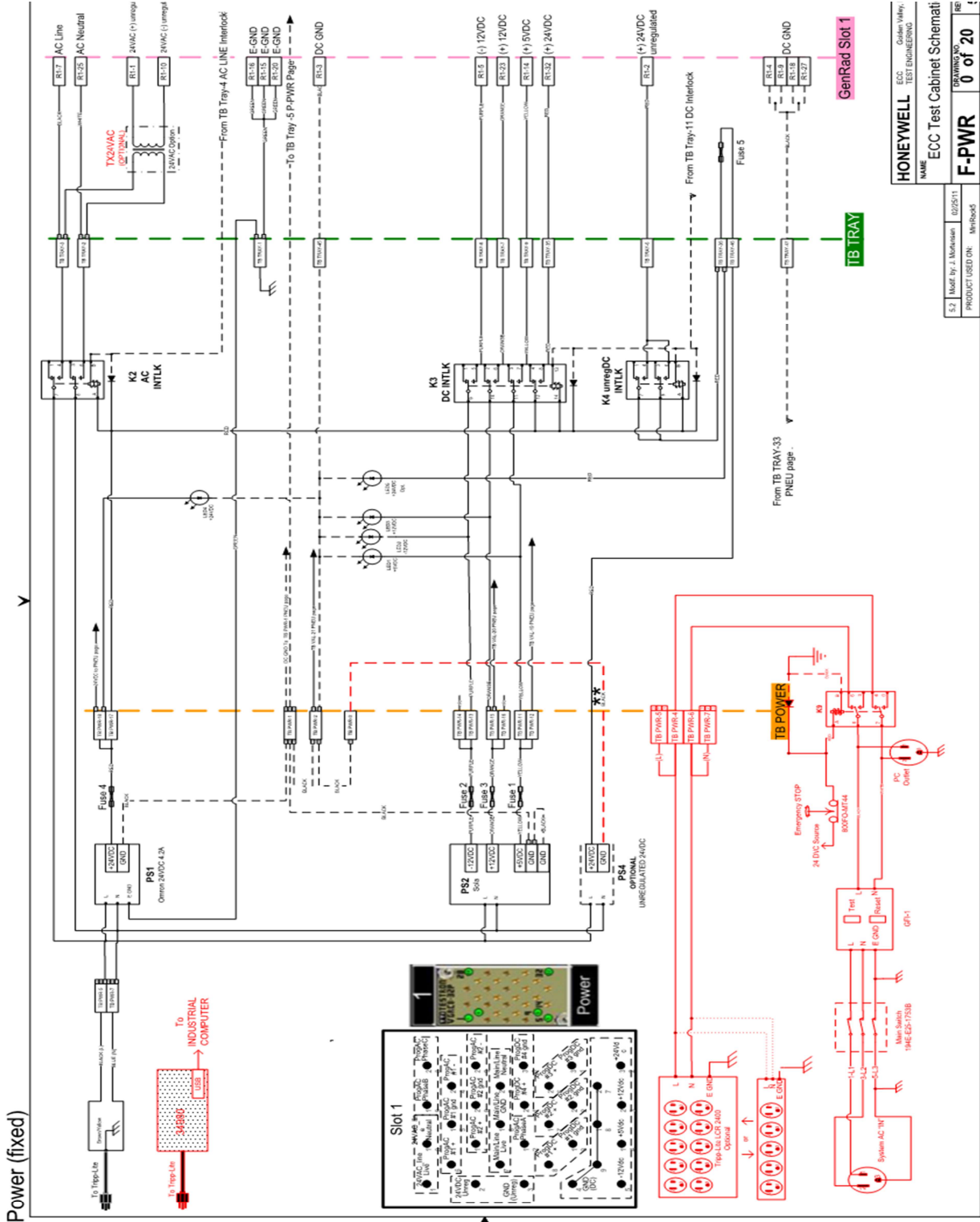
# ANEXOS

## ANEXO i



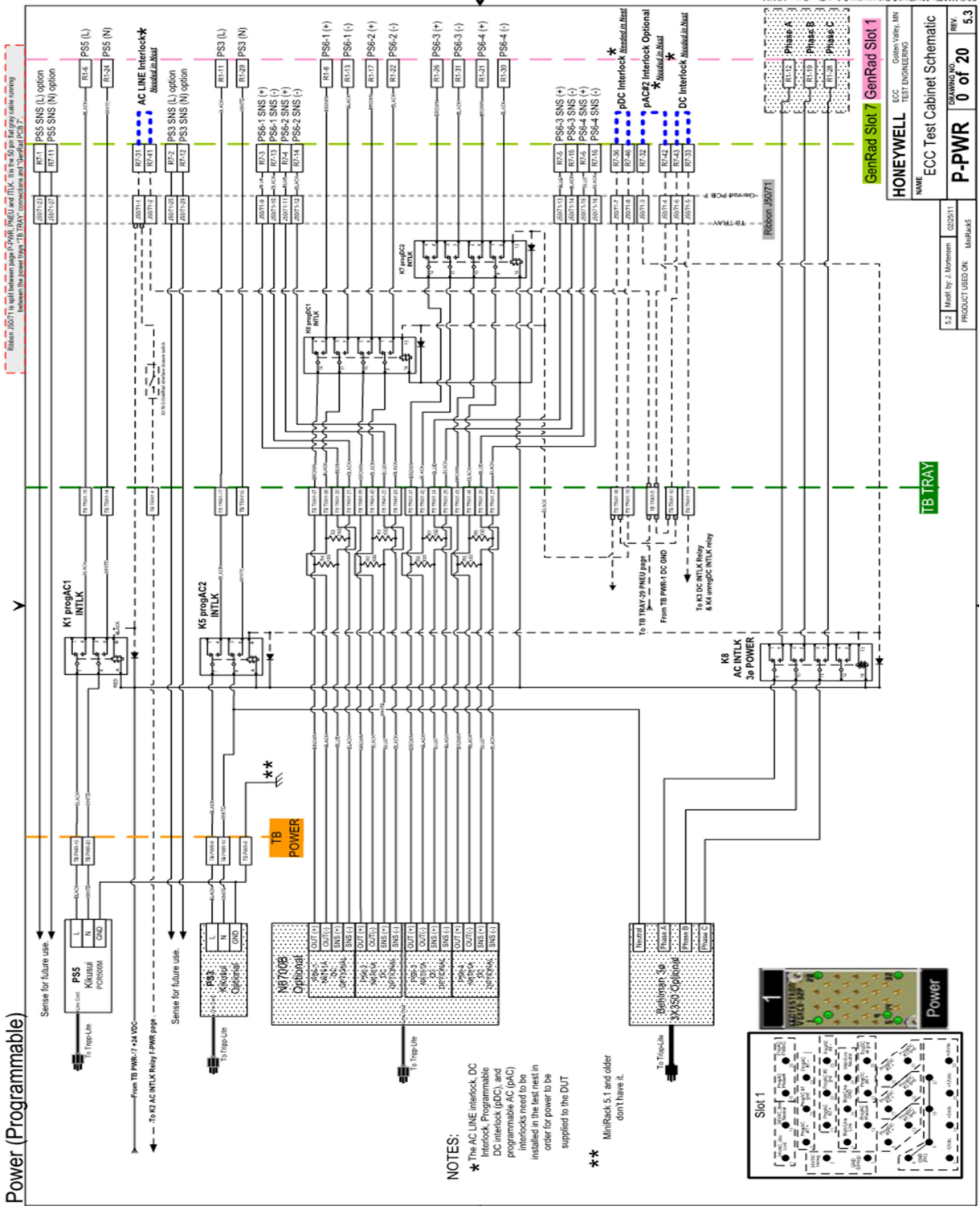


ANEXO v



HONEYWELL ECC Golden Valley TEST ENGINEERING	
NAME: ECC Test Cabinet Schemati	
DATE: 02/25/11	REV: 0 of 20
F-PWR	
5.2 Made By: J. Makusian	PRODUCT USED ON: MtrRad5



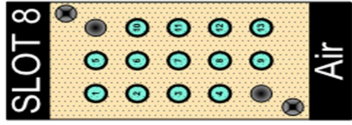
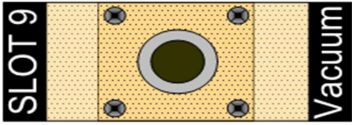
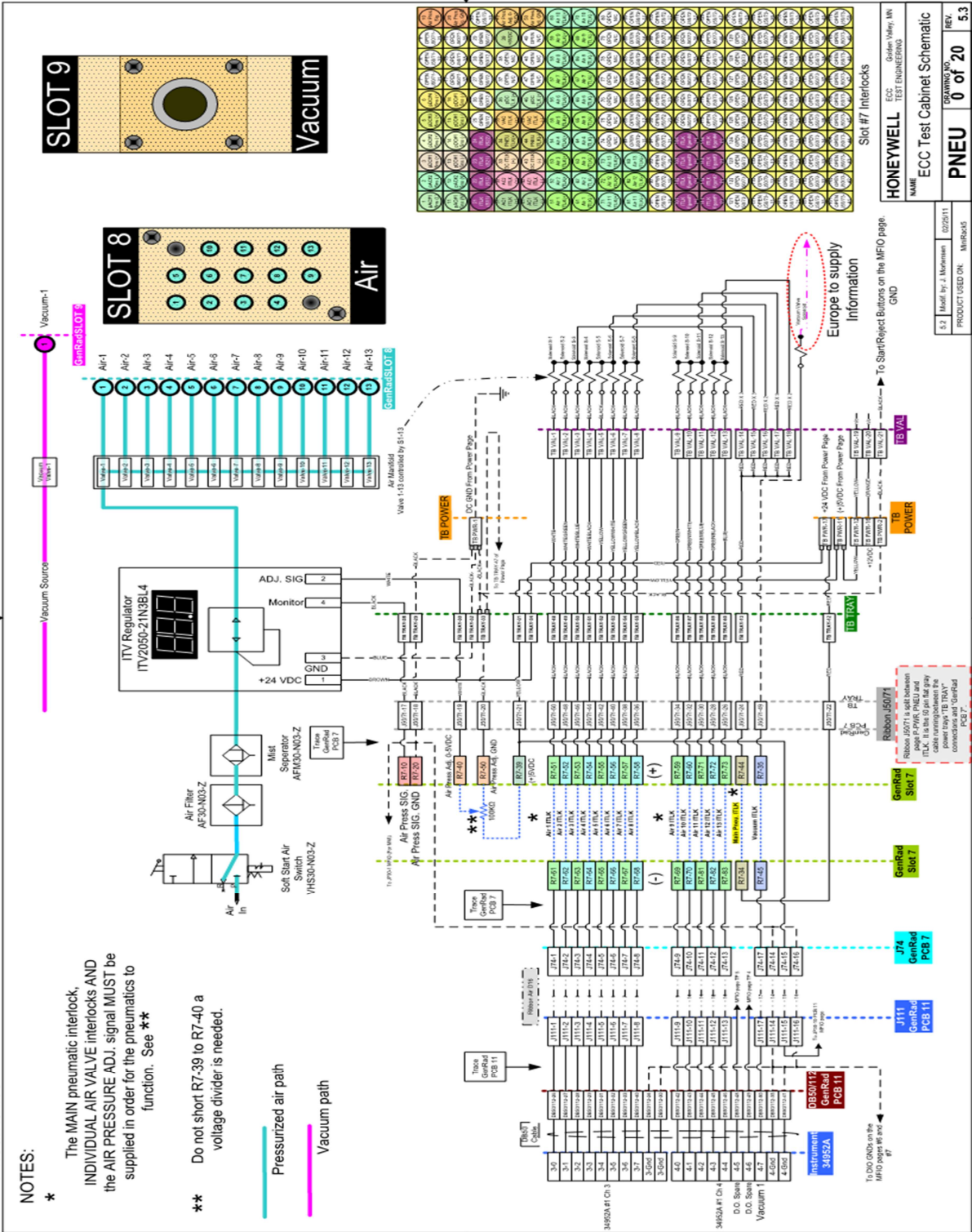


### Pneumatics

**NOTES:**

- \* The MAIN pneumatic interlock, INDIVIDUAL AIR VALVE interlocks AND the AIR PRESSURE ADJ. signal MUST be supplied in order for the pneumatics to function. See \*\*
- \*\* Do not short R7-39 to R7-40 a voltage divider is needed.

Pressurized air path  
Vacuum path



Slot #7 Interlocks

Slot	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	2005-1	2005-2	2005-3	2005-4	2005-5	2005-6	2005-7	2005-8	2005-9	2005-10	2005-11	2005-12	2005-13	2005-14	2005-15	2005-16	2005-17
2	2006-1	2006-2	2006-3	2006-4	2006-5	2006-6	2006-7	2006-8	2006-9	2006-10	2006-11	2006-12	2006-13	2006-14	2006-15	2006-16	2006-17
3	2007-1	2007-2	2007-3	2007-4	2007-5	2007-6	2007-7	2007-8	2007-9	2007-10	2007-11	2007-12	2007-13	2007-14	2007-15	2007-16	2007-17
4	2008-1	2008-2	2008-3	2008-4	2008-5	2008-6	2008-7	2008-8	2008-9	2008-10	2008-11	2008-12	2008-13	2008-14	2008-15	2008-16	2008-17
5	2009-1	2009-2	2009-3	2009-4	2009-5	2009-6	2009-7	2009-8	2009-9	2009-10	2009-11	2009-12	2009-13	2009-14	2009-15	2009-16	2009-17
6	2010-1	2010-2	2010-3	2010-4	2010-5	2010-6	2010-7	2010-8	2010-9	2010-10	2010-11	2010-12	2010-13	2010-14	2010-15	2010-16	2010-17
7	2011-1	2011-2	2011-3	2011-4	2011-5	2011-6	2011-7	2011-8	2011-9	2011-10	2011-11	2011-12	2011-13	2011-14	2011-15	2011-16	2011-17
8	2012-1	2012-2	2012-3	2012-4	2012-5	2012-6	2012-7	2012-8	2012-9	2012-10	2012-11	2012-12	2012-13	2012-14	2012-15	2012-16	2012-17
9	2013-1	2013-2	2013-3	2013-4	2013-5	2013-6	2013-7	2013-8	2013-9	2013-10	2013-11	2013-12	2013-13	2013-14	2013-15	2013-16	2013-17
10	2014-1	2014-2	2014-3	2014-4	2014-5	2014-6	2014-7	2014-8	2014-9	2014-10	2014-11	2014-12	2014-13	2014-14	2014-15	2014-16	2014-17
11	2015-1	2015-2	2015-3	2015-4	2015-5	2015-6	2015-7	2015-8	2015-9	2015-10	2015-11	2015-12	2015-13	2015-14	2015-15	2015-16	2015-17
12	2016-1	2016-2	2016-3	2016-4	2016-5	2016-6	2016-7	2016-8	2016-9	2016-10	2016-11	2016-12	2016-13	2016-14	2016-15	2016-16	2016-17
13	2017-1	2017-2	2017-3	2017-4	2017-5	2017-6	2017-7	2017-8	2017-9	2017-10	2017-11	2017-12	2017-13	2017-14	2017-15	2017-16	2017-17
14	2018-1	2018-2	2018-3	2018-4	2018-5	2018-6	2018-7	2018-8	2018-9	2018-10	2018-11	2018-12	2018-13	2018-14	2018-15	2018-16	2018-17
15	2019-1	2019-2	2019-3	2019-4	2019-5	2019-6	2019-7	2019-8	2019-9	2019-10	2019-11	2019-12	2019-13	2019-14	2019-15	2019-16	2019-17
16	2020-1	2020-2	2020-3	2020-4	2020-5	2020-6	2020-7	2020-8	2020-9	2020-10	2020-11	2020-12	2020-13	2020-14	2020-15	2020-16	2020-17
17	2021-1	2021-2	2021-3	2021-4	2021-5	2021-6	2021-7	2021-8	2021-9	2021-10	2021-11	2021-12	2021-13	2021-14	2021-15	2021-16	2021-17

HONEYWELL ECC Golden Valley, MN  
TEST ENGINEERING

NAME: ECC Test Cabinet Schematic  
REV: 0 of 20

PRODUCT USED ON: MWRKAS

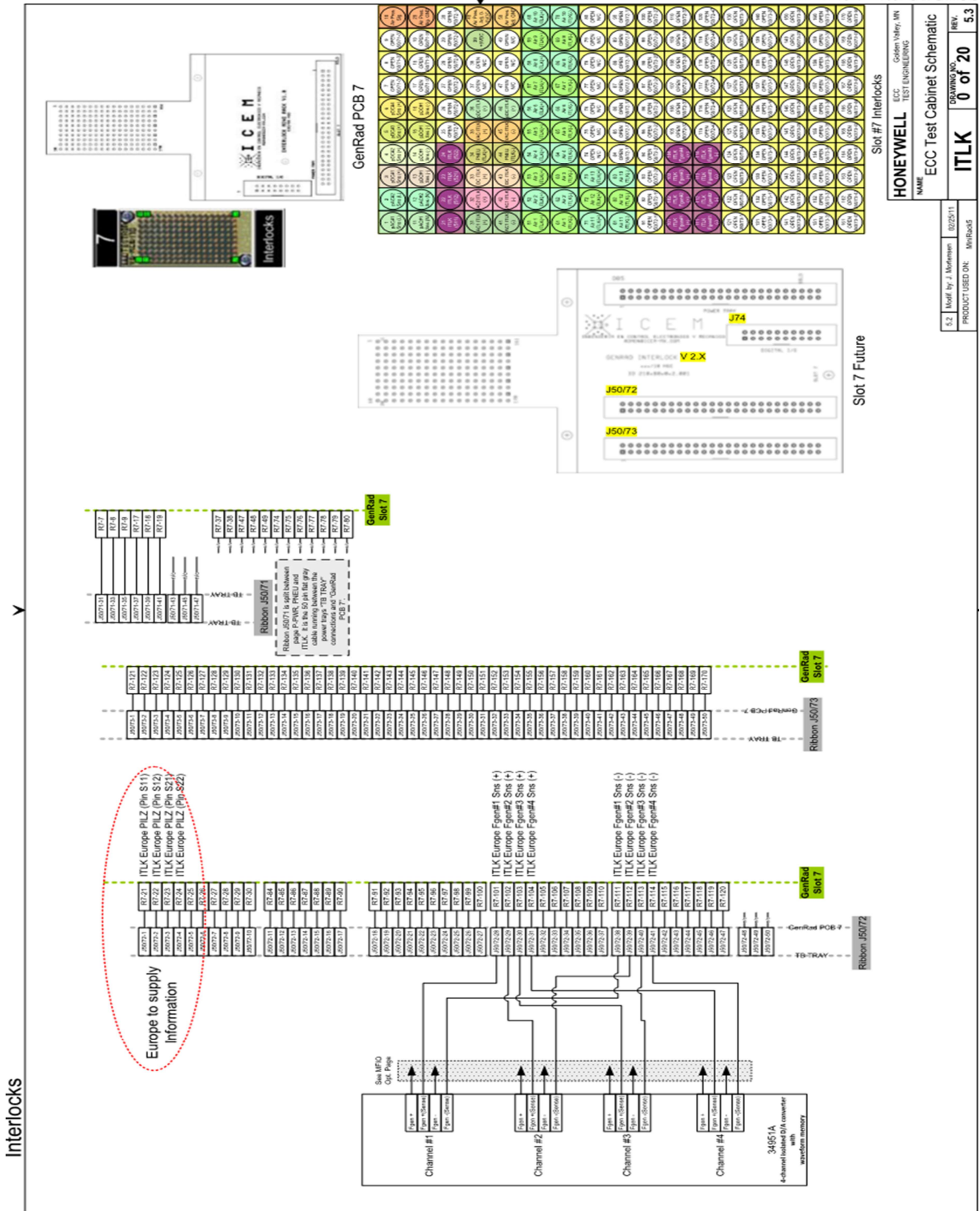
5.3 Modified by: J. Mortenson 02/28/11

Europe to supply Information

To Start/Reject Buttons on the MFC page

Ribbon 150771  
From the MFC page, set the  
Ribbon PWR (PWR) and  
Ribbon ILLK (ILLK) to the  
state shown in the table  
below. If the ILLK pin for any  
cable connector is not  
connected, connect the  
connector and Ground  
PCB F.

