

Instituto Tecnológico de Costa Rica
Escuela de Ingeniería en Electrónica



Proyecto De Graduación

Mejoras a las pruebas eléctricas utilizadas en la validación de ChipSets

Componentes INTEL Costa Rica

Mauricio Brenes Quesada

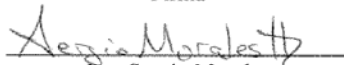
Federico Del Valle Monge

Cartago, Enero 2007

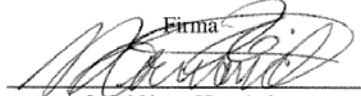
INSTITUTO TECNOLOGICO DE COSTA RICA
ESCUELA DE INGENIERIA ELECTRONICA
PROYECTO DE GRADUACIÓN
TRIBUNAL EVALUADOR

Proyecto de Graduación defendido ante el presente Tribunal Evaluador como requisito para optar por el título de Ingeniero en Electrónica con el grado académico de Licenciatura, del Instituto Tecnológico de Costa Rica.

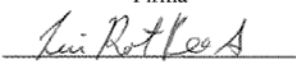
Miembros del Tribunal

Firma

Ing. Sergio Morales

Profesor lector

Firma

Ing. Néstor Hernández

Profesor lector

Firma

Ing. Roberto Pereira Arroyo

Profesor asesor

Los miembros de este Tribunal dan fe de que el presente trabajo de graduación ha sido aprobado y cumple con las normas establecidas por la Escuela de Ingeniería Electrónica

Cartago, 2 de Febrero del 2007

Declaro que el presente Proyecto de Graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas.

En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado y por el contenido del correspondiente informe final.

Componentes Intel CR
Viernes 2 de Febrero, 2007.



Federico Del Valle Monge

Cédula: 1-1096-0923

Declaro que el presente Proyecto de Graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas.

En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado y por el contenido del correspondiente informe final.

Componentes Intel CR
Viernes 2 de Febrero, 2007.



Mauricio Brenes Quesada

Cédula: 3-0363-0082

A nuestros padres con mucho amor.

*Gracias por su comprensión
y apoyo incondicional en
estos años de estudio.*

Resumen

En el departamento Ingeniería del Producto específicamente en la plataforma de validación de ChipSets de la empresa Componentes Intel de Costa Rica no se contaba con la información suficiente para realizar un análisis del porqué se están dando ciertas fallas en la línea de producción, así como si la falla que se presentaba se debía a la máquina de pruebas o en causa propia de la unidad sometida a prueba.

La principal consecuencia de que este problema no fuera resuelto es que se seguirían clasificando unidades en buen estado como unidades defectuosas. Además el origen de la falla sería una incógnita debido a la carencia de información necesaria. Finalmente el descartar unidades en buen estado hace que decaiga el índice de productividad del sitio, repercutiendo en la imagen y la confiabilidad por parte de la gerencia internacional.

Los beneficios de la solución al problema planteado se notan en la confiabilidad sobre los resultados obtenidos en las pruebas, además de la retroalimentación que se obtenida al contar con información específica de la causa de la falla, de esta manera se permite tener mas eficiencia y se es mas certero a la hora de implementar las acciones correctivas al proceso.

Para darle solución al problema se elaboraron una serie de diseños metodológicos de pruebas que cumplen con los requisitos estipulados para no afectar al producto y simultáneamente generar mayor información para el mantenimiento del módulo, así como acciones correctivas de algunas de las causas de error más comunes.

Abstract

The product engineering department, more specifically in the ChipSets validation platform of Intel Costa Rica did not have available the necessary information in order to perform an analysis based on the reason why certain flaws are happening in the production line, as well as if the flaw that presented itself was due to the test module or to the unit itself.

The main consequence of this problem not being fixed is that units that are actually in an operational state will still be qualified as defective, as well as the fact that the actual origin of the flaw would still be a mystery due to lack of information. It is also worth mentioning that the fact that operational units are discarded has a negative impact in the productivity reports of the site, as well as in the image and reliability of the company in Costa Rica and worldwide.

The benefits of the problems solution are noticeable in the reliability of the results obtained from the tests, as well as the important feedback that means to have the direct cause of the flaw. Therefore the process is more efficient and the engineers can be certain when it comes to implementing the necessary corrective actions in the process.

In order to implement a solution to the problem a series of methodological designs were implemented to run tests, these designs are oriented to be able to run and not affect in any manner the product, as well as to generate the necessary information for the maintenance of the module and corrective actions for the most common errors.

ÍNDICE

Capítulo 1. Introducción	1
1.1. Generalidades.....	1
1.2. Descripción Sintética del Problema	2
1.3. Solución Seleccionada	3
Capítulo 2. Meta y Objetivos	5
2.1. Meta	5
2.2. Objetivo General.....	5
2.3. Objetivos Específicos.....	5
Capítulo 3. Marco Teórico.....	8
3.1. Descripción del Equipo.....	8
3.2. Generalidades Técnicas	12
3.3. Distribuciones y Conceptos Estadísticos	13
3.4. Cualidades Necesarias para la Lectura del TESTINB.	16
Capítulo 4. Metodología	24
4.1. Procedimiento Metodológico.....	24
4.2. Detalle de Actividades	25
Capítulo 5. Descripción Detallada de la Solución	29
5.1. Correlación entre la Variable Tensión de la Tarjeta Madre con el Fallo de una Prueba por parte una Unidad.....	29
5.2. Solución Seleccionada	36
5.3. Implementación de Post Codes	39
Capítulo 6. Análisis de Resultados	52
6.1. Resultados para la prueba de corrección según la probabilidad de éxito.	52
6.2. Lectura de la Identificación de la Unidad.....	61
6.3. Lectura de Post Codes.....	63
Capítulo 7. Conclusiones	67
Bibliografía	68
APENDICES.....	69
Apéndice.1. Glosario	69
Apéndice.2. Ubicación del Proyecto.....	71
Apéndice.2.1. Descripción de la Empresa	71
Apéndice.2.2. Departamento Específico.....	71

INDICE DE FIGURAS

Figura 1.1. Proceso de ensamble y prueba de un dispositivo Chipset.....	2
Figura 1.1. Diagrama de Flujo Actual del Módulo de Validación	4
Figura 3.1. Dispositivo de Control de Pruebas Eléctricas. En la fotografía se muestra la tarjeta Centrix, en posición vertical, y la CelBridge en posición horizontal	9
Figura 3.2. Jerarquía funcional de un sistema operativo por capas	11
Figura 3.3. Estratificación de las pruebas realizadas en el módulo de validaciones	12
Figura 3.4. Aproximación de una Distribución según los Datos Tomados	13
Figura 3.5. a) Simétrica, b) Asimétrica izquierda.....	13
Figura 3.6. Conjunto de Datos en un Histograma que Evidencian una Distribución con dos Modas (Bimodal).....	14
Figura 3.7. Distribución de datos donde se dan algunas excepciones a considerar.....	14
Figura 3.8. Representación de la Diferencia entre la Media y la Mediana en una Distribución.....	15
Figura 3.9. Representación de una Distribución simétrica	15
Figura 3.10. Descripción de la Caja Estadística	16
Figura 3.11. Diagrama Esquemático de la Lógica Envuelta en el Funcionamiento del TESTINB	18
Figura 3.12. Diagrama de Estados de la Lógica del TESTINB.....	20
Figura 3.13: Diagrama de Bloques de del Acceso a los Fusibles.....	23
Figura 5.1. Distribución de la Tensión en la Tarjeta Madre de Pruebas.....	29
Figura 5.2. Distribución de la Tensión en la Tarjeta Madre de la Unidades que Pasan ..	30
Figura 5.3. Tendencia de las unidades validadas como buenas contra la tensión medida en la tarjeta madre	31
Figura 5.4. Grafico de Variabilidad según el Módulo, Ramal y Cabeza.....	33
Figura 5.5. Gráfico de Variabilidad según con la Identificación de la Cabeza y las Tensiones Medidas.....	35
Figura 5.6. Diagrama de Flujo de la Solución Propuesta	38
Figura 5.7. Post Card	40
Figura 5.8. Tarjeta de expansión del bus PCI.....	43
Figura 5.9. Diagrama de flujo del funcionamiento de la prueba del sistema operativo ..	45
Figura 5.10. Diagrama de flujo de la prueba del sistema operativo modificado	47
Figura 6.1. Resultados de la Prueba Sugerida Para la Sección de Probabilidad.....	53
Figura 6.2. Resultado de una Prueba Pasada por una Unidad sin Problema	53
Figura 6.3. Resultado de una Prueba Corregida Específicamente para el error de mayor incidencia.....	54
Figura 6.4. Gráfico de Barras Comparativo Representativo a la Incidencia del Éxito (Válida), Error (Bin 6205), y Recuperaciones debidas al Programa de Pruebas Sugerido	54
Figura 6.5. Resultado de un Archivo Resumen Predispuesto a ser Subido a Las Bases de Datos de la Empresa.....	62
Figura 6.6. Decodificación de una Cadena Binaria	62
Figura 6.7. Imagen de la pantalla que muestra el modulo de pruebas en caso de una falla	64

INDICE DE TABLAS

Tabla 3.1. Tabla de Descripción del Registro de Pruebas	17
Tabla 5.1. Cuadro Resumen de las Tensiones Encontradas.....	31
Tabla 5.2. Post Card Byte 0	41
Tabla 5.3. Post Card Byte 1	41
Tabla 5.4. Opciones de frecuencia de la Post Card.....	42
Tabla 6.1. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 1 del Racal 1	55
Tabla 6.2. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 2 del Racal 1	56
Tabla 6.3. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en el Racal	56
Tabla 6.4. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 1 del Racal 2.....	56
Tabla 6.5. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 2 del Racal 2.....	57
Tabla 6.6. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 3 del Racal 2.....	57
Tabla 6.7. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 4 del Racal 2.....	58
Tabla 6.8. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en el Racal 2	58
Tabla 6.9. Síntesis de la prueba 1A con el TP Sugerido para el Módulo en General.....	58
Tabla 6.10. Detalle de la Duración del Programa de Pruebas	59
Tabla 6.11. Cuadro Comparativo para del TP Sugerido Contra el de Producción Actual para un Mismo lote	59

Capítulo 1. Introducción

1.1. Generalidades

La línea de producción en las fabricas de ensamblaje y prueba (ATM, Assembly and Test Manufacturing) consta de alrededor de 32 procesos, los cuales varían dependiendo del producto que se trate, en la figura 1.1 se describe a grandes rasgos dicho proceso. En el primer bloque se reciben las obleas y se cortan liberando así el trozo de silicio con el circuito ya impreso que representa a cada unidad y se reconoce en este punto como die¹. Seguidamente en el proceso de ensamble se une el die con el sustrato, este último es una placa cuyas pistas interconectan las señales de entrada y salida con las terminales que tienen mayor tamaño para facilitar la conectividad por parte del usuario final en sus circuitos. Una vez que se ha unido el die con el sustrato mediante pegamentos especiales y varios procesos de cocción, y además de que se le hayan adherido capacitores que funcionan como filtros de ruido se genera estadísticamente una bifurcación en el proceso para que unas unidades pasen a una etapa de estrés, dependiendo de los resultados obtenidos por la muestra esta etapa se decide si el lote debe seguir el proceso o no, de manera que se pueda garantizar la calidad del producto. Luego se procede a la etapa de pruebas, en esta etapa todas las unidades son sometidas a pruebas de comportamiento a diferentes velocidades y temperaturas de manera que se estudia la funcionalidad del circuito. Nuevamente existe otra bifurcación en el proceso donde se separa estadísticamente un número de unidades del proceso que pasan a la etapa de validación, en esta etapa se prueba el funcionamiento final del circuito y de acuerdo a los resultados se establece cual es la confiabilidad del proceso de producción y se proyecta el posible número de unidades defectuosas de manera que se cumpla con las expectativas del cliente. Finalmente se imprime la descripción del circuito sobre la cerámica, se revisa

¹ Die: trozo de oblea en el que se encuentra el circuito de alto grado de integración de interés

visualmente que este bien, y finalmente embalan los lotes y se transportan para su posterior venta.

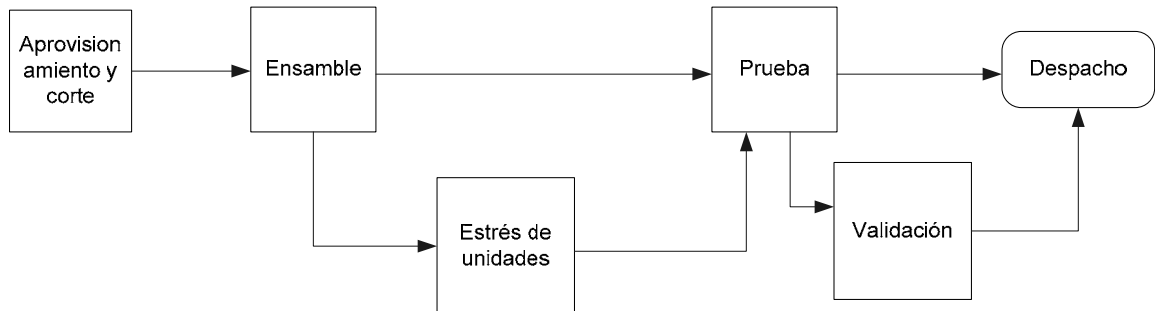


Figura 1.1. Proceso de ensamble y prueba de un dispositivo Chipset

Dichas fabricas reportan un índice de productividad, el cual tiene asignado un valor mínimo, por encima de este valor todas las fabricas se comparan y compiten entre si.

1.2. Descripción Sintética del Problema

La empresa desea gozar de la mayor confiabilidad posible en la etapa de validación de manera que cuando una unidad se rechace no se deba a problemas de la máquina que valida, ejemplo de ello desgate en los componentes o una mala calibración.

Se desea asociar de la manera más certera posible cuando una falla o serie de fallas se debe a algún parámetro particular en el dispositivo de pruebas o bien si debido al historial de la unidad se puede garantizar que la falla es propia de la unidad.

De resolver el problema se lograría actuar a tiempo cuando se detecta un fallo en el dispositivo de prueba de manera que los técnicos a cargo puedan realizar la calibración correspondiente, así como el cambio de componentes pertinente. Simultáneamente, y aún de mayor interés, es el hecho de que se evita rechazar unidades buenas, ya que los desajustes en las máquinas tienden

a ello. Adicionalmente se mejoraría la eficiencia en la reparación del equipo ya que se contaría con las alarmas pertinentes que permiten ubicar el problema más certeramente, y a tiempo.

1.3. Solución Seleccionada

Se ha decidido alterar el diagrama de flujo con el que funciona el módulo de validación, de manera que se puedan incorporar nuevas propiedades que permitan evidenciar más fácilmente la causa del rechazo de una unidad. Para alcanzar dicha confiabilidad técnica se pretende extraer la identificación de cada unidad de manera que se tenga realimentación sobre el proceso de producción y mediante estudios posteriores dar con la causa del daño de una unidad.

Simultáneamente para evitar un fallo de prueba debido a errores de contacto eléctrico o posiciones falseadas de la unidad bajo prueba, estudiar algunas variables envueltas en el proceso que condicionan la salud de la prueba o sirven como indicador para detectar un error en la misma. Adicionalmente, cuando la prueba se encuentra en progreso se pretende dar seguimiento a las pruebas de arranque ejecutadas por el BIOS, de manera que de darse una falla se cuente con el código de fallo (POST-Code), y sirva como herramienta para discernir entre fallas debidas a causa del estado del equipo o propias del integrado bajo prueba.

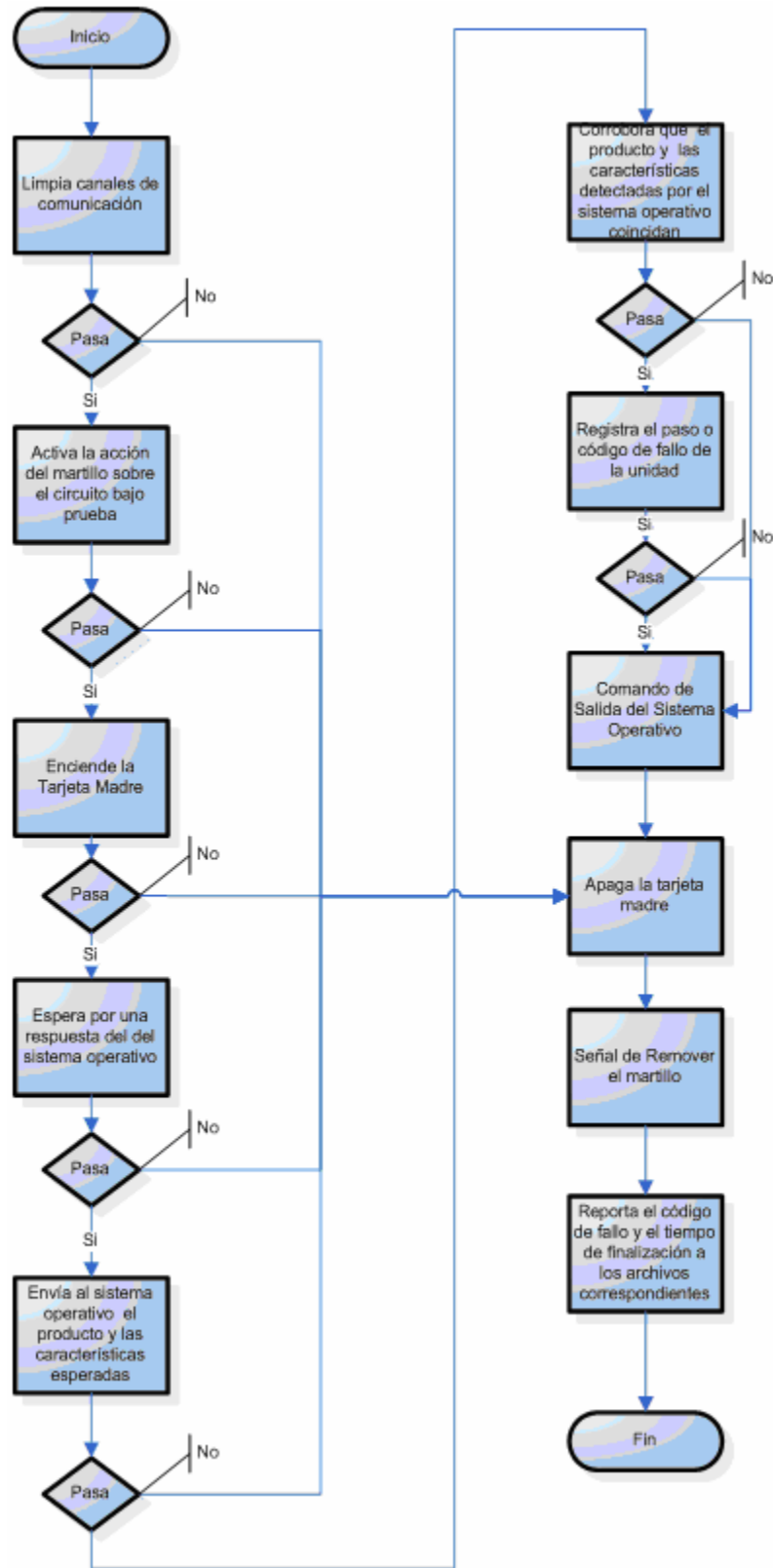


Figura 1.1. Diagrama de Flujo Actual del Módulo de Validación

Capitulo 2. Meta y Objetivos

2.1. Meta

Detectar de manera eficiente, y exacta los problemas que afecten a la producción en términos: de ubicación y atención a tiempo.

