

157
2el.

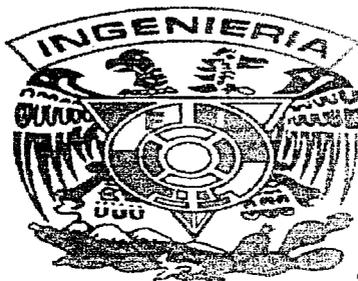


**UNIVERSIDAD NACIONAL
AUTONOMA DE MEXICO**

FACULTAD DE INGENIERIA

**DISEÑO Y CONSTRUCCION DE UN
SENSOR DE CONTACTOS OCLUSALES.**

**TESIS PROFESIONAL
QUE PARA OBTENER EL TITULO DE
INGENIERO MECANICO - ELECTRICISTA
P R E S E N T A :
ADOLFO SANTA FE DUEÑAS**



DIRECTOR: DR. VICTOR M. CASTAÑO MENESES

MEXICO, D.F. 1997 1997

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Dedicatorias

A mis amados padres, María Elena Dueñas y Adolfo Santa Fe.
Mis que imagen,
anhelo un corazón bondadoso semejante al de la semilla de mi origen.

Con todo mi Amor, a Paula Cecilia Guadarrama.
La vida es más hermosa,
si el agradable aroma de una flor te alienta en todo tiempo.

A mis hermanas, Laura y Fabiola.

A todos mis tíos,
especialmente a Héctor Santa Fe, Isabel Chávez,
Arturo Manzano, Verónica Santa Fe,
Juan Santa Fe y Luisa.

A mis abuelos,
Esperanza Alvarado y Bartola Díaz.

A la memoria de mis abuelos,
Felipe Dueñas y Pedro Santa Fe.

A mis maestros,
en especial al maestro Efraín y a la maestra Isabel.

A mi amigo Jesucristo,
"Conmigo está el consejo y el ser; yo soy la inteligencia; mía es la fortaleza".
Proverbios, 8, 14.

AGRADECIMIENTOS

Quiero agradecer a mis compañeros y amigos la C.D. Cristina Muñoz Ledo Belmonte, el I.Q. José Luis Rangel J. y el D.I. Marco Antonio Peniche, por su devoción para trabajar con excelente calidad y en equipo en el desarrollo de este proyecto.

Agradezco también, al Dr. Victor M. Castaño Meneses y al C.D.M.O. Arturo Fernández Pedrero, por sus contribuciones y guía constante para la elaboración de esta tesis.

Agradezco al Sr. Edgar Méndez. Sus valiosas ideas y entusiasmo fueron un recurso de motivación para desarrollar este trabajo.

Un agradecimiento especial a mi amigo M. en I. Lauro Santiago Cruz por sus recomendaciones y consejos que influyeron absolutamente en el éxito de este proyecto.

INTRODUCCIÓN.....	Págs. <u> </u>
-------------------	--------------------

Capítulo 1	DEFINICIÓN DEL PROBLEMA	1
-------------------	--------------------------------	----------

1.1 Fisiología Dental	1
1.2 Desarrollo de la Oclusión Primaria	4
1.3 Tipos de relaciones Oclusales	7
1.3.1 Oclusión balanceada bilateral	7
1.3.2 Oclusión balanceada unilateral	7
1.3.3 Protección Canina	8
1.4 Colocación de contactos oclusales	9
1.4.1 Relación cúspide - fosa	9
1.4.2 Cúspide - reborde marginal	9
1.4.3 Relación cúspide - fosa y cúspide - reborde marginal	10
1.4.4 Resumen de los diferentes tipos de oclusión	11
1.5 Factores del desarrollo que afectan la oclusión	12
1.6 Técnicas Clásicas de diagnóstico	13
1.7 Justificación	14

INDICE

Capítulo 2	PRINCIPIOS DE TRANSDUCCIÓN	15
-------------------	-----------------------------------	-----------

2.1	Transducción Capacitiva	16
2.2	Transducción Inductiva	17
2.3	Transducción Reluctiva	17
2.4	Transducción Electromagnética	18
2.5	Transducción Piezoeléctrica	19
2.6	Transducción Resistiva	19
2.7	Transducción Potenciométrica	20
2.8	Transducción Fotovoltaica	21
2.9	Transducción Termoelectrónica	21
2.10	Transducción por Ionización	22

Capítulo 3	CONVERTIDORES ANALÓGICO - DIGITAL	23
-------------------	--	-----------

3.1	Convertidor digital - analógico	23
3.2	Convertidor analógico - digital	25
	A. Convertidor de Integración	26
	B. Convertidor Flash o paralelo	29
	C. Convertidor por Aproximaciones sucesivas	31

Capítulo 4	ADQUISICIÓN Y PROCESAMIENTO DE LA SEÑAL	35
-------------------	--	-----------

4.1	Transducción	37
4.2	Muestreo	41
4.3	Conversión A/D (Digitalización)	42
4.4	Comunicación con la computadora	44

4.5 Diagramas de interconexión	46
4.5.1 Tarjeta de muestreo	46
4.5.2 Tarjeta de adquisición de datos	48

Capítulo 5	PROGRAMACIÓN (SOFTWARE)	51
-------------------	--------------------------------	-----------

5.1 Adquisición y procesamiento de datos	51
5.1.1 Programa PUERTOS.CPP	52
5.2 Visualización	55
5.2.1 Programa OCLUSAL.GBL	57
5.2.2 Programa OCLUSAL.BAS	62
5.2.3 Programa OCLUSAL.FRM	75

RESULTADOS	91
------------------	----

CONCLUSIONES	93
--------------------	----

GLOSARIO	95
----------------	----

BIBLIOGRAFÍA	97
--------------------	----



INTRODUCCIÓN

La palabra Ingeniero (del ita. *ingegnere* y éste del latín *ingenium*), significa sujeto dotado de la facultad de discurrir o inventar con prontitud o facilidad. Tradicionalmente se define la Ingeniería como la aplicación de los conocimientos científicos a la invención, perfeccionamiento y utilización de la técnica industrial en todas sus acepciones. También se puede definir como la ciencia que trata de encauzar las grandes fuentes de energía existentes en la naturaleza hacia el servicio y conveniencia del hombre. Pero estas definiciones, válidas hasta principios de siglo, han quedado anticuadas a causa de un fenómeno frecuente en nuestra época: la superación del trabajo individual por el trabajo en equipo. Así, en la actualidad, denominamos Ingeniería a la función específica que una o varias personas expertas aporta al proceso de una realización tecnológica o científica. Estos expertos se ocupan primordialmente en aplicar y dirigir hacia fines prácticos y económicos los descubrimientos científicos y también dan pauta de nuevas investigaciones científicas y tecnológicas [1,2].

La invención de herramientas eficientes y de procesos básicos fue el primer paso necesario en el desarrollo de la Ingeniería. A lo largo de la historia, el talento del Ingeniero estuvo generalmente confinado al diseño y construcción de maquinaria militar y de grandes edificaciones. Sin embargo, la fascinación por el conocimiento de como funcionan las cosas, permitió la expansión de la Ingeniería en otros ámbitos. Hablar de hombres como Villard de Honnecourt o Leonardo Da Vinci, quienes hicieron bosquejos de invenciones realizables e irrealizables para su época, es hablar de Ingenieros realmente creativos. Fue a principios del Renacimiento cuando el conocimiento de la Ingeniería empezó a tomar carácter científico, aunado al acumulamiento de experiencias empíricas.

INTRODUCCIÓN

Los avances científicos, el desarrollo de materiales y nuevas invenciones, crearon nuevos problemas y nuevas posibilidades, haciendo a la Ingeniería cada vez más especializada [1,2].

La Ingeniería eléctrica y electrónica se originó del desarrollo en el campo de la electricidad y de investigaciones realizadas por físicos como James Clerk MAXWELL y Heinrich HERTZ. El campo de la Ingeniería Química emergió como resultado del estudio de las reacciones químicas involucradas en la industria y en respuesta a la necesidad de producir en masa productos químicos de uso industrial. Otros campos de la Ingeniería que se desarrollaron a fines del siglo XIX y principios del XX fueron la aeronáutica, Ingeniería Aeroespacial, metalúrgica, agrícola, Industrial y la Ingeniería de Materiales. Actualmente la Ingeniería está repartida en las principales áreas industriales (petróleo, cerámica, energía nuclear, reciclaje de residuos industriales, computadoras) y también en las áreas de salud, transporte y planeación urbana entre muchas otras. De este modo, la Ingeniería causa efectos profundos en el desarrollo de las áreas económicas, políticas y sociales de un país [2].

El proyecto que se presenta a continuación es un ejemplo de la interacción de la Ingeniería con otras áreas, como son las áreas de Odontología, Química y Física de polímeros. Este proyecto surgió de la necesidad de entender la naturaleza del contacto total en la dentición natural, a fin de obtener un correcto diagnóstico y mantenimiento de la longevidad de la propia dentición natural, para diagnóstico y tratamiento de disfunción mandibular [3] y para la planeación de la odontología restauradora entre otras cosas.

Cuando se realizan cambios en las superficies oclusales dentales debido a adaptación funcional o restauraciones nuevas se puede causar pérdida de equilibrio y crearse oclusión dental patológica [4]. Los contactos oclusales dentales ocurren durante los movimientos funcionales y parafuncionales de la mandíbula, que están algunas veces relacionados con expresiones faciales u orales de emoción [5]. Estos contactos dentales funcionales y parafuncionales están influidos por la actividad colectiva de los músculos

mandibulares, coordinados por el sistema nervioso central y periférico. Cualquier alteración en la fisiología neural normal muscular tendrá probablemente cierto efecto en los contactos oclusales dentales [6].

A pesar de la importancia de los contactos oclusales dentales, poca literatura ha sido presentada en lo que respecta a la investigación de la calidad, número y áreas de contacto en la dentición, con alineación normal [4]. Al parecer, solamente a partir de los 80's y de que comienza a evolucionar y a desarrollarse un sistema informático exprofeso, se atiende al estudio de este particular considerando la legítima preocupación del profesional de la odontología por acceder a un medio más exacto de medición y registro de contactos oclusales dentales.

Las variaciones en la presión oclusal dental pueden deprimir al diente que contacta primero, permitiendo a otros dientes contactar. Si los múltiples contactos ocurren de esta manera, el análisis cuantitativo de estos no distingue entre el diente que contacta primero y el diente que después toca. Para determinar con precisión la armonía oclusal dental, es necesario establecer la secuencia del contacto dental dado por una relación articular [7], sin embargo, actualmente no se cuenta con un método y/o con un equipo específico y confiable para comprobar las relaciones oclusales dentales de los pacientes.

Para solucionar este problema se necesitan conocimientos de fisiología dental, electrónica digital y analógica, así como de principios de transducción. Con tal fin, se proponen los siguientes capítulos a desarrollar:

1. Definición del Problema

En este capítulo se dará una descripción de la fisiología dental, de la oclusión dental y de los sistemas clásicos de diagnóstico.

2. Tipos de transductores.

Este capítulo describe los principios de transducción más comunes.



INTRODUCCIÓN

3. Convertidores Analógico - Digital.

En este capítulo se describen diferentes tipos de convertidores analógico - digital.

4. Adquisición y procesamiento de la señal.

Este capítulo describe el método de muestreo de las señales proporcionadas por la etapa de transducción, también describe el procesamiento de éstas para presentarlas gráficamente en la pantalla de una computadora personal.

5. Programación (Software).

Este capítulo describe la manera en que se presentan los registros en la pantalla de una computadora, también se enlistan los programas de computadora elaborados para tal fin.



DEFINICIÓN DEL PROBLEMA

1.1 Fisiología Dental.

La boca es la puerta de entrada del aparato digestivo. Consta de la entrada y la cavidad bucal propiamente dicha, limitada por arcadas óseas o maxilares. La mandíbula inferior realiza en los carnívoros un desplazamiento vertical, y en los herbívoros un movimiento lateral; la del hombre tiene una disposición intermedia que le permite todo tipo de masticación gracias a la ayuda de músculos masticadores: el temporal, el masetero, el malar y los pterigoideos (a ambos lados). Los dientes y molares están especializados y su morfología se adapta a su función, que es la de cortar, triturar o despedazar los alimentos para permitir que los jugos digestivos penetren mejor en ellos [8]. Entre otras de sus funciones, los animales los utilizan para pelear y defenderse. La forma de los dientes, su estructura, su número y su modo de acoplamiento varían considerablemente [2].

En los dientes humanos se distinguen externamente tres partes: la corona, el cuello y la raíz. En el corte longitudinal de la figura 1.1 observamos en la corona una capa de esmalte (esta parte es extremadamente dura), y en el cuello y la raíz una capa de cemento que

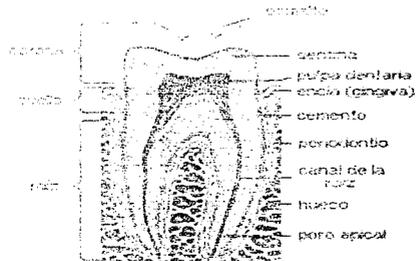


Figura 1.1



recubren a la dentina. La dentina deja una cavidad más o menos llena de pulpa a la cual llegan terminaciones nerviosas a través del poro apical (raíz). La raíz se incrusta en una cavidad del hueso (alveolo), en donde el cemento se acopla al hueso mediante la membrana periodontal (periodontio), compuesta por fibras colágenas. El periodontio, elástico, y que se puede comprimir, actúa como una auténtica articulación que permite pequeños movimientos a los dientes en sus alveolos, amortiguando de este modo los choques en el momento de producirse la masticación. El diente también se sujeta por la encía (gingiva); en esta conexión hay una pequeña depresión llamada grieta gingival [2].