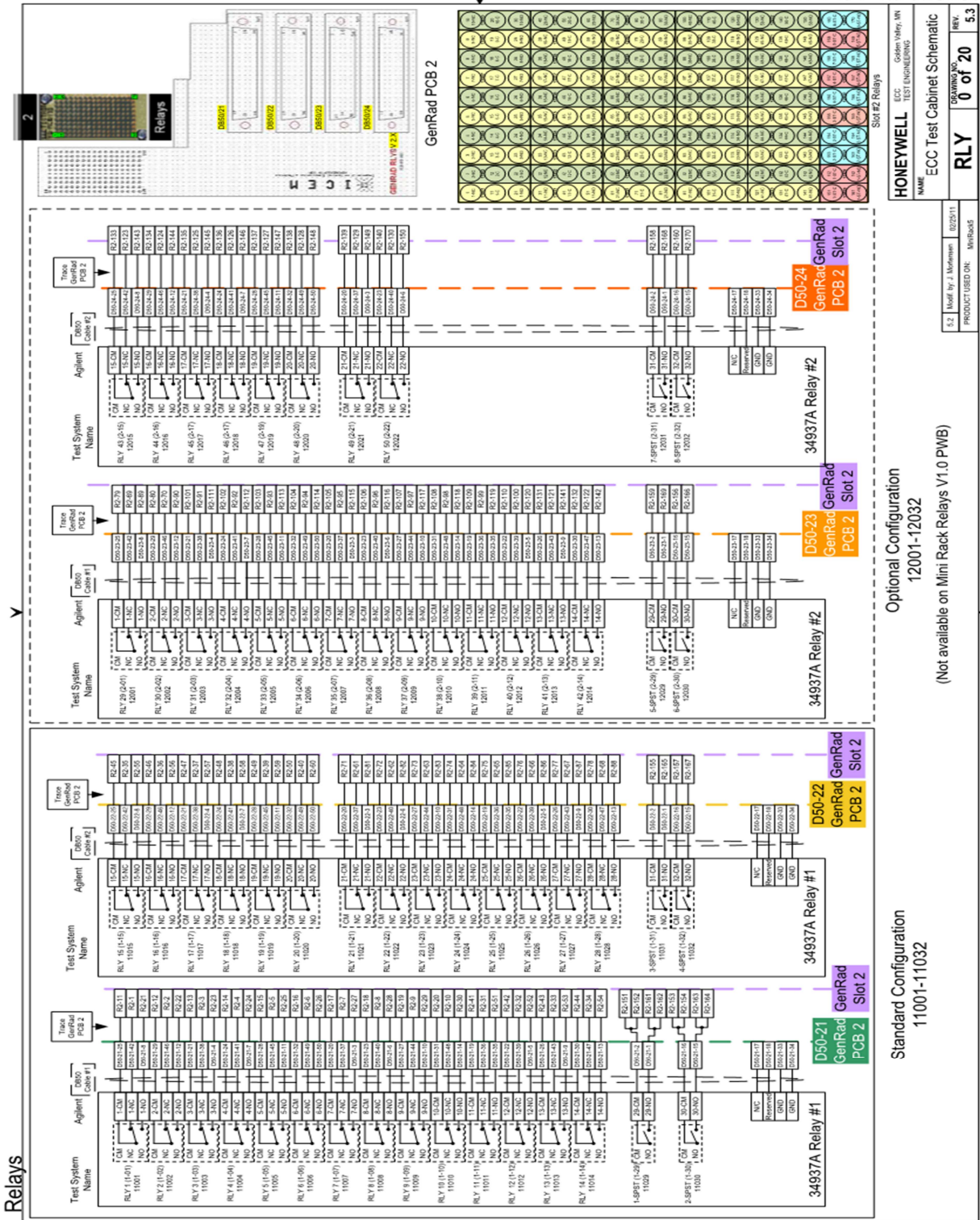
DB50117 GenRad PCB 11  
J11-01 GenRad PCB 11  
J11-02 GenRad PCB 11  
J11-03 GenRad PCB 11  
J11-04 GenRad PCB 11  
J11-05 GenRad PCB 11  
J11-06 GenRad PCB 11  
J11-07 GenRad PCB 11  
J11-08 GenRad PCB 11  
J11-09 GenRad PCB 11  
J11-10 GenRad PCB 11  
J11-11 GenRad PCB 11  
J11-12 GenRad PCB 11  
J11-13 GenRad PCB 11  
J11-14 GenRad PCB 11  
J11-15 GenRad PCB 11  
J11-16 GenRad PCB 11  
J11-17 GenRad PCB 11



**HONEYWELL** ECC Golden Valley, MN  
 TEST ENGINEERING  
 NAME  
**ECC Test Cabinet Schematic**  
**ITLK** DRAWING NO: **0 01 20** REV: **5.3**  
 5.2 Mod by J. Macomber 10/20/11  
 PRODUCT USED ON: WTRBox5

FILE NAME: MTRBox5 Schematics 5.4 VTRKey Data: 4/21/11





HONEYWELL  
ECC  
TEST ENGINEERING  
Golden Valley, MN

NAME  
ECC Test Cabinet Schematic

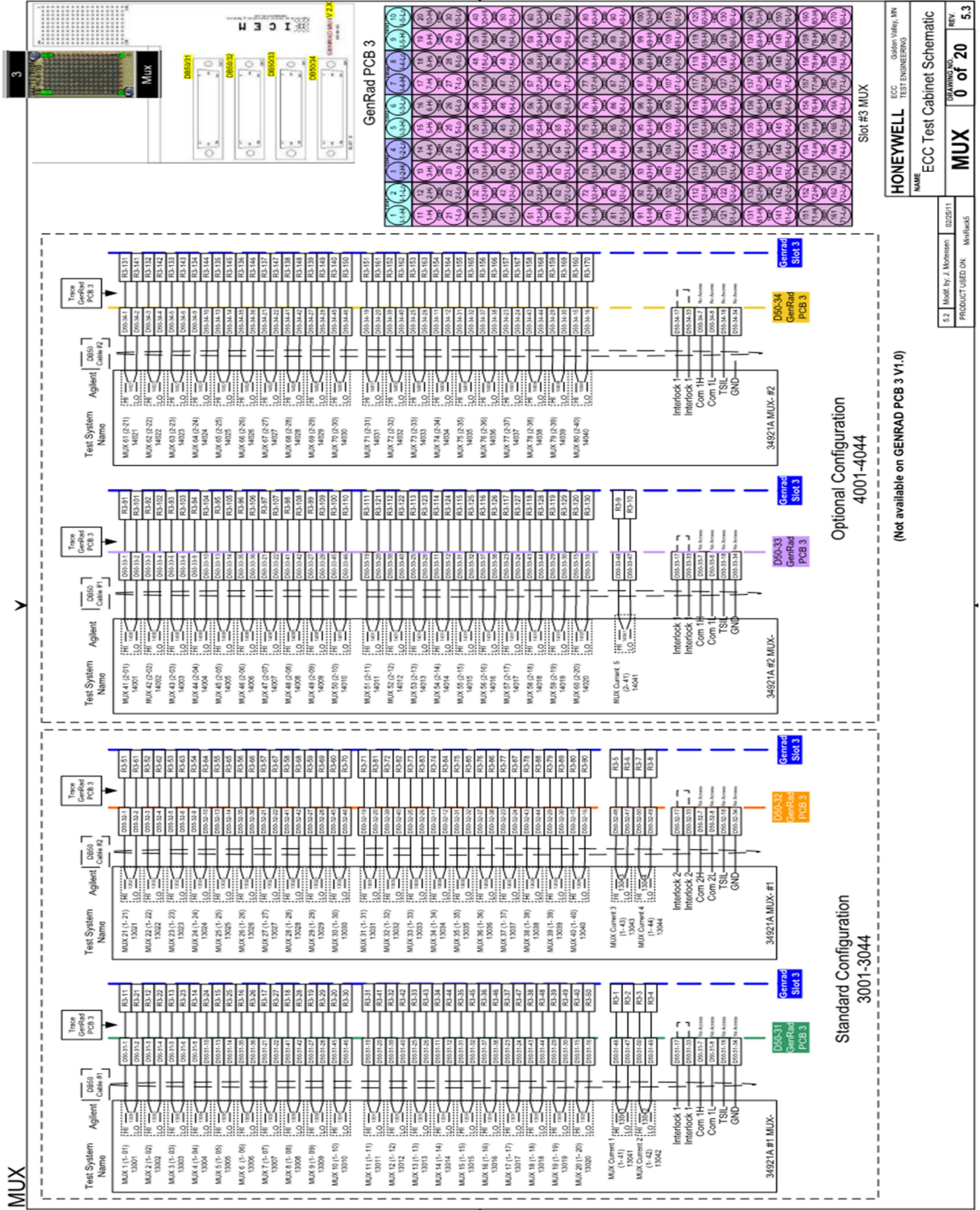
REV  
0 of 20

DATE  
08/29/11

PRODUCT USED OR  
MANUFACT

Standard Configuration  
11001-11032

Optional Configuration  
12001-12032  
(Not available on Mini Rack Relays V1.0 PWB)



MUX

Y

3

NAME: HONEYWELL ECC TEST ENGINEERING  
 G05en Valley, MN  
 ECC Test Cabinet Schematic  
 0 of 20 REV  
 5.1 Model by J. Moennesson  
 PRODUCT USED ON: MUX  
 PARTS:

Standard Configuration  
 3001-3044

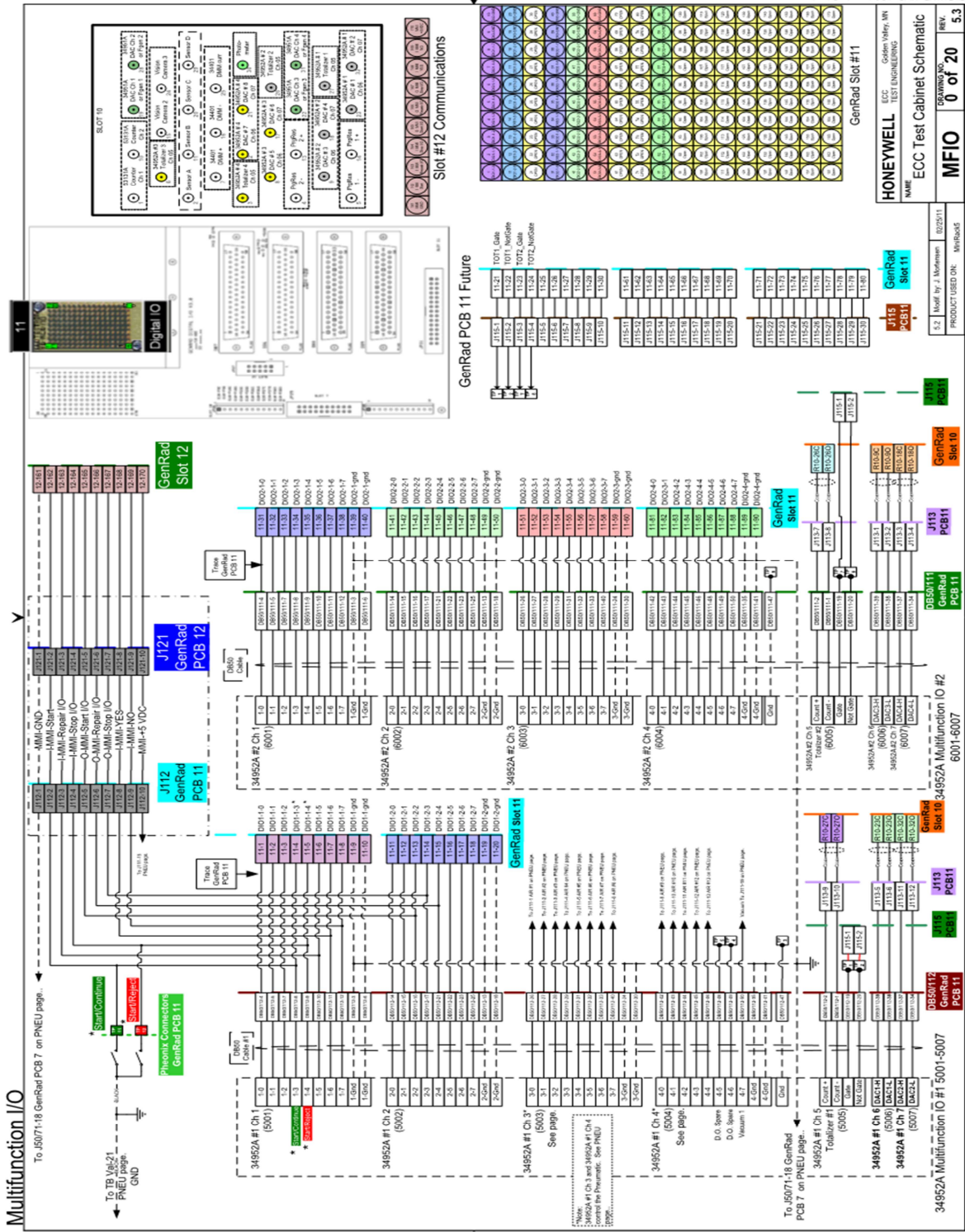
Optional Configuration  
 4001-4044

(Not available on GENRAD PCB 3 V1.0)

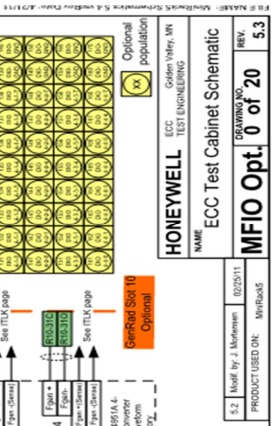
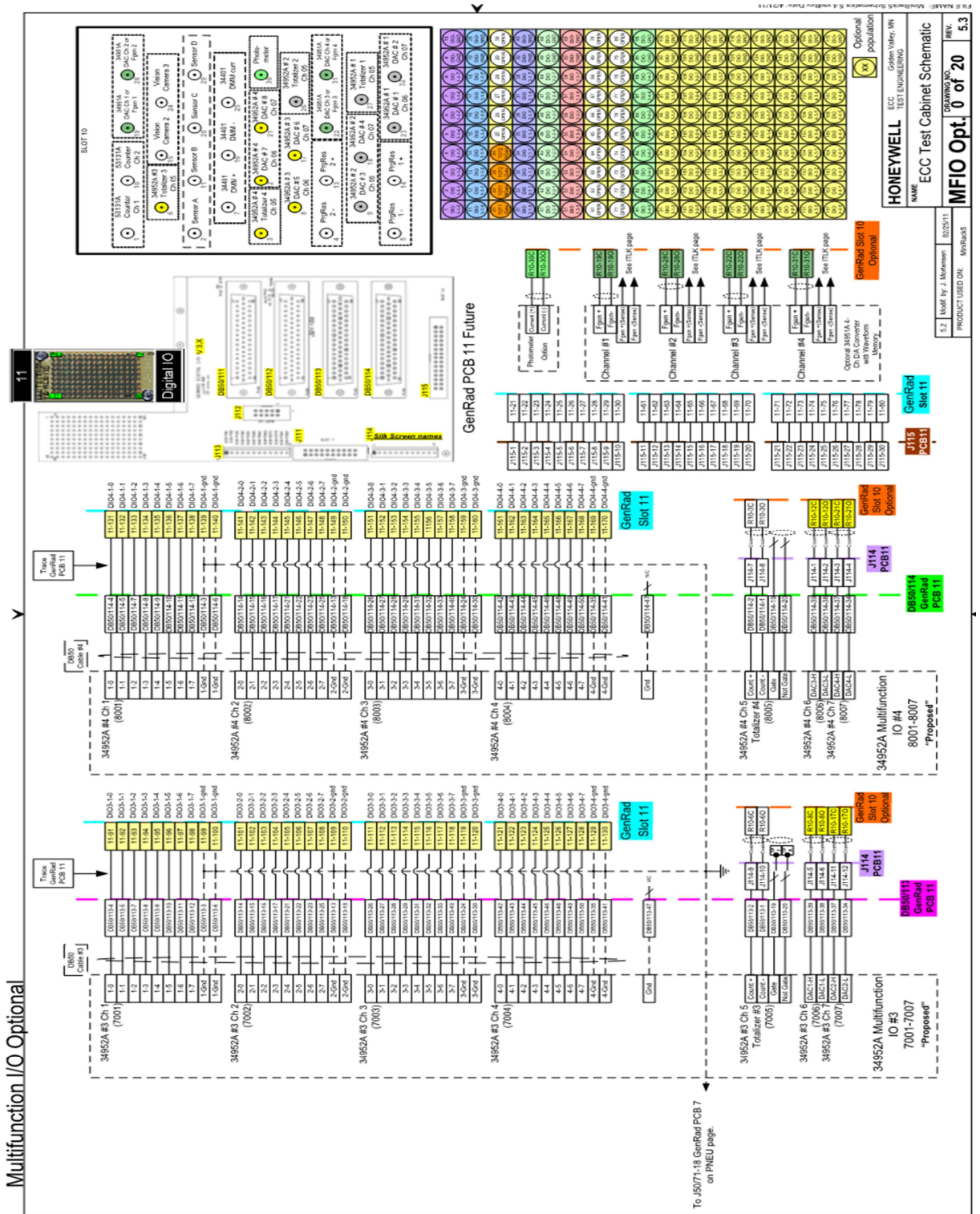
Slot #3 MUX

GENRAD PCB 3





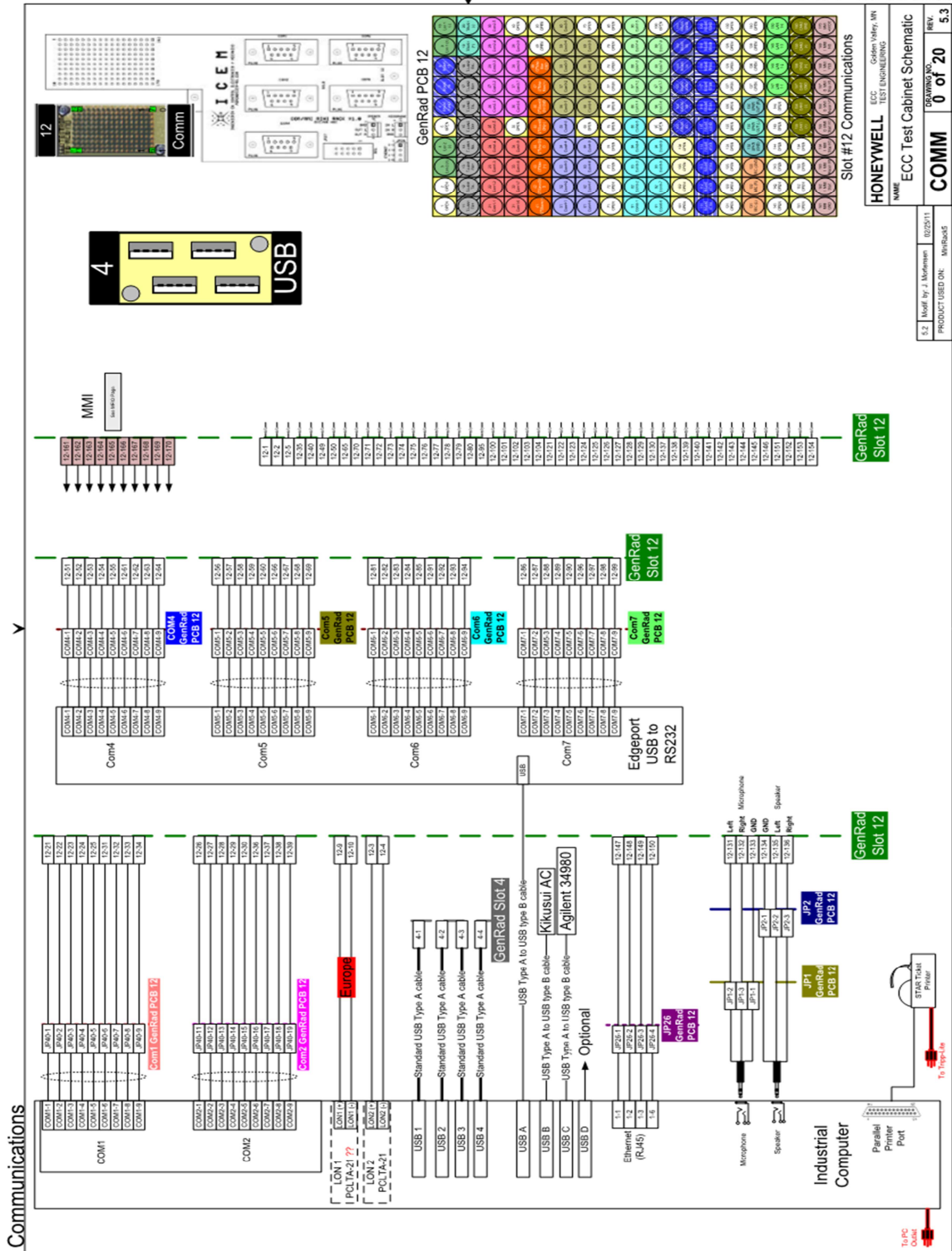
**GENRAD**  
 HONEYWELL  
 TEST ENGINEERING  
 ECC Test Cabinet Schematic  
 M/FIO 0 of 20  
 REV. 5.3  
 5.2 Modif by J. Normand 02/2011  
 PRODUCT USED ON: MFB36A5



**HONEYWELL**  
 ECC - Golden Valley, MN  
 NAME: **ECC Test Cabinet Schematic**  
 DRAWING NO. **0 of 20**  
 REV. **5.3**