2.2. Objetivo General

Desarrollar un conjunto de pruebas eléctricas que alerte sobre el estado de los procesos mediante un análisis más exhaustivo de los errores en el producto, y consecutivamente generar sospechas sobre el estado de la máquina y confiabilidad de las pruebas mediante la extracción de otros parámetros de interés.

2.3. Objetivos Específicos

- Diseño de un algoritmo que permita la optimización y mejora del software de prueba utilizado actualmente para la validación de los productos en lo que respecta al reporte de fallos funcionales y códigos posteriores de BIOS.

Actividad: Modificar el software de prueba utilizado actualmente con el fin de que se reporten los fallos funcionales y códigos posteriores de BIOS de una manera mas especifica para así poder ubicar mas eficientemente la raíz de un problema dado.

Indicador: El software obtenido una ves implementados los cambios proyectados.

- Diseño de un software de prueba que en sus rutinas de inicialización incorpore el chequeo en tiempo real de los voltajes aplicados a los

diferentes canales del modulo de pruebas y a la vez implemente las respectivas advertencias en caso de que se presentase una falla.

Actividad: Realizar el diseño de un software que al detectar un fallo en el voltaje de alguno de los canales del modulo de pruebas presente una advertencia al usuario de las posibles consecuencias que podría acarrear el mismo.

Indicador: El software obtenido a raíz de la implementación del diseño realizado.

- Validar la posibilidad de ubicar cada unidad mediante su código de identificación de manera que permita rastrear la procedencia de la misma en el módulo de validación tal como se hace en otras partes del proceso.

Actividad: Realizar una investigación que valide las opciones de implementar en el módulo de validación la lectura de la identificación de la máquina.

Indicador: Detalle de las cambios a realizar.

- Realizar un análisis estadístico de toda la información obtenida para determinar tanto posibles fallas en el proceso de producción como el estado de los componentes de la máquina de validación.

Actividad: Recopilar toda la información obtenida de las pruebas realizadas una vez implementados los cambios en el proceso de producción, para así poder realizar el análisis estadístico de la misma.

Indicador: Las tablas con los resultados obtenidos del análisis realizado.

- Incorporación de la información obtenida a las bases de datos que genera cada prueba.

Actividad: Apegándose al formato utilizado actualmente para el reporte de la realización de las pruebas en el proceso de validación, incorporar los resultados obtenidos una vez implementados los cambios.

Indicador: Los reportes obtenidos una vez que se incorporen los cambios respectivos.

Capítulo 3. Marco Teórico.

En este capítulo se describen algunas de las herramientas y principios básicos bajo los cuales estas trabajan.

3.1. *Descripción del Equipo*

El equipo existente en la etapa de validación (CPV) consiste en el módulo **EVERTECH**, este módulo es un sistema electromecánico con un área de aproximadamente 1.25m x 2.5m y una altura de 2m. El sistema consta de 2 brazos robóticos: uno de ellos toma las unidades que se encuentran adecuadamente distribuidas sobre una bandeja especializada para dicha función, y las coloca sobre un sistema rotacional donde las unidades giran de manera que el segundo brazo robótico las tome y las coloque sobre cada una de las cabezas de prueba (TIU, Test Interface Unit). Cada TIU consiste básicamente en una tarjeta madre especializada para la colocación mecánica a presión de cada unidad. Esta tarjeta incorpora algunas características que normalmente no se encuentran en otra tarjeta ya que esta está especialmente diseñada para someter el componente Chipset a prueba.

Entre las características especiales se destaca el hecho que hay algunos códigos de estado del BIOS que son desplegados en siete segmentos, además esta tarjeta tiene puertos de prueba.

Esta tarjeta madre se comunica con el dispositivo de prueba conocido como Centrix, este dispositivo se muestra en la figura 4.1. De la misma figura se puede notar que este se encuentra colocado sobre una tarjeta de adquisición de datos, la cual se conoce como Celbridge, que es precisamente dicha tarjeta la que tiene salidas tanto a la tarjeta madre de pruebas como a la unidad de control central en la cual se programan las pruebas requeridas. La unidad Centrix puede generar tanto señales digitales como señales analógicas, adicionalmente esta unidad puede mostrar señales, lo cual sirve de lectura de diferentes niveles de tensión en los puntos designados de la tarjeta madre. Los

resultados se transmiten mediante puerto serie a la unidad de comando de pruebas.

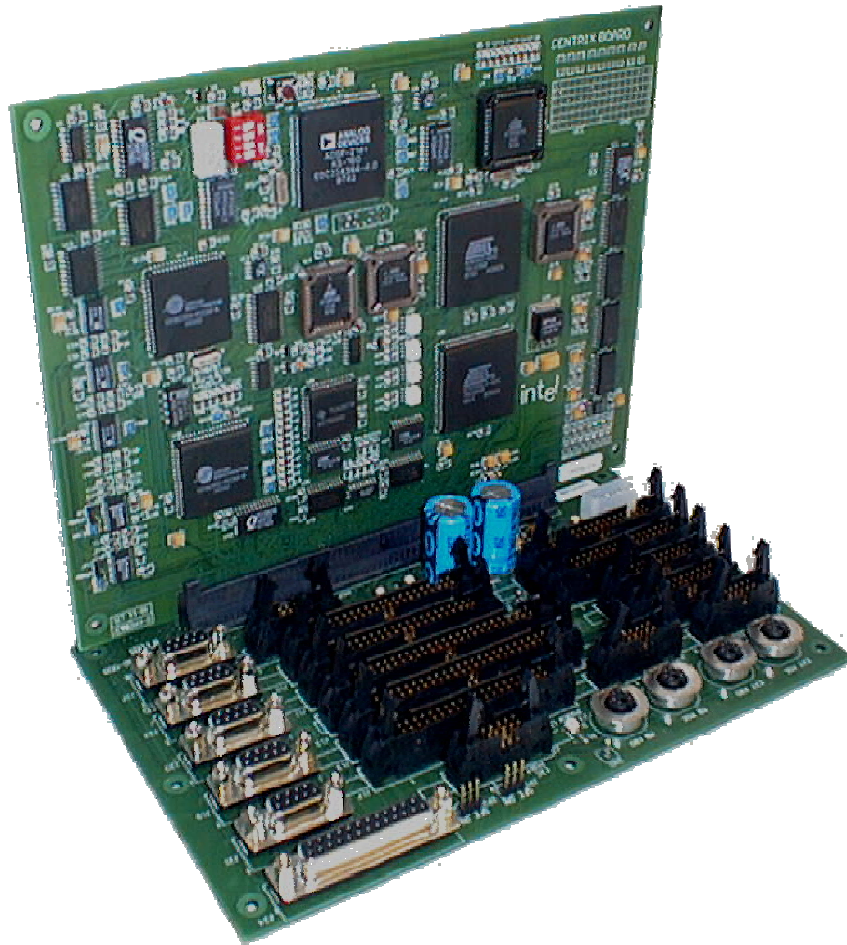


Figura 3.1. Dispositivo de Control de Pruebas Eléctricas. En la fotografía se muestra la tarjeta Centrix, en posición vertical, y la CelBridge en posición horizontal

Es importante mencionar los conceptos de BIOS y POST, ya estos acrónimos se mencionan constantemente en este documento.

Se necesita aclarar en primera instancia que el significado de BIOS es sistema básico de entrada/salida y su nombre se deriva de su significado en

ingles (Basic Input/Output System), este es un grupo de programas que están permanente guardados en un integrado de la tarjeta madre.

Al momento de realizar la inicialización, el BIOS inicializa y prueba el sistema, durante la ejecución, el se encarga de proveer el acceso a dispositivos del sistema que el sistema operativo y ciertos programas puedan necesitar.

El BIOS interactúa directamente con el hardware, es el software de capa mas baja en una PC y funciona como una interfase entre el hardware y las otras capas de software, aislando las anteriores de los detalles del hardware. Maneja interfases de bajo nivel con dispositivos como el teclado, video, impresoras, discos y diskettes. Es generalmente asociado con dispositivos de entrada y salida y contiene drivers para manejar sus tareas relacionadas con el hardware.

El BIOS también contiene rutinas de interrupción del sistema, tales como la interrupción del teclado, la del timer, la del reloj en tiempo real, etc.

En cuanto al auto examen al iniciar que llamaremos POST por su nombre en ingles (Power On Self Test) cabe mencionar que es un programa de diagnostico incorporado en el sistema. El POST inicializa componentes instalados en el sistema, prueba el hardware del sistema, detecta la presencia o ausencia de dispositivos, realiza reportes de de la configuración del sistema así como de el resultado de el diagnostico realizado y también carga un sistema de operación.

En el caso de que el sistema sea reiniciado, el CPU busca y ejecuta la primera instrucción, de una dirección específica, esta primera instrucción es típicamente un salto al POST, para que este sea ejecutado.

El POST detecta, chequea e inicializa los componentes instalados, realiza una serie de pruebas para determinar si la máquina es funcional.

El POST chequea los registros del CPU, el controlador de teclado, el temporizador de intervalos programable, los integrados DMA, el refrescamiento de la RAM, inicializa los DMA esclavo y maestro, carga los vectores de interrupciones, entre otros.

Es importante mencionar que los dispositivos PC modernos tienen un ROM-BIOS guardado en una memoria integrada FLASH, el cual contiene el POST y el BIOS.

La PC tiene un sistema operativo en capas, la jerarquía funcional de lo anterior se puede observar en la siguiente figura:

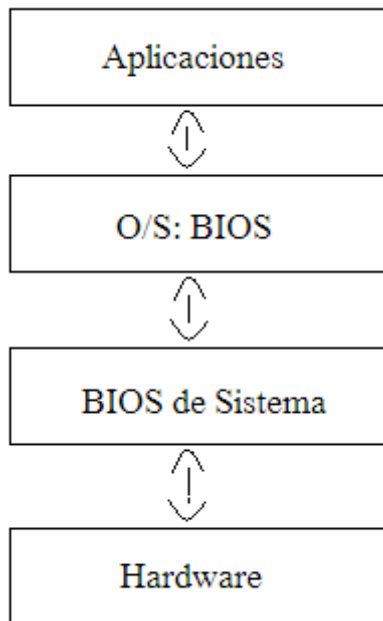


Figura 3.2. Jerarquía funcional de un sistema operativo por capas.

El sistema operativo realiza operaciones a un nivel mas alto y maneja la organización lógica del sistema. En la figura se presenta como el BIOS realiza operaciones de entrada y salida dirigiendo directamente los registros de hardware de los dispositivos integrados.

El System BIOS es llamado también la porción residente por el fabricante en cada PC, por otra parte se nota una segunda parte del BIOS que es llamada no residente, y que es leída al DRAM del disco cuando la PC carga el O/S.

3.2. Generalidades Técnicas

A nivel de validación se distinguen dos niveles de pruebas: unas de carácter eléctrico/lógico y otras de comportamiento.

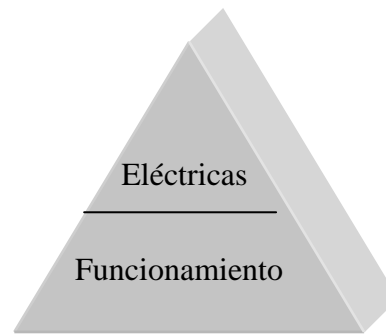


Figura 3.3. Estratificación de las pruebas realizadas en el módulo de validaciones

Las pruebas eléctricas permiten incorporar funciones predefinidas de la tarjeta Centrix tales como medición de tensión, frecuencia o temperatura, también se puede definir el valor de tensión de las salidas analógicas o el valor numérico de salida de las señales digitales con el fin de dirigir la prueba. Por ejemplo las pruebas coordinan el prensado mediante la activación de un optoacoplador, con el encendido de la tarjeta madre mediante la activación de otro, dichas acciones son generadas mediante señales de la tarjeta Centrix.

De esta tarjeta se han explotado algunas funciones que son las que se le han denominado de comportamiento, esto para motivos de este documento, ello debido a que la tarjeta Centrix puede transmitir comandos simulando un teclado serial hacia la tarjeta madre y así cargar algunas instrucciones al sistema operativo. Estas instrucciones corren escritas, algo que es común realizar en sistemas operativos como UNIX y LINUX, con lo cual, estos escritos se encargan de simular el comportamiento convencional del Chipset. Entre estas pruebas hay de manera que se pueda demostrar la clasificación del Chipset.

3.3. Distribuciones y Conceptos Estadísticos

Una distribución es un patrón de variación en un conjunto de datos. Las distribuciones se describen por su forma, centro y dispersión. La forma nos dice si la distribución es simétrica, si hay más de un pico, o si hay excepciones. El centro dice los valores típicos, comúnmente medidos por la media o la mediana. La dispersión dice cuanta variación se espera, típicamente medida por la desviación estándar o el rango.

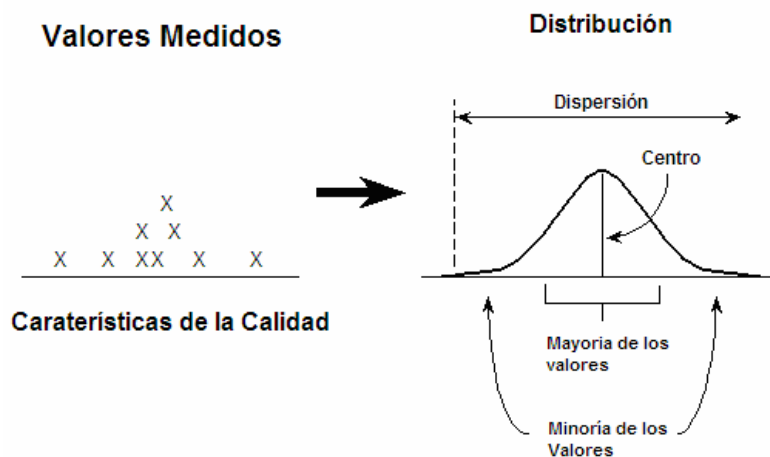


Figura 3.4. Aproximación de una Distribución según los Datos Tomados

Se distinguen dos tipos de distribución, la simétrica y la asimétrica

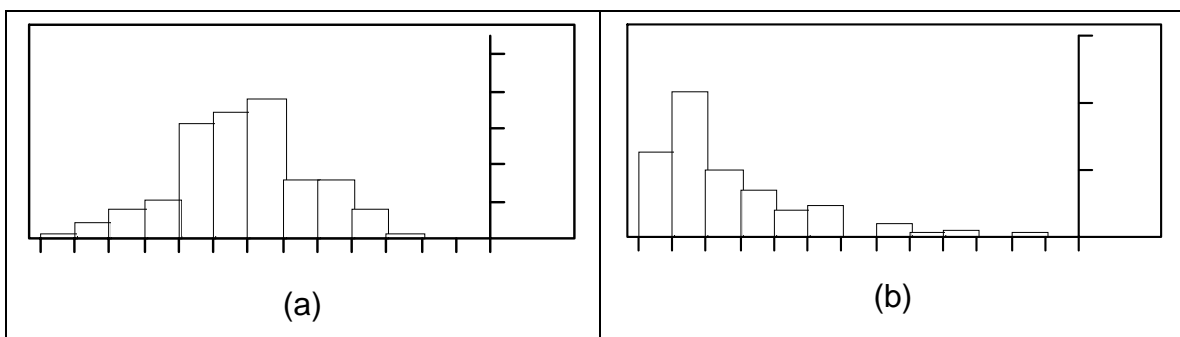


Figura 3.5. a) Simétrica, b) Asimétrica izquierda

Moda se define como el valor de mayor ocurrencia en un conjunto de datos. Modalidad se refiere a un número de picos en un conjunto de datos.

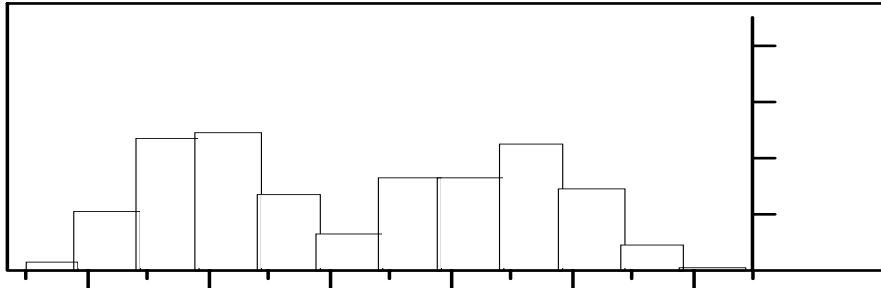


Figura 3.6. Conjunto de Datos en un Histograma que Evidencian una Distribución con dos Modas (Bimodal)

Las excepciones son observaciones en el conjunto de datos que caen fuera del resto de la distribución. Una vez determinadas las excepciones se debe dar explicación al porqué estos datos son diferentes.