La primera dentición humana (dentición primaria), emerge paulatinamente entre los 6 meses y 2 años de edad. Entre los 6 y 13 años de edad, la segunda dentición (dentición permanente) reemplaza secuencialmente a la primera. Y entre los 18 y 21 años de edad, erupcionan 4 molares más, comúnmente llamados muelas del juicio.

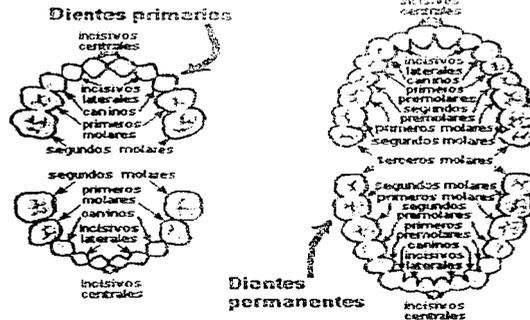


Figura 1.2

Los 20 dientes primarios (4 incisivos centrales, 4 incisivos laterales, 4 caninos (colmillos) y 8 molares) empiezan a desarrollarse antes del nacimiento. Éstos son



reemplazados eventualmente por los 32 dientes permanentes: 4 incisivos centrales, 4 incisivos laterales, 4 caninos, 8 bicúspidos (premolares) y 12 molares. Estos 12 molares incluyen las 4 muelas del juicio, que se desarrollan después de los otros 8 molares, en cuanto la mandíbula incrementa su tamaño. Si este incremento es insuficiente, las muelas de juicio pueden llegar a impactarse con los otros molares y algunas veces son causa de dolor e inflamación.

Los dientes superiores (maxilares) y los dientes inferiores (mandibulares) están formados de tal manera que ajusten mutuamente cuando el maxilar y la mandíbula estén en oclusión (cierran), como se muestra en la figura 1.3. Este ajuste es esencial para que los dientes funcionen apropiadamente en el despedazamiento de los alimentos, con los dientes de enfrente (incisivos y caninos); y en la trituración y masticación, con los dientes de atrás (molares y premolares). Los molares y los premolares tienen superficies grandes con puntas (cúspides) y surcos que normalmente logran una buena oclusión (cierre) [2].

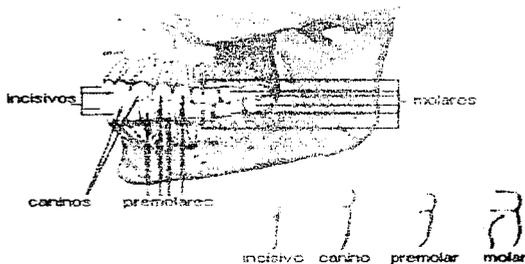


Figura 1.3



1.2 Desarrollo de la oclusión primaria.

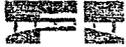
Salzmann y Sheffer han afirmado que la oclusión óptima o normal es un mito en la arcada primaria (dientes primarios), también señalan que para que la oclusión óptima ocurra, se necesita que no existan antecedentes hereditarios; que exista un ambiente óptimo y favorable; y una historia de desarrollo sin ningún accidente, enfermedad o hecho que pudiera modificar el patrón normal de crecimiento [9].

Durante la vida prenatal, la región orofacial se desarrolla mucho antes que las extremidades. Las funciones vitales como los reflejos respiratorios, reflejos de cierre mandibular, reflejos de deglución, succión y soplar, deberán ser completamente operativos al momento del nacimiento, ya que todos estos reflejos se desarrollan entre las semanas 14 y 32 de la vida intrauterina.

La musculatura orofacial y mandibular es responsable de la relación de posición vital que mantienen las vías aéreas. El mantenimiento fisiológico de estas vías es de vital importancia desde el primer día de vida extrauterina. Todas las funciones conocidas de la mandíbula, se adaptan a la posición mandibular y de la lengua, lo que hace posible una vía aérea libre.

La deglución infantil es parte de un reflejo de succión altamente complicado. Tanto la succión como la deglución deben estar desarrolladas desde el nacimiento para que el infante se pueda alimentar. La deglución infantil normalmente permanece durante el primer año de vida, y es diferente a la deglución del adulto, la cuál aparece más tarde.

Los primeros movimientos de la masticación son irregulares y pobremente coordinados. La guía sensorial, durante este periodo de aprendizaje, es proporcionada por receptores de la articulación temporomandibular, ligamento periodontal, lengua, mucosa oral, y en alguna medida por los músculos. Los movimientos individuales de la



mandíbula durante el ciclo de la masticación, son un patrón desarrollado e integrado por muchos elementos funcionales y altamente adaptativos en niños y jóvenes. Los cambios masticatorios son mucho más difíciles de adquirir a medida que pasa el tiempo.

La relación neuromuscular con la mandíbula, es importante para el desarrollo de la oclusión primaria. Las interdentaciones ocurren en secuencia, comenzando en la parte anterior con la erupción de los incisivos. La mandíbula infantil está ligeramente retraída, pero la acción de la lengua, normalmente coloca a los incisivos en función de borde a borde (sin traslape vertical) aunque ellos no se encuentren en completa posición de descanso. Los incisivos primarios, frecuentemente ocluyen en relación borde a borde, lo cual deja a la mandíbula libre para moverse hacia delante y lateralmente desde la relación céntrica. La atrición (desgaste por fricción) comienza inmediatamente cuando el infante explora la nueva sensación de contacto dental.

Durante el segundo año de vida, los caninos primarios erupcionan dentro de los arcos dentales del niño. Los caninos usualmente no son funcionales en un movimiento protrusivo (deslizamiento de la mandíbula hacia delante hasta tener contactos dentales borde a borde) y requerirán para su función, de los movimientos mandibulares laterales. Estos dientes, cuando erupcionan en el segmento anterior en un arco ancho, funcionan más como incisivos adicionales.

Los dientes inferiores proceden de ordinario a los superiores correspondientes, y cada diente primario sucesivo erupciona en secuencia. Esto constituye la oclusión en forma ordenada, con cada diente erupcionado dentro del ambiente propio, dando así una guía a su relación oclusal correcta. La lengua, los labios y los carrillos, continúan dando una guía bucolingual, mientras que en los últimos dientes que erupcionan constituyen una guía mesial (próxima a la línea media de la boca). Finalmente, se forma un límite oclusal por los efectos combinados de los dientes antagonistas, la lengua y la retracción intrínseca de las fibras gingivales.



Los caninos y los incisivos, son los únicos dientes que normalmente llevan a cabo una parte significativa de sus funciones con el contacto dental directo. Los incisivos son dientes cortantes y funcionan como tijeras, con el contacto dental moviéndose mesialmente o desde el contacto inicial. Esta función no representa completa utilidad si no hasta cuando los dientes posteriores sirvan para la masticación de la comida; pero un infante podrá usar los incisivos activamente aún sin la presencia de comida.

La erupción de los primeros molares completa la dentición primaria. Hasta ahora, los patrones básicos funcionales han sido ya iniciados por los caninos y los incisivos. Conforme los nuevos dientes van apareciendo, los músculos aprenden a adaptarse a los movimientos oclusales funcionales necesarios. Existe menos variabilidad en la relación oclusal de la dentición primaria que la dentición permanente; ya que la oclusión primaria ha sido establecida durante períodos más inestables de adaptación del desarrollo, los dientes son guiados dentro de su posición oclusal por la matriz funcional de los músculos durante el crecimiento más activo del esqueleto facial.

La mayoría de las denticiones primarias se completan entre el segundo y tercer año de vida, sin embargo, hay cambios en la relación oclusal y éstos podrán ser vistos hasta la etapa de la dentición mixta.

Generalmente los arcos dentales primarios son de forma oval. Una razón común para la alteración del arco, es el resultado de varios síndromes genéticos, como el paladar fisurado. Los arcos primarios también muestran un espacio interdental generalizado; estos espacios son encontrados en la región anterior y son necesarios para la estabilidad incisal (de los bordes cortantes de los dientes caninos e incisivos) entre los dientes primarios y los dientes permanentes [9].



1.3 Tipos de relaciones oclusales dentales.

Las relaciones entre los dientes maxilares y mandibulares varían tanto en posición céntrica como en las excursiones laterales y el movimiento protrusivo.

Se consideran tres tipos de relaciones oclusales. Estas se fueron conociendo de forma secuencial como sigue:

1.3.1 OCLUSIÓN BALANCEADA BILATERAL

La oclusión balanceada bilateral propone que exista el máximo número de contactos dentarios en los movimientos excursivos. En relación céntrica todos los dientes mandibulares y maxilares tienen contacto, con excepción de los dientes anteriores. Todos los dientes posteriores del lado de trabajo, así como en el lado de balance tienen contacto. En protrusión los dientes anteriores maxilares tienen contacto con los mandibulares, así mismo el último molar de cada arco tiene contacto con su antagonista.

Se ha establecido que la oclusión balanceada óptima, en la cual cada faceta de cada cúspide tiene contacto con su cúspide antagonista y sus inclinaciones durante todos los movimientos, no es esencial en dentaduras completas.

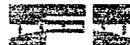
El concepto de oclusión balanceada bilateral no es aplicable a la de dentición natural. A la existencia de contactos en el lado de balance en una dentición natural, se le denomina interferencia en balance y es considerado como perjudicial.

1.3.2 OCLUSIÓN BALANCEADA UNILATERAL

La oclusión balanceada unilateral, también es conocida como oclusión de función de grupo, es ampliamente aceptada y utilizada en la restauración de la dentición natural. Ya que no implica contactos en el lado de balance, no es aplicable a dentaduras completas.

En la oclusión balanceada unilateral todos los dientes posteriores, así como los anteriores tienen contacto con su antagonista en relación céntrica y/u oclusión céntrica.

DEFINICIÓN DEL PROBLEMA

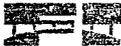


En el lado de trabajo, solo las cúspides bucales mandibulares tienen contacto con la inclinación lingual de las cúspides bucales maxilares, cuando la mandíbula se separa de la posición céntrica en una excursión. No existe contacto entre las cúspides linguales ni maxilares ni mandibulares, en el lado de trabajo. La existencia de contactos entre las cúspides bucales y linguales mandibulares con las cúspides bucales y linguales de los dientes maxilares en lado de trabajo, es conocida como la estabilización dental cruzada, que no es deseable cuando se restaura la dentición natural, según algunos autores.

Contrario a la oclusión balanceada bilateral, la oclusión balanceada unilateral, no establece ningún contacto en el lado de balance. Estos contactos en el lado de balance son considerados perjudiciales en este tipo de oclusión. Los dientes anteriores inferiores tienen contacto en las superficies palatinas de los dientes anteriores superiores en el movimiento protrusivo, eliminando así los contactos entre los dientes posteriores[9].

PROTECCIÓN CANINA

También conocida como desoclusión canina. Debido a la ausencia de contactos friccionales en los dientes posteriores, este tipo de oclusión, propicia un mínimo desgaste en los dientes. Los dientes posteriores maxilares y mandibulares tienen contacto en posición céntrica. La relación céntrica y la oclusión céntrica coinciden, y los dientes anteriores apenas se tocan en la posición céntrica. Los dientes anteriores, incluyendo los caninos e incisivos, tienen contacto en el lado de trabajo. No existe contacto entre los dientes posteriores en los movimientos protrusivos, tampoco existe contacto entre los dientes posteriores en el lado de balance, ya que los dientes anteriores guían a la mandíbula en este movimiento [9].



1.4 Colocación de contactos oclusales dentales.

Se han reportado tres tipos de colocación de cúspides utilizadas en la odontología restauradora.

1.4.1 RELACIÓN CUSPIDE-FOSA

La relación cúspide-fosa es también conocida como oclusión de diente a diente. Esta se encuentra raramente en la dentición natural. Este tipo de relación proporciona una excelente distribución de las fuerzas oclusales y estabilidad a la dentición restaurada. Cada cúspide funcional de los dientes mandibulares contacta con la fosa de los dientes maxilares antagonistas y cada cúspide funcional de los dientes maxilares contacta con la fosa de los dientes antagonistas mandibulares.

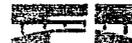
1.4.2 CÚSPIDE-REBORDE MARGINAL.

Este tipo de asentamiento cuspidero es también conocido como cúspide en contra de áreas interproximales. Esta es una relación de un diente contra dos dientes. Cada cúspide bucal mandibular contacta con los rebordes marginales de los dientes maxilares antagonistas, con excepción de las siguientes cúspides, las cuales contactan en la fosa central de los dientes antagonistas maxilares.

1. - Cúspide distal y distobucal del primer molar inferior.
2. - Cúspide distal del segundo molar inferior.

En la relación de la cúspide contra el borde marginal, cada cúspide lingual maxilar, contacta con los dos rebordes marginales de los dientes mandibulares antagonistas a excepción de las siguientes cúspides, las cuales contactan en la fosa central de los dientes antagonistas mandibulares.

1. - Cúspide mesiolingual del primer molar superior.
2. - Cúspide distolingual del segundo molar superior.



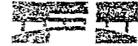
1.4.3 RELACIÓN CÚSPIDE-FOSA Y CÚSPIDE REBORDE MARGINAL.

La relación cúspide-fosa y cúspide-reborde marginal puede ser utilizada ampliamente cuando se restauran varios dientes antagonistas o en reconstrucciones totales de la boca. Esta puede requerir la combinación de las dos técnicas cuando las restauraciones sean antagonistas a la dentición natural.

Adicionalmente, el clínico puede escoger o combinar las dos técnicas por razones funcionales de estabilidad y estética [9].