5.2 | Modified by J. Mofmann | 02/25/11  
 PRODUCT USED ON: AIRPAC

**MPIO Opt.**



HONEYWELL ECC Golden Valley, MN  
TEST ENGINEERING

NAME: ECC Test Cabinet Schematic

DRAWING NO.: 0 of 20

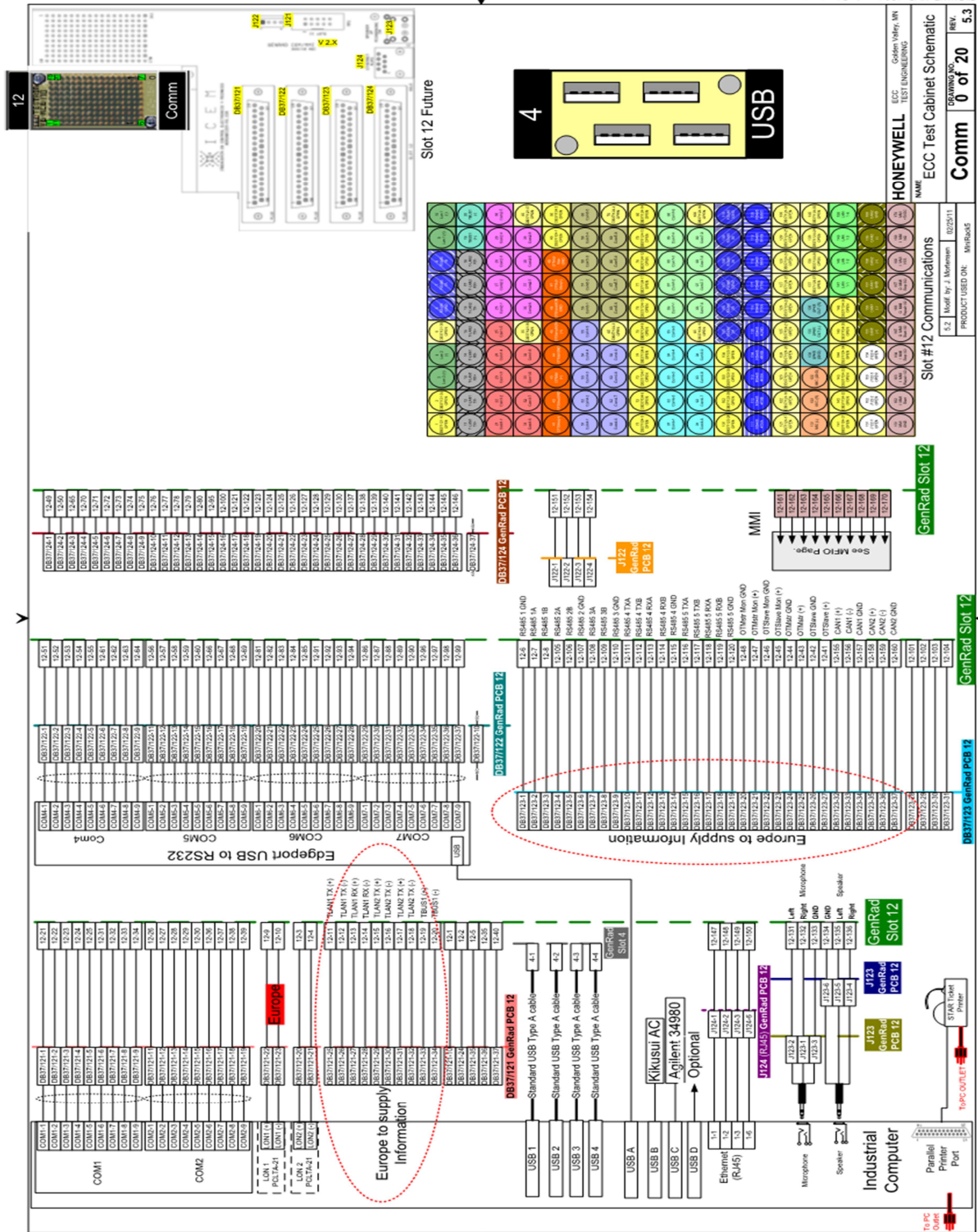
REV: 5.3

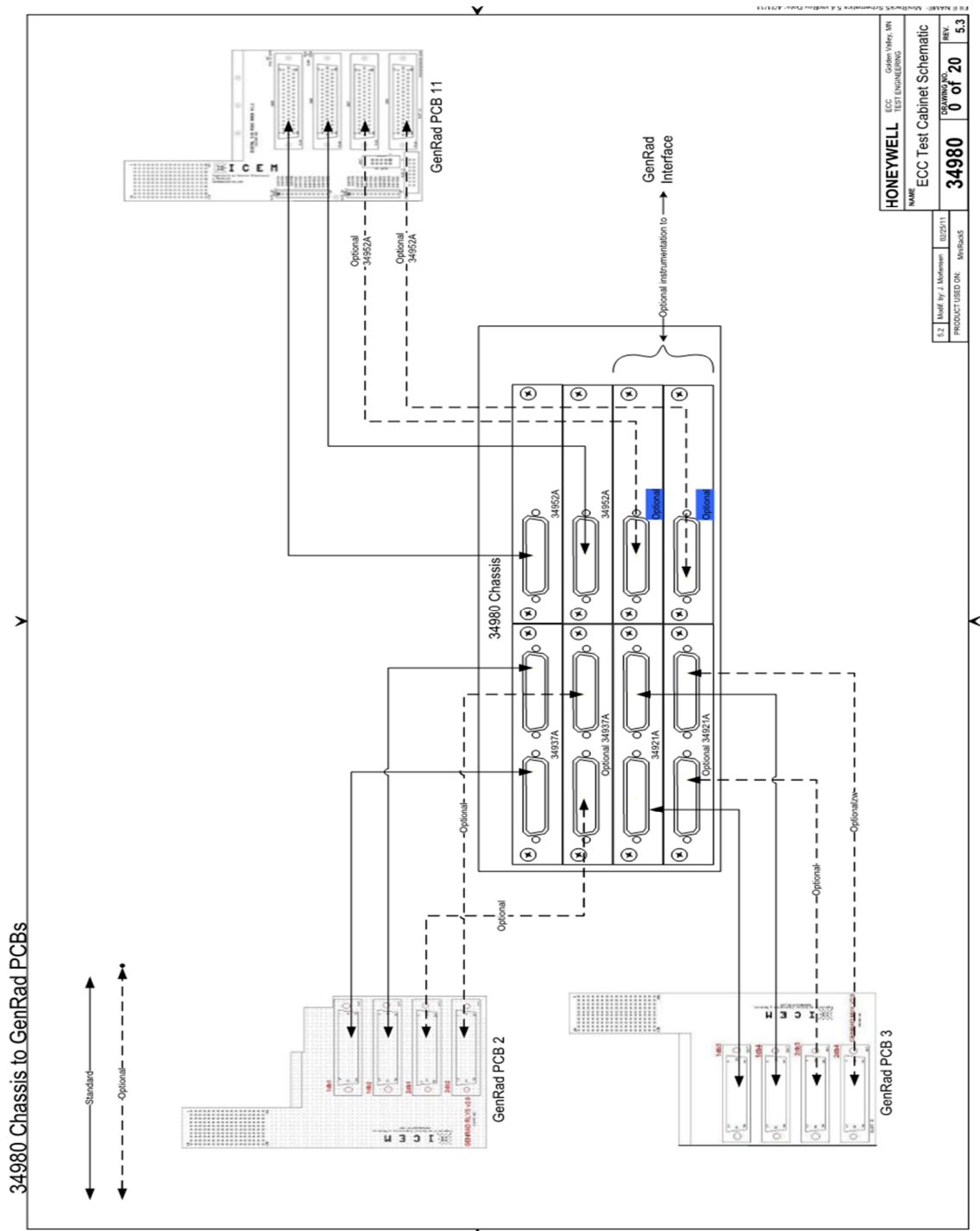
COMM

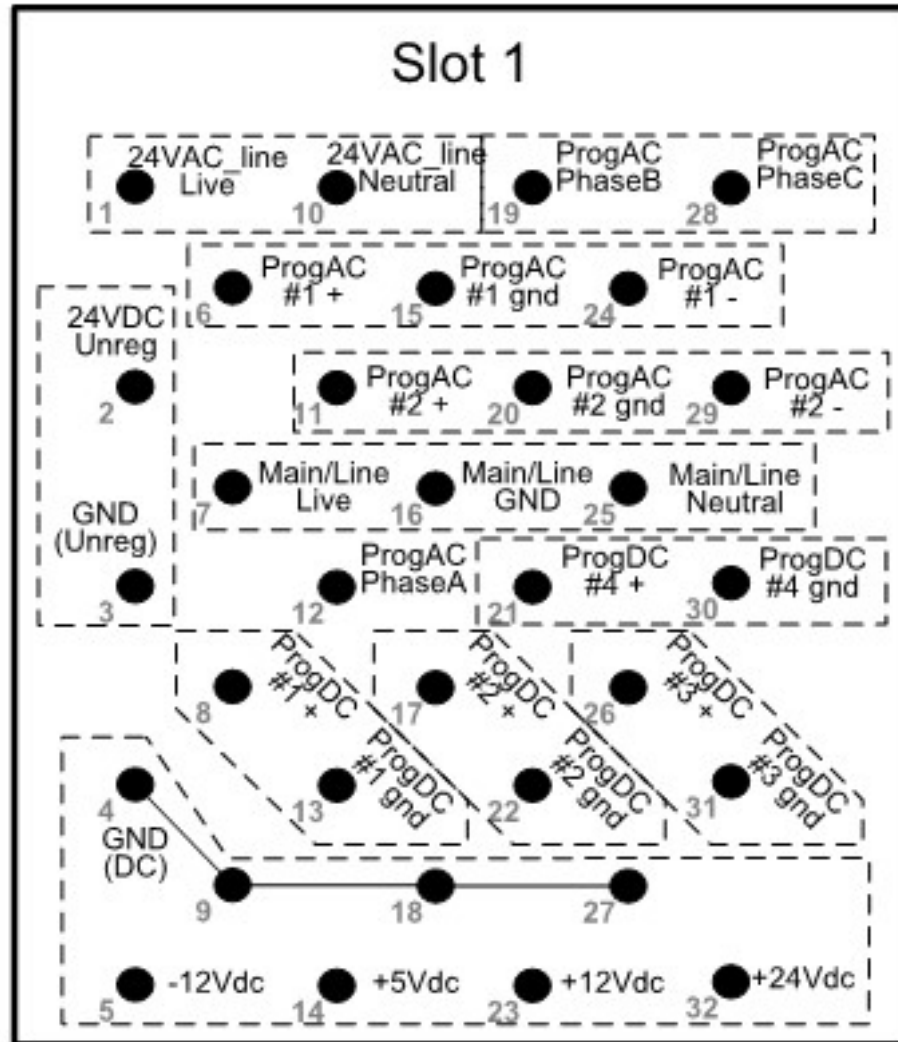
5.2 Modif by J. Matsumura 10/20/11

PRODUCT USED ON: Workbooks











ANEXO xv

13041		13042		13043		13044		14041	
1 I-1-Hi	2 I-1-Lo	3 I-2-Hi	4 I-2-Lo	5 I-3-Hi	6 I-3-Lo	7 I-4-Hi	8 I-4-Lo	9 I-5-Hi	10 I-5-Lo
11 1-1-Hi 13001	12 2-1-Hi 13002	13 3-1-Hi 13003	14 4-1-Hi 13004	15 5-1-Hi 13005	16 6-1-Hi 13006	17 7-1-Hi 13007	18 8-1-Hi 13008	19 9-1-Hi 13009	20 10-1-Hi 13010
21 1-Lo	22 2-Lo	23 3-Lo	24 4-Lo	25 5-Lo	26 6-Lo	27 7-Lo	28 8-Lo	29 9-Lo	30 10-Lo
31 11-Hi 13011	32 12-Hi 13012	33 13-Hi 13013	34 14-Hi 13014	35 15-Hi 13015	36 16-Hi 13016	37 17-Hi 13017	38 18-Lo 13018	39 19-Hi 13019	40 20-Hi 13020
41 11-Lo	42 12-Lo	43 13-Lo	44 14-Lo	45 15-Lo	46 16-Lo	47 17-Lo	48 18-Lo	49 19-Lo	50 20-Lo
51 21-Hi 13021	52 22-Hi 13022	53 23-Hi 13023	54 24-Hi 13024	55 25-Hi 13025	56 26-Hi 13026	57 27-Hi 13027	58 28-Hi 13028	59 29-Hi 13029	60 30-Hi 13030
61 21-Lo	62 22-Lo	63 23-Lo	64 24-Lo	65 25-Lo	66 26-Lo	67 27-Lo	68 28-Lo	69 29-Lo	70 30-Lo
71 31-Hi 13031	72 32-Hi 13032	73 33-Hi 13033	74 34-Hi 13034	75 35-Hi 13035	76 36-Hi 13036	77 37-Hi 13037	78 38-Hi 13038	79 39-Hi 13039	80 40-Hi 13040
81 31-Lo	82 32-Lo	83 33-Lo	84 34-Lo	85 35-Lo	86 36-Lo	87 37-Lo	88 38-Lo	89 39-Lo	90 40-Lo
91 41-Hi 14001	92 42-Hi 14002	93 43-Hi 14003	94 44-Hi 14004	95 45-Hi 14005	96 46-Hi 14006	97 47-Hi 14007	98 48-Hi 14008	99 49-Hi 14009	100 50-Hi 14010
101 41-Lo	102 42-Lo	103 43-Lo	104 44-Lo	105 45-Lo	106 46-Lo	107 47-Lo	108 48-Lo	109 49-Lo	110 50-Lo
111 51-Hi 14011	112 52-Hi 14012	113 53-Hi 14013	114 54-Hi 14014	115 55-Hi 14015	116 56-Hi 14016	117 57-Hi 14017	118 58-Hi 14018	119 59-Hi 14019	120 60-Hi 14020
121 51-Lo	122 52-Lo	123 53-Lo	124 54-Lo	125 55-Lo	126 56-Lo	127 57-Lo	128 58-Lo	129 59-Lo	130 60-Lo
131 61-Hi 14021	132 62-Hi 14022	133 63-Hi 14023	134 64-Hi 14024	135 65-Hi 14025	136 66-Hi 14026	137 67-Hi 14027	138 68-Hi 14028	139 69-Hi 14029	140 70-Hi 14030
141 61-Lo	142 62-Lo	143 63-Lo	144 64-Lo	145 65-Lo	146 66-Lo	147 67-Lo	148 68-Lo	149 69-Lo	150 70-Lo
151 71-Hi 14031	152 72-Hi 14032	153 73-Hi 14033	154 74-Hi 14034	155 75-Hi 14035	156 76-Hi 14036	157 77-Hi 14037	158 78-Hi 14038	159 79-Hi 14039	160 80-Hi 14040
161 71-Lo	162 72-Lo	163 73-Lo	164 74-Lo	165 75-Lo	166 76-Lo	167 77-Lo	168 78-Lo	169 79-Lo	170 80-Lo

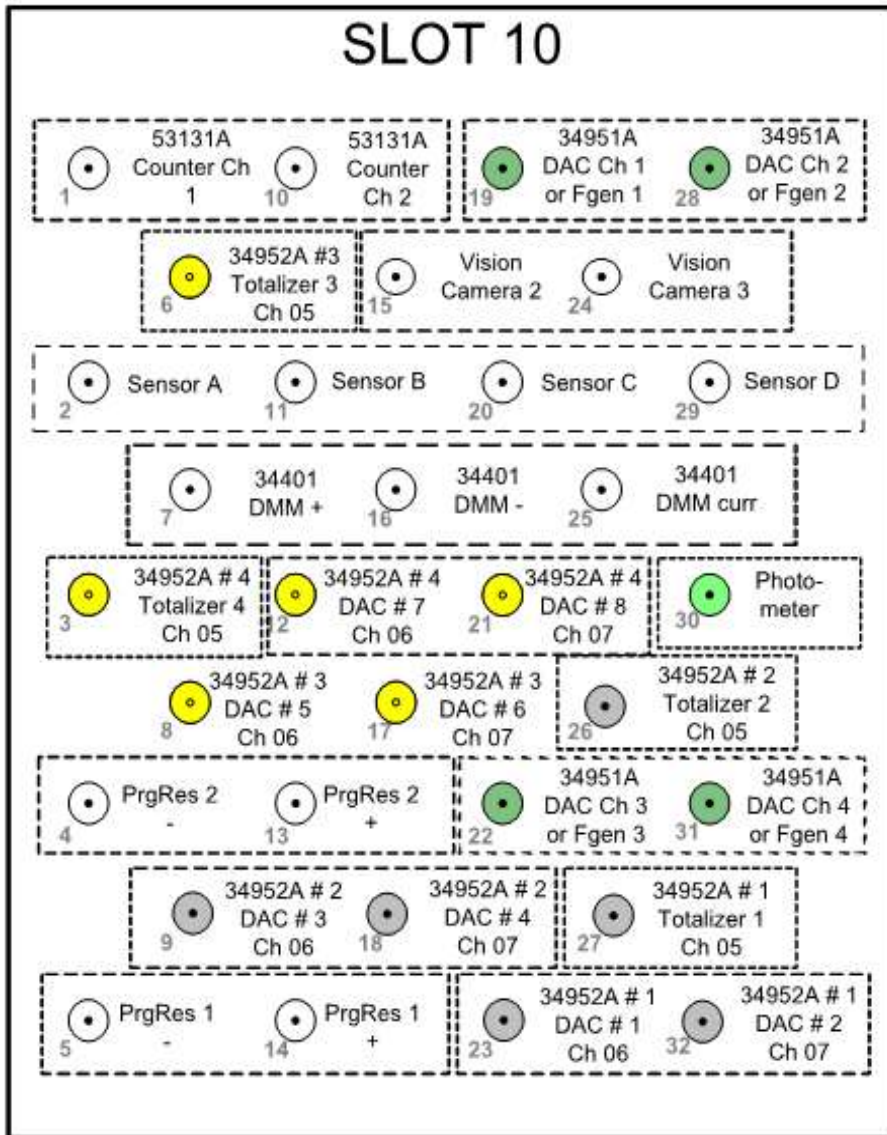





1 1-NC 11001	2 2-NC 11002	3 3-NC 11003	4 4-NC 11004	5 5-NC 11005	6 6-NC 11006	7 7-NC 11007	8 8-NC 11008	9 9-NC 11009	10 10-NC 11010
11 1-C 11011	12 2-C 11012	13 3-C 11013	14 4-C 11014	15 5-C 11015	16 6-C 11016	17 7-C 11017	18 8-C 11018	19 9-C 11019	20 10-C 11020
21 1-NO 11021	22 2-NO 11022	23 3-NO 11023	24 4-NO 11024	25 5-NO 11025	26 6-NO 11026	27 7-NO 11027	28 8-NO 11028	29 9-NO 11029	30 10-NO 11030
31 11-NC 11031	32 12-NC 11032	33 13-NC 11033	34 14-NC 11034	35 15-NC 11035	36 16-NC 11036	37 17-NC 11037	38 18-NC 11038	39 19-NC 11039	40 20-NC 11040
41 11-C 11041	42 12-C 11042	43 13-C 11043	44 14-C 11044	45 15-C 11045	46 16-C 11046	47 17-C 11047	48 18-C 11048	49 19-C 11049	50 20-C 11050
51 11-NO 11051	52 12-NO 11052	53 13-NO 11053	54 14-NO 11054	55 15-NO 11055	56 16-NO 11056	57 17-NO 11057	58 18-NO 11058	59 19-NO 11059	60 20-NO 11060
61 21-NC 11061	62 22-NC 11062	63 23-NC 11063	64 24-NC 11064	65 25-NC 11065	66 26-NC 11066	67 27-NC 11067	68 28-NC 11068	69 29-NC 11069	70 30-NC 11070
71 21-C 11071	72 22-C 11072	73 23-C 11073	74 24-C 11074	75 25-C 11075	76 26-C 11076	77 27-C 11077	78 28-C 11078	79 29-C 11079	80 30-C 11080
81 21-NO 11081	82 22-NO 11082	83 23-NO 11083	84 24-NO 11084	85 25-NO 11085	86 26-NO 11086	87 27-NO 11087	88 28-NO 11088	89 29-NO 11089	90 30-NO 11090
91 31-NC 11091	92 32-NC 11092	93 33-NC 11093	94 34-NC 11094	95 35-NC 11095	96 36-NC 11096	97 37-NC 11097	98 38-NC 11098	99 39-NC 11099	100 40-NC 11100
101 31-C 11101	102 32-C 11102	103 33-C 11103	104 34-C 11104	105 35-C 11105	106 36-C 11106	107 37-C 11107	108 38-C 11108	109 39-C 11109	110 40-C 11110
111 31-NO 11111	112 32-NO 11112	113 33-NO 11113	114 34-NO 11114	115 35-NO 11115	116 36-NO 11116	117 37-NO 11117	118 38-NO 11118	119 39-NO 11119	120 40-NO 11120
121 41-NC 11121	122 42-NC 11122	123 43-NC 11123	124 44-NC 11124	125 45-NC 11125	126 46-NC 11126	127 47-NC 11127	128 48-NC 11128	129 49-NC 11129	130 50-NC 11130
131 41-C 11131	132 42-C 11132	133 43-C 11133	134 44-C 11134	135 45-C 11135	136 46-C 11136	137 47-C 11137	138 48-C 11138	139 49-C 11139	140 50-C 11140
141 41-NO 11141	142 42-NO 11142	143 43-NO 11143	144 44-NO 11144	145 45-NO 11145	146 46-NO 11146	147 47-NO 11147	148 48-NO 11148	149 49-NO 11149	150 50-NO 11150
151 1-ST-C 11151	152 1-ST-C 11152	153 2-ST-C 11153	154 2-ST-C 11154	155 3-ST-C 11155	156 6-ST-C 11156	157 4-ST-C 11157	158 7-ST-C 11158	159 5-ST-C 11159	160 8-ST-C 11160
161 1-ST-NO 11161	162 1-ST-NO 11162	163 2-ST-NO 11163	164 2-ST-NO 11164	165 3-ST-NO 11165	166 6-ST-NO 11166	167 4-ST-NO 11167	168 7-ST-NO 11168	169 5-ST-NO 11169	170 8-ST-NO 11170



ANEXO xxixi

1 pAC#1 Sns (L)	2 pAC#2 Sns (L)	3 pDC#1 Sns (+)	4 pDC#2 Sns (+)	5 pDC#3 Sns (+)	6 pDC#4 Sns (+)	7 OPEN J50/71- 31	8 OPEN J50/71- 32	9 OPEN J50/71- 33	10 Air Pres. Sig.
11 pAC#1 Sns (N)	12 pAC#2 Sns (N)	13 pDC#1 Sns (-)	14 pDC#1 Sns (-)	15 pDC#1 Sns (-)	16 pDC#1 Sns (-)	17 OPEN J50/71- 37	18 OPEN J50/71- 38	19 OPEN J50/71- 39	20 Air Pres. Sig. GND
21 ITLK (S11)	22 ITLK (S12)	23 ITLK (S21)	24 ITLK (S22)	25 OPEN J50/72-8	26 OPEN J50/72-8	27 OPEN J50/72-7	28 OPEN J50/72-8	29 OPEN J50/72-8	30 OPEN J50/72- 10
31 AC1 ITLK(+)	32 AC2 ITLK(+)	33 DC ITLK (+)	34 PNEU ITLK(+)	35 VAC ITLK(+)	36 pDC ITLK(+)	37 OPEN N/C	38 OPEN N/C	39 +5VDC	40 Air Pres. Adj. 0- 5VDC
41 AC1 ITLK(-)	42 AC1 ITLK(-)	43 DC ITLK(-)	44 PNEU ITLK(-)	45 VAC ITLK(-)	46 pDC ITLK(+)	47 OPEN N/C	48 OPEN N/C	49 OPEN N/C	50 Air Pres. Adj. GND
51 Air 1 ITLK(+)	52 Air 2 ITLK(+)	53 Air 3 ITLK(+)	54 Air 4 ITLK(+)	55 Air 5 ITLK(+)	56 Air 6 ITLK(+)	57 Air 7 ITLK(+)	58 Air 8 ITLK(+)	59 Air 9 ITLK(+)	60 Air 10 ITLK(+)
61 Air 1 ITLK(-)	62 Air 2 ITLK(-)	63 Air 3 ITLK(-)	64 Air 4 ITLK(-)	65 Air 5 ITLK(-)	66 Air 6 ITLK(-)	67 Air 7 ITLK(-)	68 Air 8 ITLK(-)	69 Air 9 ITLK(-)	70 Air 10 ITLK(-)
71 Air 11 ITLK(+)	72 Air 12 ITLK(+)	73 Air 13 ITLK(+)	74 OPEN N/C	75 OPEN N/C	76 OPEN N/C	77 OPEN N/C	78 OPEN N/C	79 OPEN N/C	80 OPEN N/C
81 Air 11 ITLK(-)	82 Air 12 ITLK(-)	83 Air 13 ITLK(-)	84 OPEN J50/72- 11	85 OPEN J50/72- 12	86 OPEN J50/72- 13	87 OPEN J50/72- 14	88 OPEN J50/72- 15	89 OPEN J50/72- 16	90 OPEN J50/72- 17
91 OPEN J50/72- 18	92 OPEN J50/72- 19	93 OPEN J50/72- 20	94 OPEN J50/72- 21	95 OPEN J50/72- 22	96 OPEN J50/72- 23	97 OPEN J50/72- 24	98 OPEN J50/72- 25	99 OPEN J50/72- 26	100 OPEN J50/72- 27
101 ITLK Fgen#1 (+)	102 ITLK Fgen#2 (+)	103 ITLK Fgen#3 (+)	104 ITLK Fgen#4 (+)	105 OPEN J50/72- 32	106 OPEN J50/72- 33	107 OPEN J50/72- 34	108 OPEN J50/72- 35	109 OPEN J50/72- 36	110 OPEN J50/72- 37
111 ITLK Fgen#1 (-)	112 ITLK Fgen#2 (-)	113 ITLK Fgen#3 (-)	114 ITLK Fgen#4 (-)	115 OPEN J50/72- 42	116 OPEN J50/72- 43	117 OPEN J50/72- 44	118 OPEN J50/72- 45	119 OPEN J50/72- 46	120 OPEN J50/72- 47
121 OPEN J50/73-1	122 OPEN J50/73-2	123 OPEN J50/73-3	124 OPEN J50/73-4	125 OPEN J50/73-5	126 OPEN J50/73-6	127 OPEN J50/73-7	128 OPEN J50/73-8	129 OPEN J50/73-9	130 OPEN J50/73- 10
131 OPEN J50/73- 11	132 OPEN J50/73- 12	133 OPEN J50/73- 13	134 OPEN J50/73- 14	135 OPEN J50/73- 15	136 OPEN J50/73- 16	137 OPEN J50/73- 17	138 OPEN J50/73- 18	139 OPEN J50/73- 19	140 OPEN J50/73- 20
141 OPEN J50/73- 21	142 OPEN J50/73- 22	143 OPEN J50/73- 23	144 OPEN J50/73- 24	145 OPEN J50/73- 25	146 OPEN J50/73- 26	147 OPEN J50/73- 27	148 OPEN J50/73- 28	149 OPEN J50/73- 29	150 OPEN J50/73- 30
151 OPEN J50/73- 31	152 OPEN J50/73- 32	153 OPEN J50/73- 33	154 OPEN J50/73- 34	155 OPEN J50/73- 35	156 OPEN J50/73- 36	157 OPEN J50/73- 37	158 OPEN J50/73- 38	159 OPEN J50/73- 39	160 OPEN J50/73- 40
161 OPEN J50/73- 41	162 OPEN J50/73- 42	163 OPEN J50/73- 43	164 OPEN J50/73- 44	165 OPEN J50/73- 45	166 OPEN J50/73- 46	167 OPEN J50/73- 47	168 OPEN J50/73- 48	169 OPEN J50/73- 49	170 OPEN J50/73- 50