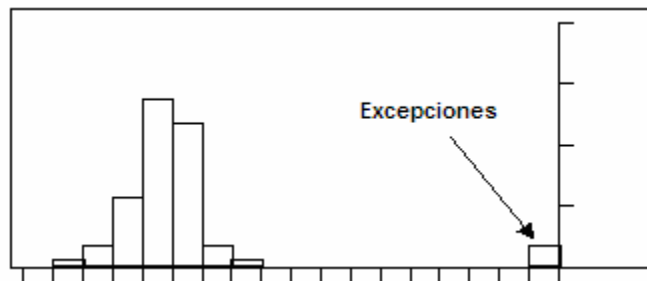


Figura 3.7. Distribución de datos donde se dan algunas excepciones a considerar

La mediana es el punto central donde habría un balance geométrico sobre el plano de una distribución estadística. Mientras que la media es el punto donde el 50% de los datos se encuentran sobre este y el otro 50% por debajo del mismo.

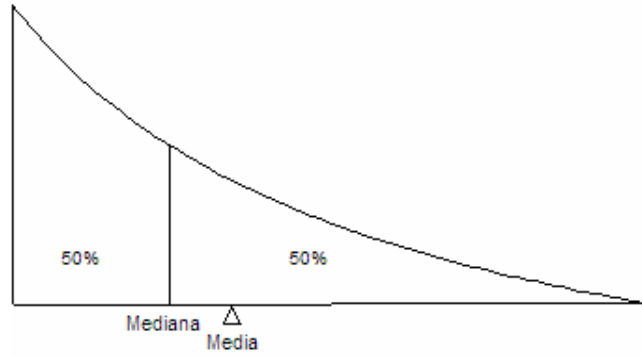


Figura 3.8. Representación de la Diferencia entre la Media y la Mediana en una Distribución

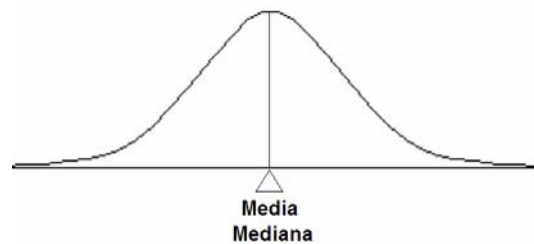


Figura 3.9. Representación de una Distribución simétrica

Si la distribución es simétrica la media y la mediana se muestran próxima una de la otra.

Entre los gráficos utilizados para las distribuciones estadísticas se encuentra la caja estadística, en esta los extremos de la caja se representan por los cuartiles Q1 y Q3. El centro del diamante es la mediana, y la línea central la media. Los extremos se encuentran a 1.5 del grosor de la caja, representados por líneas verticales, donde si un punto se encuentra fuera de esta línea se considera una potencial excepción.

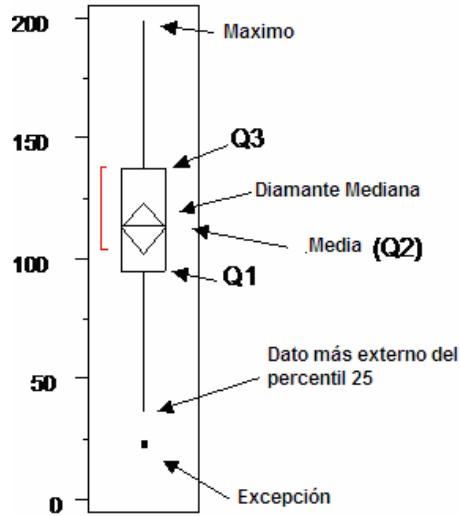


Figura 3.10. Descripción de la Caja Estadística

3.4. Cualidades Necesarias para la Lectura del TESTINB.

El Chipset como circuito interno lo que tiene en lugar de JTAG es otra lógica digital que se conoce como TESTINB, esta se centra en un único pin exclusivo para pruebas y se apoya en otras señales para coordinar su funcionamiento.

Para inicializar un ciclo de TESTINb un dato válido debe ser ingresado en el registro de control a través del pin TESTINb. Este registro se encuentra dividido en dos secciones, estas se conocen como de comando y de índice. La idea es que cuando el comando introducido indique que se debe realizar una instrucción de lectura o escritura, la sección de index indicará al registro que se utilizará como fuente o destino. Los bits [12:10] indican el tipo de operación, los tres tipos son leer de los registros, escribir a los registros, o solicitud de acceso a una nueva región donde se encuentra registros de funcionamiento especial. El modo de pruebas soporta el acceso a múltiples registros a través del bit de habilitación GBTMEN (GloBal Test Mode ENable). Cuando este bit se encuentra en bajo el usuario puede programar múltiples registros sin que ninguna prueba sea invocada, con lo cual cuando se introduce la última escritura y se pone en alto el bit GBTMEN los registros previamente escritos surtirán efecto simultáneamente.

Tabla 3.1. Tabla de Descripción del Registro de Pruebas

Campo de Bit	Valor defecto y Acceso	Descripción
[17]	0b	Debe ser 0
[16:11]	000000b Sólo escritura	Index (selección de 64 registros) 000000 – Reg0 000001 – Reg2 111111 – Reg63
[10:0]	000h Sólo escritura	Comando: [10:9] – 01 Leer 11 Escribir [8] – HTunit requisición [7:2] – Reservado [1] – Habilidad de la carga [0] – Habilidad del modo global de pruebas

La activación del modo de pruebas empieza cuando se detecta un flanco negativo en el pin testinb, seguido por 18bits de datos, mas un bit de parada. El flanco negativo activa la máquina de estados, y la información es introducida al registro serie de control. El bloque de control contiene una máquina de estados y varios bloques lógicos, los cuales regulan las operaciones de leer y de escribir.

Para motivos de la lectura luego del bit de inicio se introducen 18 bits de datos. Seguidamente se libera el pin de TESTINB, seguidamente, la información solicitada será liberada serialmente a través del TESTINb. Todos los Registros de Prueba están conectados a un MUX 32X1 cuya selección se encuentra interconectada a la sección INDEX del registro de control. La información se

carga en un registro temporal de 32 bits de manera paralela, finalmente, es expulsada serialmente hacia el pin TESTINb.

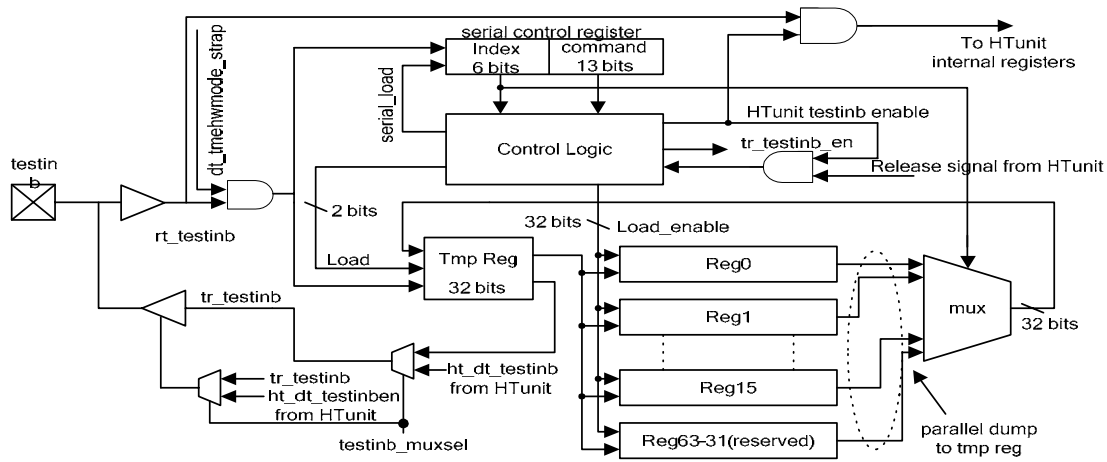


Figura 3.11. Diagrama Esquemático de la Lógica Envuelta en el Funcionamiento del TESTINB

La máquina de estados regula las operaciones de leer y escribir. Hay seis estados: IDLE, LOAD_CMD, READ_TMP, WRITE_TMP, SHIFT_OUT, y HTUNIT_CNTL. La máquina de estados se mantiene en el estado IDLE hasta que se detecte la caída de tensión en el pin testinb. Luego de que se detecta, la máquina de estado cambia a LOAD_CMD, y permanece en este estado hasta que 19 bits sean cargados serialmente en el registro de control. Esta situación se garantiza con la implementación de un contador, el cual evitará a la máquina de estados que vaya a otro estado hasta que se haya alcanzado la cuenta de 19. Luego de ello existen cuatro posibles transiciones. La primera es de vuelta a IDLE. Ello puede ocurrir si un comando incorrecto es cargado. La segunda transición posible es del estado LOAD_CMD a WRITE_TMP, en este, los datos ingresados serialmente (desde el testinb) pasa al registro de Tmp de 32 bits, y el contenido del registro Tmp pasa es almacenado en el registro indicado por la sección de Index del registro de control cuando la máquina de estado pasa del estado WRITE_TMP a IDLE.

La tercera transición es de LOAD_CMD a READ_TMP. Este estado permite que los datos del registro de pruebas seleccionado por el Index sea cargado en el registro Tmp mediante un bus de 32 bits. Además permite poner en tercer estado al testinb. En el siguiente flanco, la máquina de estados pasa a SHIFT_OUT. En este punto el controlador de salida es activado, y los 32 bits de datos en el registro de lectura son sacados serialmente a través del pin testinb. Esta operación es garantizada mediante el uso de un contador, lo cual prevendrá a la máquina de estados de transitar a IDLE hasta que la cuenta de 32 se haya alcanzado. Una vez que toda la información ha salido, la máquina regresa a IDLE. En la transición de SHIFT_OUT a IDLE el controlador de salida es deshabilitado. En este punto otra operación de lectura/escritura puede ser llevada a cabo a través del pin testinb.

La cuarta transición de LOAD_CMD es hacia HTUNIT_CNTL. Este estado permite al HTunit, tener control completo del pin testinb. Una vez en este estado la máquina de estados espera la señal que libera del control sobre el testinb de HTunit devuelta al DTunit, esta señal debe durar cinco ciclos de reloj, posteriormente la máquina de estados transita al estado IDLE. La HTunit puede no necesariamente remitir una señal de liberar a la DTunit si los registros TM ya no necesitan ser programados.

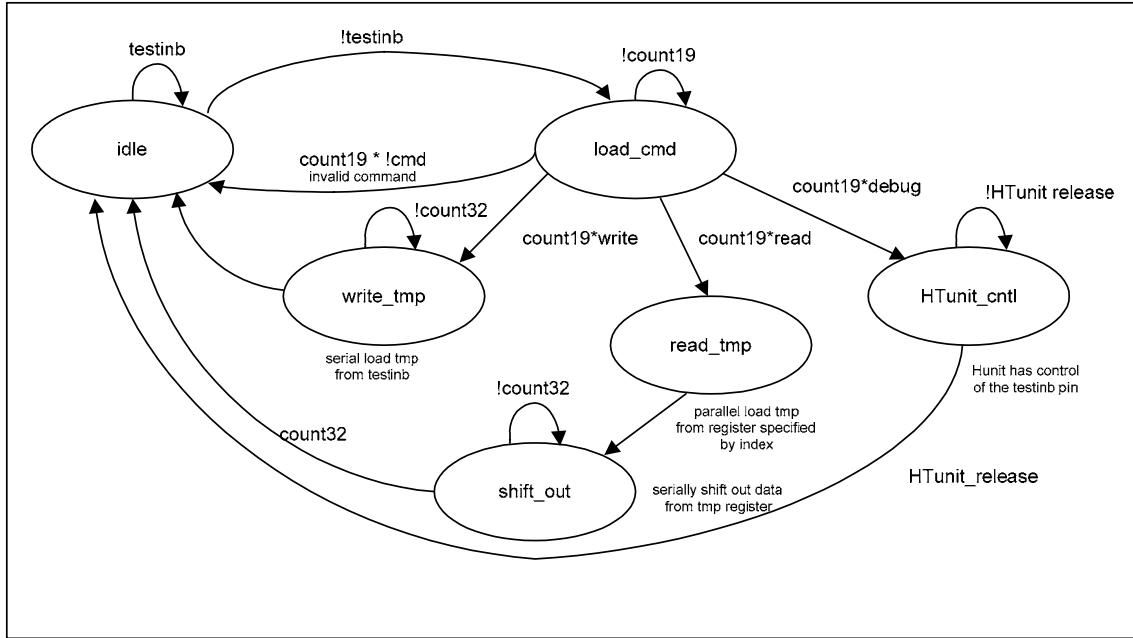


Figura 3.12. Diagrama de Estados de la Lógica del TESTINB

Existen dos maneras de leer los fusibles, el primero y más utilizado involucra realizar la tarea a través del TESTINB ingresando al modo de prueba y posteriormente localizando los fusibles de interés mediante el FSB. El segundo es a través de los registros de configuración a través de ciclos del FSB.

Los pines del FSB para las operaciones de prueba se interpretan de la manera mostrada en tabla 3.1.

Tabla 3.1. Interpretación de las Señales del Bus Frontal en Modo de Prueba.

FSB Pins	Direction	Description	DT Output Entity
H_DB_0	Input	Global Program Enable	dt_pf_pgmen
H_DB_1	Input	Local Program Enable	dt_pf_pgmenin
H_DB_2	Input	Program Enable Clock	dt_pf_pgmenclk
H_DB_3	Input	Data In	dt_pf_datain
H_DB_4	Input	Data In Clock	dt_pf_dataclk
H_DB_5	Input	Sense	dt_pf_sense
H_DB_6	Input	Favor 0	dt_pf_favor0
H_DB_7	Input	Favor 1	dt_pf_favor1
H_DB_8	Input	ALT0	dt_pf_alt0
H_DB_9	Input	ALT1	dt_pf_alt1
H_DB_10	Output	Program Enable Out	dt_fusepgmenout
H_DB_11	Output	Data Out	dt_fusedata

El primer método consiste en invocar el modo de pruebas y utilizar el FSB para propósitos de lectura de los fusibles. Esto se hace acertando la señal “sense” (h_db[5]) luego de que se ha ingresado al modo de prueba la información se extrae serialmente. Este método se hace principalmente con la programación de los fusibles, de manera que se garantice que lo programado se lee correctamente. El segundo método es mediante la configuración de registros mediante ciclos del FSB.

Método 1

Los siguientes pasos detallan la secuencia requerida para los fusibles de Calistoga. Estos pasos se hacen comúnmente luego de programar los fusibles, para corroborar si estos se programaron correctamente.

1. Invocar el modo de pruebas. Esto se hace mediante el pin de testinb, mediante la escritura del bit 23 del registro 01 en el Test Mode Entry de Calistoga.
2. Acertar la señal "sense" para leer la información programada. Reiniciar datain_clk y empezar la salida de datos a través del pin "data_out" (h_db[10]) del FSB.

Método 2

La segunda manera en la que pueden ser leídos los fusibles es a través de los ciclos de configuración del FSB. Durante el reset la información paralela de los bloques de fusibles es captada y escrita en los registros de los bloques DT y HT correspondientes. Debido a ello, para leer los valores de la información fuseada, se debe identificar las direcciones de Offset de los registros o el registro requerido y utilizar la configuración del FSB para acceder al valor almacenado.

Existen varios registros de estado en la unidad DT que capturan los valores de los fusibles durante la secuencia de arranque. Tal como se muestra en la figura 3.13, existen 3 registros TME que almacenan los valores de los fusibles para todos los fusibles tanto de capacidad (DIFF) como de calificación (SE). La lectura de estos fusibles se puede hacer a través del pin Testinb o el protocolo de acceso a registros a través del FSB.