1.4.4 RESUMEN DE LOS DIFERENTES TIPOS DE OCLUSIÓN DENTAL

	CONTACTOS EN CÉNTRICA		CONTACTOS EN TRABAJO		CONTACTOS EN PROTRUSIVA		CONTACTOS EN BALANCE
	Dientes posteriores	Dientes anteriores	Dientes posteriores	Dientes anteriores	Dientes posteriores	Dientes anteriores	
Oclusión bilateral balanceada	si	apenas tocan	si	no	últimos dientes posteriores	si	si
Oclusión unilateral balanceada	si	pueden o no existir	si	no	no	si	no
Oclusión protegida por los caninos	si	apenas tocan	no	si	no	si	no



1.5 Factores del desarrollo que afectan la oclusión dental.

En el desarrollo de la dentición del adulto, rara vez se encuentra una oclusión ideal. Las anomalías se presentan en la etapa de formación de los dientes. Si ninguno de los dientes se desarrolla, el paciente padecerá de anodoncia (ausencia de todos los dientes). Si falta un solo diente o alguno de ellos, el paciente presentará oligodoncia (ausencia de algunos dientes).

Otra condición observada durante la etapa de inicio y proliferación del desarrollo dental, son los dientes extras o supernumerarios.

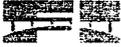
Durante el tiempo de desarrollo del germen dentario, pueden observarse alteraciones como dientes con corona bifida y una sola raíz.

No es difícil ver como los dientes ausentes contribuirán al desarrollo de maloclusiones. La ausencia de dientes tiene una importante relación en cuanto a la forma del arco y la inclinación de los dientes. De la misma manera, muchos dientes pueden ser nocivos para el arco dental en desarrollo. El tiempo normal de erupción de los dientes permanentes es de gran importancia para el correcto desarrollo de la relación intra-arco.

Otra condición o factor local que puede causar la maloclusión, es la pérdida prematura de los dientes primarios. La pérdida no programada de uno o más dientes, puede ser la causa de que se pierda el delicado equilibrio, impidiendo que haya una oclusión saludable y normal.

Otro factor local que debe ser considerado, es la prolongada retención de los dientes primarios. La prolongada retención de los dientes primarios puede desviar la erupción del diente permanente a una posición de maloclusión. Si la ausencia de los dientes es congénita, el tratamiento de elección será la extracción de los dientes primarios retenidos. Esto es para proporcionar una "guía" a los dientes permanentes y lograr que erupcionen en posición normal.

La oclusión usualmente sufre de contactos oclusales anormales, desvíos prematuros y una guía cuspídea pobre. Los dientes pequeños en un alveolo normal, con



los diastemas resultantes, requieren ordinariamente de correcciones estéticas muy severas. Cuando existe pobre relación inter-arco, la fase restaurativa se hace por ello más difícil; el odontólogo debe entonces decidir entre la estética y la función oclusal apropiada.

Lo último por discutir es la caries dental. La caries tiende a llevarnos a la pérdida prematura de los dientes primarios o permanentes y subsecuentemente a la inclinación axial anormal, sobre erupción, pérdida ósea y así sucesivamente a la formación inadecuada de los arcos dentales. Es básico que las lesiones cariosas sean eliminadas, no solo para prevenir la infección y la pérdida de los dientes, sino también para mantener la integridad de los arcos dentales en desarrollo [9].

1.1.0 **Técnicas clásicas de diagnóstico.**

El área total de contacto oclusal en posición intercuspídea, ha sido medida por diferentes métodos para diferentes propósitos, incluyéndose para tal efecto el uso de cera, de compuestos modelados y de papeles indicadores [10].

La revisión de la literatura revela que a principios de los 80's solo existían métodos no prácticos para precisar el análisis cuantitativo de los contactos oclusales. Años después el método Foto-oclusión (Photocclusion method) fue introducido para cuantificar el análisis de las tensiones inducidas en una lámina de medición fotoplástica, referida posteriormente como una lámina de memoria [7].

En 1986 Gazit [11] realizó un estudio que tenía por finalidad diversos análisis:

- Examinar clínicamente la reproductividad de la técnica foto-oclusión comparandola con dos registros oclusales consecutivos y registros realizados a intervalos de un mes;
- Examinar la reproductividad de los papeles de articular o indicadores bajo mismas condiciones y
- Comparar la localización y registro de los contactos con esas dos técnicas.

DEFINICIÓN DEL PROBLEMA



Sus resultados demostraron que ninguna técnica es altamente reproducible; sin embargo el método foto-oclusión es más reproducible que el de papel de articular.

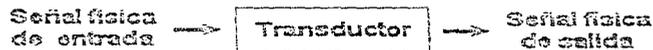
En 1991 Wayne examina la confiabilidad y validez de los últimos sensores y establecen el promedio de la fuerza del umbral requerido para grabar un contacto oclusal interceptivo conocido. El proyecto fue diseñado para medir repetidamente la validez de los resultados idénticos. Los datos obtenidos sugieren que el dentista únicamente debería asentar dos registros con cada sensor, por que el tercero no es exacto [12].

Justificación

A causa de todo lo anteriormente expuesto y dada la importancia de esta temática en la salud dental, conociendo que las alteraciones en las relaciones oclusales pueden causar desgaste excesivo de los dientes, desequilibrio de músculos masticatorios, trastornos en las articulaciones, establecimiento de hábitos orales nocivos y fracturas de cúspides dentarias entre muchas otras cosas, fue como nos encontramos con profundas fuerzas motivadoras para el desarrollo del presente proyecto que apoyará a la práctica dental en lo que a prevención diagnóstico y tratamiento de estas alteraciones corresponde, debido a que con un dispositivo de detección y registro de información oclusal se puede conocer y formar un historial clínico respecto al inicio y desarrollo de un trabajo dental particular, a fin de poder evaluarlo con la mayor fidelidad posible y de tal manera que sea posible detectar y corregir probables fallos.

PRINCIPIOS DE TRANSDUCCIÓN

Un transductor es un dispositivo que transforma una señal física de entrada a una señal física de salida. El tipo de energía en la entrada puede ser igual o diferente al de la salida. El propósito de la transformación es que ésta permite el procesamiento indirecto de la señal física de entrada.



Un ejemplo de transductor es el mecanismo de un termostato, el cual generalmente se basa en la contracción y dilatación que experimentan los cuerpos con las variaciones de temperatura. Así, el termostato puede estar construido por una barra curva bimetalica (transductor) que establece o interrumpe varios contactos, según la variación de su curvatura por la temperatura (señal física de entrada). Entonces mediante la deformación (señal física de salida) de la barra bimetalica se puede cuantificar la temperatura y procesar esta información permitiendo el establecimiento o interrupción de dispositivos que aumenten o disminuyan la temperatura que censa el termostato [2].

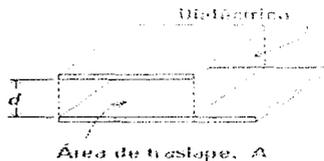
Existen muchos tipos de transductores y estos pueden ser clasificados por su principio de transducción [13]. Los principios de transducción más comunes son los siguientes:



2.1 Transducción capacitiva

Los elementos de transducción *capacitiva* convierten un cambio de la magnitud a medir en un cambio de capacidad. Un condensador o capacitor está compuesto por un par de conductores separados por un dieléctrico, el cual puede ser un líquido, un gas, un sólido, el vacío o una combinación de estos elementos. Considerando, por ejemplo, el transductor capacitivo de la figura 2.1. La capacitancia o capacidad (en picofaradios) de este arreglo está dada por:

Figura 2.1



donde d = distancia entre electrodos,
 A = área de traslape,
 ϵ = constante dieléctrica.

La capacitancia de un condensador puede cambiar debido al movimiento de uno de sus electrodos con respecto al otro (alterando la distancia y/o el área de traslape entre ellos), o mediante cambios en el dieléctrico situado entre los dos electrodos (figura 2.2).

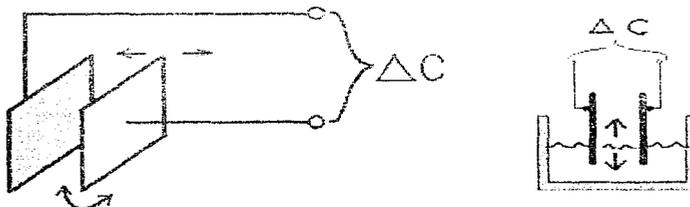


Figura 2.2



El transductor capacitivo puede usarse para medir desplazamiento mediante la variación de la distancia o del área de traslape; también puede usarse para medir niveles de líquido mediante la variación de la constante dieléctrica de acuerdo con el nivel del líquido.

2.2 Transducción inductiva

En la transducción *inductiva*, se convierte un cambio de la magnitud a medir en un cambio de la inductancia de un devanado. Los cambios de inductancia pueden ser ocasionados por el movimiento de un núcleo ferro magnético dentro de un devanado único o por cambios de flujo introducidos externamente en un devanado con un núcleo fijo (figura 2.3).

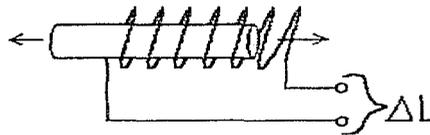


Figura 2.3

El transductor inductivo podría usarse para medir desplazamiento, mediante el movimiento del núcleo dentro del devanado.

2.3 Transducción reluctiva

Los elementos de transducción *reluctiva* convierten un cambio de la magnitud a medir, en un cambio de tensión de C.A. debido a la reluctancia del campo magnético (resistencia magnética) entre dos o más devanados (o porciones separadas de uno o más devanados), con una excitación de C.A. aplicada al sistema de devanados. El cambio en la reluctancia se realiza, usualmente, mediante el movimiento del núcleo magnético dentro del sistema de devanado. Esta categoría incluye los elementos de reluctancia variable, transformador diferencial y puente de inductancias.



Por ejemplo, un transformador lineal es un transformador compuesto por un núcleo magnético que puede moverse libremente dentro de un arreglo lineal de tres bobinas como se muestra en la figura 2.4

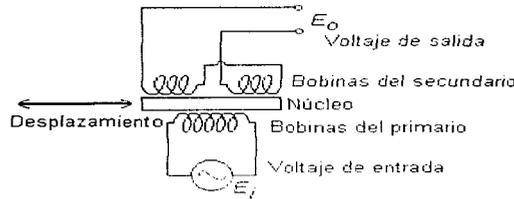


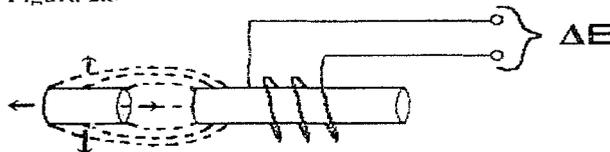
Figura 2.4

Al aplicarse un voltaje alterno en el devanado primario se obtendrá un voltaje de salida de las bobinas del devanado secundario, dependiendo del acoplamiento magnético entre el núcleo y las bobinas. Este acoplamiento depende, a su vez, de la posición del núcleo, mientras el núcleo esté cerca del centro, se tendrá una respuesta bastante lineal en el voltaje de salida. Si este transformador se opera dentro de su gama lineal, entonces se le llama Transformador Lineal Diferencial Variable (LVDT). El transformador lineal se usa para medir desplazamiento mediante el movimiento del núcleo dentro de las bobinas.

2.4 Transducción electromagnética

La transducción *electromagnética* convierte un cambio de la magnitud a medir, en una fuerza electromotriz (tensión de salida) inducida en un conductor debido a un cambio en el flujo magnético en ausencia de excitación (figura 2.5).

Figura 2.5



El cambio en el flujo magnético se realiza usualmente en un movimiento relativo entre un electromagneto y un imán o una porción de material magnético.

2.5 Transducción piezoeléctrica

Los elementos de transducción *piezoeléctrica* convierten un cambio en la magnitud a medir en un cambio en la carga electrostática (Q) o tensión (E) generada por ciertos cristales cuando se encuentran sometidos a un esfuerzo mecánico. El esfuerzo se desarrolla mecánicamente, debido a fuerzas de tensión o de compresión, o esfuerzos cortantes ejercidos directamente sobre el cristal por un elemento sensor o por un elemento de enlace mecánico ligado al elemento sensor (figura 2.6).

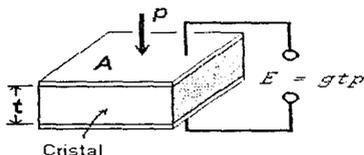


Figura 2.6

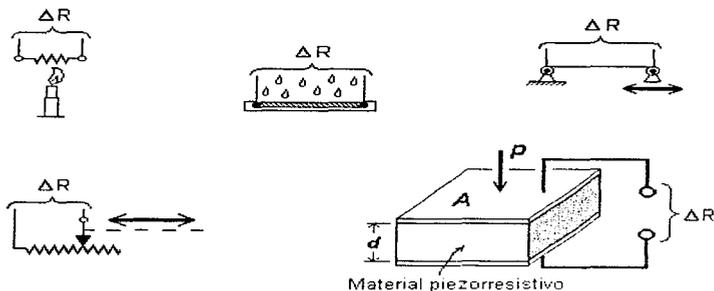
El voltaje de salida lo da:

$$E = g t p$$

Donde t es el espesor del cristal piezoeléctrico en metros, p la presión aplicada en newtons por metro cuadrado y g la sensibilidad de voltaje.

2.6 Transducción resistiva

En la transducción *resistiva*, se convierte un cambio en la magnitud a medir en un cambio en la resistencia. Los cambios de resistencia pueden realizarse en los conductores así como en los semiconductores por diversos medios, como calentamiento o enfriamiento, aplicación de esfuerzos mecánicos (para utilizar el *efecto piezorresistivo*), por incidencia luminosa sobre materiales fotoconductores, por humidificación o deshumidificación de ciertas sales electrolíticas, o por movimientos en el brazo de la escobilla de un reóstato (figura 2.7).