-  Required on the std configuration
-  Optional instrumentation
-  Optional Population



ANEXO xix

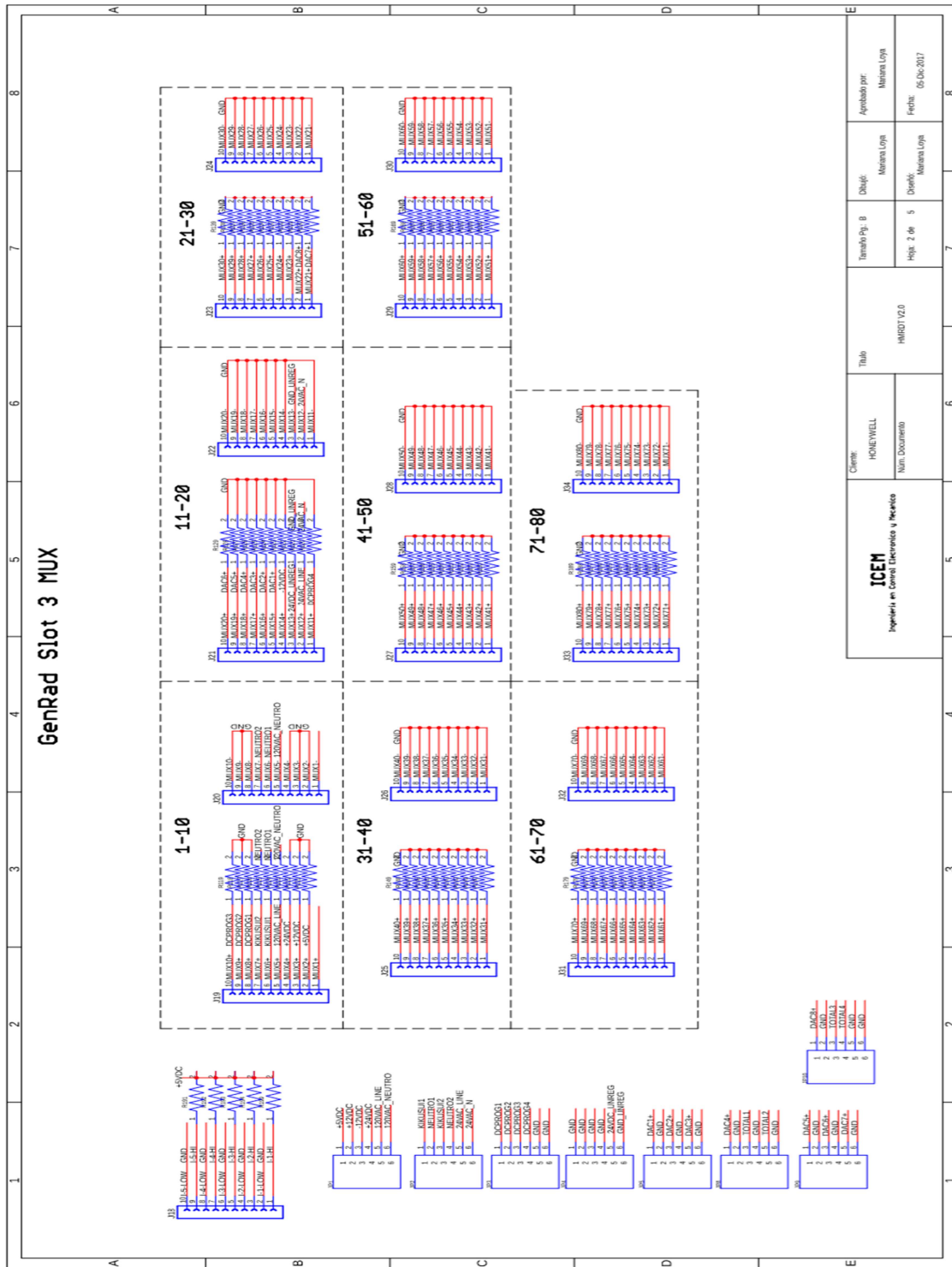
DIO Optional Configuration

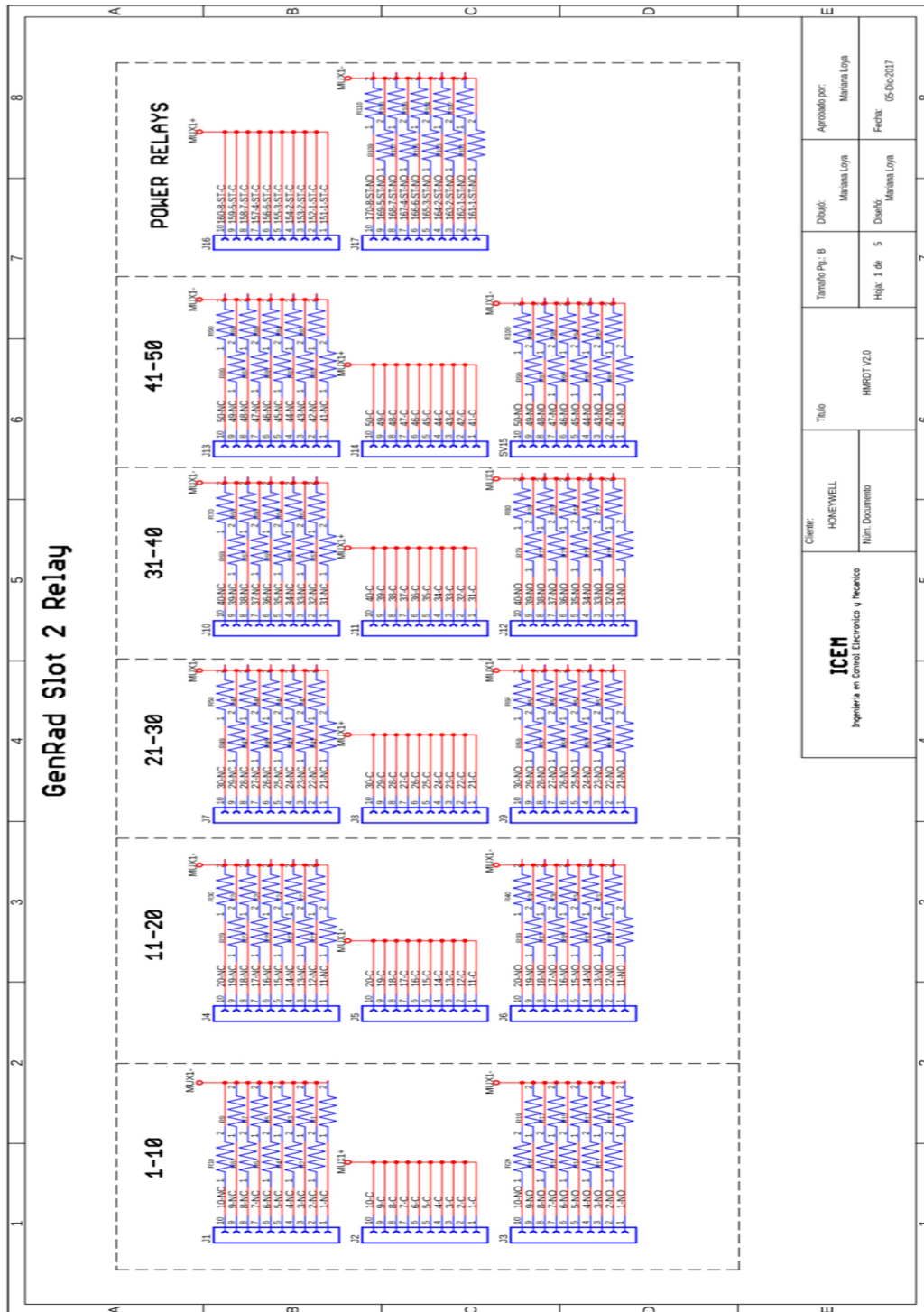
1 DIO 1-1-0	2 DIO 1-1-1	3 DIO 1-1-2	4 DIO 1-1-3	5 DIO 1-1-4	6 DIO 1-1-5	7 DIO 1-1-6	8 DIO 1-1-7	9 DIO- GND	10 DIO- GND
11 DIO 1-1-0	12 DIO 1-1-1	13 DIO 1-1-2	14 DIO 1-1-3	15 DIO 1-1-4	16 DIO 1-1-5	17 DIO 1-1-6	18 DIO 1-1-7	19 DIO- GND	20 DIO- GND
21 TOT1 Gate	22 TOT1 NGate	23 TOT2 Gate	24 TOT2 NGate	25 OPEN	26 OPEN	27 OPEN	28 OPEN	29 OPEN	30 OPEN
31 DIO 2-1-0	32 DIO 2-1-1	33 DIO 2-1-2	34 DIO 2-1-3	35 DIO 2-1-4	36 DIO 2-1-5	37 DIO 2-1-6	38 DIO 2-1-7	39 DIO- GND	40 DIO- GND
41 DIO 2-2-0	42 DIO 2-2-1	43 DIO 2-2-2	44 DIO 2-2-3	45 DIO 2-2-4	46 DIO 2-2-5	47 DIO 2-2-6	48 DIO 2-2-7	49 DIO- GND	50 DIO- GND
51 DIO 2-3-0	52 DIO 2-3-1	53 DIO 2-3-2	54 DIO 2-3-3	55 DIO 2-3-4	56 DIO 2-3-5	57 DIO 2-3-6	58 DIO 2-3-7	59 DIO- GND	60 DIO- GND
61 OPEN	62 OPEN	63 OPEN	64 OPEN	65 OPEN	66 OPEN	67 OPEN	68 OPEN	69 OPEN	70 OPEN
71 OPEN	72 OPEN	73 OPEN	74 OPEN	75 OPEN	76 OPEN	77 OPEN	78 OPEN	79 OPEN	80 OPEN
81 DIO 2-4-0	82 DIO 2-4-1	83 DIO 2-4-2	84 DIO 2-4-3	85 DIO 2-4-4	86 DIO 2-4-5	87 DIO 2-4-6	88 DIO 2-4-7	89 DIO- GND	90 DIO- GND
91 DIO 3-1-0	92 DIO 3-1-1	93 DIO 3-1-2	94 DIO 3-1-3	95 DIO 3-1-4	96 DIO 3-1-5	97 DIO 3-1-6	98 DIO 3-1-7	99 DIO- GND	100 DIO- GND
101 DIO 3-2-0	102 DIO 3-2-1	103 DIO 3-2-2	104 DIO 3-2-3	105 DIO 3-2-4	106 DIO 3-2-5	107 DIO 3-2-6	108 DIO 3-2-7	109 DIO- GND	110 DIO- GND
111 DIO 3-3-1	112 DIO 3-3-1	113 DIO 3-3-2	114 DIO 3-3-3	115 DIO 3-3-4	116 DIO 3-3-5	117 DIO 3-3-6	118 DIO 3-3-7	119 DIO- GND	120 DIO- GND
121 DIO 3-4-0	122 DIO 3-4-1	123 DIO 3-4-2	124 DIO 3-4-3	125 DIO 3-4-4	126 DIO 3-4-5	127 DIO 3-4-6	128 DIO 3-4-7	129 DIO- GND	130 DIO- GND
131 DIO 4-1-0	132 DIO 4-1-1	133 DIO 4-1-2	134 DIO 4-1-3	135 DIO 4-1-4	136 DIO 4-1-5	137 DIO 4-1-6	138 DIO 4-1-7	139 DIO- GND	140 DIO- GND
141 DIO 4-2-0	142 DIO 4-2-1	143 DIO 4-2-2	144 DIO 4-2-3	145 DIO 4-2-4	146 DIO 4-2-5	147 DIO 4-2-6	148 DIO 4-2-7	149 DIO- GND	150 DIO- GND
151 DIO 4-3-0	152 DIO 4-3-1	153 DIO 4-3-2	154 DIO 4-3-3	155 DIO 4-3-4	156 DIO 4-3-5	157 DIO 4-3-6	158 DIO 4-3-7	159 DIO- GND	160 DIO- GND
161 DIO 4-4-0	162 DIO 4-4-1	163 DIO 4-4-2	164 DIO 4-4-3	165 DIO 4-4-4	166 DIO 4-4-5	167 DIO 4-4-6	168 DIO 4-4-7	169 DIO- GND	170 DIO- GND



ANEXO xx

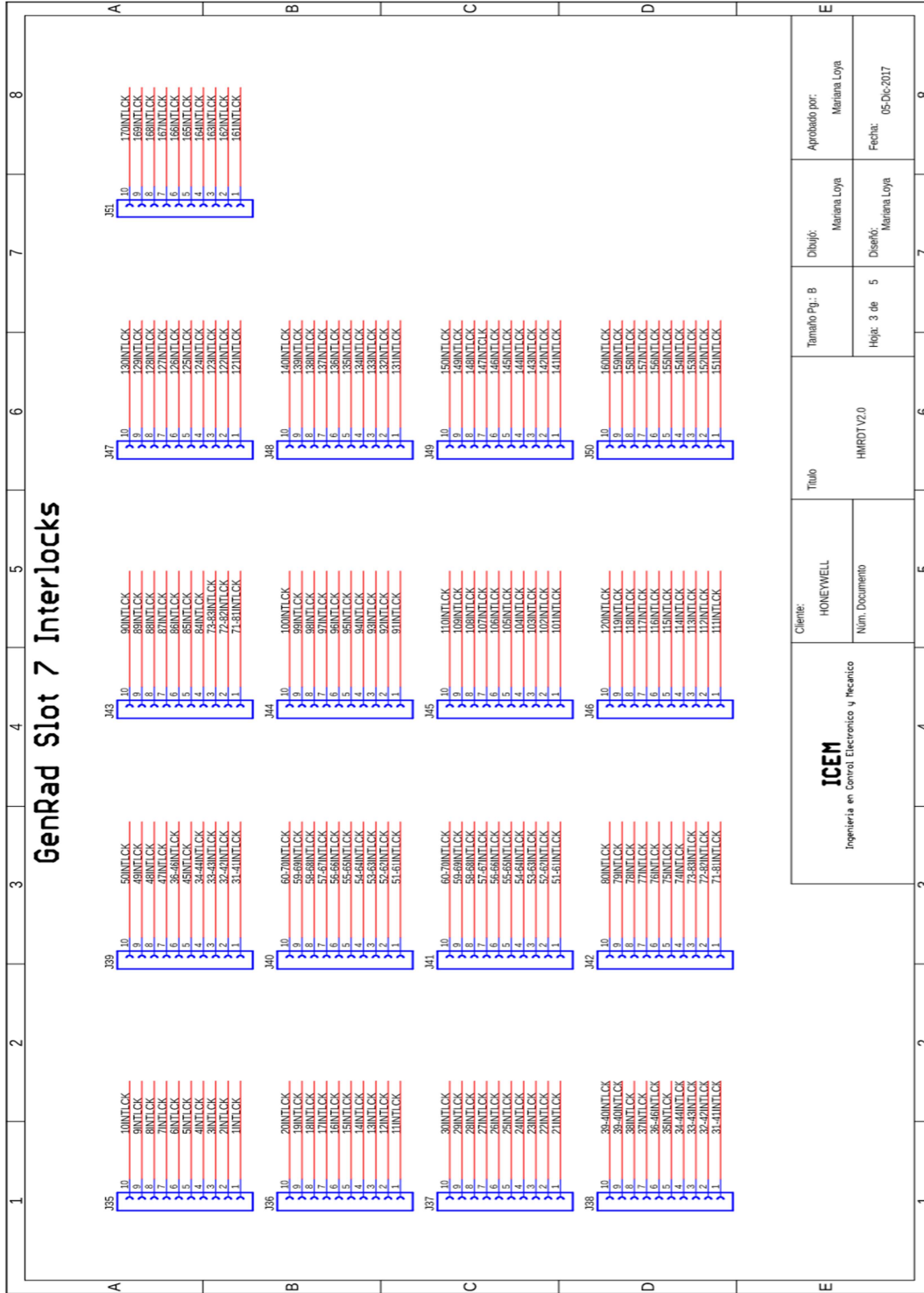
1 JP40-10 OPEN	2 JP40-24 OPEN	3 Lon 2 (+)	4 Lon 2 (-)	5 JP40-35 OPEN	6 RS485 1GND	7 RS485 1A	8 RS485 1B	9 Lon 1 (+)	10 Lon 1 (-)
11 T-LAN1 TX+	12 T-LAN1 TX-	13 T-LAN1 RX+	14 T-LAN1 RX-	15 T-LAN2 TX+	16 T-LAN2 TX-	17 T-LAN2 RX+	18 T-LAN2 RX-	19 TBUS1 (+)	20 TBUS1 (-)
21 Com1-1	22 Com1-2	23 Com1-3	24 Com1-4	25 Com1-5	26 Com2-1	27 Com2-2	28 Com2-3	29 Com2-4	30 Com2-5
31 Com1-6	32 Com1-7	33 Com1-8	34 Com1-9	35 JP40-36 OPEN	36 Com2-6	37 Com2-7	38 Com2-8	39 Com2-9	40 JP40-37 OPEN
41 OTSlave (+)	42 OTSlave GND	43 OTMstr (+)	44 OTMstr GND	45 OTSlave MON(+)	46 OTSlave MON GND	47 OTMstr Mon(+)	48 OTMstr Mon GND	49 JP43-1 OPEN	50 JP43-2 OPEN
51 Com4-1	52 Com4-2	53 Com4-3	54 Com4-4	55 Com4-5	56 Com5-1	57 Com5-2	58 Com5-3	59 Com5-4	60 Com5-5
61 Com4-6	62 Com4-7	63 Com4-8	64 Com4-9	65 JP43-3 OPEN	66 Com5-6	67 Com5-7	68 Com5-8	69 Com5-9	70 JP43-4 OPEN
71 JP43-5 OPEN	72 JP43-6 OPEN	73 JP43-7 OPEN	74 JP43-8 OPEN	75 JP43-9 OPEN	76 JP43-10 OPEN	77 JP43-11 OPEN	78 JP43-12 OPEN	79 JP43-13 OPEN	80 JP43-14 OPEN
81 Com6-1	82 Com6-2	83 Com6-3	84 Com6-4	85 Com6-5	86 Com7-1	87 Com7-2	88 Com7-3	89 Com7-4	90 Com7-5
91 Com6-6	92 Com6-7	93 Com6-8	94 Com6-9	95 JP43-15 OPEN	96 Com7-6	97 Com7-7	98 Com7-8	99 Com7-9	100 JP43-16 OPEN
101 JP42-28 OPEN	102 JP42-29 OPEN	103 JP42-30 OPEN	104 JP42-31 OPEN	105 RS485 2A	106 RS485 2B	107 RS485 2GND	108 RS485 3A	109 RS485 3B	110 RS485 3GND
111 RS485 4TXA	112 RS485 4TXB	113 RS485 4RXA	114 RS485 4RXB	115 RS485 4GND	116 RS485 5TXA	117 RS485 5TXB	118 RS485 5RXA	119 RS485 5RXB	120 RS485 5GND
121 JP43-17 OPEN	122 JP43-18 OPEN	123 JP43-19 OPEN	124 JP43-20 OPEN	125 JP43-21 OPEN	126 JP43-22 OPEN	127 JP43-23 OPEN	128 JP43-24 OPEN	129 JP43-25 OPEN	130 JP43-26 OPEN
131 MIC (L)	132 MIC (R)	133 MIC (GND)	134 SPKR (GND)	135 SPKR OUT (L)	136 SPKR OUT (R)	137 JP43-27 OPEN	138 JP43-28 OPEN	139 JP43-29 OPEN	140 JP43-30 OPEN
141 JP43-31 OPEN	142 JP43-32 OPEN	143 JP43-33 OPEN	144 JP43-34 OPEN	145 JP43-35 OPEN	146 JP43-36 OPEN	147 LAN 1-1	148 LAN 1-2	149 LAN 1-3	150 LAN 1-6
151 JP4-1 OPEN	152 JP44-2 OPEN	153 JP44-3 OPEN	154 JP44-4 OPEN	155 CAN1 (+)	156 CAN1 (-)	157 CAN1 GND	158 CAN2 (+)	159 CAN2 (-)	160 CAN2 GND
161 MMI GND	162 I_MMI Start	163 L_MMI Repeat I/O	164 I_MMI Stop I/O	165 O_MMI Start I/O	166 O_MMI Repeat I/O	167 O_MMI Stop I/O	168 I_MMI YES	169 I_MMI NO	170 MMI +5VDC