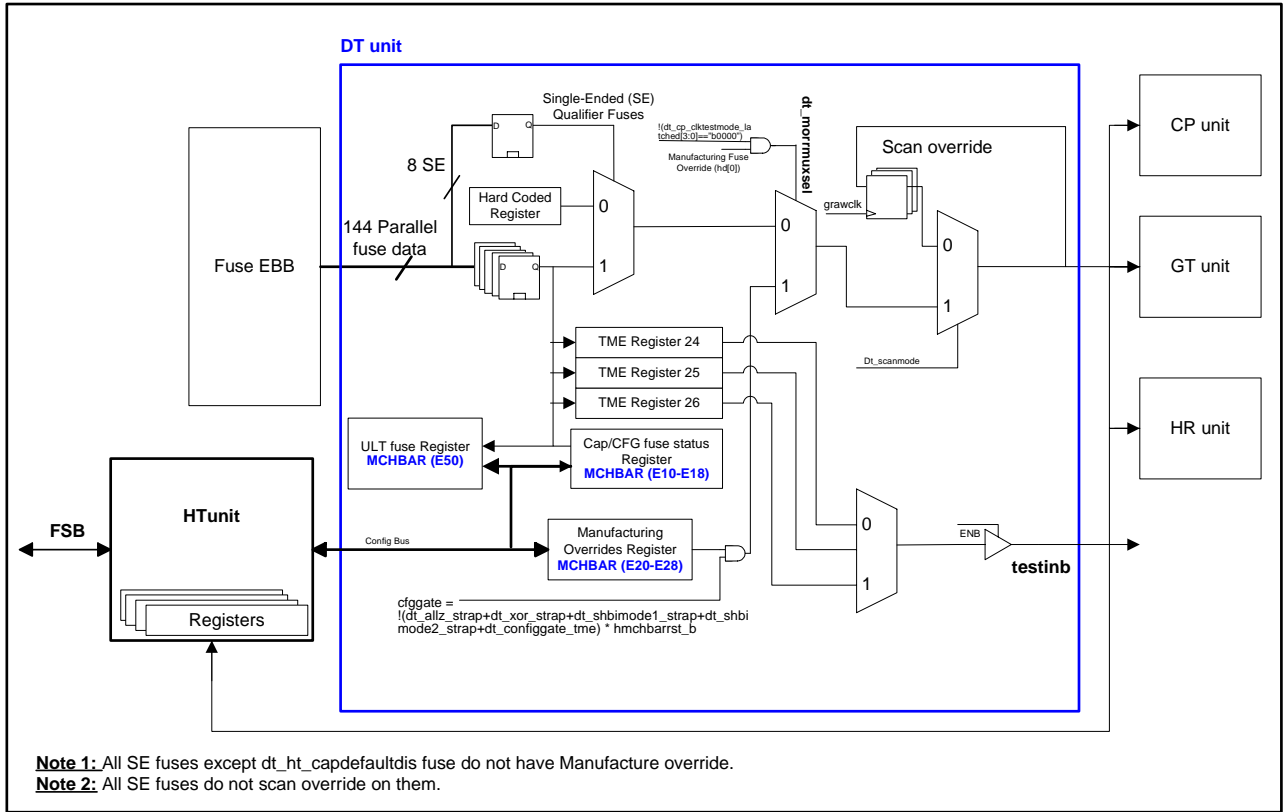


Figura 3.13: Diagrama de Bloques de del Acceso a los Fusibles

Capítulo 4. Metodología

En el presente capítulo se describe cual es la manera de proceder bajo la cual se ha logrado detallar el problema y realizar el análisis adecuado hasta implementar una solución viable.

4.1. Procedimiento Metodológico

- 1) Investigación y capacitación ha sido parte importante para el desarrollo del proyecto. Ello involucra conocer la labor de la maquinaria a nivel de usuario para la operatividad de la solución, así como a nivel técnico para sugerir mejoras, evaluar las posibilidades, y tener criterios aptos para considerar modificaciones. Dado el profundo carácter de programación que envuelve el módulo, es importante mantener contacto con los ingenieros de producto encargados de la confiabilidad en términos de productividad y calidad, ello realizado mediante la elaboración de los programas de prueba con los que son probadas cada una de las unidades.
- 2) Análisis de la situación en variables medibles. Una vez que se conocen los índices utilizados para detectar la productividad se busca incidir sobre ellos mediante la correlación sobre una variable física controlable con la que se pueda variar el comportamiento de estos.
- 3) Desarrollo de programas de pruebas, junto con experimentos que permitan corroborar el funcionamiento esperado del módulo de validación.
- 4) Análisis de los resultados obtenidos a nivel estadístico para demostrar que existe impacto mediante el nuevo flujo desarrollado.
- 5) Documentación y distribución de los alcances del proyecto.

4.2. Detalle de Actividades

1. Familiarización con el proceso de producción, y herramientas de ingeniería y estadística que se utilizan para el control de la calidad, productividad, y control de fallas dentro de la empresa. Adicionalmente certificación en lo concerniente a la accesibilidad de los servidores de interés, así como capacitación sobre la ubicación y manipulación propia del sistema operativo sobre el cual estos trabajan.

Explicación:

Se pretende evaluar la situación actual de la empresa, sin la implementación de las mejoras, y posteriormente, cual es el impacto una vez que las mejoras son implementadas.

En este punto cabe explicar que en INTEL existen herramientas estadísticas que se utilizan para generar gráficos y tablas de casi cualquier variable que se desee con respecto a condiciones de ubicación temporal, espacial, sector de manufactura o prueba, o algún parámetro físico deseado en conjunto con cualquier otro criterio de interés. Ello permite sintetizar información relevante para los diversos estudios a los que constantemente se somete la empresa.

En cuanto al acceso de la información, en INTEL hay un gran número de servidores a nivel mundial, algunos de los cuales tienen acceso restringido consecuentemente se requiere alguna justificación para autorizar su acceso, debido a ello es necesario ubicar cuales son los servidores a los que se requiere acceso, y obtener los permisos respectivos.

Además, para la manipulación de la información es necesario conocer el conjunto de instrucciones que permiten el filtrado requerido dado el sistema operativo con que trabaja el servidor.

2. Entrevistas con ingenieros que conocen del proceso de prueba y validación, así como con ingenieros que han llevado a cabo proyectos que tienen alguna vinculación con el tema. Simultáneamente compilación de toda la información técnica posible que amplíe el panorama de posibilidades y delimite los alcances de una manera predefinida de proceder.

Explicación:

En lo que concierne a experiencias previas relacionadas con el proyecto, se sabe que en el área de TEST la lectura del ULT de cada unidad es posible, por lo tanto conviene indagar cómo se logra obtener esta información con las personas a cargo. Además se sabe que el ingeniero Richard Marín ha logrado la lectura mediante un dispositivo externo pero de un producto muy diferente que es el CPU.

Toda compilación de información es gran ayuda para el progreso del proyecto. En cuanto a la formación teórica referente al Chipset, así como con la tarjeta madre, permite enmarcar dentro del panorama de factibilidad el discernir la mejor manera de hacer las cosas, y consecuentemente proponer alguna de ser el caso.

3. Esquematización del proceso de pruebas en la etapa de validación.

Explicación:

Existen diferentes niveles de prueba en la etapa de validación: se distinguen las pruebas eléctricas, y las pruebas de comportamiento, en esta última se espera una respuesta predeterminada frente a una excitación convencional. Básicamente lo que se hace es correr un escrito de LINUX, sistema operativo de inicio de cada cabeza de pruebas, el cual

se encarga de devolver cierta información de interés. Las instrucciones y escritos son llamados mediante el puerto serie, y la información de interés es devuelta mediante el mismo. No hay pruebas de respuesta ante diferentes niveles de tensión generados por una fuente, ni pruebas de temperatura.

Conociendo la programación de escritos en LINUX, es necesario estructurar, documentar, y comprender, la manera en que dichos escritos hacen uso de si mismos, este proceso se plantea como parte de la metodología debido a la complejidad y cantidad de escritos envueltos en el desarrollo de las pruebas.

4. Relación entre el ensamblado de la solución de lectura con la capacidad de obtener dicha lectura mediante el sistema operativo.

Explicación:

Es necesario conocer la estructura de un Chipset y algunas de sus funciones a nivel de lenguaje ensamblador, de manera que se evalúe la factibilidad de obtener la información con las propiedades de hardware que se manejan.

5. Destacar los resultados del proyecto de manera explicativa como cuantitativa.

Explicación:

Analizar el desempeño de las pruebas, en especial si dichas pruebas van a ser acogidas por el proceso de producción. Ello implica analizar la duración de la prueba de manera que afecte en el menor grado posible la producción, así como que el engranaje que implique la prueba

en el sistema no vaya a afectar otros resultados esperados del sistema de pruebas.

Capítulo 5. Descripción Detallada de la Solución

5.1. Correlación entre la Variable Tensión de la Tarjeta Madre con el Fallo de una Prueba por parte una Unidad

Una vez estudiados los recursos de medición de variables del módulo de pruebas de validación se ha optado por estudiar la relación existente entre la tensión suministrada a la tarjeta madre, y la posibilidad de que debido a la tensión reportada la unidad vaya a fallar. Se recurre a un análisis estadístico que permita relacionar el registro de una tensión dada con la posibilidad de que dado dicha tensión la prueba vaya a ser fallida o no.

La unidad de validación mide tensiones discretas en la tarjeta madre (MB), por consiguiente luego de un filtrar los datos de producción se ha obtenido que la distribución de incidencia es la que se muestra en la figura 5.1. Se puede notar que el comportamiento se aproxima a una distribución normal.

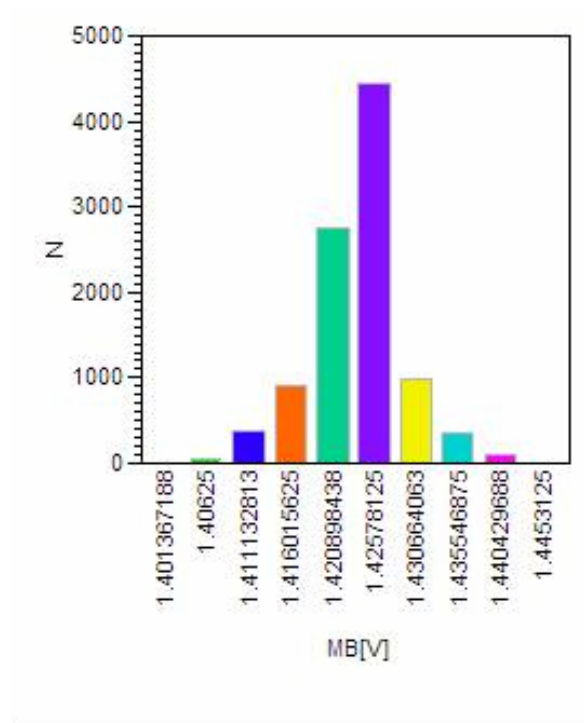


Figura 5.1. Distribución de la Tensión en la Tarjeta Madre de Pruebas

Si se analizan las unidades que pasan, de acuerdo a la tensión, se observa la distribución de la figura 5.2, en esta se puede notar que se repite la forma de la figura 5.1, esto es, el porcentaje de validación de unidades en las diferentes tensiones que se miden es similar en cada una de ellas.

A la hora cuantificar el porcentaje de unidades válidas en función de la tensión se registran los resultados mostrados en la tabla 5.1, donde si se grafica la tensión contra el porcentaje de unidades válidas se observa el comportamiento mostrado en la figura 5.3.

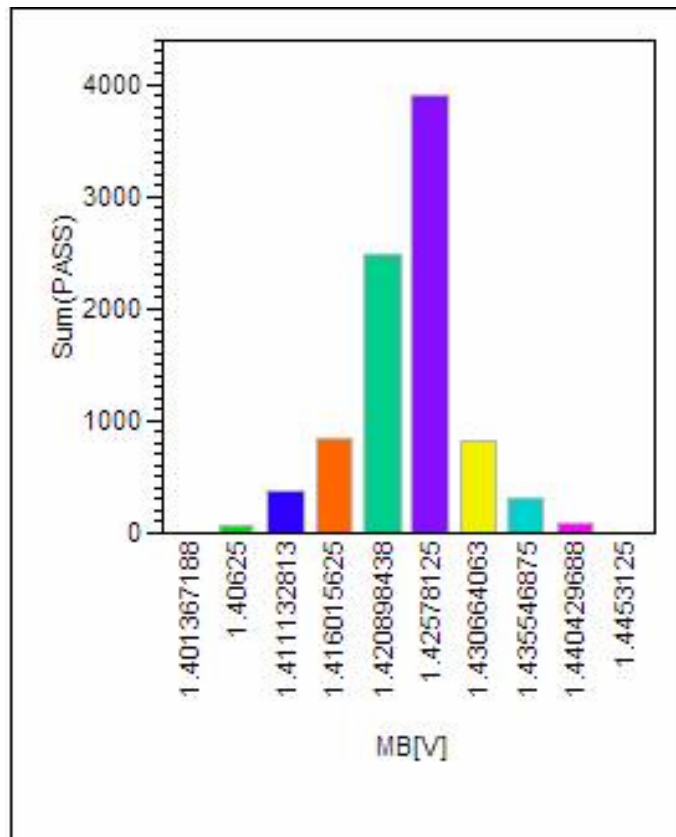


Figura 5.2. Distribución de la Tensión en la Tarjeta Madre de la Unidades que Pasan

Tabla 5.1. Cuadro Resumen de las Tensiones Encontradas

Tensión Tarjeta Madre	Total Encontrado	Unidas Válidas	Pocentaje de unidades válidas	Frecuencia
1.401367188	7	7	100	0.00069238
1.406250000	71	69	97.1831	0.00702275
1.411132813	391	381	97.44246	0.03867458
1.416015625	932	865	92.81116	0.09218595
1.420898438	2770	2498	90.18051	0.27398615
1.425781250	4456	3920	87.97127	0.44075173
1.430664063	989	842	85.1365	0.09782394
1.435546875	372	325	87.36559	0.03679525
1.440429688	121	102	84.29752	0.01196835
1.445312500	1	1	100	0.00009891

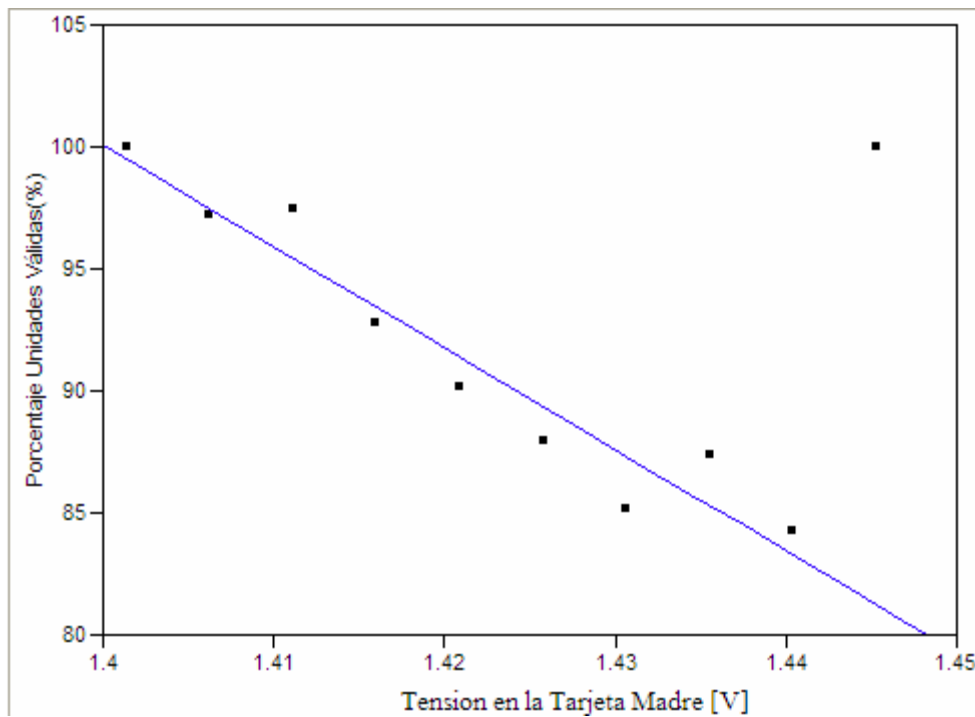


Figura 5.3. Tendencia de las unidades validadas como buenas contra la tensión medida en la tarjeta madre

La figura 5.3 es muy importante ya que se puede correlacionar la tensión en la tarjeta madre con el porcentaje de unidades validadas, la línea presente en el gráfico es la recta de menor ajuste, no se contempla el punto de mayor tensión (una unidad). De este gráfico se puede inducir cual es la probabilidad de que dada una tensión en la tarjeta madre la unidad vaya a fallar.

El sesgo de los datos se debe a lo que se conoce como sesgo del equipo, a nivel digital, esto es la discretización debida a los intervalos de detección y representación posibles del medio de medición. Razón por la cual el registro de tensiones se agrupa tal como se muestra en la tabla 5.1.

Si se observa la frecuencia de incidencia de cada tensión se puede notar que el 85% de las pruebas registran tensiones por debajo de 1.43V, con lo cual el 15% restante al estar por encima de esta tensión y al asociarlo con la figura 5.3, se puede notar que sobre esta tensión la probabilidad de éxito de pasar la prueba disminuye.

La recta de mejor ajuste para los puntos del gráfico de la figura 3 corresponde a la ecuación siguiente:

$$Y = 681.69 - 415.45 X$$

Y: Probabilidad de éxito (%)

X: Tensión en la tarjeta madre (V)

También es importante analizar el comportamiento de las tensiones por TIU, lo cual se muestra en el gráfico de variabilidad de la figura 5.4. La línea punteada representa el promedio total de todas las tensiones, la línea verde es el promedio en cada racial, y debajo de este se grafica la desviación estándar en cada subgrupo.

Se puede notar que los promedios por grupo de módulo no se desvían del promedio general, ni entre ellos, además la desviación estándar es relativamente la misma, esto en parte garantiza el funcionamiento estándar esperado de cada TIU con lo cual las consideraciones que se tomen para una cabeza se pueden

aplicar a las demás. Adicionalmente se puede observar que en el promedio de las tensiones no alcanza en ningún momento 1.43V.