Dentro de este tipo de transducción, también están incluidas las bandas extensiométricas o galgas (figura 2.8). Éstas son los dispositivos más usados para medir la deformación.

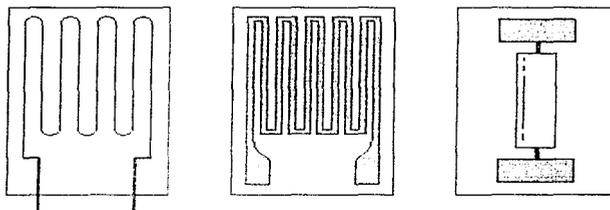


Figura 2.8

2.7 Transducción potenciométrica

Los elementos de transducción *potenciométrica* convierten un cambio en la magnitud a medir en un cambio en la *relación de tensiones*, mediante un cambio en la posición de un contacto móvil (*escobilla*) sobre un elemento resistivo, en cuyos bordes se ha aplicado una excitación.



La relación dada por la posición de la escobilla es básicamente una *relación de resistencias* (figura 2.9).

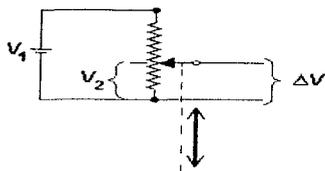


Figura 2.9

2.8 Transducción fotovoltaica

La transducción *fotovoltaica* convierte un cambio en la magnitud a medir en un cambio en la tensión generada, cuando la iluminación incidente sobre una unión entre ciertos materiales distintos cambia (figura 2.10).

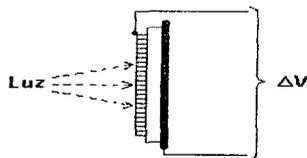


Figura 2.10

2.9 Transducción termoelectrica

Los elementos de transducción *termoelectrica* convierten un cambio en la magnitud a medir en un cambio en la fuerza electromotriz generada por la diferencia de temperaturas existente entre las uniones de dos materiales distintos seleccionados (debido al efecto Seebeck) [13].



Figura 2.11

El elemento termoelectrico básico mostrado en la figura 2.11, puede ir a las terminales de un voltímetro (en las que permanece una temperatura T_2).

2.10 Transducción por ionización

En la transducción por ionización (figura 2.12), se convierte un cambio en la magnitud a medir, en un cambio en la corriente de ionización, ejercida sobre un gas entre dos electrodos [13].

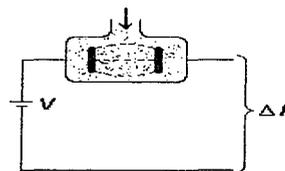


Figura 2.12

CONVERTIDORES ANALÓGICO -DIGITAL

Todas las variables físicas como la temperatura, la humedad, el olor, el sonido, son señales analógicas o continuas de la naturaleza que varían constantemente en el tiempo. La velocidad de variación puede ser relativamente lenta, como la temperatura ambiente, o muy rápida, como las ondas sonoras. Con la tecnología analógica actual, no es fácil manipular, procesar o almacenar información con buena exactitud; es por esta razón, que se recurre a la digitalización de las señales, con el objeto de utilizar las ventajas en el procesamiento de datos que proporciona la tecnología digital. La intercomunicación entre estas dos tecnologías se hace mediante los *convertidores analógico - digital* y *digital - analógico* [14].

3.1 Convertidor digital - analógico

Supongamos que se tiene un convertidor digital - analógico (DAC) con 4 entradas digitales y una salida analógica, el diagrama que lo representa es el de la figura 3.1.

**Figura 3.1**



Cada entrada digital (bit) puede tener solamente dos valores eléctricos que representan un "1" lógico y un "0" lógico (encendido y apagado). Si los consideramos como cifras de un número binario, D_0 es el bit menos significativo (LSB) y D_3 es el bit más significativo (MSB). Con 4 entradas se pueden tener hasta 16 combinaciones diferentes (palabras binarias) de acuerdo a la siguiente fórmula:

$$R=2^n$$

donde, n es el número de bits de entrada o entradas digitales del convertidor y R es el número de combinaciones posibles en las entradas digitales [14]. Las 16 combinaciones posibles en la entrada, ordenadas de acuerdo a los números binarios, y el correspondiente nivel de voltaje en la salida analógica, son los que se muestran en la tabla siguiente:

Entradas				Salida
D_3	D_2	D_1	D_0	V_o
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

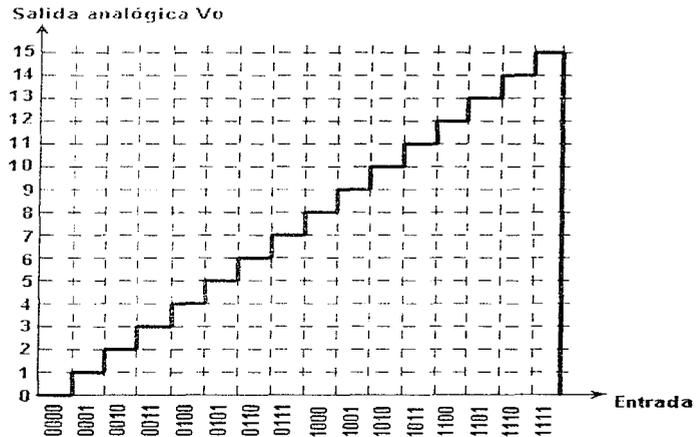


Figura 3.2



El voltaje analógico de salida máximo o voltaje de salida a escala completa ($V_{O_{FS}}$) de este convertidor se obtiene cuando la palabra en la entrada es 1111 (figura 3.2). Suponiendo que $V_{O_{FS}}$ es de 15 V y dado que son 16 los niveles posibles de voltaje de salida, entonces la diferencia de voltaje entre un nivel y su inmediato sería de 1 V. A esta diferencia de voltaje se le llama resolución y está dada por la siguiente fórmula:

$$Res = \frac{V_{O_{FS}}}{2^n - 1}$$

Donde, n es el número de bits de entrada del convertidor.

3.2 Convertidor analógico-digital

Supongamos que queremos medir un voltaje de corriente directa y que se tiene un instrumento que cuenta con lo siguiente: (1) un dispositivo que da una señal (pulso) cada segundo (Reloj); (2) la capacidad de generar un voltaje interno de referencia y de disminuirlo linealmente de 10 V a 0 V a razón de 0.1 V cada pulso de reloj; (3) la capacidad de comparar el voltaje de referencia al voltaje que queremos medir y generar una señal que active a un contador de pulsos de reloj cuando el voltaje de referencia es menor o igual al que queremos medir; (4) la capacidad de generar otra señal que pare el conteo de pulsos cuando el voltaje de referencia llega a 0 V.

Si por ejemplo, queremos medir un voltaje de 4.7 volts, primero se hace disminuir linealmente el voltaje de referencia desde 10 V hasta 0 V (figura 3.3). El contador de pulsos estará inactivo mientras no se genere la señal que indica que el voltaje de referencia es igual o menor al voltaje a medir; cuando el voltaje de referencia es igual a 4.7 volts, se le indica al contador que empiece la cuenta de pulsos de reloj. A partir de allí, transcurrirán 47 segundos hasta que el voltaje de referencia llegue a 0 V debido a que este voltaje disminuye 0.1 V por segundo, y hasta entonces se le indicará al contador que interrumpa su conteo. Así, los pulsos contados serían 47 y como en cada pulso se disminuyó el voltaje de referencia en 0.1 V, entonces se multiplica 0.1 V por 47 y se obtiene el voltaje medido [13].

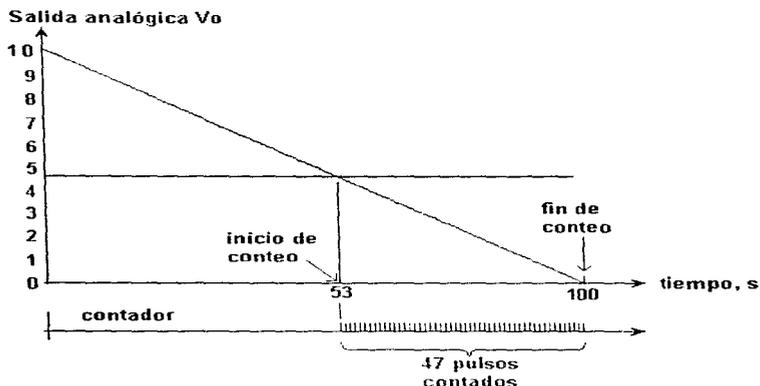
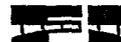


Figura 3.3

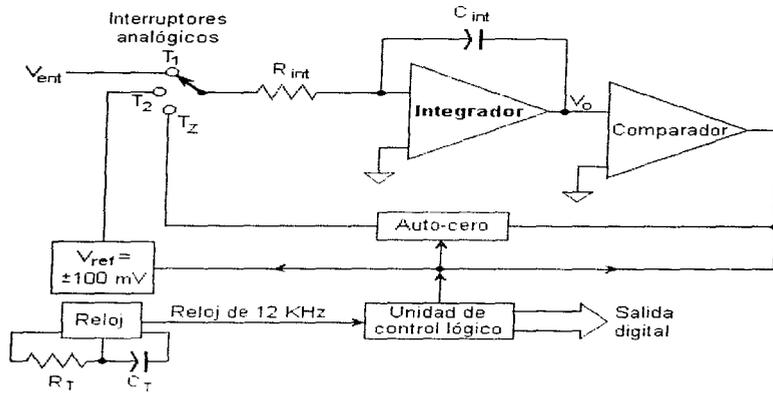
Éste es el principio básico de operación de un convertidor analógico-digital (ADC). Existen varios tipos de ADC's, clasificados según sus tiempos de conversión. El convertidor de *integración lenta* normalmente requiere 300 ms para efectuar una conversión. Es el más indicado para medir voltajes de CD de variación lenta. El siguiente tipo es el convertidor por *aproximaciones sucesivas*, tiene tiempos de conversión de unos cuantos microsegundos y puede digitalizar señales de audio. Los más rápidos de todos son los convertidores tipo *flash* y también son los más costosos, estos convertidores pueden digitalizar señales de video.

A. Convertidor de integración

Los más comerciales de este tipo, son los convertidores de doble rampa, como el 7106 y el 7107 cuyo diagrama de bloques es el que se muestra en la figura 3.4.



Figura 3.4



Este convertidor realiza la conversión analógica-digital en tres fases (integración de señal, integración de referencia y fase de puesta a cero).

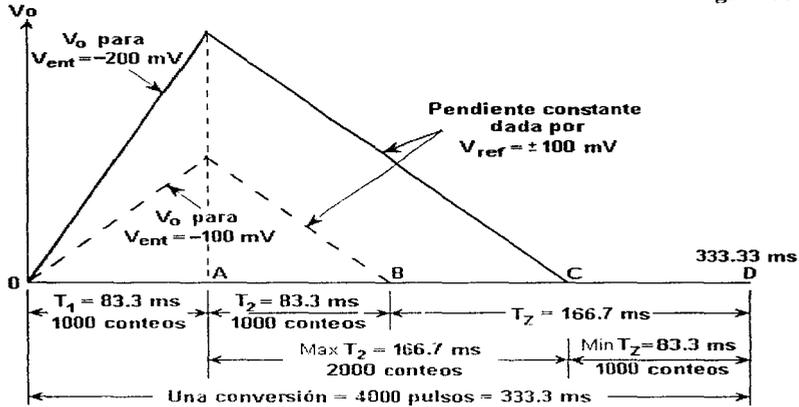
Integración de señal T_1

En esta fase se conecta el voltaje analógico que queremos digitalizar (V_{ent}) al integrador por medio de interruptores analógicos. La salida V_o del integrador será una rampa de pendiente positiva o negativa según la polaridad de V_{ent} ; (Figura 3.5). Esta fase tiene una duración de 1000 pulsos de reloj fijada por la unidad lógica. Durante este tiempo, se carga un capacitor C_{ref} con un voltaje de 100 mV y una polaridad contraria a V_{ent} .

Después de los 1000 pulsos se desconecta V_{ent} del integrador y se conecta V_{ref} , el cual se obtiene de C_{ref} .



Figura 3.5



Integración del voltaje de referencia T_2

En esta fase el voltaje de salida del integrador tendrá una nueva pendiente constante que lo llevará nuevamente a cero debido a que V_{ref} es constante y de polaridad diferente a la de V_{ent} . Cuando V_o llega a cero, se termina esta fase y se obtiene un tiempo T_2 cuantificado en pulsos de reloj. T_2 es proporcional al voltaje máximo obtenido en la salida del integrador y por lo tanto a V_{ent} .

$$T_2 = T_1 \frac{V_{ent}}{V_{ref}}$$

Al inicio de esta fase, la unidad de control conecta el reloj a un contador que cuantifica los pulsos de reloj y lo desconecta cuando termina T_2 , es decir cuando V_o llega a cero. El valor alcanzado en el contador se convierte en la salida digital mediante:

$$Salida\ digital = T_2$$

Sustituyendo T_2 ,

$$Salida\ digital = T_1 (V_{ent} / V_{ref})$$



sustituyendo T_1 :

$$\text{Salida digital} = (1000 \text{ pulsos})(V_{ent} / 100 \text{ mV})$$

o bien:

$$\text{Salida digital} = (10 \text{ pulsos / mV})V_{ent}$$

Así, para $V_{ent} = 100 \text{ mV}$, la salida digital sería 1000 por lo que se debe colocar un punto decimal para mostrar 100.0 unidades de mV.