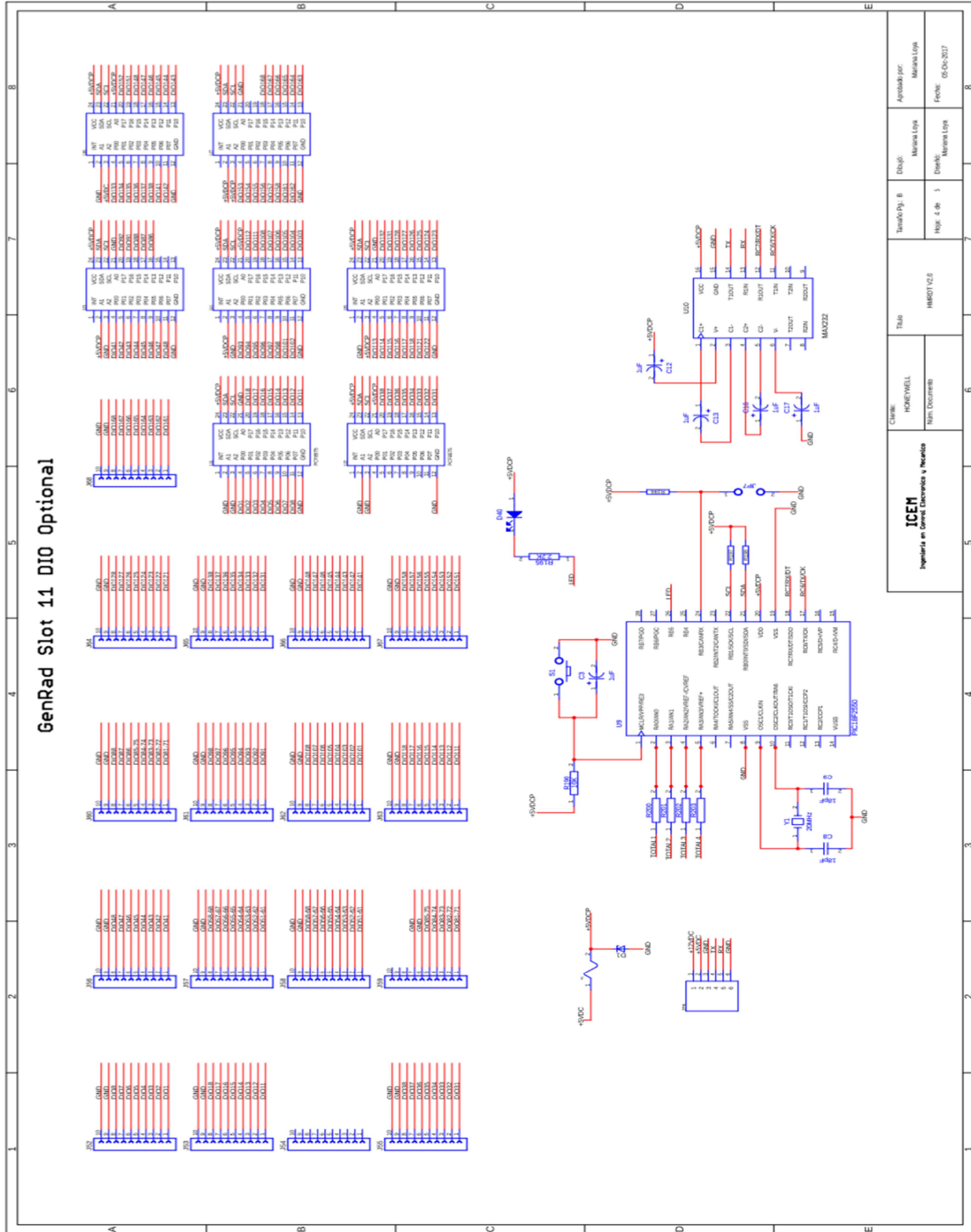




<b>ICEM</b> Ingeniería en Control Electrónico y Mecánico	Cientista:	HONEYWELL	Título HMROT V2.0	Aprobado por: Mariana Loya
	Num. Documento			
			Hoja: 1 de 5	Fecha: 05-Dic-2017

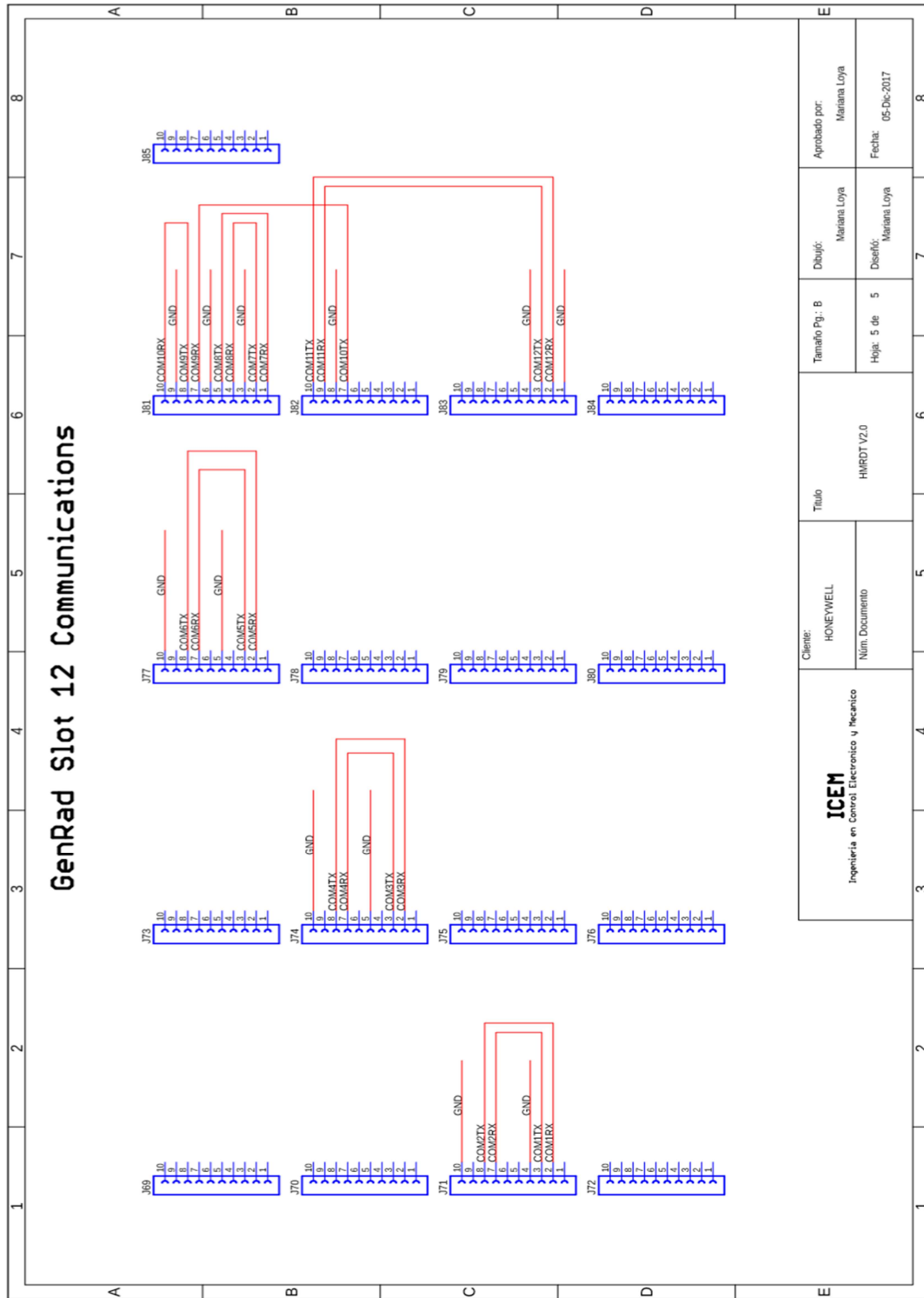


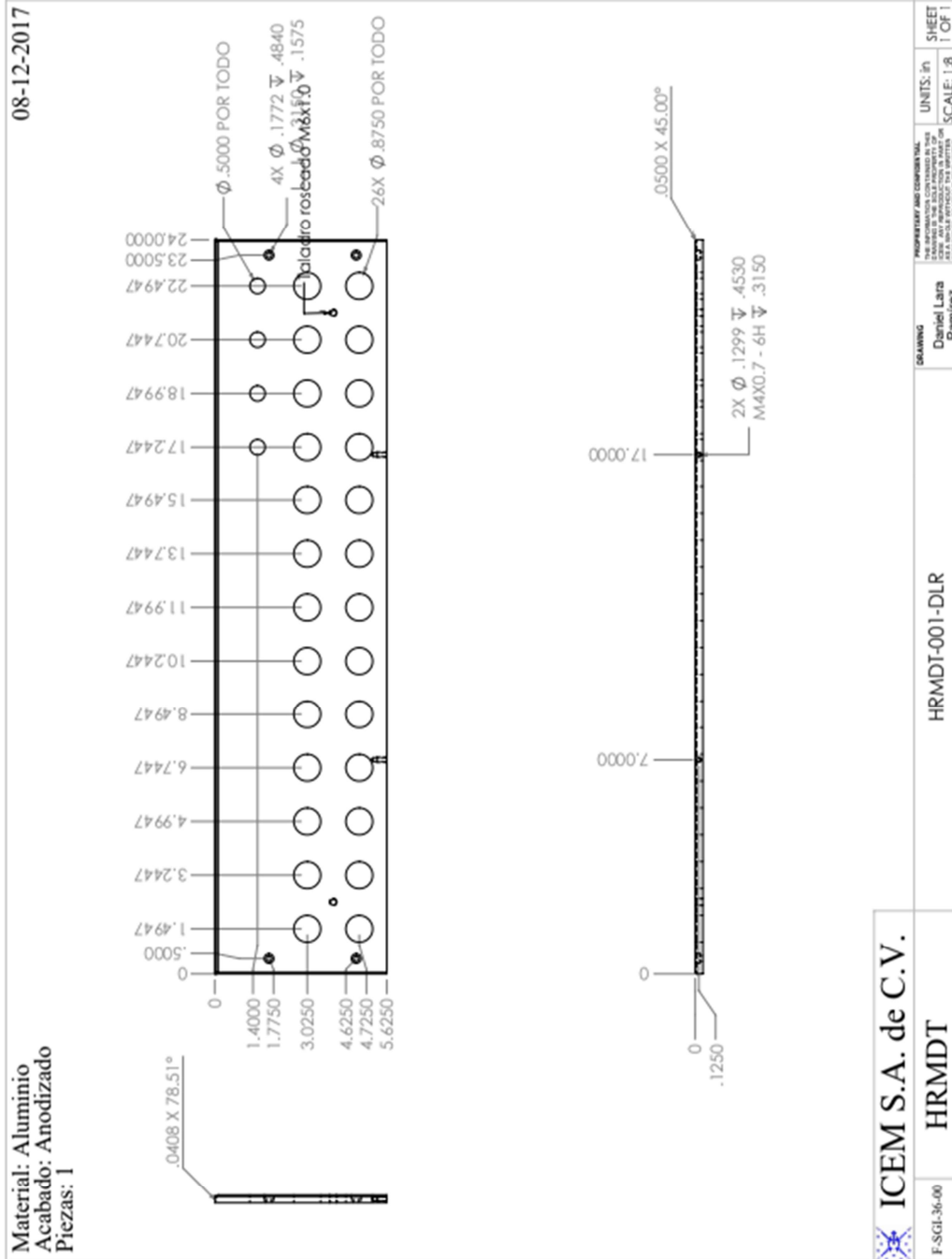


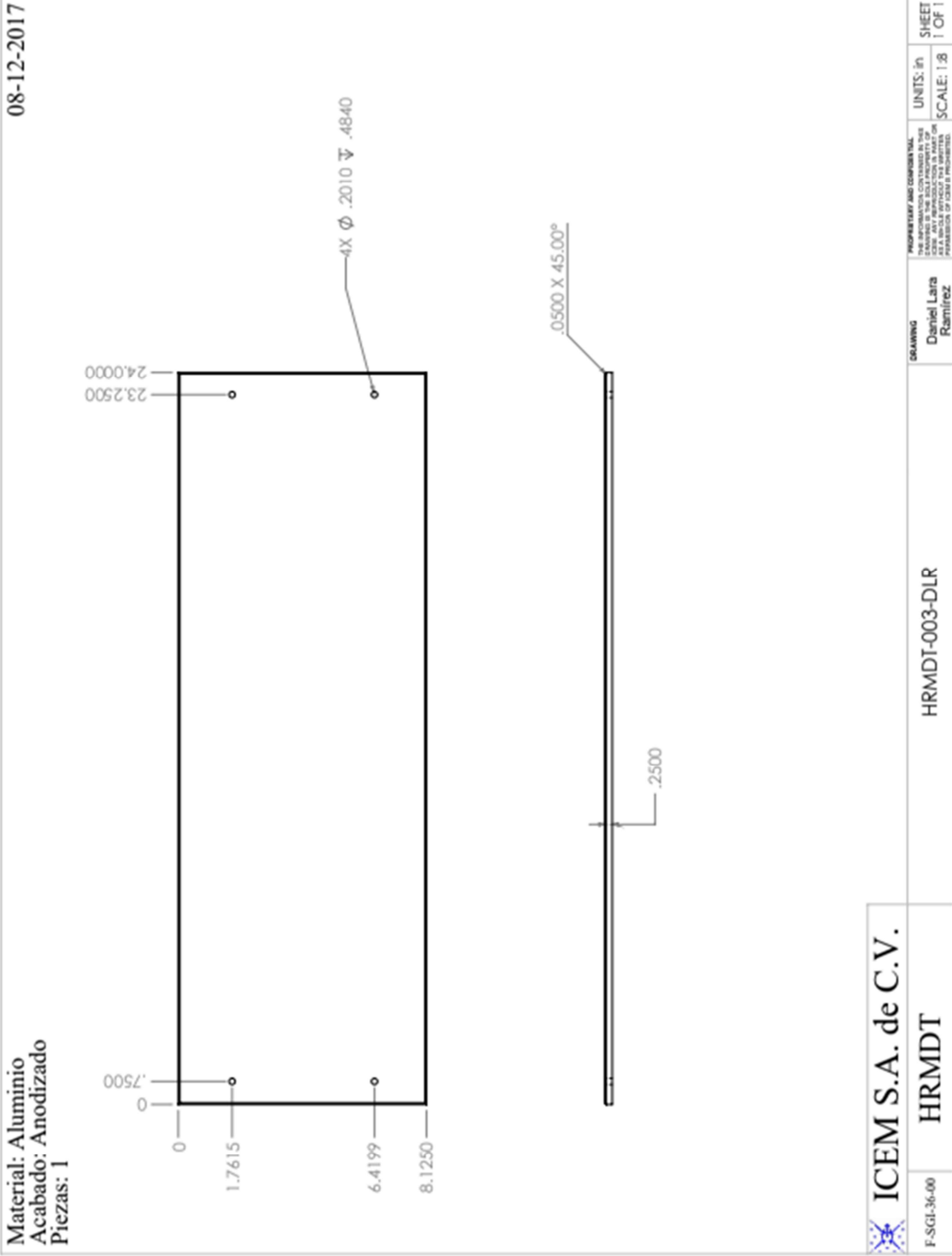


<b>ICEM</b> Ingeniería en Control Electrónico e Informática	Curso:	NOISEHELL	Título:	INM07023	Trabajo Práctico:	B	Diseño:	Marcelo Loyola	Aprobado por:	Marcelo Loyola
		Tema Documento:					Hoja:	4 de 1	Día:	05/02/2017









**ICEM S.A. de C.V.**

F-SGL-36-00

HRMDT

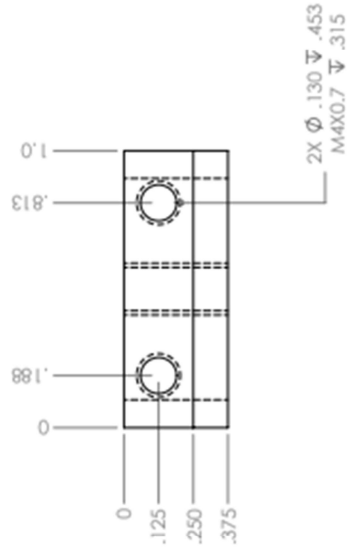
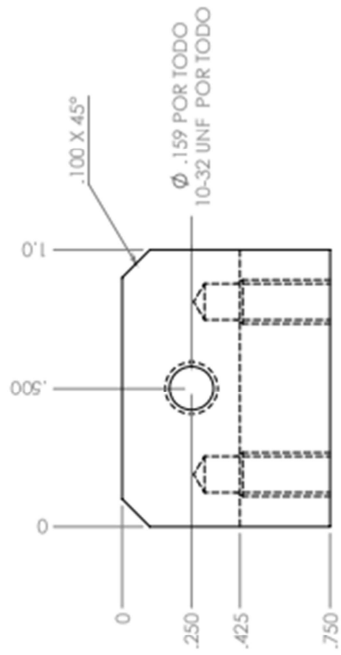
HRMDT-003-DLR

Daniel Lara Ramirez

UNITS: in  
 SCALE: 1:8  
 SHEET 1 OF 1

08-12-2017

Material: Aluminio  
 Acabado: Anodizado  
 Piezas: 4



 ICEM S.A. de C.V.

HRMDT

HRMDT-005-DLR

REV A

DESIGNER  
 Sergio Ricardo  
 Porfirio Carlos

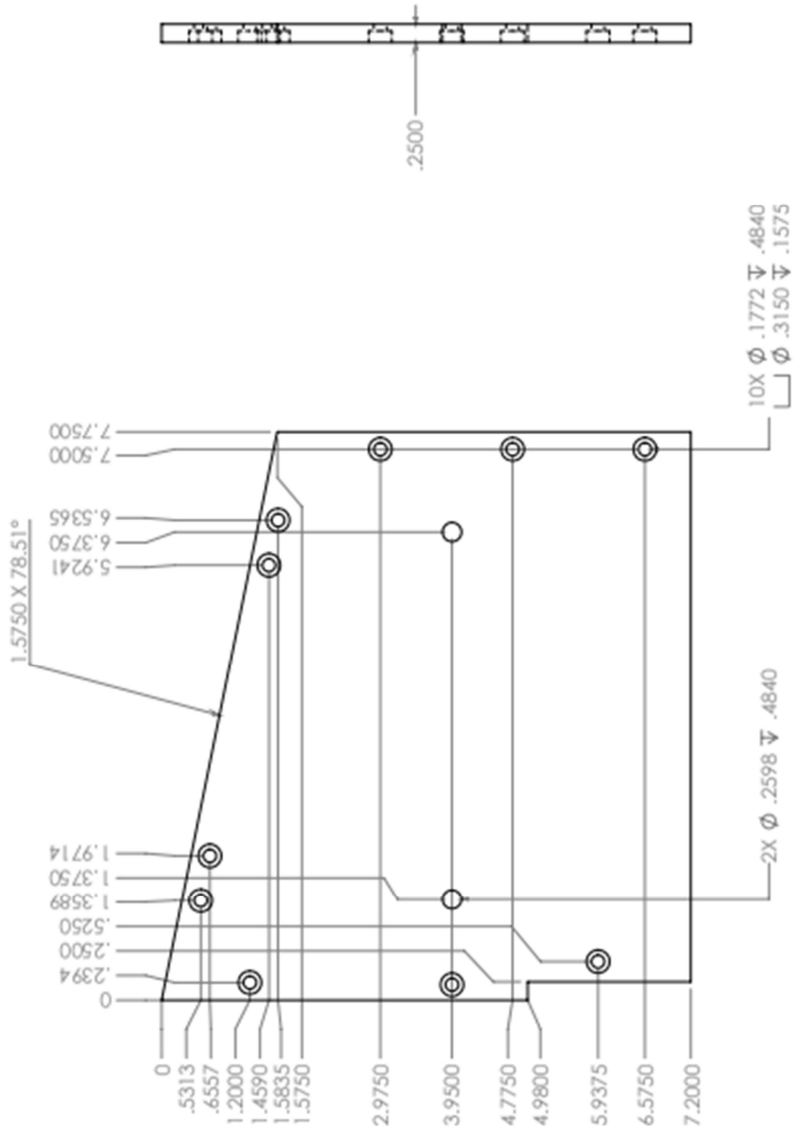
AUTHORIZE  
 Luis Antonio  
 Montoya Jaquez

PROPRIETARY AND CONFIDENTIAL  
 THIS DRAWING IS THE PROPERTY OF ICEM S.A. ALL RIGHTS ARE RESERVED. REPRODUCTION OR TRANSMISSION OF THIS DRAWING WITHOUT THE WRITTEN PERMISSION OF ICEM IS PROHIBITED.

UNITS: in  
 SCALE: 2:1  
 SHEET  
 1 OF 1

08-12-201

Material: Aluminio  
 Acabado: Anodizado  
 Piezas: 1

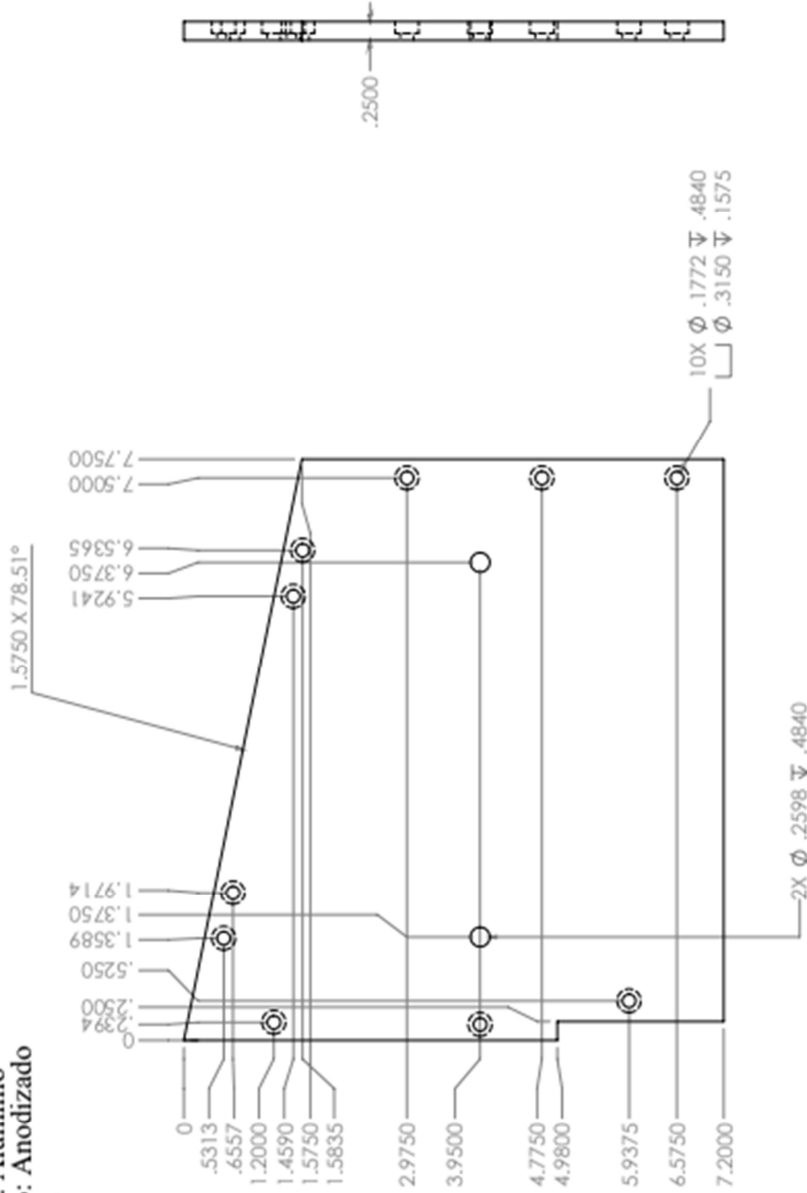


**ICEM S.A. de C.V.**  
 F-SGI-36-00  
**HRMDT**

HRMDT-002-DLR  
 Daniel Lara Ramirez  
 UNITS: in  
 SCALE: 1:2  
 SHEET 1 OF 1

08-12-2017

Material: Aluminio  
 Acabado: Anodizado  
 Piezas: 1



UNITS: in  
 SCALE: 1:2

DRAWING  
 Daniel Lara  
 Ramirez

PROPRIETARY AND CONFIDENTIAL  
 DRAWING IS THE SOLE PROPERTY OF  
 ICHEM S.A. ANY REPRODUCTION OR  
 ALTERATION OF THIS DRAWING OR  
 PORTION OF THIS IS PROHIBITED.