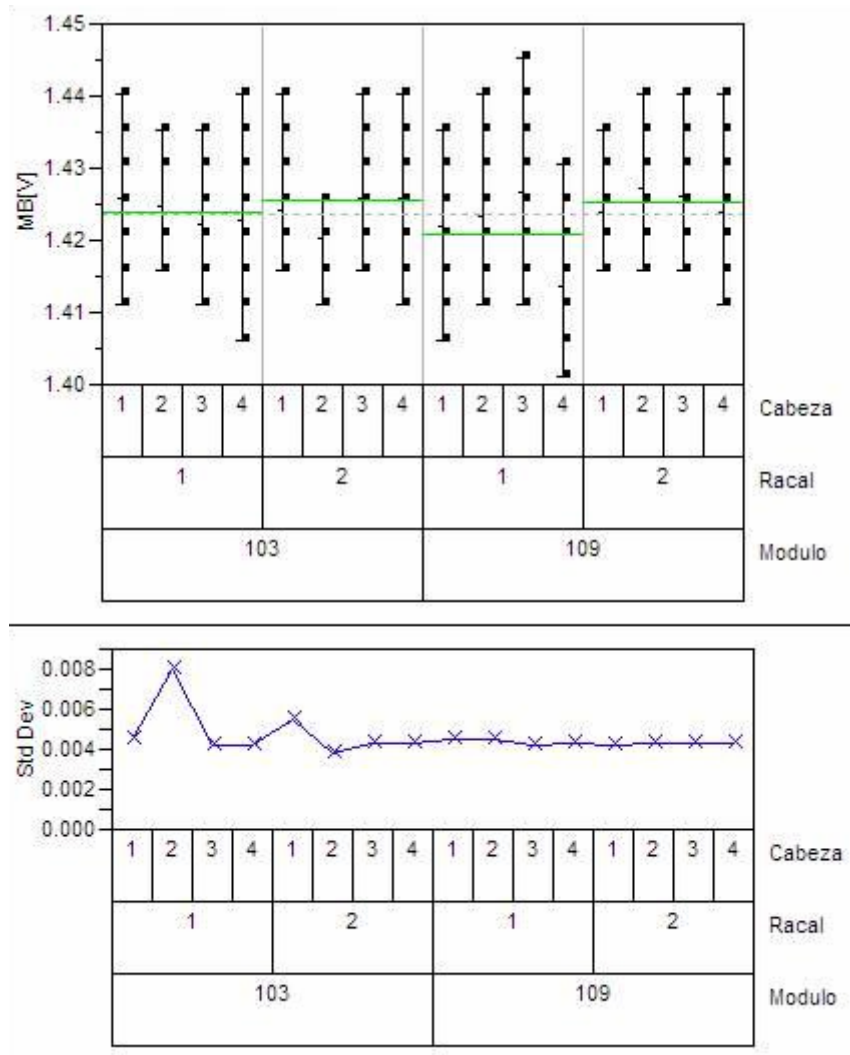


Figura 5.4. Grafico de Variabilidad según el Módulo, Ramal y Cabeza.

La figura 5.5 agrupa bajo la identificación de cada cabeza y tensión discretizada el resultado de cada porcentaje de unidades validadas como buenas. De aquí es interesante notar que es mas influyente el estado de la cabeza que la tensión que se ha medido. Aún así sin descargar los resultados debidos a la tensión de la figura 5.4 se puede notar que cuando un promedio de tensión de una cabeza se encuentra por debajo del promedio de tensión general

el porcentaje de unidades validadas como buenas de la figura 5.5 será mayor. Por ejemplo obsérvese el modulo 109, racal 1 y cabeza 4 de la figura 5.4, nótese que el promedio se encuentra por debajo del promedio general, ahora refiérase a la figura 5.5 y refiérase a la cabeza M109R1C4, donde se puede notar que el porcentaje de unidades validadas como buenas de 98%. Otro caso es en el sentido inverso, nótese de la figura 5.5 la cabeza M109R2C2, donde el porcentaje de unidades validadas como buenas es de 65%, esto es muy bajo, seguidamente de referirse al módulo 109, racal2 y cabeza 2 de la figura 5.4, se su puede notar que su promedio de tensión se encuentra por encima de la media de tensiones general con lo cual se cumple con lo esperado y estipulado por el gráfico de la figura 5.3 donde se muestra que a las tensiones mayores registradas la probabilidad de fallo del integrado bajo prueba es mayor.

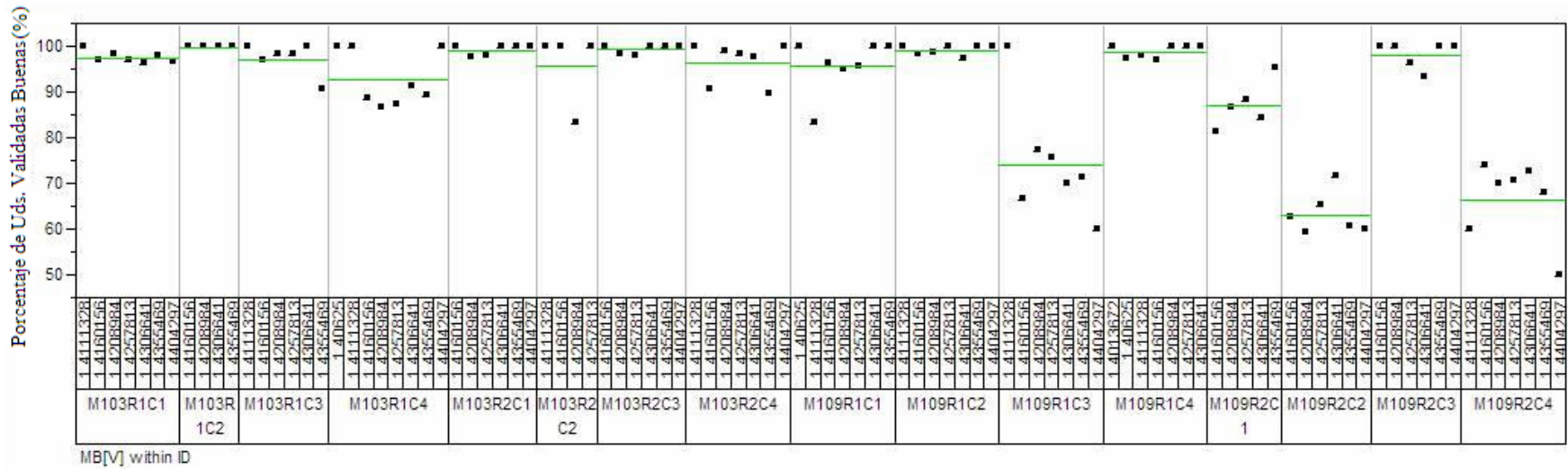


Figura 5.5. Gráfico de Variabilidad según con la Identificación de la Cabeza y las Tensiones Medidas.

5.2. Solución Seleccionada

Sustentado en el análisis estadístico expuesto en la sección anterior, se ha decidido modificar el diagrama de flujo del programa de pruebas de manera tal que sustentándose sobre la probabilidad de falla el sistema pueda discernir a partir de la tensión medida si es conveniente realizar la prueba o mejor se reinicia, esto considerando que debido a una mala colocación del integrado, o la presión insuficiente, o algún otro inconveniente mecánico, la probabilidad de falla es mayor.

En la figura 5.6 se muestra el diagrama de flujo a implementar. El primer bloque adicionado es el que mide la tensión y con ella se le asocia una probabilidad de éxito a la prueba.

El promedio para una muestra de 10110 datos es de 1.4238V, pero la tensión de mayor incidencia es de 1.4257V, y con la intensidad de que el sólo el 15% de las unidades tengan que recurrir a una actuación múltiple se coloca el límite de tensión a 1.4258V. Así cuando se detecta que el valor registrado de tensión supera dicho valor se sospecha que la probabilidad de éxito será menor, con lo cual se procede a calcular el promedio de 60 mediciones, y si estas superan el promedio preestablecido, 1.4238V, entonces se procede a actuar nuevamente esperando corregir el problema mecánico, o de presión del martillo que hace que la unidad falle. Tal como se puede notar del diagrama de flujo la tarjeta madre se apaga, y se remueve el martillo, y consiguientemente se repite la prueba.

Las correcciones al flujo del programa se realizan en el lenguaje Basic Script en el cual se ingresan las instrucciones a realizar, ejecutadas por parte de la tarje de pruebas CENTRIX, especialmente diseñada para difusión y captura de señales tanto analógicas como digitales.

A nivel del programa de prueba se ha incorporado una nueva rutina que se encarga de asociar la probabilidad de éxito con el nivel de tensión que es medido, esta se representa en el diagrama de flujo como el primer bloque en

color rojo, de manera secuencial se procede a evaluar si la primera tensión medida es superior a 1.4257V, según la línea de mejor ajuste de la figura 5.3 se establece que para valores de tensión mayores a este la probabilidad de pasar la prueba será menor al 90%, lo cual se considera como un valor bajo según los índices de productividad manejados en la empresa. Si la tensión se encuentra por encima de este límite se procede a estimar la media para descartar una lectura incorrecta, en este punto la media debe superar los 1.4238V, esto es suficiente para tomar la decisión de apagar la tarjeta madre y volver a iniciar el proceso, tal como se muestra en la figura 5.3.

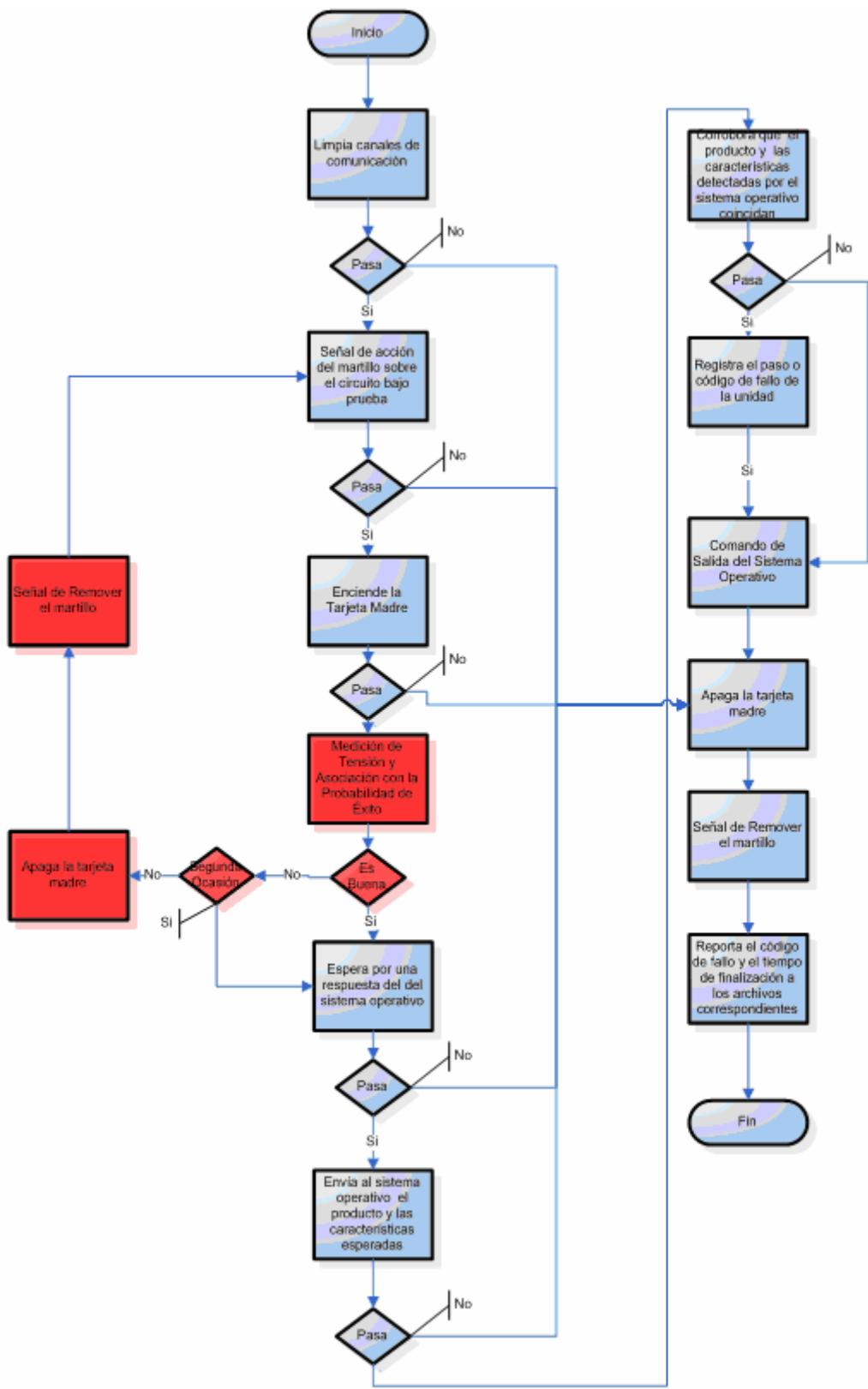


Figura 5.6. Diagrama de Flujo de la Solución Propuesta

5.3. Implementación de Post Codes

Para realizar una explicación detallada de cómo se alcanzó la solución del problema, podemos mencionar los siguientes dos apartados.

- Realizar los cambios necesarios al Hardware del módulo de pruebas para lograr tener acceso a los PostCodes que se almacenan en el puerto 80 de la tarjeta madre.
- Modificar el programa de pruebas que se utiliza actualmente para lograr la correcta lectura de los PostCodes.

A continuación se presenta una explicación detallada de cada uno de los apartados presentados anteriormente.

Realizar los cambios necesarios al Hardware del módulo de pruebas para lograr tener acceso a los Post Codes que se almacenan en el puerto 80 de la tarjeta madre.

Cuando se presenta una falla en el circuito que está siendo probado, el último PostCode que se generó está almacenado en el puerto 80 de la tarjeta madre y está siendo mostrado en dos displays de siete segmentos que se encuentran en la misma.

La necesidad que se presenta es la de tener un registro de este último PostCode que se genera, para lograr lo anterior se utilizó una tarjeta de adquisición de datos llamada Post Card, la cual realiza un monitoreo de los contenidos del puerto 80 así como de las variaciones de voltaje de la tarjeta madre, dicha tarjeta lee los contenidos del puerto 80 y registra cualquier cambio, puede ser leída a través de una conexión con el puerto serie.

La tarjeta utilizada se muestra en la siguiente figura.

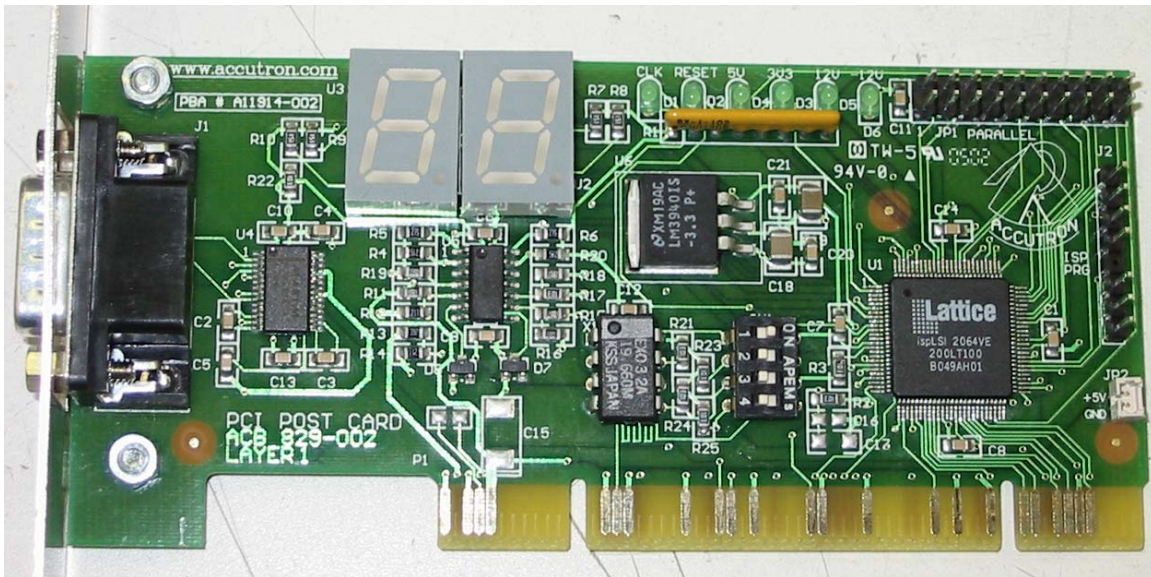


Figura 5.7. Post Card

La tarjeta mostrada anteriormente debe ser colocada en uno de los espacios designados en la tarjeta madre para la conexión de dispositivos que funcionan utilizando el bus (Peripheral Component Interconnect), PCI por sus siglas en inglés. En funcionamiento normal de la computadora, así como durante la inicialización del sistema la Post Card muestra en su display de siete segmentos doble el código hexadecimal correspondiente al Post Code que se encuentra en ese momento en el puerto 80.

La Post Card se comunica vía puerto serie con la tarjeta de adquisición de datos que contiene el modulo de pruebas, la cual a su vez realiza la comunicación con la tarjeta que contiene el programa de alto nivel realizado en Visual Basic Script y que determina los pasos a seguir durante el desarrollo de la prueba. Los cambios realizados a dicho programa de alto nivel para lograr la lectura necesaria se mencionaran mas adelante.

La comunicación a través de la Post Card es realizada a una tasa de 38400 Baudios, transmitiendo 8 bits y sin presencia de un bit de paridad. A continuación se muestra el formato de los dos bytes que son enviados por la

Post Card cuando se reciben nuevos datos en el bus PCI. También es posible solicitar estos datos enviando un carácter vía puerto serie a la Post Card.

Tabla 5.2. Post Card Byte 0

Simbolo	bit	Valor/ Estado	Descripción
ID del paquete	7	0	Identificación del Byte.
D6	6	Alto	PCI Data 6
D5	5	Alto	PCI Data 5
D4	4	Alto	PCI Data 4
D3	3	Alto	PCI Data 3
D2	2	Alto	PCI Data 2
D1	1	Alto	PCI Data 1
D0	0	Alto	PCI Data 0

Tabla 5.3. Post Card Byte 1

Símbolo	bit	Valor/ Estado	Descripción
ID del paquete	7	1	Identificación del Byte.
D7	6	Alto	PCI Data 7
GOODRST	5	Alto	PCI Rst
GOODCLK	4	Alto	PCI Reloj
GOODM12	3	Alto	PCI -12 Voltaje
GOODP12	2	Alto	PCI 12 Voltaje
GOOD5V	1	Alto	PCI 5 Voltaje
GOOD3V3	0	Alto	PCI 3.3v Voltaje

Como se notó en la figura 5.7 la Post Card cuenta con un Dip Switch que permite al variarlo cambiar la frecuencia a la cual la tarjeta esta transmitiendo la información, en este caso se utilizo la opción de 38400 baudios puesto que el modulo de pruebas realiza toda comunicación a esta taza. A continuación se muestran las frecuencias a las que la Post Card permite trabajar y las diferentes configuraciones de las opciones del Dip Switch para obtenerlas.