Auto-cero

Durante la última fase, T_3 , la unidad lógica conecta al integrador un capacitor de puesta a cero C_{AZ} . Este capacitor se carga a un voltaje aproximadamente igual al voltaje de error promedio debido a C_{ent} y al voltaje de offset de los amplificadores operacionales que al ser conectado al integrador, cancela cualquier voltaje de error en C_{ref} [15].

B. Convertidor flash o paralelo

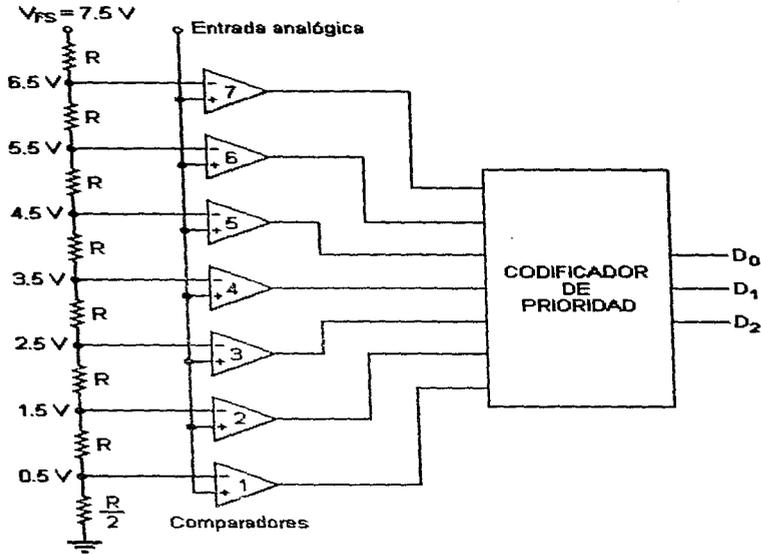
Es el más rápido de los convertidores A/D. Está compuesto por un conjunto de resistencias en serie; que establecen diferentes voltajes de referencia para un grupo de comparadores. Además, estos comparadores tienen como entrada común (paralela), el voltaje analógico V_{ent} que se quiere digitalizar.

Las salidas de los convertidores se conectan a un codificador de prioridad, el cual genera un código binario que representa la salida analógica.

Por ejemplo, un convertidor tipo flash de 3 bits de salida (figura 3.6), tiene 8 resistencias que establecen diferencias de voltaje de 1 V a partir de 0.5 V hasta 6.5 V.



Figura 3.6



Entrada analógica (V)	SALIDA DE LOS COMPARADORES							Salida digital		
	1	2	3	4	5	6	7	D ₂	D ₁	D ₀
0 - 0.5	BAJA	BAJA	BAJA	BAJA	BAJA	BAJA	BAJA	0	0	0
0.5 - 1.5	ALTA	BAJA	BAJA	BAJA	BAJA	BAJA	BAJA	0	0	1
1.5 - 2.5	ALTA	ALTA	BAJA	BAJA	BAJA	BAJA	BAJA	0	1	0
2.5 - 3.5	ALTA	ALTA	ALTA	BAJA	BAJA	BAJA	BAJA	0	1	1
3.5 - 4.5	ALTA	ALTA	ALTA	ALTA	BAJA	BAJA	BAJA	1	0	0
4.5 - 5.5	ALTA	ALTA	ALTA	ALTA	ALTA	BAJA	BAJA	1	0	1
5.5 - 6.5	ALTA	ALTA	ALTA	ALTA	ALTA	ALTA	BAJA	1	1	0
6.5 o más	ALTA	ALTA	ALTA	ALTA	ALTA	ALTA	ALTA	1	1	1



Así, si $V_{ent} = 4 \text{ V}$, las salidas de los comparadores 1 a 4 serán altas y la de los comparadores 5 a 7 serán bajas. Entonces, la salida digital será "100" de acuerdo a la tabla. El tiempo de conversión de los convertidores tipo flash sólo está limitado por el tiempo de respuesta de las compuertas lógicas del codificador y de los comparadores; por lo que es el más apropiado para digitalizar señales de alta frecuencia como video.

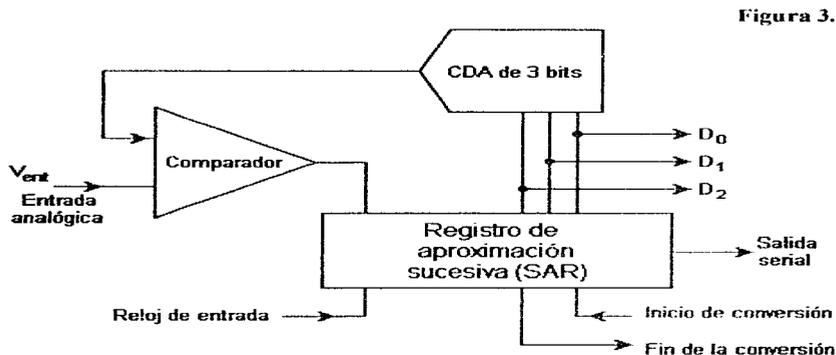
El precio de estos convertidores aumenta considerablemente por cada bit de salida adicional. El convertidor de la figura tiene 3 salidas digitales y requiere 7 comparadores, pero si se necesitaran 8 bits en la salida digital se requerirían 255 comparadores de acuerdo a:

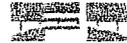
$$\text{Número de comparadores} = 2^n - 1$$

Donde n es el número de bits en la salida digital.

C. Convertidor por aproximaciones sucesivas

En la figura 3.7, se muestra el diagrama de bloques de un ADC por aproximaciones sucesivas de 3 bits.





Esta compuesto por un convertidor digital - analógico (DAC), un comparador y un registro de aproximaciones sucesivas (SAR). Contiene también 3 líneas de control: *inicio de conversión*, que da la orden de empezar el ciclo de conversión analógico a digital; *fin de conversión*, que indica cuando el ciclo de conversión a terminado; y el *reloj*, que establece los tiempos en el ciclo de conversión.

Cuando se inicia la conversión, el SAR manda una serie de números digitales al DAC y éste los transforma en una salida analógica (V_o). Esta salida analógica se compara con el voltaje analógico de entrada V_{ent} . El resultado de esta comparación decide el estado final para cada bit de la salida digital. Se inician las comparaciones con el bit más significativo (MSB) y se terminan con el menos significativo (LSB) como sigue:

Supongamos que el voltaje representado por el bit menos significativo (LSB) cuando es transformado en el DAC es de 1 V, entonces el segundo bit representara 2 V y el tercero (MSB) representara 4 V. Si, por ejemplo, queremos convertir 2.7 V en una salida digital, se evaluara primero el bit más significativo: el valor que representa este bit (4 V) se compara con 2.7 V (V_{ent}). Si V_{ent} es mayor entonces el bit comparado quedara como un "1" (activado) durante las siguientes comparaciones, en caso contrario quedara como un "0" (desactivado). En este caso como 2.7 V es menor que 4 V, el bit queda entonces desactivado ("0"). Después se compara el siguiente bit, así el SAR aplica un "010" al DAC (considerando que el primer bit ya quedo como "0") con lo que se tiene en su salida un $V_o = 2$ V. Como $V_{ent} = 2.7$ V es mayor a V_o , entonces el bit queda como un "1". Por último, para comparar el tercer bit, el SAR aplica un "011", obteniéndose en la salida del DAC un $V_o = 3$ V. Como 2.7 V es menor a V_o , el último bit queda entonces como un "0". Así, la palabra digital en la salida que representa al voltaje analógico de entrada V_{ent} , es finalmente, "010" (figura 3.8) [15].

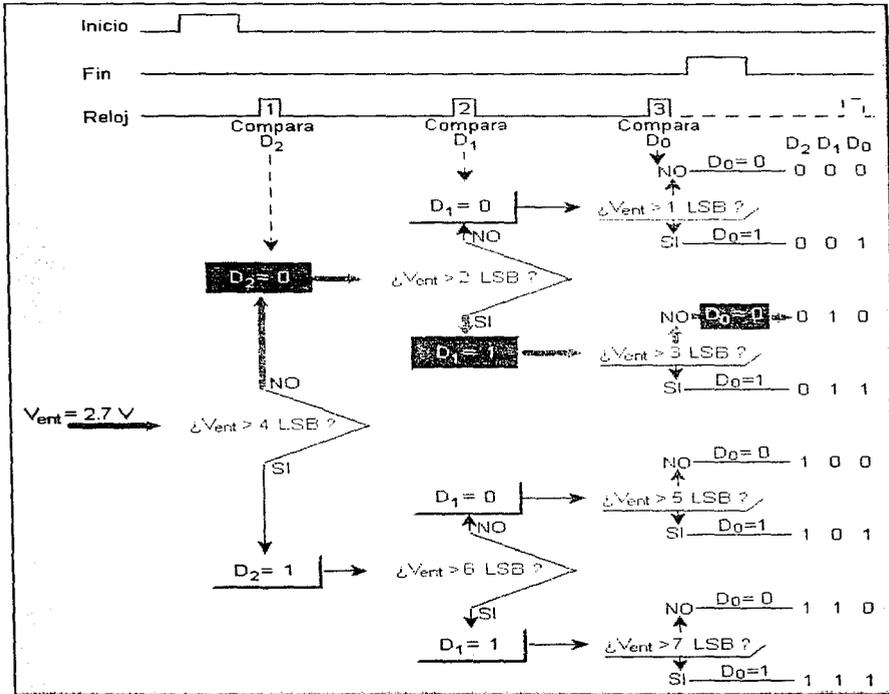
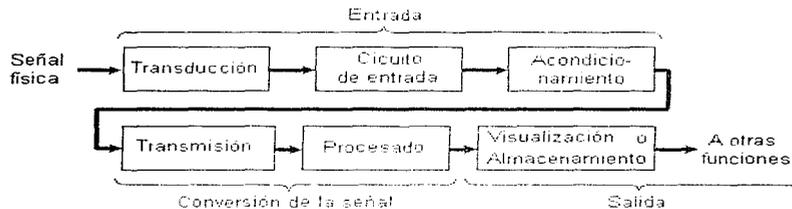


Figura 3.8

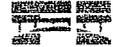
ADQUISICIÓN Y PROCESAMIENTO DE LA SEÑAL

El objetivo de cualquier sistema de obtención y procesamiento de datos es coleccionar los datos, procesarlos en la forma deseada y registrar los resultados de manera adecuada para almacenamiento, presentación o proceso adicional subsecuente. Los elementos principales de un sistema de adquisición de datos se muestran en el diagrama de bloques de la figura 4.1.

Figura 4.1



La etapa de entrada consta de transductores apropiados y un circuito de entrada, además de circuitos adicionales y el acondicionamiento de la señal como sea necesario (amplificadores, filtros, etc.). La segunda etapa es la de conversión de la señal, en la cual la información se prepara para ser transmitida, si se requiere (como en las situaciones en que los transductores están lejos de donde se desea mostrar los datos), lo mismo que el equipo de transmisión y recepción y cualesquiera procesadores de datos; un ejemplo de ésta última, es la conversión de una señal de la forma analógica en digital. La etapa final



es la de salida con dos funciones primarias: mostrar y almacenar datos. Algunos ejemplos son el despliegue y almacenamiento de la información.

Si en la entrada se cuenta con varios transductores, el sistema de procesamiento y obtención de datos debe estar equipado para captar y analizar canales múltiples de entrada de datos. Normalmente se emplea un barredor (o explorador / programador para trabajar con canales múltiples. El barredor es un dispositivo que muestra los canales de datos, de modo que solo se necesita una etapa de conversión y salida [13].

En este proyecto, el sistema de adquisición de datos incluye una etapa de transducción para convertir las señales de presión en el plano oclusal en señales eléctricas; una etapa de muestreo (barredor), que permite seleccionar los canales de entrada de la etapa de transducción. Como la distancia entre la etapa de transducción y la de procesado es corta (más o menos 4 m.) no se requiere una etapa de transmisión, ni de recepción [23]. Se cuenta con una etapa de conversión analógica - digital y con una de procesamiento para poder presentar los datos de manera gráfica. Durante la etapa de visualización y almacenamiento los datos son presentados en la pantalla de una computadora y pueden ser almacenados como archivo o en forma gráfica mediante su impresión (figura 4.2).

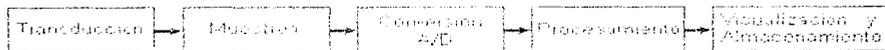
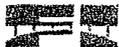


Figura 4.2

A continuación se describen brevemente estas etapas:



4.1 Transducción

Para censar la presión en el plano oclusal, fue necesario utilizar una matriz de transductores sensibles a la presión. Las características requeridas para que esta matriz de transductores sea confiable en la adquisición de los datos y confortable para el paciente, son las siguientes:

- Ser desechable y económica, o en su defecto esterilizable. Así evitamos el contagio de enfermedades, pues se introduce en la boca del paciente.
- Con forma ergonómica para comodidad del paciente.
- Tener un grosor menor a 2 mm. De esta forma aseguramos un espacio interoclusal pequeño y una mayor aproximación en la detección de los puntos prematuros de contacto en la oclusión dental.
- Ser mecánicamente resistente, dado que en una mordida se pueden superar los 400 Kg/cm².
- Contener un mínimo de 30 celdas por centímetro cuadrado para poder localizar los puntos de contacto con buena precisión.
- Ser lo suficientemente flexible para asegurar independencia de transducción entre las celdas.

Con estos requerimientos, el diseño de las celdas transductoras se enfocó en el uso de transductores piezorresistivos, debido principalmente a su tamaño, fácil manejo y a su bajo costo. Además, se contó con la colaboración del Instituto de Física en la elaboración de un compuesto que se utilizaría como material piezorresistivo.