HRMDT-006-DLR

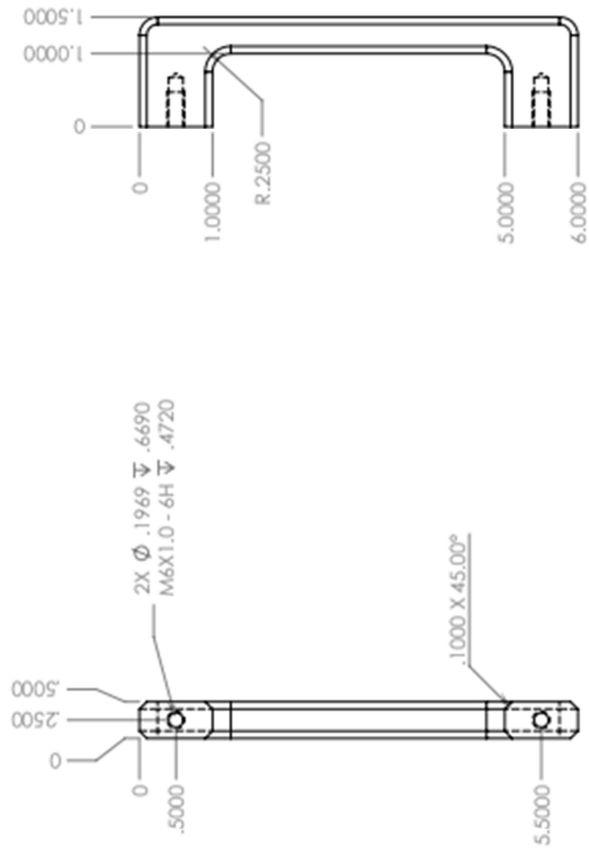
HRMDT

ICEM S.A. de C.V.  
 F-SGI-36-00

SHEET  
 1 OF 1

08-12-2017

Material: Aluminio  
 Acabado: Anodizado  
 Piezas: 2



**ICEM S.A. de C.V.**  
 F-SGI-36-00 HRMDT

HRMDT-007-DLR

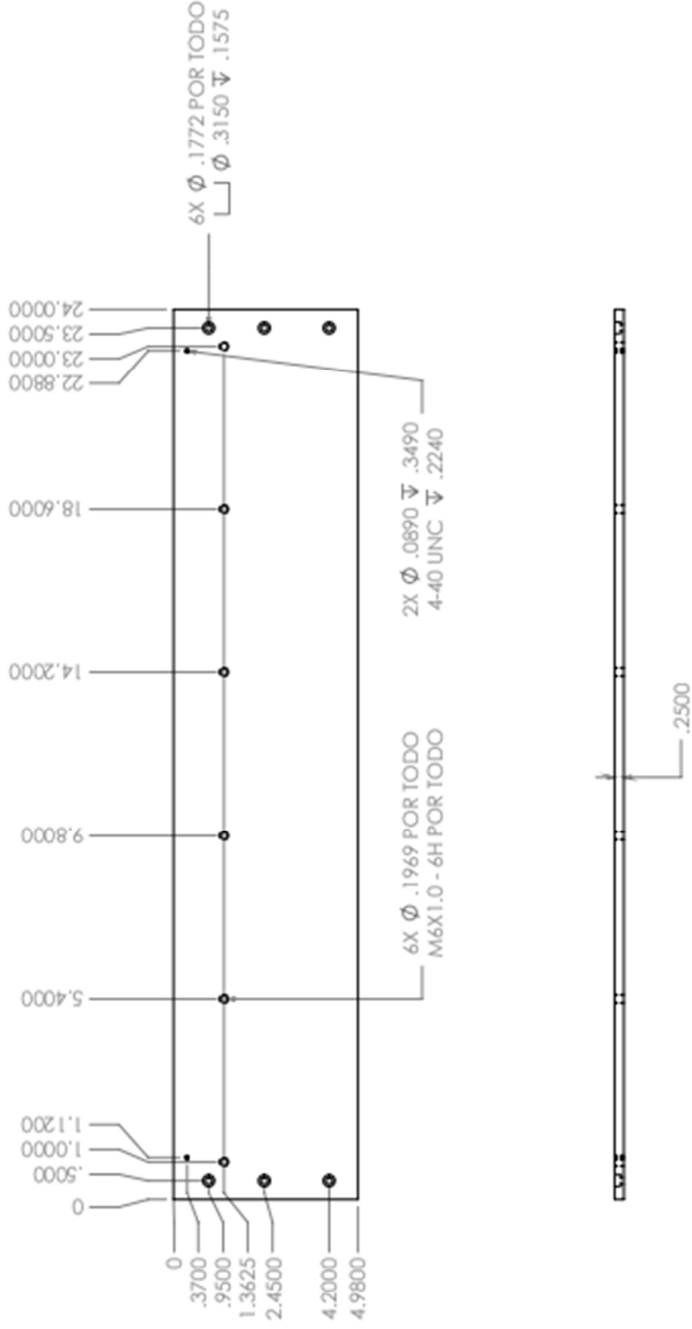
DRAWING  
 Daniel Lara

PROPRIETARY AND CONFIDENTIAL  
 THIS DRAWING IS THE PROPERTY OF  
 ICHEM S.A. DE C.V. NO PART OF THIS  
 DRAWING IS TO BE REPRODUCED OR  
 TRANSMITTED IN ANY FORM OR BY  
 ANY MEANS, ELECTRONIC OR MECHANICAL,  
 INCLUDING PHOTOCOPYING, RECORDING,  
 OR BY ANY INFORMATION STORAGE AND  
 RETRIEVAL SYSTEM, WITHOUT THE  
 PERMISSION OF ICHEM S.A. DE C.V.

UNITS: IN  
 SCALE: 1:2  
 SHEET  
 1 OF 1

08-12-2017

Material: Aluminio  
 Acabado: Anodizado  
 Piezas: 1



**ICEM S.A. de C.V.**  
 F-SGI-36-00

HRMDT

HRMDI-009-DLR

DRAWING  
 Daniel Lara  
 Ramirez

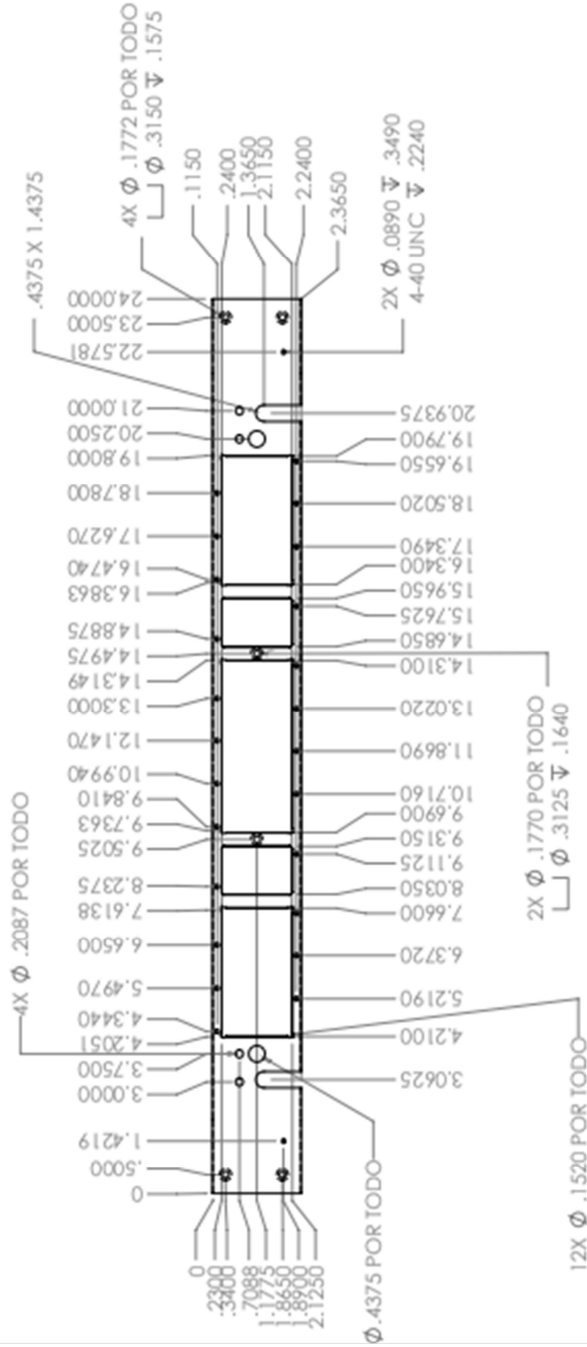
PROPERTY AND COPYRIGHT  
 THE INFORMATION CONTAINED IN THIS  
 DRAWING IS THE PROPERTY OF ICEM S.A.  
 ANY REPRODUCTION IN WHOLE OR  
 PART WITHOUT THE WRITTEN  
 PERMISSION OF ICEM IS PROHIBITED.

UNITS: in  
 SCALE: 1:3  
 SHEET  
 1 OF



08-12-2017

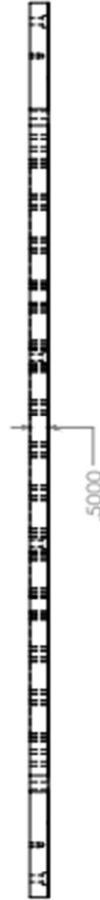
Material: Cold Rolled  
Acabado: Zincado  
Piezas: 1



CHAFLAN EN LOS CORTES

.0100 X 45.00°

.0100 X 45.00°



ICEM S.A. de C.V.  
F-SGI-36-00

HRMDT

HRMDT-011-DLR

DRAWING  
Daniel Laja  
Ramirez

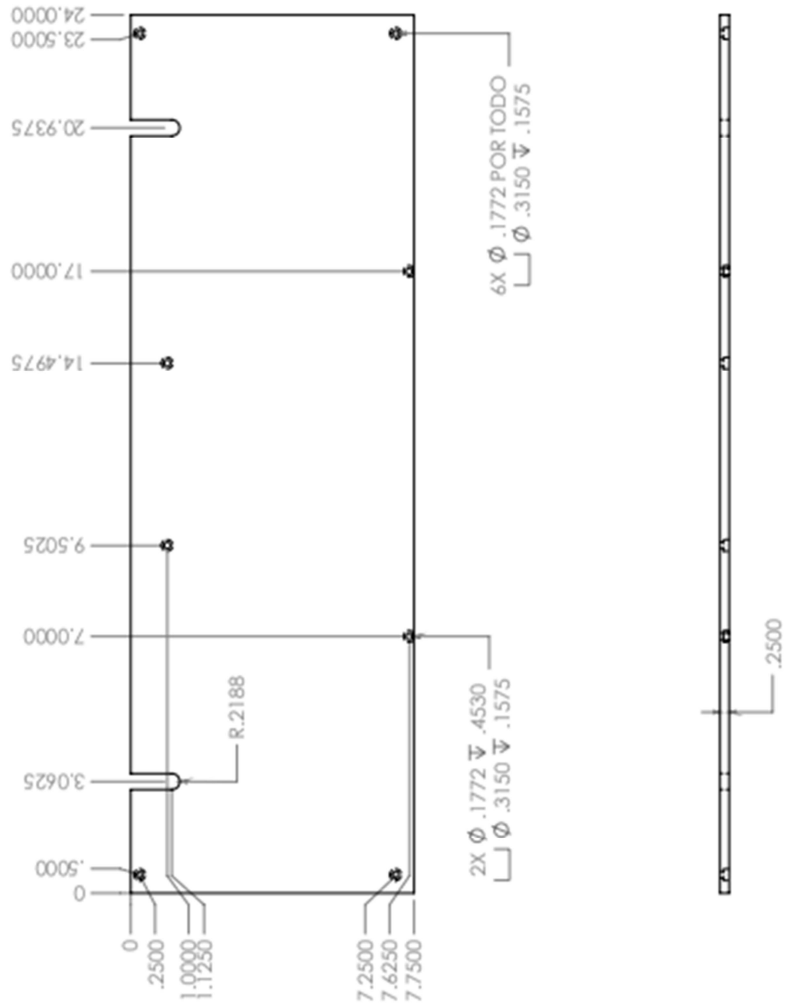
PROPRIETARY AND CONFIDENTIAL  
DRAWING IS THE SOLE PROPERTY OF  
ICEM S.A. DE C.V. ALL RIGHTS RESERVED.  
REPRODUCTION OF THIS IS PROHIBITED.

UNITS: in  
SCALE: 1:8

SHEET  
1 OF 1

08-12-2017

Material: Aluminio  
 Acabado: Anodizado  
 Piezas: 1



**ICEM S.A. de C.V.**  
 F-SGI-36-00

HRMDT

HRMDT-013-DUR

UNITS: in  
 SCALE: 1:8

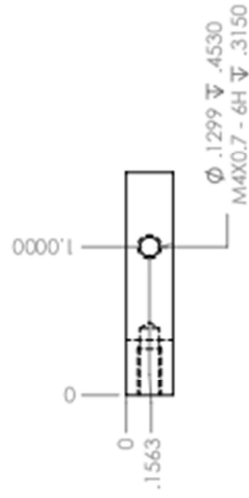
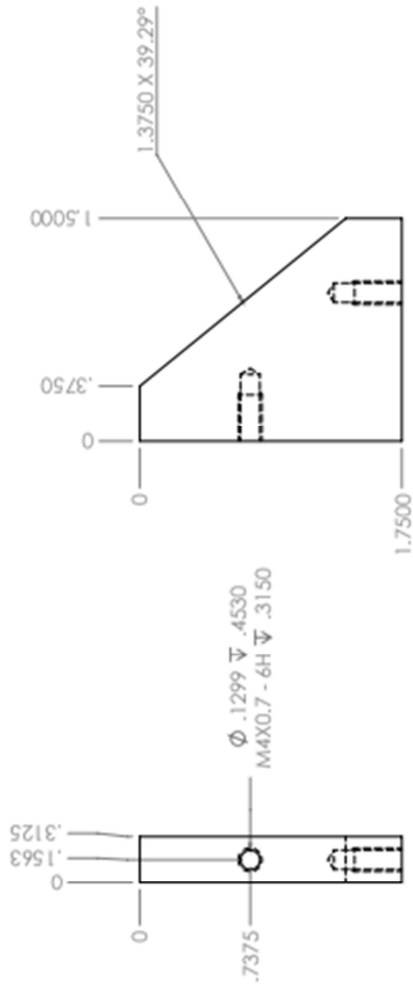
SHEET  
 1 OF 1

PROPERTY AND CONFIDENTIALITY  
 THIS DRAWING IS THE SOLE PROPERTY OF  
 ICEM S.A. DE C.V. NO TO BE REPRODUCED  
 OR COPIED IN ANY MANNER WITHOUT THE WRITTEN  
 PERMISSION OF ICEM S.A. DE C.V.

DRAWING  
 Daniel Lara  
 Ramirez

11-12-2017

Material: Aluminio  
 Acabado: Anodizado  
 Piezas: 2



ICEM S.A. de C.V.

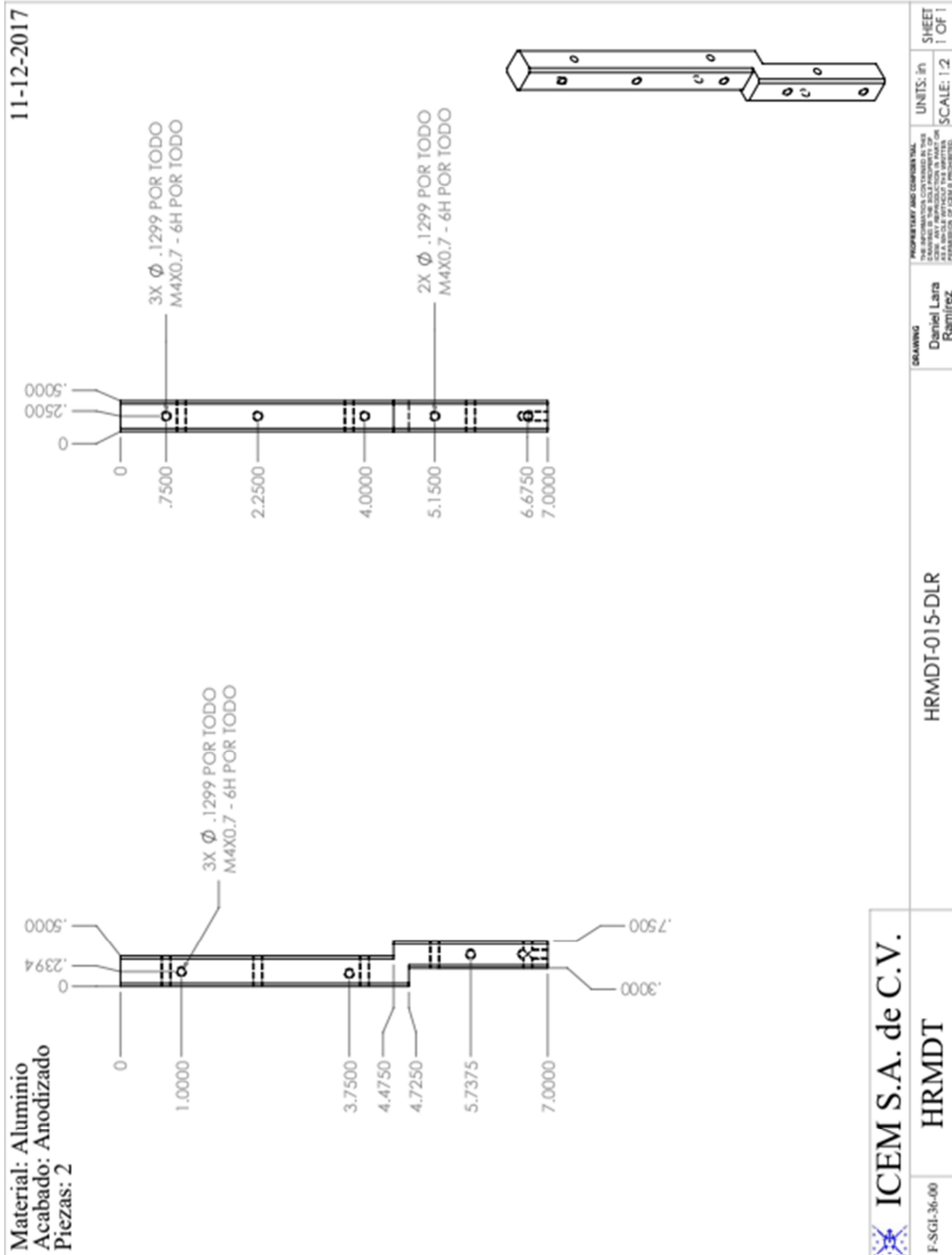
F-SGI-36-00  
 HRMDT

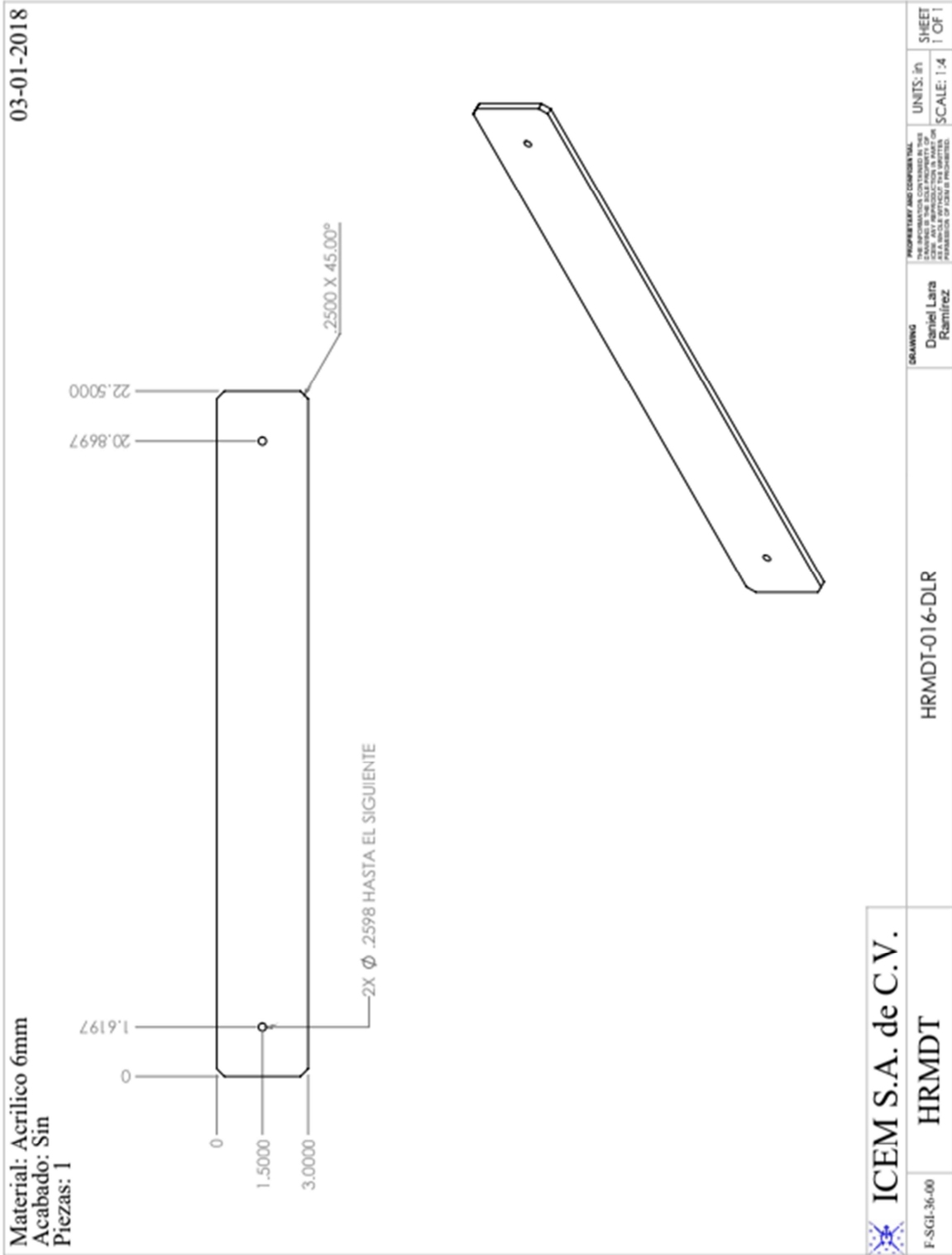
HRMDT-014-DLR

DRAWING  
 Daniel Lara  
 Ramirez

PROPERTY AND COPYRIGHT ARE  
 RESERVED BY ICEM S.A. DE C.V.  
 FOR ANY REPRODUCTION IN ANY FORM OR  
 MANNER WITHOUT THE WRITTEN  
 PERMISSION OF ICEM S.A. DE C.V.