Tabla 5.4. Opciones de frecuencia de la Post Card

SW1 (1)	SW1 (2)	SW1 (3)	SW1 (4)	Baud Rate
OFF	OFF	OFF	ON/OFF	2400 baud
OFF	OFF	ON	ON/OFF	4800 baud
OFF	ON	OFF	ON/OFF	9600 baud
OFF	ON	ON	ON/OFF	19200 baud
ON	OFF	OFF	ON/OFF	38400 baud (Default)

Al momento de la conexión de la Post Card al sistema se noto que la tarjeta madre poseía únicamente dos espacios para la adición de dispositivos que utilizaran el bus PCI para su funcionamiento, y ambos estaban siendo ocupados por tarjetas utilizadas en el proceso de pruebas para realizar funciones específicas.

Surgió la necesidad de realizar una adición a la tarjeta madre de una tarjeta de expansión que permitiera conectar la Post Card y utilizarla normalmente en el funcionamiento del modulo, para no remover ninguno de los otros dispositivos ya conectados.

La tarjeta de expansión que se incorporo al sistema fue proporcionada por la empresa, y no requirió de ningún trámite ya que se encontraba en las reservas de material referente al modulo de pruebas, únicamente no se estaba utilizando debido a que no era necesaria para las pruebas anteriormente.

La tarjeta de expansión del bus PCI, cuenta con tres nuevas conexiones para dispositivos que lo utilicen, así como para otros puertos que se pueden utilizar en caso de que sea necesario. La siguiente imagen muestra la tarjeta de expansión incorporada a la tarjeta madre del modulo de pruebas.

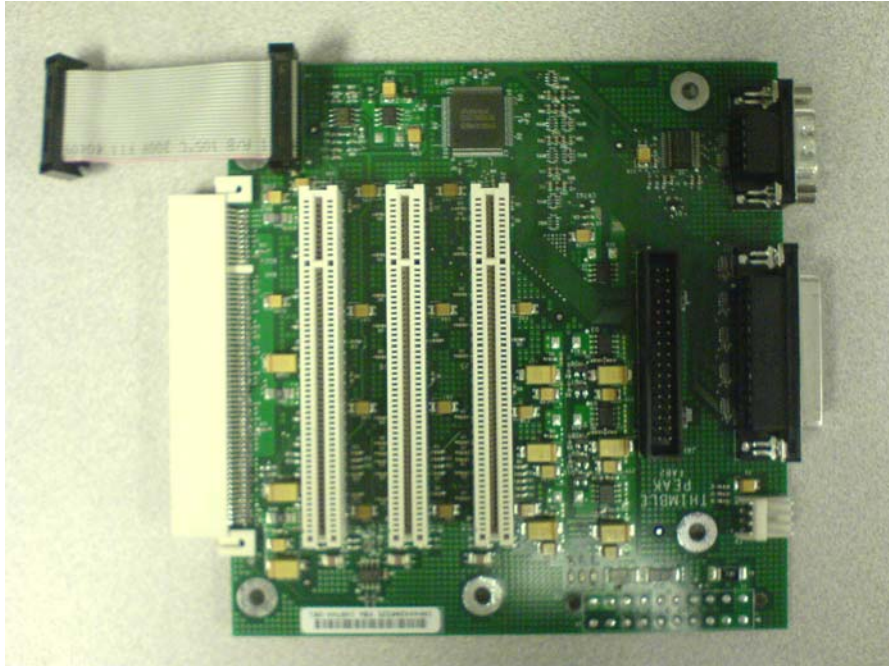


Figura 5.8. Tarjeta de expansión del bus PCI.

Una vez incorporadas la tarjeta de expansión así como la Post Card al modulo de pruebas, se realizaron pruebas que indicaron que la tarjeta madre no estaba tomando en cuenta la Post Card en cuanto a la lectura de los Post Codes, para solucionar lo anterior se debió acceder a la configuración del Bios en la inicialización de la computadora del modulo, una vez ahí se modifico la opción que determina donde serán enviados los ciclos del puerto 80, ya que esta estaba predeterminada para enviar los datos al bus LPC y no al PCI, por ende mostraba siempre el resultado en la tarjeta madre.

Una vez realizado el cambio anterior la Post Card funcionaba correctamente y la lectura dependería completamente de la programación que se realizara en el programa de alto nivel del modulo.

Modificar el programa de pruebas utilizado con el fin de lograr la correcta lectura de los PostCodes.

En el programa de pruebas que se ejecuta sobre cada unidad, una de las pruebas mas importantes es la que se realiza con el fin de corroborar el correcto funcionamiento del sistema operativo, esta prueba es realizada cuando se esta dando el final de la inicialización del sistema, un tiempo prudencial después de haber encendido la tarjeta madre, y permite estar seguro de que el circuito a prueba al haber aprobado este paso puede ser estresado y así comprobar que podrá funcionar correctamente bajo uso normal. Por motivos de confidencialidad a esta prueba se le llamara prueba del sistema operativo, lo anterior siendo un nombre usado únicamente como referencia puesto que la empresa no permite la divulgación de los detalles de las pruebas que se ejecutan en producción.

A continuación se muestra un diagrama de flujo que explica el funcionamiento de la prueba del sistema operativo.

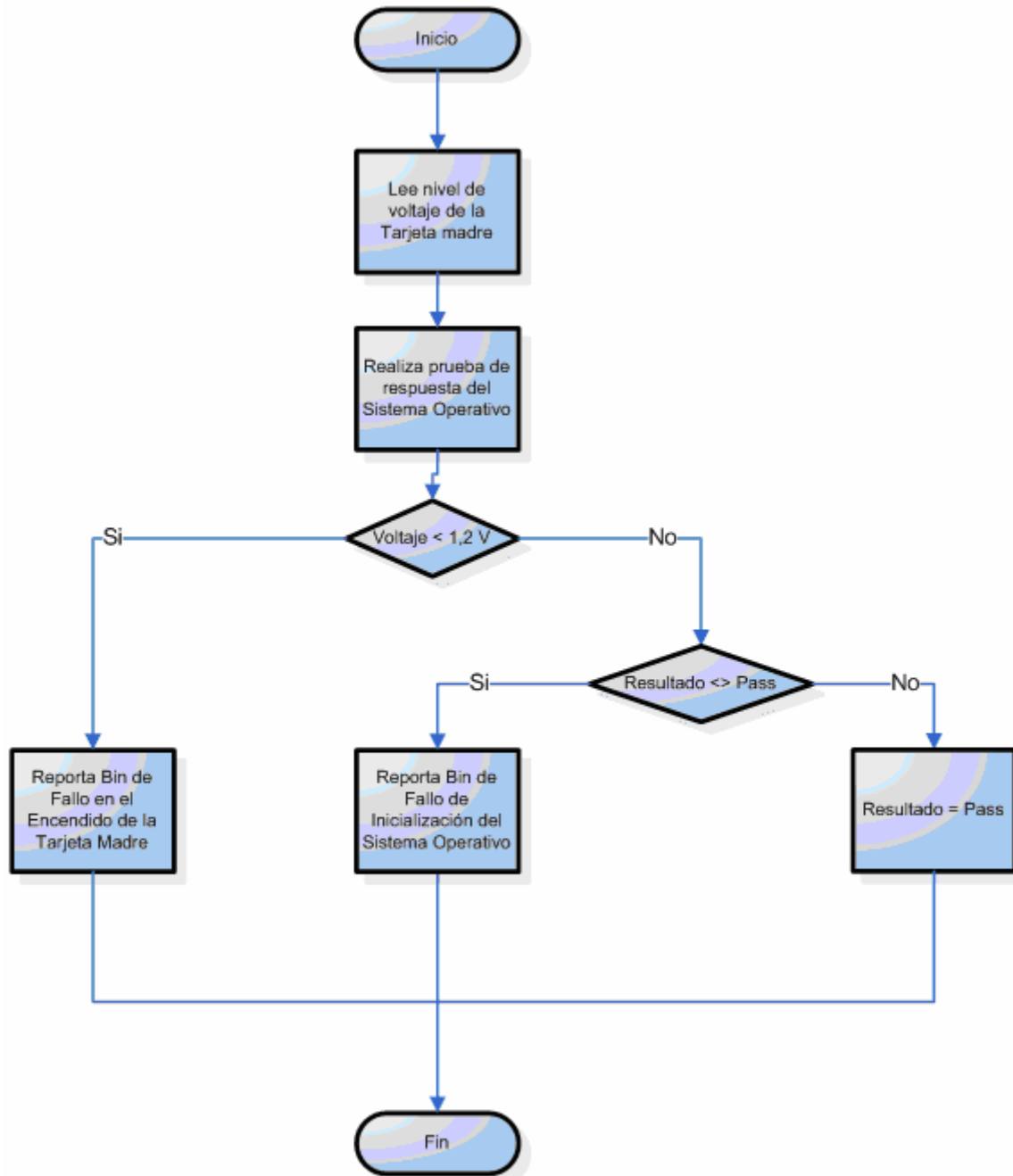


Figura 5.9. Diagrama de flujo del funcionamiento de la prueba del sistema operativo.

En el diagrama se menciona que en caso de una falla el programa reporta el “Bin” de fallo respectivo, dicho Bin es un código que establece empresa para cada uno de los fallos que puede presentar la unidad bajo prueba, esto para

tener un mejor control y realizar los estudios estadísticos correspondientes de una manera mas eficiente que si se conociera cada fallo por su nombre.

Si la unidad bajo prueba logra alcanzar con éxito la conclusión del anterior flujo, se entiende que la etapa de inicialización del sistema operativo un éxito y la unidad ahora esta lista para mas pruebas. Si en algún punto del flujo presentado en la figura 5.9 se diera una falla en la unidad, el ultimo Post Code generado estaría siendo almacenado en el puerto 80 de la tarjeta madre del modulo y se mostraría en su display.

Para lograr la correcta lectura de los Post Codes generados durante la prueba del sistema operativo y tener acceso a ellos en caso de una falla se modifiko el flujo de la prueba agregando un estado, a continuación se muestra el flujo resultante al cambio.

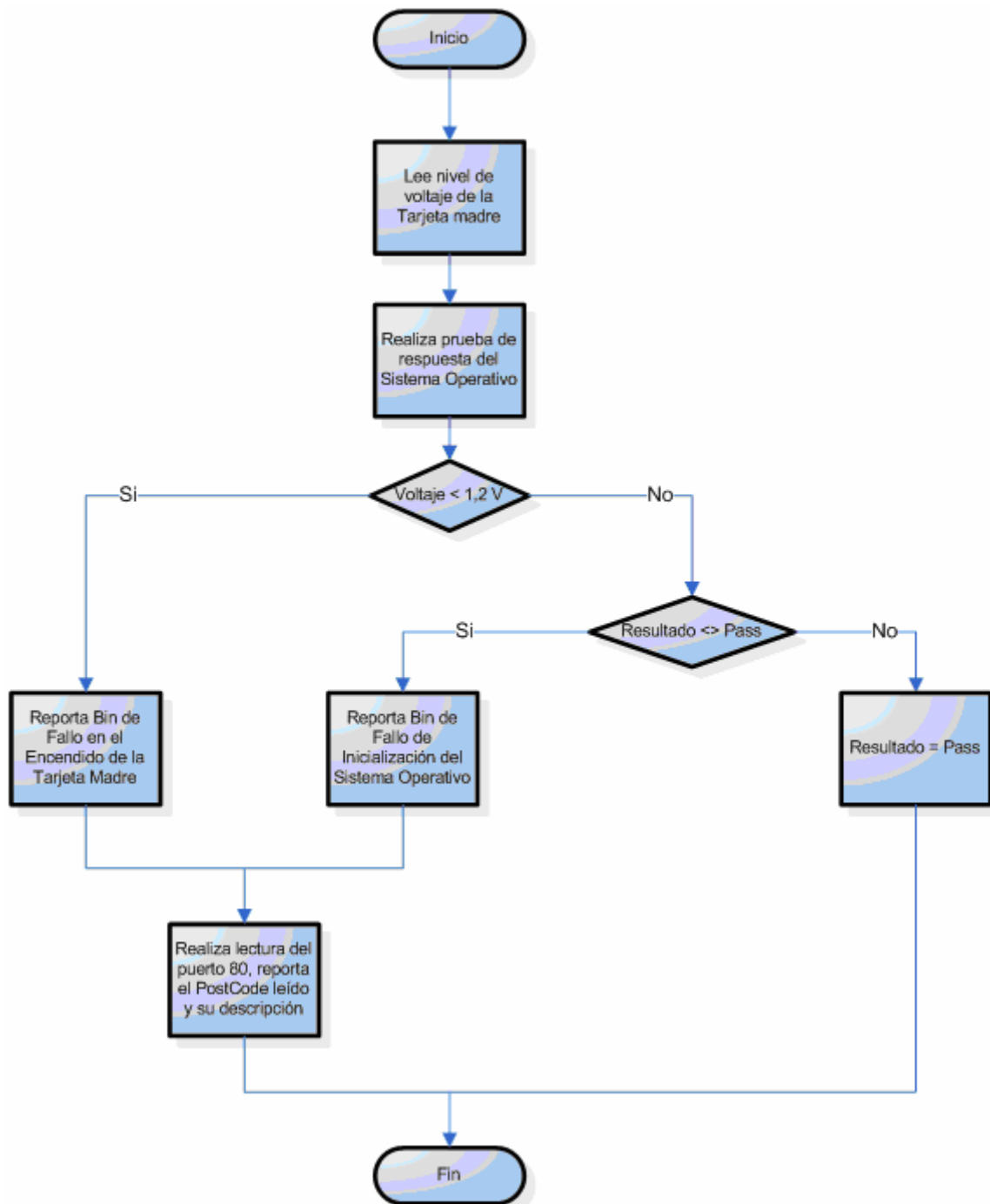


Figura 5.10. Diagrama de flujo de la prueba del sistema operativo modificado.

En caso de que se de una falla en el proceso, antes de terminar el programa de pruebas, se realiza la acción de lectura del puerto 80 de la tarjeta madre por medio de la lectura de la Post Card adicionada al sistema.

El cambio que se realizó al programa de alto nivel que es ejecutado por el módulo de pruebas, consistió en realizar la lectura del puerto serial que está conectado a la Post Card y almacenar los datos leídos en una variable, la cual contiene el Post Code que se generó en el momento que se realizó la lectura. Como se ha mencionado antes cada Post Code representa una acción que el sistema operativo está realizando en el momento de generarlo. Al programa también se incorporó una tabla con todos los Post Codes que se pueden generar en caso de una falla y su descripción detallada, esto con el fin de que en el momento que una falla se presente y se deba realizar un reporte de que Post Code fue el generado en el momento de la misma, también se pueda adjuntar al reporte la descripción de dicho Post Code, lo anterior con el fin de agudizar el nivel de descripción de la falla que se tiene y poder resolver de una manera más eficiente los problemas que se puedan presentar durante la realización de la prueba a una cantidad grande de unidades en producción normal.

Los reportes que se incorporan a la base de datos de las pruebas en caso de una falla cuentan ahora con un indicador de cuál fue el punto específico en el proceso de inicialización en el que el sistema no respondió, pudiendo ser este la lectura de un registro específico o bien la inicialización de algún puerto.

Una vez logrado el objetivo de hacer accesibles los Post Codes en el reporte presentado en caso de una falla de la unidad bajo prueba, se procedió a realizar cambios en el flujo de la misma para poder tomar decisiones basadas en dichos Post Codes. Lo anterior debido a que hay unos Post Codes que por lo general y basándose en los reportes estadísticos generados en la empresa, son un indicador de que la falla no fue exactamente del sistema operativo, sino que puede estar relacionada con un problema mecánico o bien físico de la posición del martillo del módulo sobre el circuito de prueba.

Se encontraron cinco Post Codes que en caso de que la prueba los reportara, si el sistema actuaba de nuevo sobre la unidad a prueba había una gran probabilidad de que la unidad pasara la misma. Por lo que se incorporaron al programa los pasos necesarios para que en caso de que se presentaran,

automáticamente el programa volviera a actuar sobre el circuito y evitarse una posterior prueba de validación en la que se iba a hacer exactamente lo mismo.

El flujo resultante a dichos cambios se presenta a continuación.