El material piezorresistivo funciona como una resistencia variable, si conectado a un voltaje fijo y a una resistencia en serie, con el fin de formar un divisor de voltaje (figura 4.3), se le aplica una fuerza de compresión, este material disminuye su resistencia eléctrica y la caída de voltaje en la resistencia cambia de manera proporcional a la presión aplicada.

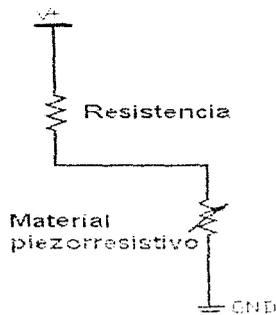
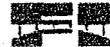


Figura 4.3

Inicialmente, se pensó en hacer la matriz de transductores usando una placa conectada a un voltaje fijo y con un circuito impreso en forma matricial, en donde cada elemento de la matriz sería una celda transductora con su propia línea de selección hacia una resistencia conectada a tierra y con su propio material piezorresistivo como se muestra en la figura 4.4.

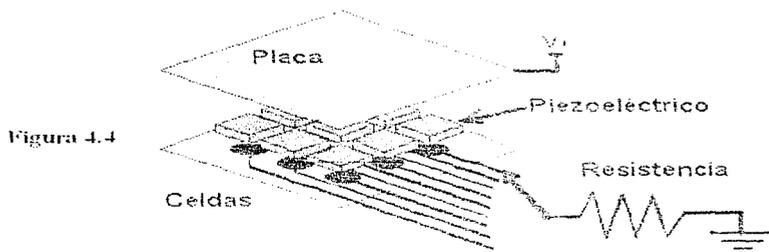
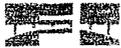


Figura 4.4

Pero se desechó esta opción, porque al necesitarse 30 celdas/cm², cada celda mediría 1.8 mm, por lado y la densidad de líneas en el circuito impreso de la matriz sería muy alta. Además, se tendrían demasiadas líneas hacia la resistencia y la selección de las celdas requeriría de muchos dispositivos.



Por esta razón, se decidió diseñar las celdas sensoras pensando en la manera de disminuir el número de líneas para seleccionárlas. De este modo, la matriz transductora diseñada consiste de dos plantillas conductoras (una con líneas verticales y otra con líneas horizontales), además del compuesto piezorresistivo (figura 4.5).

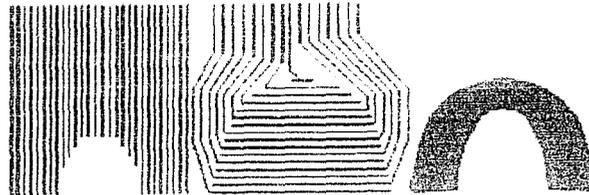


Figura 4.5

Empalmando los tres elementos con el compuesto piezorresistivo entre las dos plantillas, resulta un arreglo de aproximadamente 3000 celdas sensoras de presión como se muestra en la figura 4.6.

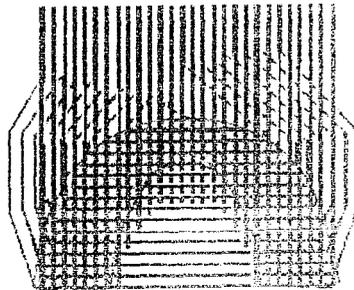
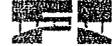


Figura 4.6



Este arreglo permite seleccionar una celda sensora mediante las líneas vertical y horizontal correspondientes de las dos plantillas, como se muestra en la figura 4.7.

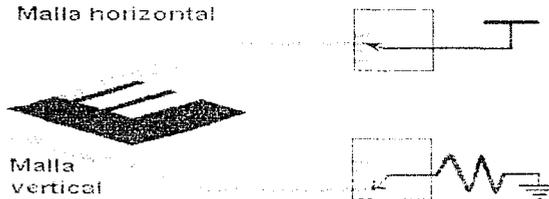
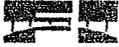


Figura 4.7

Además, se puede utilizar una lamina de material piezorresistivo y no es necesario tener un pedazo de este para cada celda (también es posible utilizar tinta semiconductor impresa en una de las plantillas). Esto es posible gracias a que con este arreglo, la resistencia eléctrica que se presenta entre líneas conductoras de la misma plantilla es mucho mayor con respecto a la que se presenta entre las dos plantillas, lo que nos permite considerar solo la que se presenta entre la línea horizontal y la línea vertical que corresponden a la celda transductora seleccionada.

Las plantillas conductoras son dos circuitos flexibles que para su elaboración se consideran varias opciones:

- se pueden realizar con una lamina flexible de cobre recubierta en una de sus caras por mylar. Como este material es muy caro (aproximadamente 100 dolares por dm^2), se podría utilizar solo si se piensa en esterilizar las plantillas para ser usadas varias veces. También es un material muy rígido que no permite una autonomía aceptable entre las celdas transductoras. La ventaja que se tiene con este material, es que es mecánicamente muy resistente, por lo que sería uno de los apropiados si se quiere hacer un registro de presiones cuando el paciente muerde con mucha fuerza.



- para hacer más flexibles las plantillas se propuso pegar una laminilla de cobre a un plástico flexible como el nylon, aun así, no se logró buena independencia entre las celdas.

- finalmente, nos encontramos con un material de deshecho de la industria empacadora de alimentos, el cual está conformado de nylon y aluminio. Comparándolo con los anteriores, este material es muy barato (1 peso por m²), lo que permite que las plantillas sean desechables. Este material proporciona mayor flexibilidad que los demás y es de fácil adquisición.

4.2 Muestreo

El objetivo de esta etapa es seleccionar cada una de las celdas transductoras y medir la caída de voltaje que se presenta en estas. La selección de una celda específica, puede realizarse, como se muestra en la figura-4.7, mediante el uso de interruptores analógicos. Los circuitos utilizados en esta etapa son los siguientes:

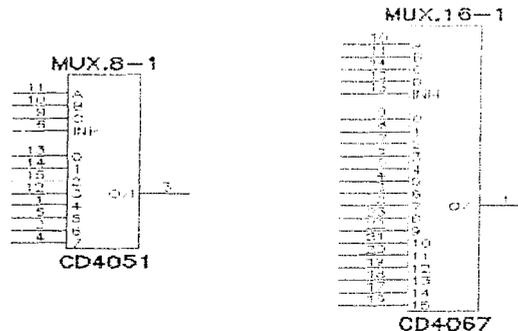


Figura 4.8

El CD4051 es un multiplexor analógico 8:1, y el CD4067 es un multiplexor analógico 16:1.



Los pines marcados con las letras A, B, C y D, son las líneas de selección; los pines marcados con los numeros 0-15 son los canales de entrada / salida; I/O es el canal común y el pin INH es la línea que habilita al circuito [16].

4.3 Conversión A/D o Digitalización

En esta etapa, el voltaje muestreado de las celdas transductoras es digitalizado por medio de un convertidor analógico - digital (ADC) de 8 bits. El ADC utilizado es el ADC804, cuyo diagrama esquemático se muestra a continuación:

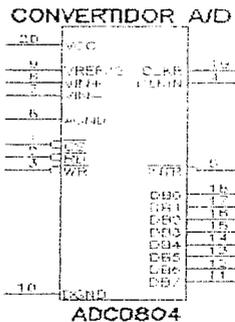


Figura 4.9

Vin+ y Vin- .- son los pines de entrada analógica diferencial.

AGND .- tierra analógica

CS .- línea de habilitación del circuito

RD .- Lectura.

WR .- Escritura.

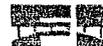
CLKR y CLKIN .- Configuración del reloj.

INTR .- interrupcion (fin de conversión)

DB0 - DB7 .- salidas digitales.

Vref/2 .- Voltaje de referencia del convertidor.

Este convertidor A/D es de aproximaciones sucesivas, es de fácil adquisición, funciona parecido a un puerto de entrada o una memoria y no necesita ninguna interfase para ser conectado a un puerto de un microprocesador. Además, tiene una resolución de 8 bits y un tiempo de conversión de 100 μ S, suficientes para satisfacer nuestros requerimientos de muestreo.



4.4 Comunicación con la computadora

Para comunicar los dispositivos anteriores (convertidor y multiplexores) con la computadora, se accede directamente al bus de datos de esta por medio de una de sus ranuras de expansión y usando una tarjeta de adquisición de datos. La tarjeta diseñada utiliza los circuitos que se muestran en la figura 4.11.

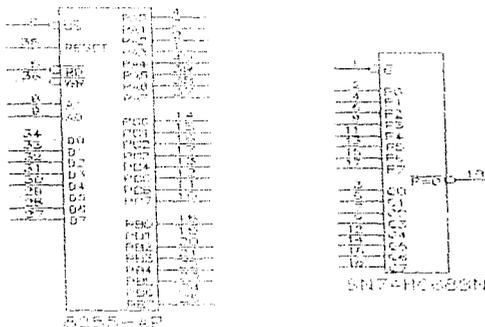


Figura 4.11

El circuito integrado S255-AP es una interfase programable de puertos (PPI). Este circuito genera 3 puertos de 8 bits programables como entradas o salidas [17].

- PA0-PA7 .- líneas del puerto A.
- PB0-PB7 .- líneas del puerto B.
- PC0-PC7 .- líneas del puerto C.
- D0-D7 .- Bus de datos.
- A0-A7 .- líneas de selección.
- CS.- habilitación del circuito.
- RD .- lectura.
- WR .- escritura.
- Reset .- reinicializa al circuito.



El circuito integrado 74HC688 es un comparador de 8 bits [18].

P0-P7 .- Primer grupo de bits

Q0-Q7 .- Segundo grupo de bits.

G .- habilitación del circuito.

P=Q .- resultado de la comparación.

La tarjeta de adquisición de datos se diseñó para bus PC. Este bus es compatible con el bus AT-ISA y el EISA lo que permite conectar la tarjeta a cualquier computadora que tenga alguno de estos tipos de buses. El bus PC tiene disponibles los siguientes elementos [19].

-Un bus de datos, que consta de 8 líneas de Entrada/Salida (D0-D7).

-Un bus de direcciones, que consta de 20 líneas (A0-A19) que determinan la memoria direccionable.

-Líneas de interrupción (IRQ2-IRQ7).- Que indican al procesador que atienda a algún periférico.

-Niveles de voltaje (+5V, -5V, +12V, -12V y GND).

-Líneas de control de lectura y escritura.

-Líneas para acceso directo a memoria (DMA).

-Reloj del sistema.

-Reset.

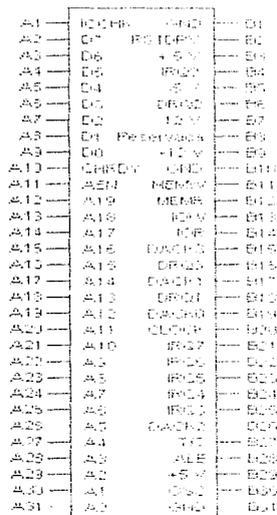
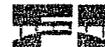


Figura 4.12



4.5 Diagramas de interconexión

4.5.1 Tarjeta de muestreo

Como se muestra en la figura 4.7 se necesitan dos interruptores analógicos para seleccionar una celda específica (un interruptor para seleccionar la línea horizontal y otro para la línea vertical correspondientes de las plantillas). En la matriz de transducción diseñada, se tiene un total de 48 líneas horizontales y 61 líneas verticales. Las líneas horizontales se conectan a los multiplexores analógicos CD4051; estos multiplexores tienen 8 canales, por lo que se utilizan 6 multiplexores para seleccionar las 48 líneas de la plantilla horizontal. Las líneas verticales se conectan a los multiplexores CD4067, como estos multiplexores cuentan con 16 canales analógicos, se utilizan 4 para seleccionar las 61 líneas de la plantilla vertical.

Las líneas verticales conectadas a los canales de los multiplexores CD4067, son seleccionadas por medio de las líneas de selección A, B, C y D (terminales 10 a 13) y por la línea de habilitación (terminal 15). Esta selección conecta a una sola línea o pista de la plantilla vertical a un voltaje fijo (VCC) y desconecta a las pistas restantes.

Las líneas horizontales conectadas a los canales de los multiplexores CD4051, son seleccionadas por medio de las líneas de selección A, B y C (terminales 9 a 11) de los 6 multiplexores y mediante las líneas de selección A, B y C de un séptimo multiplexor CD4051, el cual tiene conectado en 6 de sus canales, las salidas de los otros 6 multiplexores. La salida del séptimo multiplexor (DEC7) proporciona el potencial de la resistencia R_F del divisor de voltaje. Esta salida se conecta en serie con la línea horizontal seleccionada y por ende a la celda transductora seleccionada. Este voltaje es proporcional a la presión ejercida sobre la celda transductora seleccionada y se envía al convertidor A/C para ser digitalizado.

El diagrama esquemático de esta etapa de muestreo se presenta en la figura siguiente:



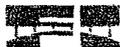
4.5.2 Tarjeta de adquisición de datos

Para poder acceder al bus de la PC desde una de sus ranuras de expansión, se tienen 32 direcciones disponibles (300h-31Fh) [19]. La tarjeta diseñada utiliza 4 direcciones consecutivas dentro de este rango; una para cada uno de los 3 puertos generados por el 8255 y una más para el control de estos puertos.

Para identificar estas direcciones se utilizan los 10 bits menos significativos del Bus PC (A0-A9). Los bits A2-A9 son comparados por el comparador 74HC688 para saber si la dirección que indica el bus está entre las citadas anteriormente. Como algunas de las direcciones disponibles para acceder al bus pueden estar ya ocupadas por otras tarjetas (como de video o de red), se tendrían que escoger 4 direcciones consecutivas libres para el correcto funcionamiento de la tarjeta de adquisición de datos. Esta selección puede realizarse mediante dos miniswitches que activan o desactivan las entradas de un grupo de bits del comparador. Las entradas del otro grupo de bits del comparador son activadas por los bits de dirección del bus PC. La salida del comparador se activa cuando ambos grupos de bits coinciden. El resultado de esta comparación activa o desactiva al 8255 y al convertidor.