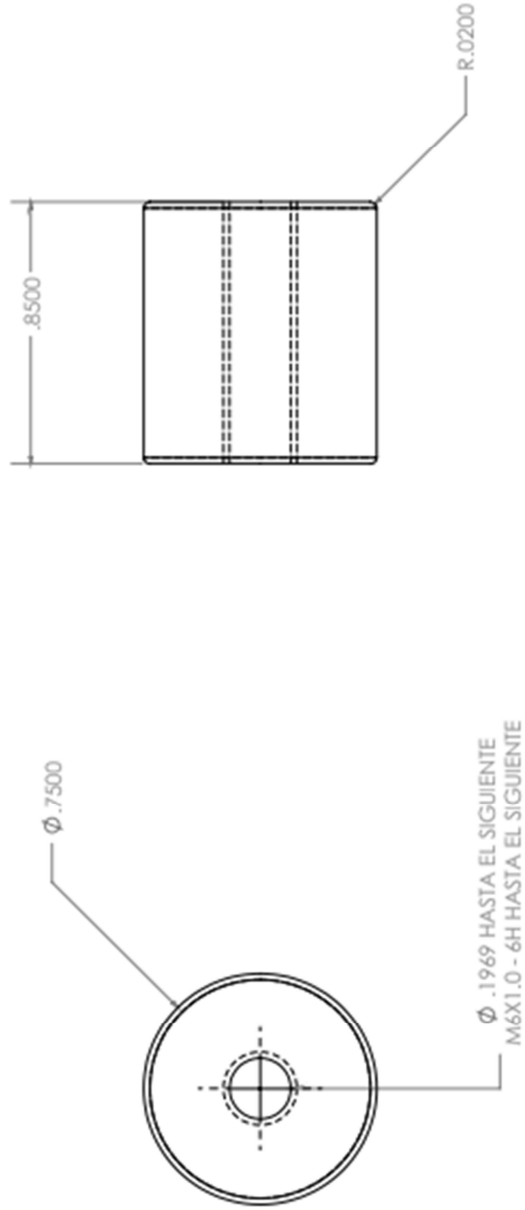
UNITS: in  
 SCALE: 1:1  
 SHEET  
 1 OF 1






03-01-2018

Material: Aluminio  
Acabado: Anodizado  
Piezas: 2



 ICEM S.A. de C.V.

F-SGI-36-00

HRMDT

HRMDT-017-DLR

DRAWING  
Daniel Lara  
Ramirez

PROPRIETARY AND CONFIDENTIAL  
THE INFORMATION CONTAINED IN THIS  
DRAWING IS UNCLASSIFIED EXCEPT WHERE  
NOTED. ANY REPRODUCTION IN WHOLE OR  
PART WITHOUT THE WRITTEN PERMISSION OF  
ICEM IS PROHIBITED.

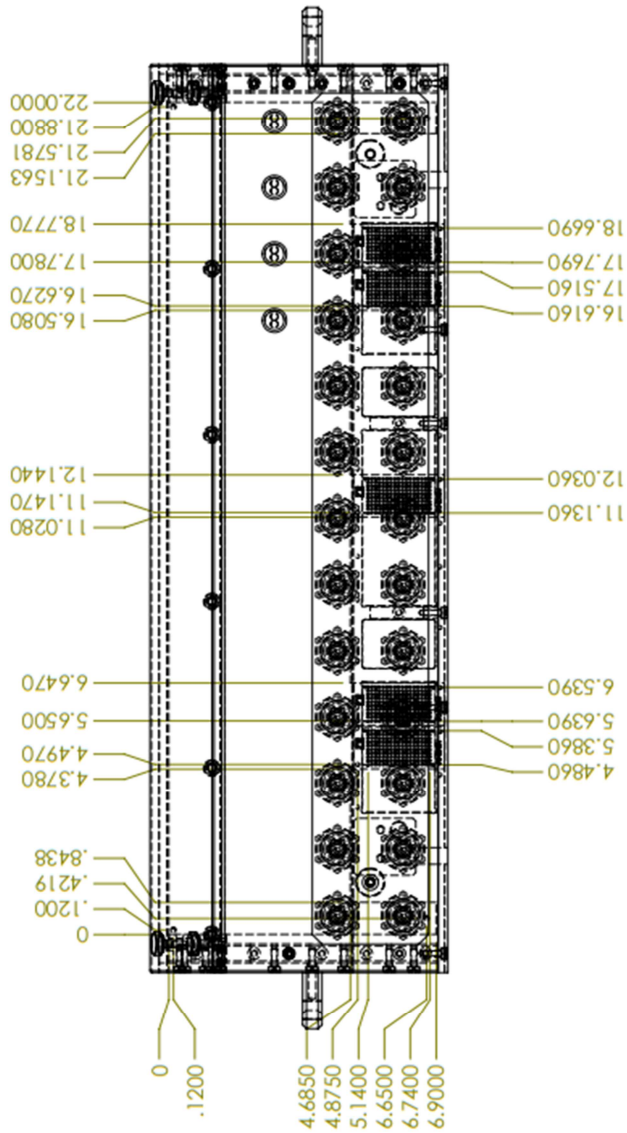
UNITS: in  
SCALE: 2:1  
SHEET  
1 OF

08-12-2017

Material: Tablilla HRMDT

Acabado:

Piezas: 1



**ICEM S.A. de C.V.**

F-SGI-36-00

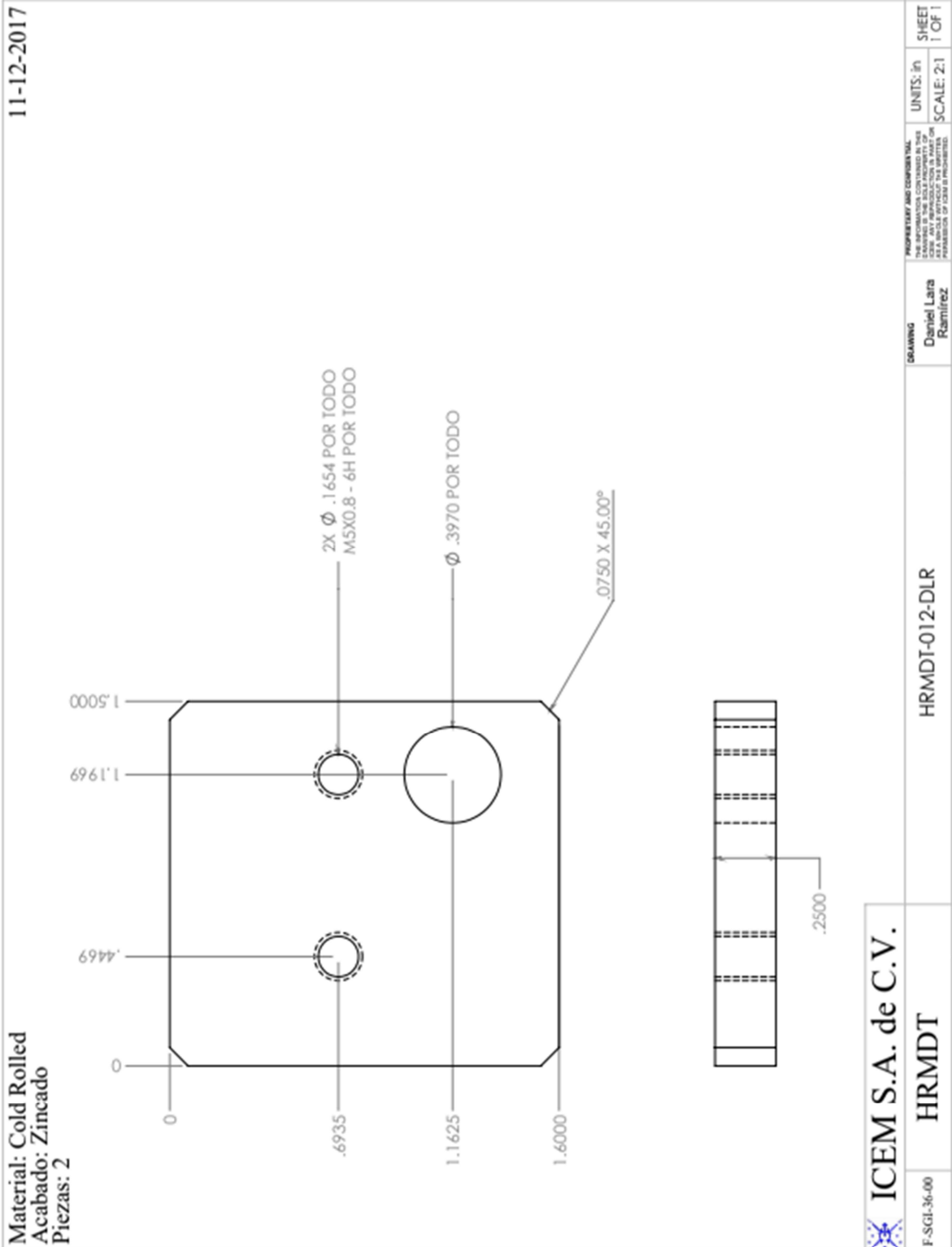
HRMDT

HRMDT-ASSY

DRAWING  
Daniel Lara  
Ramirez

PROPRIETARY AND CONFIDENTIAL  
THIS DRAWING IS THE SOLE PROPERTY OF  
ICEM S.A. AND IS TO BE KEPT AS SUCH  
AS A WHOLE WITHOUT THE WRITTEN  
CONSENT OF ICEM S.A.

UNITS: in  
SCALE: 1:8  
SHEET  
1 OF 1





## ANEXO XLI

### HMRDT MAIN TEST

```

/*****
/*
/* File: HMRDTMain.cpp
/* Test Program for HMRDT Device
/*
/* Copyright © 2006 Honeywell. All Rights Reserved.
/*
/* Rev: 1.00 10/05/06
/* Latest Rev by: HW
/*
/* Written by: HW 10/05/06
/*
*****/

#include "TestAll.h"
#pragma hdrstop

#include "HMRDTMainTest.h"
#include "HMRDTMainForm.h"
#include "MsgForm.h"
#include "Database.h"

//-----

// Application Name constant
const String APP_INI_NAME = "HMRDT.ini";
```

```

// Label constants

const String DEVICE_LABEL_FILE = "HMRDTLabel.dat";
const String DEVICE_LABEL_FORM = "HMRDTForm.btw";

//-----

//-----

//-----

#pragma package(smart_init)
#pragma resource "*.dfm"
TMainTest *MainTest;

//-----

__fastcall TMainTest::TMainTest( TComponent* Owner )
: TTestMainChild( Owner )
{
    // Setup the TestDB variable and databases
    TestDB    = new TTestDB( NULL );
    pOS_Data  = ((TTestMainBase *)Owner)->pOS_Data;
    pErrors   = new ERROR_DB( INI_ERROR_FILE, this );
    pLimits   = new LIMITS_DB( INI_LIMIT_FILE, this );
    pMeasure  = new TMeasureDB( INI_MEASURE_FILE, this );
    pStats    = new DEVICE_STATS( INI_STAT_FILE, INI_ALT_STAT_FILE, this );
    pFailExpert = new FAIL_DB( INI_FAIL_EXPERT_FILE, this );
    pLangString = new TLangStringDB( INI_LANG_FILE );
}

//-----

```

```

void TMainTest::GetCurOS_Info( void )
{
    // Code to load OS Number information from the database, gets called whenever a new model is selected
    // Get Options
}
//-----

void TMainTest::PrintLabel( void )
{
    String data_record;

    // Set to false for debug until printer is available
    bool print_it = false;

    // Build the data record
    data_record =
        FIELD_FIRST( OS_Number )          + // Field 1
        FIELD_NEXT( DateCode.GetCurDateCode() ) + // Field 2
        FIELD_END;

    // Print the file
    PrintBartenderLabel( DEVICE_LABEL_FORM, DEVICE_LABEL_FILE, data_record, print_it );
}
//-----

void TMainTest::PassStamp( void )
{
    // Stamp code to be executed if device passes
}

```

```

//-----

void TMainTest::Shutdown( void )
{
    // Shutdown instruments here, get called at end of every test
    AppendVarDataHeaderLine( "Test End" );
    // Turn off power

    // Open Relays
    // Digital Outputs Off
    // Air Off
}
//-----

//-----

/* Cut and paste code for test functions
//-----

// Power Supplies
    // AcSource->SetVoltage( 0.0 );
    // AcSource->SetFreq( 60.0 );
    // AcSource->SetCurLimit(1);
    // AcSource->SetMode(AC_MODE);
    // AcSource->On();
    // DcSource->SetVoltage( 0.0 );
    // DcSource->On();

// Close Relays
    // MeasureSystem->Close( RLY_DIGITAL_INPUT_2);

// Open Relays
    // MeasureSystem->Open( RLY_DIGITAL_INPUT_2);

```

```

// Digital Outputs
    // MeasureSystem->SetBit(DIO_ELPROTRONIC_RELAY, 0);
// Digital Inputs
    // MeasureSystem->ReadBit(DIO_NEST_CLOSED, IsNestClosed, 1);
// Air On
    // MeasureSystem->SetAir(AIR_NEURON_JACK_STEP2_6, 1);
// Air Off
    // MeasureSystem->SetAir(AIR_NEURON_JACK_STEP2_6, 1);
// DAC channel
    // MeasureSystem->DacOutput(DAC_10VCALSIGNAL, 10.0);
// Limits
    // pLimits->InLimits(LN_DIGITAL_OUTPUT_2_ON);           //reading the mux and checking
limits
    // pLimits->InLimits(LN_PRESS_JACK_INSERTED, reading);
    // pLimits->InString(LN_DIGITAL_INPUTS_NOT_WORKING, Results);
    // pLimits->InLimits(LN_LON_JACK_INSERTED, reading, "", LER_NONE); // Limit Use with no
launching errors
// Read MUX
    // MeasureSystem->ReadMux(reading, MUX_LON_JACK_SENSOR);
// Language Strings
    // LANG_STR( ILS_DIGITAL_INPUTS_TEST )
// Stat database fx
    // pStats->SetString( "ReportedStatus", "Success");
    // pStats->SetVar(DB_START_BOARD_TIME, msec);
// Errors
    // ERR_RETURN( LN_NO_BARCODE); //Launch an error
    // IF_FAIL_RETURN;           //to abort if an error has been
occured
// Data Variable edit box

```

```
// AppendVarDataTextLine(""); //adding an enter
// AppendVarDataHeaderLine( "Header in bold" ); //adding a header comment from the RC file
*/
//-----

//-----

//-----EXAMPLES-----
//-----

/*
// Use Language Strings

// Open Relays

// Close Relays

// Digital Outputs Off

// Digital Outputs On

// Air Off

// Air On

// Limit Use

*/
//-----
```

```

// Test Functions
//-----

void TMainTest::Test( void )
{

    //let this instruction be the first in the Test Sequence
    // ERR_RETURN( BoardCanBeTested() ); // Data Server Collection. Checking in the Server if the Unit
can be tested.

    TEST5VDC();
    TESTN12VDC();
    TEST12VDC();
    TEST24VDC();
    //TEST120VAC();
    TESTMUXES();
    TESTRELAYS();
    TESTMUXESCURRENT();
    TESTRELAYSCURRENT();
    TESTDIOS();
    TESTDAC();
    TESTDCPROGCH1();
    TESTDCPROGCH2();
    //TESTDCPROGCH3();
    //TESTDCPROGCH4();
    TESTDCKIKUSUI1();
    //TESTDCKIKUSUI2();

```

```

}

void TMainTest:: TEST5VDC(void)
{
    double Result5VDC;
    AppendVarDataHeaderLine("5VDC Testing");
    MeasureSystem->ReadMux(Result5VDC,MUX5VDC,1);
    pLimits->InLimits(LN_5VDC, Result5VDC);

}

void TMainTest::TESTN12VDC(void)
{
    double ResultN12VDC;
    AppendVarDataHeaderLine("-12VDC Testing");
    MeasureSystem->ReadMux(ResultN12VDC,MUXN12VDC,1);
    pLimits->InLimits(LN_N12VDC,ResultN12VDC);

}

void TMainTest::TEST12VDC(void)
{
    double Result12VDC;
    AppendVarDataHeaderLine("12VDC Testing");
    MeasureSystem->ReadMux(Result12VDC,MUX12VDC,1);
    pLimits->InLimits(LN_12VDC,Result12VDC);

}

```



```

void TMainTest::TEST24VDC(void)
{
    double Result24VDC;
    AppendVarDataHeaderLine("24VDC Testing");
    MeasureSystem->ReadMux(Result24VDC,MUX24VDC,1);
    pLimits->InLimits(LN_24VDC,Result24VDC);
}

void TMainTest::TEST120VAC(void)
{
    double Result120VAC;
    AppendVarDataHeaderLine("120VAC Testing");
    MeasureSystem->ReadMux(Result120VAC,MUX120VAC,1);
    pLimits->InLimits(LN_120VAC,Result120VAC*118.18);
}

void TMainTest::TESTDCPROGCH1(void)
{
    double ResultDC;
    DcSource->SetVoltage(1,1);
    DcSource->SetCurLimit(0.5,1);
    DcSource->On(1);
    AppendVarDataHeaderLine("DC Prog CH1 Testing");
    Timer.Delay_msec(500);
    MeasureSystem->ReadMux(ResultDC,MUXDCPROG1,1);
    pLimits->InLimits(LN_DCPROG1,ResultDC);
}

void TMainTest::TESTDCPROGCH2(void)

```

```

{
    double ResultDC;
    DcSource->SetVoltage(2,2);
    //DcSource->SetCurLimit(0.5,2);
    DcSource->On(2);
    AppendVarDataHeaderLine("DC Prog CH2 Testing");
    Timer.Delay_msec(500);
    MeasureSystem->ReadMux(ResultDC,MUXDCPROG2,1);
    pLimits->InLimits(LN_DCPROG2,ResultDC);
}
////////////////////////////////optional////////////////////////////////
void TMainTest::TESTDCPROGCH3(void)
{
    double ResultDC;
    DcSource->SetVoltage(1,3);
    DcSource->On(3);
    AppendVarDataHeaderLine("DC Prog CH3 Testing");
    MeasureSystem->ReadMux(ResultDC,MUXDCPROG3,1);
    pLimits->InLimits(LN_DCPROG3,ResultDC);
}

void TMainTest::TESTDCPROGCH4(void)
{
    double ResultDC;
    DcSource->SetVoltage(1,4);
    DcSource->On(4);
    AppendVarDataHeaderLine("DC Prog CH4 Testing");
    MeasureSystem->ReadMux(ResultDC,MUXDCPROG4,1);
    pLimits->InLimits(LN_DCPROG4,ResultDC);
}

```

```

}
////////////////////////////////////

void TMainTest::TESTDCKIKUSUI1(void)
{
    double ResultDCK;
    AcSource->SetMode(DC_MODE);
    AcSource->SetDCOffset(1);
    AcSource->On();
    AppendVarDataHeaderLine("Kikusui 1 Testing");
    Timer.Delay_msec(500);
    MeasureSystem->ReadMux(ResultDCK,MUXKIKUSUI1,1);
    pLimits->InLimits(LN_KIKUSUI1,ResultDCK);
}

void TMainTest::TESTDCKIKUSUI2(void)
{
    double ResultDCK;
    AcSource2->SetMode(DC_MODE);
    AcSource2->SetDCOffset(1);
    AcSource2->On();
    AppendVarDataHeaderLine("Kikusui 2 Testing");
    Timer.Delay_msec(500);
    MeasureSystem->ReadMux(ResultDCK,MUXKIKUSUI2,1);
    pLimits->InLimits(LN_KIKUSUI2,ResultDCK);
}

```

```
////////////////////////////////////
```

```
void TMainTest::TESTMUXES(void)
```

```
{
```

```
    bool MUXCARD2;
```

```
    int MUXTOTEST,MUX;
```

```
    double ResMUX;
```

```
    MUXCARD2= pOS_Data->GetTable()->FieldByName("MUXCARD2")->AsBoolean;
```

```
    MUXTOTEST=40;
```

```
    if (MUXCARD2) MUXTOTEST=80;
```

```
    AppendVarDataHeaderLine("Testing Muxes 23 to "+IntToStr(MUXTOTEST));
```

```
        for (MUX = 23; MUX <= MUXTOTEST; MUX++) {
```

```
            AppendVarDataTextLine("Testing Mux "+IntToStr(MUX));
```

```
            MeasureSystem->ReadMux(ResMUX,MUX);
```

```
            pLimits->InLimits(MUX,ResMUX);
```

```
        }
```

```
}
```

```
void TMainTest::TESTRELAYS(void)
```

```
{
```

```
    bool RELAYCARD2;
```

```
    int RELAYTOTEST, RELAY;
```

```
    double ResRELAY;
```

```

RELAYCARD2= pOS_Data->GetTable()->FieldByName("RELAYCARD2")->AsBoolean;
RELAYTOTEST=28;
if (RELAYCARD2)RELAYTOTEST=50;
AppendVarDataHeaderLine("Testing Relays 1 to "+IntToStr(RELAYTOTEST));

    for (RELAY = 1; RELAY <= RELAYTOTEST; RELAY++) {
        AppendVarDataTextLine("Testing Relay "+IntToStr(RELAY));
        MeasureSystem->ReadMux(ResRELAY, RELAYMUX);
        if (RELAYCARD2) pLimits->InLimits(LN_RELAYOPEN2, ResRELAY);
        else pLimits->InLimits(LN_RELAYOPEN, ResRELAY);
        MeasureSystem->Close(RELAY,1);
        MeasureSystem->ReadMux(ResRELAY, RELAYMUX);
        pLimits->InLimits(LN_RELAYCLOSED, ResRELAY);
        MeasureSystem->Open(RELAY,1);
    }

}

void TMainTest::TESTMUXESCURRENT(void)
{
    bool MUXCARD2;
    int MUXTOTESTCURRENT,MUXCURRENT;
    double ResMUXCURRENT;

    MUXCARD2= pOS_Data->GetTable()->FieldByName("MUXCARD2")->AsBoolean;
    MUXTOTESTCURRENT=4;