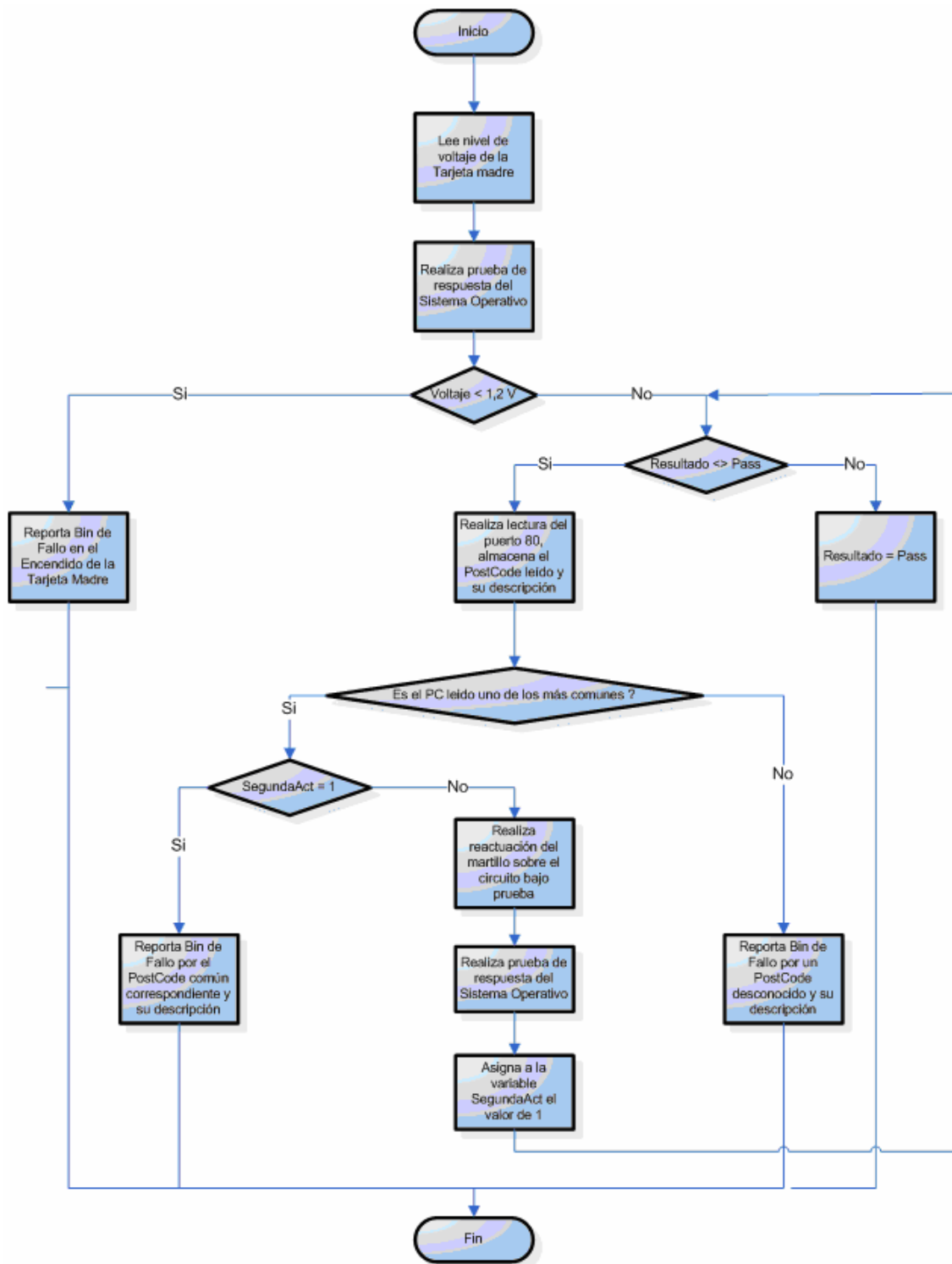


Figura 5.11. Diagrama de flujo final de la prueba del sistema operativo.

En el diagrama anterior se tiene que si sucede una falla, después de leer el Post Code que se genero se realiza una prueba que verifica si dicho Post Code es uno de los más comunes habilitando el camino del programa que

permite que se actúe el martillo del modulo sobre el circuito bajo prueba si así lo fuera. Una vez reactuado el martillo se corre la prueba del sistema operativo, y si falla de nuevo, se reporta como fallo por el Post Code definido como uno de los mas comunes, también en este caso se reporta la descripción detallada del Post Code.

La variable SegundaAct permite saber si ya se realizo una segunda actuación sobre el circuito a prueba.

En caso de que el Post Code generado no sea uno de los llamados “comunes”, se reporta un fallo por Post Code desconocido y se presenta de igual manera su descripción, lo anterior esta accesible en ambas ocasiones que se corre la prueba del sistema operativo en este flujo.

Capítulo 6. Análisis de Resultados

6.1. Resultados para la prueba de corrección según la probabilidad de éxito.

El programa de pruebas implementado pretende contener el rechazo de una unidad debido a una mala conexión. En la figura 6.1, resaltado, líneas 6.00, 6.67, 6.69, 8.23, se muestran las acciones tomadas por el programa de pruebas en respuesta al resultado probabilístico (previa asociación a la medición de tensión). En este caso luego de volver a actuar sobre la unidad, se da una falla conocida como bin 6205, que tiene nuevamente como posible causa de falla el hecho de que aún la unidad este en una mala posición sobre la base, con lo cual se actúa por segunda vez, líneas 36.95 y 37.64, en este caso se puede notar que el sistema arrancó, línea 90.72, y finalmente la unidad pasa la prueba, línea 95.27.

El estudio estadístico pretende reducir cualquier otra causa de error distinta del de arranque conocido como bin 6205, pero en caso de detectarse específicamente ese error el sistema siempre volverá a reaccionar una vez a manera de contención del mismo, que es la situación ejemplificada en la figura 1.

En la figura 6.2 se muestra la respuesta del sistema cuando una unidad pasa de manera convencional. En figura 3 se muestra los mismo, pero esta vez con una reactuación debido a que se ha encontrado el bin 6205, y posteriormente se muestra que se ha recuperado la unidad.

0.08 : ===== BEGIN TESTING =====	65.22 : ERROR: LPTRead failed
0.08 : Test start at: 12/21/2006 1:45:21 PM	65.22 : Failed SystemBootUp
0.09 : LPG A.2	65.22 : CMOS may be corrupted! Hitting the <F2> key.
0.09 : Test program executing on testhead: 2	90.22 : Typing SVOS at LILO.
0.34 : PASS: SetUp	90.69 : Listening for os prompt on parallel port...
1.47 : PASS: ActuateTCU	90.72 : PASS: SystemBootUp
1.56 : Turning Relay ON...	91.03 : Setting dBin/iBin to PASS.
3.22 : Forcing power button LOW	91.05 : PASS: SetPassBin
4.22 : Forcing power button HIGH	91.39 : SVOS logout command was sent
4.24 : Power Up passed.	94.23 : PASS: FS
4.25 : PASS: PowerOnRelay	94.25 : PASS: SVOS
6.00 : Passed to deactivate the TIU.	94.36 : PASS: PowerOffRelay
6.00 : Passing probability too low according to voltage study. Actuating againg	95.14 : Passed to deactivate the TIU.
6.67 : Re-actuating over same unit	95.16 : PASS: DeactuateTCU
6.69 : Turning on again the MB	95.22 : Bin: 100
8.23 : PASS: Reliability	95.22 : Test end at: 12/21/2006 1:46:57 PM
8.25 : PASS: StartUp	95.22 : Elapsed test time: 1.6 minutes
8.33 : Listening for boot OK on parallel port...	95.22 : Elapsed test time: 96s
34.22 : Timed out waiting for code read.	95.22 : Pass:1 out of:1
36.95 : Passed to deactivate the TIU.	95.22 : Percentage:100%
36.95 : Bin 6205 detected. Deactuating	95.23 : PASS: ReportBinNumber
37.64 : Re-actuating over same unit	95.25 : PASS: Shutdown
65.22 : Timed out waiting for code read.	95.27 : PASS: Lakeport CPVs

Figura 6.1.Resultados de la Prueba Sugerida Para la Sección de Probabilidad

728.80 : ===== BEGIN TESTING =====	757.28 : PASS: SystemBootUp
728.80 : Test start at: 12/21/2006 1:13:39 PM	757.59 : Setting dBin/iBin to PASS.
728.81 : LPG A.2	757.61 : PASS: SetPassBin
728.81 : Test program executing on testhead: 2	757.95 : SVOS logout command was sent
729.06 : PASS: SetUp	760.47 : PASS: FS
729.19 : Turning Relay ON...	760.48 : PASS: SVOS
730.45 : Forcing power button LOW	760.59 : PASS: PowerOffRelay
731.45 : Forcing power button HIGH	760.69 : Bin: 100
731.47 : Power Up passed.	760.69 : Test end at: 12/21/2006 1:14:11 PM
731.48 : PASS: PowerOnRelay	760.69 : Elapsed test time: 0.5 minutes
731.53 : PASS: StartUp	760.69 : Elapsed test time: 32s
731.61 : Listening for boot OK on parallel port...	760.69 : Pass:38 out of:48
754.20 : read 18	760.69 : Percentage:79.1666666666667%
754.20 : Listening for os prompt on parallel port...	760.70 : PASS: ReportBinNumber
757.27 : System entered SVOS after disk check. Read hex 200 on the LPT port	760.72 : PASS: Shutdown
	760.73 : FAIL: Lakeport CPVs

Figura 6.2. Resultado de una Prueba Pasada por una Unidad sin Problema

```

823.30 : ===== BEGIN TESTING
=====
823.30 : Test start at: 12/21/2006 1:15:13 PM
823.31 : LPG A.2
823.31 : Test program executing on testhead: 2
823.56 : PASS: SetUp
823.69 : Turning Relay ON...
825.45 : Forcing power button LOW
826.45 : Forcing power button HIGH
826.47 : Power Up passed.
826.48 : PASS: PowerOnRelay
826.53 : PASS: StartUp
826.61 : Listening for boot OK on parallel port...
852.45 : Timed out waiting for code read.
855.16 : Passed to deactuate the TIU.
855.16 : Bin 6205 detected. Deactuating
855.84 : Re-actuating over same unit
881.20 : read 18
881.20 : Listening for os prompt on parallel port...

884.23 : System entered SVOS after disk check. Read
hex 200 on the LPT port
884.25 : PASS: SystemBootUp
884.56 : Setting dBin/iBin to PASS.
884.58 : PASS: SetPassBin
884.92 : SVOS logout command was sent
887.47 : PASS: FS
887.48 : PASS: SVOS
887.59 : PASS: PowerOffRelay
887.69 : Bin: 100
887.69 : Test end at: 12/21/2006 1:16:18 PM
887.69 : Elapsed test time: 1.1 minutes
887.69 : Elapsed test time: 65s
887.69 : Pass:39 out of:50
887.69 : Percentage:78%
887.70 : PASS: ReportBinNumber
887.72 : PASS: Shutdown
887.73 : FAIL: Lakeport CPPS

```

Figura 6.3. Resultado de una Prueba Corregida Específicamente para el error de mayor incidencia.

En una prueba llevada a cabo donde se realizaron 69 corridas sobre una misma unidad, se detectó el bin 6205 18 veces, esto es un 26% de las veces, de estas hubo 6 recuperaciones, lo cual corresponde a un 33% de las veces en que se detectó el error, y a un 8% del total de corridas.

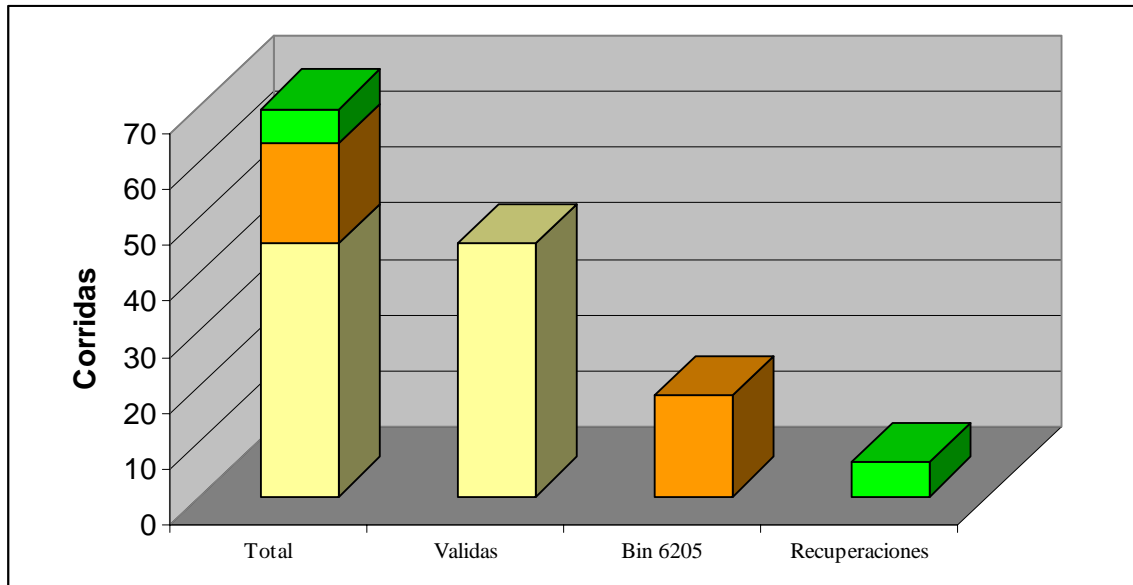


Figura 6.4. Gráfico de Barras Comparativo Representativo a la Incidencia del Éxito (Válida), Error (Bin 6205), y Recuperaciones debidas al Programa de Pruebas Sugerido

Una recuperación del 8%, es muy significativo, ya que para un lote de 3500 unidades implica un ahorro de 2 horas y 20 minutos en unidades que no se correrán una segunda vez. Lo cual aumentaría la capacidad de producción, no solo en un 8% que correspondiente al tiempo ganado con motivo de que no se tuvo que validar por segunda vez las ahora unidades recuperadas, sino que adicionalmente se gana el tiempo correspondiente a la manipulación de la unidad tanto por el operario como por la máquina.

En un segundo experimento a mayor volumen se ha utilizado un módulo completo para hacer la verificación de que con la reactuación en efecto se recuperan unidades. El experimento consistió en tomar 300 unidades de un lote de producción y observar cuantas unidades se revalidan y pasan como buenas debido a esta acción por parte del programa de pruebas sugerido.

Los resultados obtenidos se resumen en las tablas incluidas a continuación:

Tabla 6.1. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 1 del Racal 1

Total de unidades	35
Bin 6205 presente	4
Bin 6205 recuperado	3

La cabeza 1 para una muestra de 35 unidades presentó en 4 casos el bin 6205 para una incidencia de 11.43% de las veces, de los cuales se dieron 3 recuperaciones mediante la reactuación de la TCU, esto es un 75% recuperado. Reduciéndose así a una unidad la presencia del bin 6205, lo cual implica una reducción de la incidencia del bin 6205 en la prueba de 2.86%.

Tabla 6.2. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 2 del Racal 1

Total de unidades	33
Bin 6205 presente	2
Bin 6205 recuperado	2

La cabeza 2 para una muestra de 33 unidades presentó en 2 casos el bin 6205 para una incidencia de 6.06% de las veces, de los cuales se dieron 2 recuperaciones mediante la reactuación de la TCU, esto es un 100% recuperado.

Tabla 6.3. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en el Racal 1

Total de unidades	68
Bin 6205 presente	6
Bin 6205 recuperado	5

El Racal 2 para una muestra de 68 unidades presentó en 6 casos el bin 6205 para una incidencia de 8.82% de las veces, de los cuales se dieron 5 recuperaciones mediante la reactuación de la TCU, esto es un 83.33% recuperado. Reduciéndose así a 1.47% el número de unidades rechazadas por bin 6205.

Tabla 6.4. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 1 del Racal 2

Total de unidades	48
Bin 6205 presente	6
Bin 6205 recuperado	5

La cabeza 1 para una muestra de 48 unidades presentó en 6 casos el bin 6205 para una incidencia de 12.5% de las veces, de los cuales se dieron 6 recuperaciones mediante la reactuación de la TCU, esto es un 83.33% recuperado. Reduciéndose así a una unidad la presencia del bin 6205, lo cual implica una incidencia en la prueba de 2.08%.

Tabla 6.5. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 2 del Racal 2

Total de unidades	74
Bin 6205 presente	5
Bin 6205 recuperado	5

La cabeza 2 para una muestra de 74 unidades presentó en 5 casos el bin 6205 para una incidencia de 6.7% de las veces, de los cuales se dieron 5 recuperaciones mediante la reactuación de la TCU, esto es un 100% recuperado.

Tabla 6.6. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 3 del Racal 2

Total de unidades	73
Bin 6205 presente	0
Bin 6205 recuperado	0

La cabeza 3 no mostró el bin 6205.

Tabla 6.7. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en la Cabeza 4 del Racal 2

Total de unidades	8
Bin 6205 presente	4
Bin 6205 recuperado	1

La cabeza 4 para una muestra de 8 unidades presentó en 4 casos el bin 6205 para una incidencia del 50% de las veces, de los cuales se dio 1 recuperación mediante la reactuación de la TCU, esto es un 25% recuperado.

Tabla 6.8. Síntesis del Resultado Obtenido con el Programa de Prueba Sugerido en el Racal 2

Total de unidades	203
Bin 6205 presente	15
Bin 6205 recuperado	11

El Racal 2 para una muestra de 203 unidades presentó en 15 casos el bin 6205 para una incidencia de 7.4% de las veces, de los cuales se dieron 11 recuperaciones mediante la reactuación de la TCU, esto es un 73.33% recuperado. Reduciéndose así a 2% el número de unidades rechazadas por bin 6205.

Tabla 6.9. Síntesis de la prueba 1A con el TP Sugerido para el Módulo en General

Total de unidades	271
Bin 6205 presente	21
Bin 6205 recuperado	16

El Racal 2 para una muestra de 271 unidades presentó en 21 casos el bin 6205 para una incidencia de 7.75% de las veces, de los cuales se dieron 16 recuperaciones mediante la reactuación de la TCU, esto es un 76.19% recuperado. Reduciéndose así a 1.85% el número de unidades rechazadas por bin 6205.

Tabla 6.10. Detalle de la Duración del Programa de Pruebas

Casos	Tiempo de Ejecución [s]
Bin 100	30
Bin 6205	62
Delta	32

La tabla 10 detalla el costo en tiempo de ejecutar el programa de pruebas con el cambio propuesto. Esto es, a la hora de implementar la nueva rama en el código de programa, en aquellos casos en que se detecte el bin 6205, de tomarse la rama la prueba dura 32 segundos extra cualquiera que sea el programa de pruebas en que se implemente este cambio.

Tabla 6.11. Cuadro Comparativo para del TP Sugerido Contra el de Producción Actual para un Mismo lote

TP	Datos	Bin 6205	Proporción
Sugerido	271	5	1.85%
Producción	295	26	8.81%

* Se utilizaron las mismas 300 unidades para el experimento, sometiéndose las 300 a ambos TP's. Los datos faltantes para el TP sugerido se debe a que 5 corrieron en una cabeza dañada y fueron rejects consecutivos, los otros 24 son datos extraviados debido a una mala manipulación del archivo, los 5 de producción fueron unidades no sometidas a prueba por el módulo.

A la hora de comparar el comportamiento de someter las mismas 300 unidades al programa de pruebas sugerido contra el de producción se puede notar de la tabla 11 que la incidencia del bin 6205 se reduce de aproximadamente un 9% a un 2%.