El 8255 utiliza dos líneas de direcciones (A0 y A1) para diferenciar las 4 direcciones que utiliza [17]:

Dir0	Dir1	Puerto
0	0	puerto A (entrada/salida)
0	1	puerto B (entrada/salida)
1	0	puerto C (entrada/salida)
1	1	Puerto de control (salida)



Al puerto de control se envía una palabra para programar al 8255 y a sus puertos A, B y C.

El bit 7 de esta palabra selecciona al chip. 1=Activo.

Los bits 6 y 5 seleccionan el modo de trabajo de los bits del puerto A y de los cuatro bits más significativos del puerto C. 00=Modo 0, 01=Modo 1.

El bit 4 programa al puerto A, si es 1, como entrada; si es 0, como salida.

El bit 3 programa la parte alta del puerto C (bits 4-7). 0=salida, 1=entrada.

El bit 2 selecciona modo de trabajo de los bits del puerto B y de los cuatro bits menos significativos del puerto C. 0=modo 0, 1=modo 1.

El bit 1 programa al puerto B. 0=salida, 1=entrada.

El bit 0 programa la parte baja del puerto C (bits 0-3). 0=salida, 1=entrada.

Para nuestro propósito se utilizaron dos puertos como salida (puertos B y C) y un puerto como entrada (puerto A). De esta manera, el modo de trabajo para esta configuración es el modo 0 y la palabra que se manda al puerto de control para programar a los puertos A, B y C es 90h (10010000b). Los puertos de salida seleccionan las líneas horizontales y verticales de la malla transductora; y el puerto de entrada lee los bits de salida del convertidor (pines 11 a 13), los cuales representan el voltaje analógico en la celda transductora seleccionada. Las señales de control de lectura y escritura del convertidor y del PPI son controladas por el bus PC como se muestra en el siguiente diagrama:

PROGRAMACIÓN (SOFTWARE)

El software elaborado consta de dos partes. La primera parte es la correspondiente a la adquisición y procesamiento. Esta parte se encarga de hacer el barrido de todas las celdas en la matriz de transductores y de procesar la salida del convertidor A-D para posteriormente representarla en forma gráfica. El programa elaborado para esta primera parte se realizó en lenguaje "C", debido a que este lenguaje cuenta con instrucciones de entrada - salida via hardware. La segunda parte es la correspondiente a la visualización, la cual se encarga de mostrar gráficamente en la pantalla de la PC los registros de datos resultantes de parte de adquisición y de procesamiento. Además, esta parte permite editar las gráficas en caso de ser necesario. Esta segunda parte del software fue elaborada en Visual Basic debido a la facilidad en la programación bajo ambiente Windows que proporciona este lenguaje.

Adquisición y procesamiento de datos.

Como se dijo anteriormente, esta parte fue realizada en lenguaje "C" y se encarga de adquirir los datos que proporciona el convertidor A/D y de guardar esta información en archivos gráficos. Inicialmente, se programa a los puertos B y C del PPI 8255 como puertos de entradas y al puerto A como puerto de salida por medio del puerto de control (dirección 30Fh). Mediante ciclos anidados se hace un barrido para seleccionar todas las



celdas transductoras y para leer la información que el convertidor proporciona al censar el voltaje en cada celda. La selección de una celda transductora se realiza mediante la selección de las líneas horizontales y verticales correspondientes; esta selección de líneas se hace mediante los puertos de salida (direcciones 30Dh y 30Eh). La información otorgada por el convertidor se lee mediante el puerto de entrada (dirección 30Ch). La información proporcionada por el convertidor es procesada para ser representada mediante colores. Esta información, entonces, es guardada como una grafica de colores en formato de mapa de bits (BMP) para poder ser manejada en ambiente windows y aprovechar las ventajas que proporciona este sistema operativo. A continuación se presenta el listado del programa encargado de la adquisición y del procesamiento de los datos [20].

5.1.2 Programa PUERTOS.CPP

```
#include <stdio.h>           Librerías de "C" que incluyen funciones
#include <dos.h>             utilizadas en el programa.
// Asignación de variables
unsigned char arreglo2[64][192],arreglo[64][64];
unsigned char mapa[] = {0x42,0x4d,0x36,0x30,0,0,0,0,0,0x36,0,0,0,0x28,0,
                        0,0,0x40,0,0,0,0x40,0,0,0,1,0,0x18,0,0,0,
                        0,0,0,0x30,0,0,0x0e,0x0e,0,0,0xd8,0x0e,
                        0,0,0,0,0,0,0,0,0};

// Programa principal
int main(void)
{
    FILE *stream;
    int port = 0x301,i,j,k,l=0,n,m;           El puerto de control del 8255 es el 301 h.
    unsigned char value = 0x90;            Al mandar un 90h al puerto de control del 8255
    ouporth[port, value];                 se programan los puertos B y C de este como salidas y
                                          el puerto A como entradas.
}
```



```
for(j=0;j<48;j++){
```

Se seleccionan las líneas horizontales y verticales que identifican cada celda de la matriz de transducción y se lee la información proporcionada por el convertidor.

```
    outportb(port-2, j);
```

Selección de la línea horizontal de la matriz

```
    l=4;
```

Se ignoran las primeras 4 líneas verticales

```
    for(i=0x70;i<0x80;i++){
```

```
        outportb(port-1,i);
```

Selección de la línea vertical

```
        arreglo[i][j]=inportb(port-3);
```

Lectura de la salida del convertidor

```
    };
```

```
    }
```

```
    for(i=0xb0;i<0xc0;i++){
```

```
        outportb(port-1,i);
```

```
        arreglo[i][j]=inportb(port-3);
```

```
    };
```

```
    }
```

```
    for(i=0xd0;i<0xe0;i++){
```

```
        outportb(port-1,i);
```

```
        arreglo[i][j]=inportb(port-3);
```

```
    };
```

```
    }
```

```
    for(i=0xe0;i<0xe7;i++){
```

```
        outportb(port-1,i);
```

```
        arreglo[i][j]=inportb(port-3);
```

```
    };
```

```
    }
```

```
}
```

Asignación de colores en formato RGB.

```
outportb(port, value);
```

```
for (i=0;i<64;i++){
```

```
    l=0;
```

```
    for (j=0;j<32;j++){
```

```
        k=i*3;
```

```
        if (i<4){
```

```
            arreglo[i][j]=255;
```

```
            arreglo[i][45-j]=255;
```

```
        }
```

```
        if (i<58){
```

```
            arreglo[i][j]=255;
```

```
            arreglo[i][45-j]=255;
```

```
        }
```

```
}
```

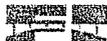


```

        if (arreglo[i][j] < 127){
            m = arreglo[i][j]-128;
            n = 1;
        }
        arreglo2[i][k] = 255;
        arreglo2[i][k+1] = arreglo[i][j];
        arreglo2[i][k+2] = 255-2*(arreglo[i][j]-2*m)+n;
        if (l == 45) {
            arreglo2[i][k] = 255;           valor asignado de Rojo en RGB
            arreglo2[i][k+1] = 255;       valor asignado de Verde en RGB
            arreglo2[i][k+2] = 255;       valor asignado de Azul en RGB
        }
        m = 0;
        n = 0;
        l++;
        if (arreglo[i][45-j] < 127){
            m = arreglo[i][45-j]-128;
            n = 1;
        }
        arreglo2[i][k] = 255;           valor asignado de Rojo en RGB
        arreglo2[i][k+1] = arreglo[i][45-j]; valor asignado de Verde en RGB
        arreglo2[i][k+2] = 255-2*(arreglo[i][45-j]-2*m)+n; valor asignado de Azul
        if (l == 45) {
            arreglo2[i][k] = 255;
            arreglo2[i][k+1] = 255;
            arreglo2[i][k+2] = 255;
        }
        m = 0;
        n = 0;
        l++;
    }
}

if ((stream = fopen("TEST.bmp", "wb")) == NULL) /* abre el archivo TEST.BMP */
{
    fprintf(stderr, "No se puede abrir el archivo.\n");
    return 1;
}
fwrite(&ima, sizeof(ima), 1, stream); /* se escribe el encabezado del archivo */
fwrite(&arreglo2, sizeof(arreglo2), 1, stream); /* se escribe la matriz en el archivo */
fclose(stream); /* se cierra el archivo */
return 0;
}

```



5.2 Visualización

La segunda parte del software fue elaborada en "Visual Basic" con el objeto de aprovechar las ventajas que proporciona el ambiente Windows, entre las que están:

- gran capacidad en la interacción de programas,
- presentación grafica de los programas o
- interacción amigable con el usuario.

El programa consta de varias ventanas (Edición, Historial, Paleta de colores y Acerca de Oclusal). En la ventana de Historial se pueden observar registros previamente guardados y en la ventana de edición es donde se puede realizar un nuevo registro y editarlo si es necesario. La ventana Edición se muestra en la figura 5.1. Cuenta con un menú principal, una area de edición, una paleta de colores, herramientas de edición, un conjunto de 6 cuadros que permite visualizar 6 graficas de registros, indicadores de la posición del raton cuando este se encuentra en el area de edición, indicador del valor promedio de presión representado por el color señalado por el raton en el área de edición, colores de edición seleccionados para cada boton del raton, un boton de captura de nuevos registros y un boton de captura dinamica de registros.

El menú principal contiene los siguientes submenús:

- Archivo.- con este submenú se pueden guardar los registros capturados y/o editados; se pueden abrir archivos que habian sido guardados anteriormente y también tiene una opción para salir del programa.
- Editar.- en este submenú se tienen varias opciones para editar los registros como rotar, invertir colores, voltear vertical y horizontalmente, copiar la imagen al portapapeles de Windows y pegar imágenes del portapapeles al área de edición.

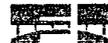
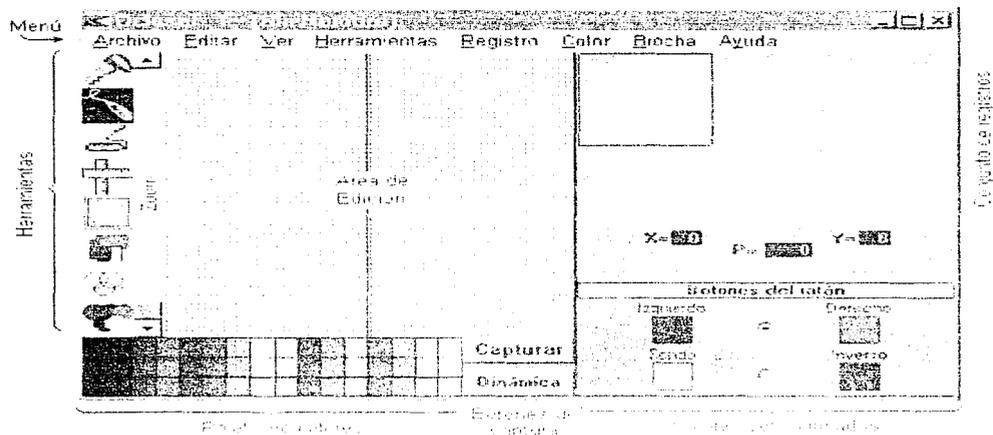


Figura 5.1



- Ver.- en este submenú se permite la visualización o no de algunas partes de la ventana principal del programa. Además, se puede activar y seleccionar el tipo de cuadrícula o malla en el área de edición, que sirve para visualizar el contorno de las celdas en la gráfica.

- Herramientas.- permite seleccionar las herramientas de edición como lápiz, líneas, circunferencias, círculos, rectángulos, rectángulos rellenos, relleno y selección de área. Esta selección de herramientas también se puede realizar haciendo clic en el área de herramientas.

- Registro.- permite seleccionar un registro del conjunto de 8 registros y mostrarlo amplificado en el área de edición.

- Color.- Abre archivos de configuración de la paleta de colores; también permite guardar nuevas configuraciones.



- Brocha.- Permite seleccionar el tamaño de la brocha de edición.
- Ayuda.- permitir obtener ayuda sobre el manejo del programa.

Con el botón de captura se puede realizar un nuevo registro. Al hacer clic en él, se ejecuta el programa de adquisición y el archivo gráfico guardado por éste es cargado en el área de edición.

El botón de captura dinámica realiza 6 capturas consecutivas de registros permitiendo visualizar un efecto dinámico.

A continuación se muestran los programas correspondientes a la visualización.