```

```

if (MUXCARD2) MUXTOTESTCURRENT=5;
AppendVarDataHeaderLine("Testing Current Muxes 1 to "+IntToStr(MUXTOTESTCURRENT));

    for (MUXCURRENT = 1; MUXCURRENT <= MUXTOTESTCURRENT;
MUXCURRENT++) {
        AppendVarDataTextLine("Testing Current Mux "+IntToStr(MUXCURRENT));
        MeasureSystem->ReadMuxCurrent(ResMUXCURRENT,MUXCURRENT);
        pLimits->InLimits(MUXCURRENT,ResMUXCURRENT);
    }
}

void TMainTest::TESTRELAYSCURRENT(void)
{
    bool RELAYCARD2;
    int RELAYTOTESTCURRENT,RELAYCURRENT;
    double ResRELAYCURRENT;

    RELAYCARD2= pOS_Data->GetTable()->FieldByName("RELAYCARD2")->AsBoolean;
    RELAYTOTESTCURRENT=4;
    if (RELAYCARD2) RELAYTOTESTCURRENT=5;
    AppendVarDataHeaderLine("Testing          Current          Relays          1          to
"+IntToStr(RELAYTOTESTCURRENT));

        for (RELAYCURRENT = 1; RELAYCURRENT <= RELAYTOTESTCURRENT;
RELAYCURRENT++) {
            AppendVarDataTextLine("Testing Current Relay "+IntToStr(RELAYCURRENT));
            MeasureSystem->OpenSPST(RELAYCURRENT,1);
            MeasureSystem->ReadMux (ResRELAYCURRENT,RELAYMUX);

```

```

        if (RELAYCARD2) pLimits->InLimits(LN_RELAYOPEN2, ResRELAYCURRENT);
        else pLimits->InLimits(LN_RELAYOPEN, ResRELAYCURRENT);
        MeasureSystem->CloseSPST(RELAYCURRENT,1);
        MeasureSystem->ReadMux(ResRELAYCURRENT, RELAYMUX);
        pLimits->InLimits(LN_RELAYCLOSED, ResRELAYCURRENT);
        MeasureSystem->OpenSPST(RELAYCURRENT,1);
    }
}

void TMainTest::TESTDIOS(void)
{
    bool DIOCARD2;
    bool DIOCARD3;
    bool DIOCARD4;
    String sDIO;
    String sENABLED;
    String rCOMANDO;
    int RESULTADO;
    int IO;
    TIMER diotimer;

    DIOCARD2= pOS_Data->GetTable()->FieldByName("DIOCARD2")->AsBoolean;
    DIOCARD3= pOS_Data->GetTable()->FieldByName("DIOCARD3")->AsBoolean;
    DIOCARD4= pOS_Data->GetTable()->FieldByName("DIOCARD4")->AsBoolean;

    AppendVarDataTextLine("Testing IO 1 to 32");
    for (IO=1; IO <= 32; IO++) {
        sDIO="DIO"+IntToStr(IO);

```

```

sENABLED= pConfigFile->ReadString("DIOTEST", sDIO, "");
if (sENABLED=="ENABLED") {
rCOMANDO= pConfigFile->ReadString("DIOICTEST", sDIO, "");

/////////

AppendVarDataTextLine("Testing IO "+ IntToStr(IO));

MeasureSystem->SetBit(IO,true,1);
diotimer.Delay_msec(50);

RESULTADO= MainForm->DIOS->Read(rCOMANDO);
// pLimits->InLimits(LN_LIMITEON, RESULTADO);
pLimits->Logical(LN_LIMITEON, RESULTADO, "IO_ON", LN_LIMITEON);

/////////

MeasureSystem->SetBit( IO,false,1);

diotimer.Delay_msec(50);
RESULTADO= MainForm->DIOS->Read(rCOMANDO);
// pLimits->InLimits(LN_LIMITEOFF, RESULTADO);
pLimits->Logical(LN_LIMITEOFF, !RESULTADO, "IO_OFF", LN_LIMITEOFF);

}
}
//////////card2//////////

```



```

if (DIOCARD2) {

AppendVarDataTextLine("Testing IO 33 to 64");
for (IO=33; IO <= 64; IO++) {
    sDIO="DIO"+IntToStr(IO);

    sENABLED= pConfigFile->ReadString("DIOTEST", sDIO, "");
    if (sENABLED=="ENABLED") {
        rCOMANDO= pConfigFile->ReadString("DIOICTEST", sDIO, "");

//////////

        AppendVarDataTextLine("Testing IO "+ IntToStr(IO));
        MeasureSystem->SetBit(IO,true,1);
        diotimer.Delay_msec(50);

        RESULTADO= MainForm->DIOS->Read(rCOMANDO);
        //pLimits->InLimits(LN_LIMITEON, RESULTADO);
        pLimits->Logical(LN_LIMITEON, RESULTADO, "IO_ON", LN_LIMITEON);
        diotimer.Delay_msec(50);

//////////

        AppendVarDataTextLine("Testing IO "+ IntToStr(IO));
        MeasureSystem->SetBit(IO,false,1);
        diotimer.Delay_msec(50);

        RESULTADO= MainForm->DIOS->Read(rCOMANDO);
        //pLimits->InLimits(LN_LIMITEOFF, RESULTADO);
        pLimits->Logical(LN_LIMITEOFF, !RESULTADO, "IO_OFF", LN_LIMITEOFF);
    }
}
}

```

```

}
}
}

////////////////////////////////card3////////////////////////////////

if (DIOCARD3) {
AppendVarDataTextLine("Testing IO 65 to 96");
for (IO=65; IO <= 96; IO++) {
    sDIO="DIO"+IntToStr(IO);

    sENABLED= pConfigFile->ReadString("DIOTEST", sDIO, "");
    if (sENABLED=="ENABLED") {
        rCOMANDO= pConfigFile->ReadString("DIOICTEST", sDIO, "");

////////////////////////////////

        AppendVarDataTextLine("Testing IO "+ IntToStr(IO));
        MeasureSystem->SetBit(IO,true,1);
        diotimer.Delay_msec(50);

        RESULTADO= MainForm->DIOS->Read(rCOMANDO);
        pLimits->Logical(LN_LIMITEON, RESULTADO, "IO_ON", LN_LIMITEON);

////////////////////////////////

        AppendVarDataTextLine("Testing IO "+ IntToStr(IO));
        MeasureSystem->SetBit(IO,false,1);
        diotimer.Delay_msec(50);
    }
}
}
}

```

```

RESULTADO= MainForm->DIOS->Read(rCOMANDO);
pLimits->Logical(LN_LIMITEOFF, !RESULTADO, "IO_OFF", LN_LIMITEOFF);

}

}

}

////////////////////////////////card4////////////////////////////////

if (DIOCARD4) {

AppendVarDataTextLine("Testing IO 97 to 128");
for (IO=97; IO <= 128; IO++) {
    sDIO="DIO"+IntToStr(IO);

    sENABLED= pConfigFile->ReadString("DIOTEST", sDIO, "");
    if (sENABLED=="ENABLED") {
        rCOMANDO= pConfigFile->ReadString("DIOICTEST", sDIO, "");

////////////////////////////////

AppendVarDataTextLine("Testing IO "+ IntToStr(IO));
MeasureSystem->SetBit(IO,true,1);
diotimer.Delay_msec(50);

RESULTADO= MainForm->DIOS->Read(rCOMANDO);
pLimits->Logical(LN_LIMITEON, RESULTADO, "IO_ON", LN_LIMITEON);

```

```
////////
```

```
AppendVarDataTextLine("Testing IO "+ IntToStr(IO));
```

```
MeasureSystem->SetBit(IO,false,1);
```

```
diotimer.Delay_msec(50);
```

```
RESULTADO= MainForm->DIOS->Read(rCOMANDO);
```

```
pLimits->Logical(LN_LIMITEOFF, !RESULTADO, "IO_OFF", LN_LIMITEOFF);
```

```
}
```

```
}
```

```
}
```

```
////////////////////////////////////
```

```
}
```

```
//-----
```

```
//-----
```

```

void TMainTest::TESTDAC(void){
double DacResult;
bool DACSOPTIONAL;
DACSOPTIONAL= pOS_Data->GetTable()->FieldByName("DACSOPTIONAL")->AsBoolean;

AppendVarDataTextLine("Testing DAC1 at 1VDC");
MeasureSystem->DacOutput(1,1,1);
MeasureSystem->ReadMux(DacResult,MUXDAC1,1);
pLimits->InLimits(LN_DAC1ON, DacResult);

////////////////////////////////////

AppendVarDataTextLine("Testing DAC2 at 2VDC");
MeasureSystem->DacOutput(2,2,1);
MeasureSystem->ReadMux(DacResult,MUXDAC2,1);
pLimits->InLimits(LN_DAC2ON, DacResult);

////////////////////////////////////

AppendVarDataTextLine("Testing DAC3 at 3VDC");
MeasureSystem->DacOutput(3,3,1);
MeasureSystem->ReadMux(DacResult,MUXDAC3,1);
pLimits->InLimits(LN_DAC3ON, DacResult);

////////////////////////////////////

AppendVarDataTextLine("Testing DAC4 at 4VDC");

```

```

MeasureSystem->DacOutput(4,4,1);
MeasureSystem->ReadMux(DacResult,MUXDAC4,1);
pLimits->InLimits(LN_DAC4ON, DacResult);

////////////////////////////////////

if (DACSOPTIONAL) {

////////////////////////////////////

AppendVarDataTextLine("Testing DAC5 at 5VDC");
MeasureSystem->DacOutput(5,5,1);
MeasureSystem->ReadMux(DacResult,MUXDAC5,1);
pLimits->InLimits(LN_DAC5ON, DacResult);

////////////////////////////////////

AppendVarDataTextLine("Testing DAC6 at 6VDC");
MeasureSystem->DacOutput(6,6,1);
MeasureSystem->ReadMux(DacResult,MUXDAC6,1);
pLimits->InLimits(LN_DAC6ON, DacResult);

////////////////////////////////////

AppendVarDataTextLine("Testing DAC7 at 7VDC");
MeasureSystem->DacOutput(7,7,1);
MeasureSystem->ReadMux(DacResult,MUXDAC7,1);
pLimits->InLimits(LN_DAC7ON, DacResult);

```

```
////////////////////////////////////

AppendVarDataTextLine("Testing DAC8 at 8VDC");
MeasureSystem->DacOutput(8,8,1);
MeasureSystem->ReadMux(DacResult,MUXDAC8,1);
pLimits->InLimits(LN_DAC8ON, DacResult);
////////////////////////////////////

}

}
```

```
MAIN TEST .H
/*****/
/*          */
/* File: HMRDTMain.h          */
/* Test Program include for HMRDT Device          */
/*          */
/* Copyright © 2006 Honeywell. All Rights Reserved.          */
/*          */
/* Rev: 1.10 06/09/09          */
/* Latest Rev by: HW          */
/*          */
/* Written by: HW 10/05/06          */
/*          */
```

```

/*****/

#ifndef HMRDTMainTestH
#define HMRDTMainTestH

//-----

#include <Classes.hpp>
#include <ComCtrls.hpp>
#include <Controls.hpp>
#include <DBCtrls.hpp>
#include <Dialogs.hpp>
#include <ExtCtrls.hpp>
#include <Menus.hpp>
#include <StdCtrls.hpp>
#include "TestmainChild.h"
#include <ADODB.hpp>
#include <DB.hpp>

//-----
//-----
// OS Defines
//-----

// Limit Defines
#define LN_5VDC 102
#define LN_N12VDC 103
#define LN_12VDC 104
#define LN_24VDC 105
#define LN_120VAC 106

```



```
#define LN_RELAYOPEN 108
#define LN_RELAYCLOSED 109
#define LN_LIMITEON 110
#define LN_LIMITEOFF 111
#define LN_DAC1ON 121
#define LN_DAC2ON 122
#define LN_DAC3ON 123
#define LN_DAC4ON 124
#define LN_DAC5ON 125
#define LN_DAC6ON 126
#define LN_DAC7ON 127
#define LN_DAC8ON 128
#define LN_DCPROG1 130
#define LN_DCPROG2 131
#define LN_DCPROG3 132
#define LN_DCPROG4 133
#define LN_KIKUSUI1 134
#define LN_KIKUSUI2 135
#define LN_RELAYOPEN2 136
```

```
//-----
```

```
// Mux Defines
```

```
#define MUX5VDC 2
#define MUXN12VDC 14
#define MUX12VDC 3
#define MUX24VDC 4
#define MUX120VAC 5
```

```
#define RELAYMUX 1
#define MUXDAC1 15
#define MUXDAC2 16
#define MUXDAC3 17
#define MUXDAC4 18
#define MUXDAC5 19
#define MUXDAC6 20
#define MUXDAC7 21
#define MUXDAC8 22
#define MUXDCPROG1 8
#define MUXDCPROG2 9
#define MUXDCPROG3 10
#define MUXDCPROG4 11
#define MUXKIKUSUI1 6
#define MUXKIKUSUI2 7
```

```
//-----
```

```
// Relay Defines
```

```
//-----
```

```
// IO Defines
```

```
//-----
```

```
// Air Defines
```

```
//-----
```

```
// DAC Defines
```

```
//-----
```

```

// Language Strings
//-----

// Function Strings
//-----

// Stat Defines
//-----

//-----

//-----

class TMainTest : public TTestMainChild
{
    __published: // IDE-managed Components

protected:
    // Test Sequence Variables

public: // User declarations
    // Misc Functions
    __fastcall virtual TMainTest( TComponent* Owner );
    void GetCurOS_Info( void );
    void PrintLabel( void );
    void PassStamp( void );
    void Shutdown( void );

```

```
void Test( void );
void TEST5VDC(void);
void TESTN12VDC(void);
void TEST12VDC(void);
void TEST24VDC(void);
void TEST120VAC(void);
void TESTMUXES(void);
void TESTRELAYS(void);
void TESTMUXESCURRENT(void);
void TESTRELAYSCURRENT(void);
void TESTDIOS(void);
void TESTDAC(void);
void TMainTest::TESTDCPROGCH1(void);
void TMainTest::TESTDCPROGCH2(void);
void TMainTest::TESTDCPROGCH3(void);
void TMainTest::TESTDCPROGCH4(void);
void TMainTest::TESTDCKIKUSUI1(void);
void TMainTest::TESTDCKIKUSUI2(void);
```

```
// Test Functions
```

```
};
```

```
//-----
```

```
extern PACKAGE TMainTest *MainTest;
```

```
//-----
```

```
#endif
```

## MAIN FORM

```

/*****
*/
/* File: HMRDTMain.cpp */
/* Test Program for HMRDT Device */
*/
/* Copyright © 2006 Honeywell. All Rights Reserved. */
*/
/* Rev: 1.10 06/09/09 */
/* Latest Rev by: HW */
/* Written by: HW 10/05/06 */
*/
*****/

#include "TestAll.h"
#pragma hdrstop

#include <system.hpp>
#include "HMRDTMainForm.h"
#include "HMRDTMainTest.h"
#include "Test_DB.h"
#include "Unit1.h"

//-----

// Application Name constant
const String APP_INI_NAME = "HMRDT.ini";

#pragma package(smart_init)

```

```

#pragma link "cspin"
#pragma resource "*.dfm"
TMainForm *MainForm;
//-----

// Create Global Instruments
String VISA_ADDR_AC_SOURCE = "ACSource";
//String VISA_ADDR_AC_SOURCE2= "ACSource2";
String VISA_ADDR_DC_SOURCE = "DCSource";

TAgilentAC *AcSource = NULL;
TAgilentAC *AcSource2= NULL;
TAgilentDC *DcSource = NULL;
//-----

// Nest constants
const int NUM_NESTS = 1;
//-----

__fastcall TMainForm::TMainForm( TComponent* Owner )
: TTestMain( Owner, APP_INI_NAME )
{
    // Setup TestMain variable
    TestMain = this;

    // Setup the TestDB variable and databases
    TestDB    = new TTestDB( NULL );
    pOS_Data  = new OS_DATA_DB( INI_OS_DATA_FILE );
    pErrors   = new ERROR_DB( INI_ERROR_FILE );
}

```

```

pLimits   = new LIMITS_DB( INI_LIMIT_FILE );
pMeasure  = new TMeasureDB( INI_MEASURE_FILE );
pStats    = new DEVICE_STATS( INI_STAT_FILE, INI_ALT_STAT_FILE );
pFailExpert = new FAIL_DB( INI_FAIL_EXPERT_FILE );
pLangString = new TLangStringDB( INI_LANG_FILE );

// Setup GPIB
try
{
    MeasureSystem = new TMeasureSystem();
    AcSource= new TAgilentAC(VISA_ADDR_AC_SOURCE);
    //AcSource2= new TAgilentAC(VISA_ADDR_AC_SOURCE2);

    DcSource   = new TAgilentDC( VISA_ADDR_DC_SOURCE, "DCSource" );
#ifdef SIMULATED
    AcSource->Reset();
    //AcSource2->Reset();
    DcSource->Reset();
    MeasureSystem->Reset();
    MeasureSystem->GetDeviceConfig();
    MeasureSystem->SetNegativeLogic( 0, true );
    MeasureSystem->ResetOutputs();
    MeasureSystem->SetErrorOffset( 1 );
    m_pObjVisionCtrl = new TUSBVisionCtrl();
    DIOS= new TDIOS;
    DIOS->Open(STATUS);
#endif
}
catch( XVisa &x )

```

```

    {
    }
    // StartStyle = SS_AUTO;
}
//-----

__fastcall TMainForm::~TMainForm( void )
{

}
//-----

void TMainForm::DisplayCurOS_Info( void )
{
    // Code to display OS Number information from the database, gets called whenever a new model is selected
}
//-----

void TMainForm::Calibration( void )
{
    // Put calibration code here
#ifdef BUILDER_VERSION_NUM >= 2009
    Application->MessageBox( L"Calibration Code", L"Calibration", MB_OK );
#else
    Application->MessageBox( "Calibration Code", "Calibration", MB_OK );
#endif
}
//-----

```



```

void TMainForm::Manual( void )
{
    // Put manual code here
    #if ( BUILDER_VERSION_NUM >= 2009 )
        Application->MessageBox( L"Manual Code", L"Manual", MB_OK );
    #else
        Application->MessageBox( "Manual Code", "Manual", MB_OK );
    #endif
    //MeasureMainForm->ShowModal();
    Form1->ShowModal();
}
//-----

void __fastcall TMainForm::FormShow(TObject *Sender)
{
    // Create children
    if (!fInitializedChildForms)
    {
        for (int i = 0; i < NUM_NESTS; i++)
            pChildList->Add( new TMainTest( this ) );
    }
    // Call base class FormShow
    TTestMain::FormShow( Sender );
}
//-----
//-----

```

MAIN FORM .H

```

/*****
/*
/* File: HMRDTMain.h
/* Test Program include for HMRDT Device
/*
/* Copyright © 2006 Honeywell. All Rights Reserved.
/*
/* Rev: 1.10 06/09/09
/* Latest Rev by: HW
/*
/* Written by: HW 10/05/06
/*

/*****

#ifndef HMRDTMainFormH
#define HMRDTMainFormH

//-----

#include <Classes.hpp>
#include <ComCtrls.hpp>
#include <Controls.hpp>
#include <DBCtrls.hpp>
#include <Dialogs.hpp>
#include <ExtCtrls.hpp>
#include <Menus.hpp>
#include <StdCtrls.hpp>
#include "TestmainChild.h"
#include "TestmainParent.h"
#include "HMRDTMainTest.h"

```

```

#include "MeasureMain.h"

#include "cspin.h"

#include "TestmainVersion.h"

#include "USBVisionCtrl.h"

#include "DIOs.h"

//-----

// Global instruments
extern TAgilentAC *AcSource;
extern TAgilentAC *AcSource2;
extern TAgilentDC *DcSource;

//-----

class TMainForm : public TTestMain
{
    __published: // IDE-managed Components
    void __fastcall FormShow(TObject *Sender);

protected:
    // OS Number Functions
    void DisplayCurOS_Info( void );

public: // User declarations
    __fastcall virtual TMainForm( TComponent* Owner );
    __fastcall virtual ~TMainForm( void );
    TUSBVisionCtrl *m_pObjVisionCtrl;
    String STATUS;

```

```
TDIOS *DIOS;

void Calibration( void );
void Manual( void );
};
//-----

extern PACKAGE TMainForm *MainForm;
//-----

#endif
```

```
DIOS CLASS
//-----

#pragma hdrstop

#include "DIOs.h"

//-----

#pragma package(smart_init)

/////////////////////////////////ICEM/////////////////////////////////

TDIOS::TDIOS( void ){

    m_pDIOSPort= new COM_IO;
    pDIOSLock = new TCriticalSection();
```

```

        sComPort = "";
    }
//----- ICEM -----

TDIOS::~TDIOS( void ){
    m_pDIOSPort->Close();
    delete m_pDIOSPort;
    if(pDIOSLock)
        delete pDIOSLock;
}

//////////OPENPORTDIOS//////////

int TDIOS::Open(String &sStatus)
{
    String sComMode;
    String sResponse;
    String sCommand;
    char cInputResponse[1024] = {0};
    char cOutputResponse[1024] = {0};
    int iQueue;
    int iPosition;

    // Open the COM Port according to Configuration File.
    sComPort = pConfigFile->ReadString ("DIOs", "DIOs_COM_PORT", "");
    sComMode = pConfigFile->ReadString ("DIOs", "DIOs_COM_MODE", "");
    m_pDIOSPort->SetCommMode (sComPort, sComMode, 8192, 8192);
    if(m_pDIOSPort->Open()){
        sStatus = "Error Opening DIOs Port " + sComPort;
    }
}

```

```

        return -1;

    }

    //Flush Input/Output Queues.
    m_pDIOSPort->FlushQueues();
    Application->ProcessMessages();

    sStatus = "Success";
    return 0;
}

//////////CLOSEPORTDIOS//////////

int TDIOS::Close(String &sStatus)
{
    if(sComPort != ""){
        if(m_pDIOSPort->Close()){
            sStatus = "Error Closing DIOs Port " + sComPort;
            return -1;
        }
    }

    sStatus = "Success";
    return 0;
}

int TDIOS::Read(String COMANDO){
    String Numerodeintegrado;
    int READBIT;
    String SREADBIT;

```

```

char RESULT[10];
String RESULTADO;
int iQueue;

////////////////////////////////////

m_pDIOSPort->FlushQueues();
m_pDIOSPort->WriteString(COMANDO);

do{
iQueue = m_pDIOSPort->GetNumInRecvQue();
}while(iQueue<1);

m_pDIOSPort->Read(RESULT, iQueue);

RESULTADO=RESULT;

if (RESULTADO.Pos("T")) {
return 1;
}
if (RESULTADO.Pos("F")) {
return 0;
}

}

////////////////////////////////////

```

```
DIOS CLASS .H
```

```
//-----
```

```

#ifndef DIOsH
#define DIOsH
//-----
#include <Syncobjs.hpp>
#include "WincomIO.h"
#include "WinTimer.h"

class TDIOS{
protected:
    COM_IO * m_pDIOSPort;
    String sComPort;
    TIMER DIOSTimer;
    TCriticalSection *pDIOSLock;

public: // User declarations
    TDIOS( void );
    ~TDIOS( void );
    int Open(String &sStatus);
    int Close(String &sStatus);
    int TDIOS::Read(String COMANDO);
};

#endif

```