Si bien es cierto el tiempo de la primera prueba se incrementa en 32 s para los casos en que se presenta el bin 6205, se compensa el mismo tiempo cuando las unidades equivocadamente rechazadas corran por segunda vez, esto sin incluir el tiempo de manipulación por parte de los operarios.

Aclarando, la prueba de producción demora entre [48 – 50] segundos para un bin 100 (unidad buena), donde la variabilidad depende de la rama que tome, debido a que de rescatar una unidad en la primera prueba tomaría 32 segundos adicionales mediante el cambio sugerido, si se rescatara de la manera convencional como se ha venido haciendo hasta la fecha, esto es corriendo nuevamente la prueba sobre la unidad, se demoraría 48 segundos en el mejor de los casos, con lo cual se puede notar que se ganan 16 segundos sólo por el hecho de realizar la prueba en el momento en que falló mediante el cambio sugerido.

Se puede apreciar que el reactuar corrige dificultades físicas en la colocación de la unidad sobre la base, por lo tanto la corrección de ciertos bines se puede llevar a cabo siguiendo esta metodología, en especial en aquel concerniente al bin 6205 que es el error que con mayor frecuencia se presenta, y se le ha dado énfasis a corregir.

Las unidades que en efecto se encuentran mal colocadas y se puede corregir con reactuación, se puede estimar en un 7% que es la diferencia proporcionada entre la incidencia de error el programa de pruebas actual y el sugerido. Esto conlleva a que si se ahorran 16 segundos, en el peor de los casos, en un lote por ejemplo de 3500 unidades se estarían rescatando 245, ello implica 3920 segundos que se pueden utilizar para probar mayor número de unidades, lo que equivale a aumentar la producción en 82 unidades, esto es en 2.3%.

6.2. Lectura de la Identificación de la Unidad

Para la lectura de la identificación de la unidad (UId), se accesa un archivo de registros creados por el sistema operativo donde se encuentra varia información concerniente a los componentes físicos del computador. El acceso en este caso es mediante un sistema operativo compatible con Linux. Este sistema carga toda la información concerniente a los integrados presentes en el mismo, entre ellos la concerniente al Northbridge y el Southbridge. Este archivo se crea única en el arranque de la unidad.

De ella interesa el valor conocido como ULT. En Linux esta información se encuentra en el directorio sv/northbridge, y en este archivo se busca el valor el patrón ULT_fr. Para ello se han utilizado las herramientas de Linux que permiten realizar la búsqueda de archivos, en especial, la instrucción Grep.

Esta información tiene que luego ser decodificada para poder interpretar los valores adecuados, esto es, el sitio en que se produjo la oblea, el año, la semana, información correspondiente al lote, y el número de oblea. Una vez leída esta información mediante puerto serie se transmite hace el computador (Racal) que recibe la información resultado de las diferentes pruebas y este, finalmente, almacena el resultado en un archivo para acceso del usuario.

La manera en que se puede sacar información a través del puerto serie utilizando Linux es mediante el código añadido a continuación.

```
echo "$SerialData" > /dev/ttyS0  
echo "ULT_COMPLETED" > /dev/ttyS0
```

La variable \$SerialData contiene el ULT leído del sistema operativo. En la figura 6.5 se puede ver el ULT leído y desplegado para uso del usuario en un archivo de propósito específico de la empresa.

```

3_prtnm_UnitID
2_lbeg
2_comnt_A!_rackid_CR1CPV115R1
2_comnt_A!_tsthseq_2
2_comnt_A!_testhead_CTGA0-T152
2_tname_TIUACTUATORCNT
2_mrslt_3062
2_lsep
2_trlot_Y62801715
2_trwafer_0
2_trxloc_+00
2_tryloc_+00
2_binn_8000
2_curibin_80
2_lend
3_lsep

```

Figura 6.5. Resultado de un Archivo Resumen Predispuesto a ser Subido a Las Bases de Datos de la Empresa

Lamentablemente esta información no concuerda con la información real, y la razón de ello está en el hecho de que el hexadecimal que es leído se encuentra corrupto. Véase por ejemplo la manera en que se ha descifrado el código en la Figura 6.6.

Esta es la manera correcta de descifrar un código binario. El sitio y el año se encuentra en un código que no es binario natural, por otro lado la semana, el Id.1 e Id.2 correspondiente al lote se encuentra en binario natural.

Sitio	Año	Semana	Id.1	Id.2
Y	6	28	287	0
0010	0010	011100	0100011111	0000

Figura 6.6. Decodificación de una Cadena Binaria

Ahora bien el código leído por el sistema operativo es un hexadecimal que luce de la siguiente manera **000149B0400000**. Dicho valor si se convierte a binario y luego con los criterios adecuados se da su conversión se obtiene el valor **Y6270441:0:+00:+00**, lo que difiere del valor origina **Y628287:0:+00:+00**.

Se ha buscado la manera coincidir de manera binario estos datos y no existe alguna que satisfaga una correcta decodificación para todas las unidades.

La propuesta que se hace e incluye es una sugerencia de que es lo que se debería hacer para poder leer el ULT, lo cual se muestra en el marco teórico, pero aún así para lograr esto se requiere de poder controlar los ciclos de reloj del bus que controla el chipset. Lo cual no es controlable mediante el lenguaje ensamblador si no solamente mediante señales de prueba especializadas que están fuera de las posibilidades de prueba de funcionalidad del módulo de validación.

6.3. Lectura de Post Codes

En cuanto a la lectura de los Post Codes que se generan en caso de una falla, primordialmente se buscaba que en el evento de que la unidad bajo prueba fallara la misma se pudiera tener un reporte de cual fue el último Post Code generado por el proceso de inicialización del sistema.

Cuando una unidad falla una prueba, se reporta el “Bin” o código de falla correspondiente en varias bases de datos con las que cuenta la empresa, una de estas bases de datos muestra su proceso conforme se va ejecutando la prueba haciendo posible ver lo que esta siendo escrito en una pantalla que se muestra junto al indicador de avance de la prueba. Esto en la pantalla con que cuenta el modulo de pruebas.

La figura 6.7 muestra el proceso que sigue la prueba y como en el evento de una falla, el punto del programa donde se da se reporta por estar marcado con una x. En el texto que se presenta en la parte inferior de la figura se puede leer que se dio el fallo, se leyó la Post Card y se observa también el Post Code encontrado así como seguidamente su descripción.

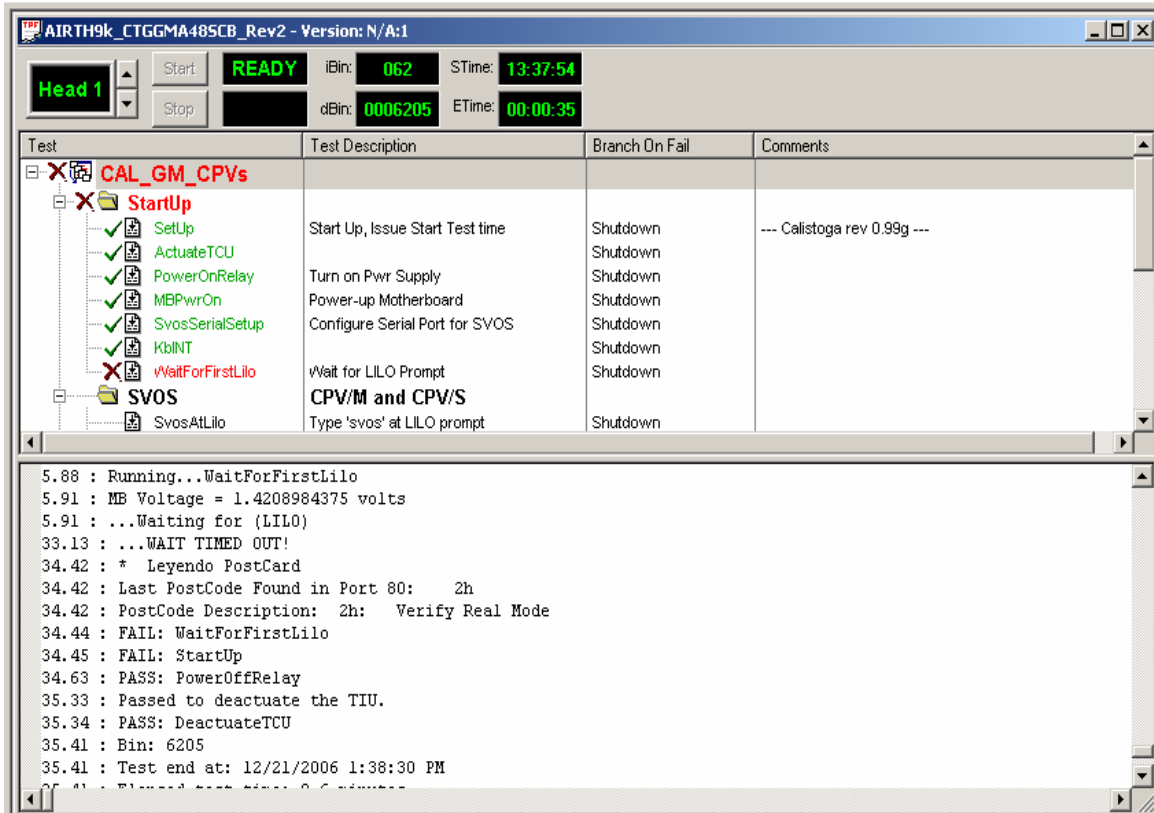


Figura 6.7. Imagen de la pantalla que muestra el modulo de pruebas en caso de una falla.

Teniendo acceso a esta información de cual fue el último Post Code generado se pueden mejorar los análisis que se realizan en caso de que un lote de muchas unidades presente un fallo específico.

Lo anterior debido a que en el pasado en caso de que se presentara una falla durante la prueba del sistema operativo, el sistema reportaba únicamente el fallo de esa prueba, pero si por ejemplo se daban mil fallos de sistema operativo seguidos, era muy extraño, y hacia pensar a los ingenieros de la empresa que era algo mas lo que estaba fallando, como por ejemplo que un puerto no estaba inicializando correctamente, la solución que se tenia a dicha situación en el pasado era revisar el modulo de pruebas, revisando todas las posibles fallas que se podían estar presentando, y una vez concluida la revisión, se corría el material por el modulo de pruebas una segunda vez. Lo anterior representaba un

gasto enorme de tiempo de ingeniería así como de tiempo que dura el producto en salir de la compañía y poder ser distribuido en el mercado.

Con la adición de la lectura de los Post Codes se puede conocer si el fallo se dio por un punto exacto durante la inicialización del sistema, y por medio del acceso a la base de datos, los ingenieros de la empresa pueden saber cual es exactamente dicho punto y tomar las medidas necesarias.

Como último cambio al programa de pruebas se incorporo la acción de que en el evento que el Post Code leído en caso de falla sea uno de los más comunes, o sea, uno de los que si se presentara una reactuación del martillo de pruebas, las posibilidades de que pase la prueba son considerables.

Teniendo certeza acerca del Post Code que evidencia en cual prueba se falla, se procedió a desarrollar un programa de pruebas que reactúa asociando el Post Code con el bin de fallo. Esto es, si la prueba falla porque no empieza el sistema operativo, se evalúa la posibilidad de que esta falla sea efectivamente debido a una mala conexión entre la unidad y la base, o bien se deba efectivamente a que la unidad se encuentra dañada. La intención de ello es de no consumir el tiempo que se demoraría en reactuar en caso de que la unidad se encuentre en efecto dañada, conociendo desde luego el Post Code.

Los resultados encontrados se dirigen hacia que se puede salvar el mismo número de unidades reactuando un menor número de veces si se conoce el Post Code.

Se ha encontrado que del código de falla perteneciente a la prueba de sistema operativo, en efecto hay aproximadamente un 70% de estos fallos que se deben a errores en la colocación de la unidad, o presión con la base, y se pueden corregir reactuando, esto sabiendo además que el Post Code es constante. Mientras que en los casos en que da el mismo error y el Post Code difiere, siendo este alguno anterior al esperado, la acción correctiva de reactuar no tiene efecto para que la unidad pase la prueba y por ende, hay mayores posibilidades que la unidad se encuentre efectivamente dañada.

Así, se ha incrementado la efectividad de la prueba ya que no se consume tiempo realizando la acción correctiva en unidades cuyo síntoma de falla no es corregible de esta manera.

En los casos en que se probó con grandes cantidades de unidades la mejora mencionada anteriormente, se notó que la reducción en la cantidad de reactuaciones es de un 10%, por lo que se salvaron la misma cantidad de unidades que se hubieran salvado simplemente al reactivar la unidad sobre los circuitos que fallaron la prueba de sistema operativo, pero con una disminución en las veces que se tuvo que reaccionar el martillo del módulo, ahorrando esto tiempo de prueba además de vida útil de los equipos del módulo de pruebas.

Capítulo 7. Conclusiones

1. Existe una relación lineal proporcional entre la tensión medida en la fuente y la probabilidad de que la unidad falle.
2. El reactivar sobre una unidad permite corregir por cuestiones físicas la mala colocación de una unidad que incursiona en condiciones de prueba deficientes.
3. La disminución con solo una reactivación por módulo pasa de aproximadamente 9% a 2% en cuanto a la cantidad de rechazos por una sola prueba.
4. La confiabilidad de la prueba aumenta de 91% a 98% con la implementación del programa de pruebas correctivo sobre cada error.
5. La capacidad de procesamiento de unidades por módulo aumenta en un 2.3%.
6. La información del punto específico del proceso de inicialización donde falla una unidad, permitió tomar medidas que aumentaron la fiabilidad de las pruebas.
7. El incorporar al programa de pruebas los Post Codes que comúnmente indican que una segunda prueba a cierta unidad defectuosa resultaría en que esta pase la prueba redujo el tiempo de prueba del modulo así como aumento su vida útil.

Bibliografía

Venegas Jiménez, Pedro. Algunos Elementos de Investigación. 1ª Ed. San José: EUNED, 1986.

Pineda, de Alvarado y de Canales. Metodología de la Investigación. 2ª Ed. Washington: PALTEX, 1994.

Arellano, Jaime. Elementos de Investigación. 8ª Ed. San José: EUNED, 1990.

Meghabghab, George. Introducción a UNIX. 1ª Ed. México: Prentice-Hall: 1997.

Mohr, James. LINUX Recursos Para El Usuario. 1ª Ed. Mexico: Prentice Hall: 1997.

Bennetts, RG. Boundary Scan Tutorial. Version 2.1. 25 Septiembre 2002

Nadolny Richard. Decoding CedarMill FuseID. INTEL.

INTEL. ITP700 Debug Port. INTEL: Febrero 2004.

Mitchell, Pat. Centrix Module Hardware Training. INTEL.

Zitterkopf, John. Product Development Engineer Training. INTEL.

APENDICES

Apéndice.1. Glosario

ATM: Assembly and Test Manufacturing, clasificación que recibe la planta de Intel ubicada en Costa Rica por realizar ensamblaje y prueba de unidades.

Die: Es el cuadro de silicio que proviene de la oblea, y es montado sobre el sustrato en la fabricación de un chip.

DPM: Defects Per Million, unidad utilizada para la medición estadística del desarrollo de la producción.

VLSI: Very Large Scale Integration

CPV: Component Platform Validation, proceso empleado para la validación del producto final que llegara al cliente.

ULT: Unit Level Traceability, información con la cual se puede rastrear la historia de una unidad específica.

TIU: Test Interface Unit, unidad mecánica que tiene contacto con la unidad para realizar las pruebas eléctricas.

Mixing: Se presenta cuando se encuentran unidades pertenecientes a un lote de producción en uno que no les corresponde.

POST: Power On Self Test, prueba que se realiza al iniciarse la máquina.

BIOS: Basic Input/Output System, realiza programas de prueba y se encarga de la comunicación entre el software y el hardware de una PC.

ChipSet: Circuito integrado construido con el fin de ser utilizado para el manejo de diferentes puertos de una PC.

Wafer: En español Oblea, es el conjunto de dies unidos que son fabricados, tiene forma de disco, contiene una gran cantidad de dies.

UNIX: Sistema operativo enfocado primordialmente para servidores.

Apéndice.2. Ubicación del Proyecto

Apéndice.2.1. Descripción de la Empresa

Componentes INTEL radicado en Costa Rica, pertenece a la reconocida empresa transnacional INTEL Corporation dedicada a la fabricación de circuitos de alto nivel de integración (VLSI), proveedor número uno a nivel mundial de procesadores, además dedicada a la producción de los integrados para el funcionamiento y comunicación con dispositivos externos del procesador. Componentes INTEL en Costa Rica se encuentra ubicado en La Rivera de Belén de la provincia de Heredia y forma parte de un conjunto de fabricas ubicadas en diversas partes del mundo que tienen la tarea de ensamblaje y prueba (Assembly and Test Manufacturing, ATM). Son seis en total: Kulim, Penang, Carite, Pudong, TBD y Costa Rica, esta última se conoce como CR AT, donde se procesa el 25% del producto INTEL.

INTEL se instaló en Costa Rica en 1997, en la actualidad cuenta con aproximadamente 5500 empleados.

Misión, satisfacer al cliente, al empleado y a los accionistas cumpliendo el compromiso de distribuir la infraestructura y avances tecnológicos esenciales para el trabajo y la vida diaria.

Apéndice.2.2. Departamento Específico

El proyecto se desarrolla en el departamento de Ingeniería del Producto (PE) de Chipset. Este departamento esta integrado por especialistas en áreas relacionadas con la computación y la electrónica.

Este departamento se encarga de garantizar la calidad del producto final en cuanto al producto Chipset. Para ello se sugieren e implementan correcciones en el proceso productivo, se verifica mediante pruebas el estado de los lotes que ingresan y salen de la fábrica, se diseñan las medidas preventivas y correctivas en cuanto a la garantía de la calidad del producto sin que se afecten los índices de productividad con que cuenta la empresa.