5.2.1 OCLUSAL.GBL (Declaración de variables globales)

DECLARACION

* Tipo usado por el compilador para manejar los cambios en los registros

Type GUARDAREGISTRO

NombreArchivo

PathCompleto

Modificado

End Type

* Declaración de Funciones API

Declare Function BitBlt Lib "GDI" (ByVal destHdc, ByVal X, ByVal Y, ByVal w, ByVal h, ByVal srcHdc, ByVal srcX, ByVal srcY, ByVal Rop) As Long

Declare Function CreateCompatibleBitmap Lib "GDI" (ByVal hDc, ByVal w, ByVal h) As Long

Declare Function CreateCompatibleDC Lib "GDI" (ByVal hDc) As Long

Declare Function DeleteDC Lib "GDI" (ByVal hDc) As Long

Declare Function DeleteObject Lib "GDI" (ByVal hObject) As Long

Declare Function ExtFloodFill Lib "GDI" (ByVal hDc, ByVal X, ByVal Y, ByVal crColor As Long, ByVal wFillType) As Long

Declare Function GetBitmapBits Lib "GDI" (ByVal hBitmap, ByVal dwCount As Long, ByVal lpBits As Long) As Long

Declare Function GetDeviceCaps Lib "GDI" (ByVal hDc, ByVal nIndex) As Long

Declare Function GetMenuItem Lib "USER" (ByVal hwnd) As Long

Declare Function GetNearestColor Lib "GDI" (ByVal hDc, ByVal RGBColor As Long) As Long



```

Declare Function GetPrivateProfileInt Lib "Kernel" (ByVal Appname As String, ByVal KeyName As String, ByVal DEFAULT As Integer, ByVal NombreArchivo As String) As Integer
Declare Function GetPrivateProfileString Lib "Kernel" (ByVal Appname As String, ByVal KeyName As String, ByVal DEFAULT As String, ByVal ReturnedString As String, ByVal MaxSize, ByVal NombreArchivo As String) As String
Declare Function GetSubMenu Lib "User" (ByVal hwnd As Integer, ByVal Position As Integer) As Integer
Declare Function GetSystemMenu Lib "User" (ByVal hwnd As Integer, ByVal bRevert As Boolean) As Integer
Declare Function GlobalLock Lib "Kernel" (ByVal mMem As Integer) As Integer
Declare Function GlobalUnlock Lib "Kernel" (ByVal mMem As Integer) As Integer
Declare Function InvertRect Lib "User" (ByVal hwnd As Integer, ByVal IpRect As RECT) As Integer
Declare Function RemoveMenu Lib "User" (ByVal hwnd As Integer, ByVal nPosition As Integer, ByVal dwFlags As Integer) As Integer
Declare Function SelectObject Lib "GDI" (ByVal hwnd As Integer, ByVal hObject As Integer) As Integer
Declare Function SetBitmapBits Lib "GDI" (ByVal hBitmap As Integer, ByVal dwCount As Integer, ByVal lpBits As Byte) As Integer
Declare Function StretchBlt Lib "GDI" (ByVal destHdc As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal w As Integer, ByVal h As Integer, ByVal srcHdc As Integer, ByVal srcX1 As Integer, ByVal srcY1 As Integer, ByVal srcW As Integer, ByVal srcH As Integer, ByVal Rop As Integer) As Integer
Declare Function TrackPopupMenu Lib "User" (ByVal hwnd As Integer, ByVal ulFlags As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal r2 As Integer, ByVal r3 As Integer) As Integer
Declare Function WinHelp Lib "User" (ByVal hwnd As Integer, ByVal HelpFile$ As String, ByVal wCommand As Integer, ByVal dwData As Integer) As Integer
Declare Function WritePrivateProfileString Lib "Kernel" (ByVal Appname As String, ByVal KeyName As String, ByVal NewString As String, ByVal NombreArchivo As String) As Integer
' Claves del archivo INI de ejemplo
Global Const APL_NOMBRE = "Oclusal"
Global Const CLAVE_COLOR = "Línea de Colores"
Global Const CLAVE_MALLA = "Malla"
Global Const CLAVE_MALLA_LINEAL = "Malla Lineal"
Global Const CLAVE_MALLA_PUNTEADA = "Malla punteada"
Global Const CLAVE AREA_ESTADO = "Área de estado"
Global Const CLAVE HERRAMIENTAS = "Herramientas"
Global Const CLAVE COLORES = "Paleta de colores"
Global Const CLAVE ZOOM = "Zoom"
Global Const CLAVE ENFOQUE = "Entoque"
Global Const CLAVE ORILLA = "Orilla del registro"
Global Const CLAVE FILTRO = "Filtro"
Global Const CLAVE LINEA_MEDIA = "Línea Media"
Global Const CLAVE POPUP = "Menú Popup"
Global Const CLAVE MOSTRAR_REGISTROS = "Mostrar registros"
Global Const CLAVE DEFAULT = "Colores de Default"
Global Const CLAVE SOLIDOS = "Colores Sólidos"
Global Const CLAVE INI = "Archivo de colores INI"
Global Const ARCHIVO_INI = "Oclusal.INI"

```



* Valor inicial del campo Nombre_Archivo para GUARDAREGISTRO.

Global Color: SIN_NOMBRE = "[Sin Nombre]"

* Índices de colores operacionales del Editor

Global Color: MOUSE_COLORES = 0

Global Color: COLORES_DE_FONDO = 1

* Variables de control para el manejo de archivos en Historial

Global Color: ARCHIVO_NO_ENCONTRADO = 53

Global Color: Nombre_Archivo_Modificado = 0

Global Color: ARCHIVO_MODIFICADO = 1

Global Color: DIR_Modificado = 2

* Tamaño de la celda de presentación en Historial 64x64 pixeles = 2

Global Color: CELDA_REGISTRO = 66&

* Índices para la Paleta de Colores

Global Color: ELEMENTO_ROJO = 0

Global Color: ELEMENTO_VERDE = 1

Global Color: ELEMENTO_AZUL = 2

* Varios

Global Color: ALTURA_EGA = 7000

Global Color: RECUADRO = 10

Global Color: DEFINIR_COLORES = 1000

Global Color: Editor de Oclusal = 0

Global Color: Ventana de Historial = 1

* Información matricial de registros y colores

Global InfoRegis(5) = GUARDAREGISTRO

Global Colores(47) =

Global MouseColores(3) =

* Variables globales de Agencia de Oclusal

Global Mundo =

Global Autores =

PROGRAMACION



* Variables globales de Historial

```
Global MáxRegistros           As Integer
Global RegistroFilas          As Integer
Global RegistroColumnas       As Integer
Global AnchoHistorialMin      As Integer
Global AlturaHistorialMin     As Integer
Global MostrandoTodosRegis    As Integer
```

* Varios

```
Global VentanaPrincipal       As Integer
Global EditorAbierto          As Integer
Global HistorialAbierto       As Integer
Global Paleta de colores Abierta As Integer
Global EditorInicializado     As Integer
Global UltimoV Modificado     As Integer
Global UltimoS Modificado     As Integer
Global NumColoresEs256        As Integer
Global Texto                   As String
Global CRLF                     As String * 2
Global A TAB                     As String * 1
```

* CONSTANTES TOMADAS DEL ARCHIVO CONSTANT.TXT DE VB

* Estado de la ventana

```
Global Const MINIMIZADO = 1
```

* Formato del portapapeles

```
Global Const CF_BITMAP = 2
```

* DragOver

```
Global Const OVER = 2
```

* Mostrar parámetros

```
Global Const MODAL = 1
```

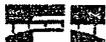
```
Global Const MODELESS = 0
```

* Colores RGB

```
Global Const NEGRO = &H0&
```

```
Global Const GRISOSCURO = &H808080
```

```
Global Const GRIS = &HC0C0C0
```



```
* Global Colors
Global Color AZUL = &HFF0000
Global Color VERDE = &HFF00&
Global Color OCEANO = &HFFF00
Global Color ROJO = &HFF&
Global Color PURPURA = &HFF00FF
Global Color AMARILLO = &HFFFF&
Global Color BLANCO = &HFFFFFF
```

* Apuntadores del ratón

```
Global Color DEFAULT = 0
Global Color CRUZ_PEQUEÑA = 2
Global Color SIZE = 5
Global Color NORTE = 10
Global Color RELOJ de ARENA = 11
```

* Variables del modo de dibujo

```
Global Color INVERSO = 6
Global Color COPY PEN = 13
```

* Valores de las propiedades del estilo de dibujo

```
Global Color SOLIDA = 0
Global Color PUNTEADA = 2
```

* Valores del estilo de relleno

```
Global Color TRANSPARENTE = 1
```

* Código de teclas

```
Global Color KEY_BACK = &H8
Global Color KEY_SHIFT = &H10
Global Color KEY_CONTROL = &H11
Global Color KEY_INSERT = &H12D
Global Color KEY_DELETE = &H12E
Global Color KEY_F1 = &H170
```

* Parámetros de shift, ctrl

```
Global Color SHIFT_MASK = 1
Global Color CTRL_MASK = 2
Global Color ALT_MASK = 4
```

* Botones del ratón

```
Global Color Botón_Izquierdo = 1
Global Color Botón_Derecho = 2
```



5.2.2 OCLUSAL.BAS

```
*Declaraciones
```

```
Defini A-Z
```

```
Dim Path,Archivo,Ayuda As String
```

```
*El programa empieza abriendo la ventana principal de edición, pero también se
* puede abrir primero la ventana de Historial, si en la línea de comando se
* anexa una H o una h
```

```
Sub Principal
```

```
    *Reconocimiento del modo de video. Si es menor a EGA se termina Oclusal
```

```
    If Screen.Height <= ALTURA_EGA Then
```

```
        MsgBox "Oclusal requiere monitor EGA o superior.", 16, "Oclusal"
```

```
        End
```

```
    Else
```

```
        *Ya que no es posible definir valores como TAB, CR, y LF a constantes
        *alfanuméricas, los valores de TAB y CR LF que se usan frecuentemente
        *en Oclusal en el despliegue de mensajes, son entonces asignados
        *a los valores globales alfanuméricos A_TAB y CRLF
```

```
        A_TAB = Chr(9)
```

```
        CRLF = Chr(13) & Chr(10)
```

```
    If Not Encontrar_Archivo_de_Ayuda() Then
```

```
        Texto = "No se encontró Oclusal.HELP en la ruta de acceso." & App.Path & CRLF & CRLF
```

```
        Texto = Texto & "Windows busca los archivos de ayuda en el PATH del sistema,"
```

```
        Texto = Texto & " por lo que necesita copiar OCLUSAL.HELP a un directorio que "
```

```
        Texto = Texto & "esté incluido en el PATH si se desea obtener ayuda durante la ejecución
```

```
de Oclusal."
```

```
        MsgBox Texto, 48, "La Ayuda de Oclusal no está disponible"
```

```
    End If
```

```
*Determina que ventana abre primero la de Edición o el Historial
```

```
    If (Command = "H") Or (Command = "h") Then Command = "E" Else
```

```
        *Edición es primero
```

```
        VentanaPrincipal = Editor_de_Oclusal
```

```
        Editor.Show
```




Seleccionar_Registro_Nuevo

Editor.Pic_ToolPalette, 0, 0, 0, 0

hDC

* Si no se descartan los cambios del registro.

Editor.Registros(Registro_Actual): 1

Amplificar_Registro 0, 0, 63, 63

hEdit

hWndEdit

Sub_Actualiza_Registro_Pic Ctrl = Control

* No se ha modificado directamente el registro, entonces se copian los mapas de bits
 * Mask e Image sobre el bitmap para que reflejen los cambios en el Area
 * de estado.

Pic_Ctrl = Editor.Pic_Image.Image

* Si hubo un cambio la bandera Modificado se activa.

h Pic_Ctrl.Tag = Editor.Reg_Deshacer.Tag + 1, hInfoRegis(Registro_Actual).Modificado = True

End Sub

Sub Actualizar_Colores_del_Ratón_Botón_N (Single, y, x, hDC, y)

* Esta rutina se invoca cuando se hace click en una paleta de colores

* para establecer el color seleccionado en Colores del ratón.

* Tambien invoca la ventana de Paleta de colores cuando se ha hecho

* Doble click en la paleta de colores del Editor.

Dim color As Long, ColorSólido As Long

* Las paletas de colores son solo dibujos de control, así que se debe

* calcular el color seleccionado basándonos en las coordenadas del ratón.

IndiceColor = Fix X / Fix y - 16

* Se obtiene el color del arreglo de colores

color = Colores(IndiceColor)

* El color muestra el modo de color de 16 bits, por eso se debe obtener el

* color sólido más próximo del color seleccionado, debido a que los colores

* de FONTC y BITMAPS, solo pueden ser sólidos.

ColorSólido = GetNearestColor(Editor.hDC, color)



```
End If
```

```
    * Si se efectúa un doble click, se inicializa esta bandera y se abre  
    * la ventana de Paleta de colores.  
    Fue_DobleClick = True  
    ColorPalette.Show
```

```
    * La inicialización de la Paleta de colores se realiza con el evento  
    * GotFocus para entrar al control.  
    ColorPalette.Paleta de Colores.Show
```

```
End If
```

```
    * Se notifica al usuario que no se puede seleccionar un color no sólido  
    * para FONDO e INVERSO.
```

```
    MsgBox "Los colores del Fondo e Inverso solo pueden ser sólidos", 16, "Error"
```

```
End If
```

```
    * Se obtienen los índices de los colores del botón:  
    * 0 - Color izquierdo  
    * 1 - Color derecho  
    * 2 - Color de fondo  
    * 3 - Color INVERSO
```

```
Index = Editor.Opt_Mouse(COLORES_DE_FONDO).Value (-2) + Botón + 1
```

```
    * Se colorea el nuevo color
```

```
MouseColores(Index) = Colores(IndiceColor)
```

```
    * Si se cambia el color de fondo automáticamente debe cambiarse  
    * el color INVERSO y viceversa.
```

```
End If
```

```
    Editor.Registros(0).Color = (1, 1, MouseColores(Index))
```

```
    MouseColores(-1 - Index - 5) = Editor.Registros(0).Color(1, 1)
```

```
    Editor.Registros(0).Color =
```

```
End If
```

```
End If
```

```
    * Cuando los colores de Fondo e INVERSO se modifican, se debe cambiar  
    * el Fondo de los registros del área de estado y también del registro  
    * de deshacer con el nuevo color de Fondo, además de redibujar el registro  
    * seleccionado en el área de Edición.
```

```
    Enmarcar_Registro_Actual
```



```

For I = 0 To 5
    Editor.Registros(I).BackColor = MouseColores(2)
Next I
Editor.Reg_Deshacer.BackColor = MouseColores(2)
Amplificar_Registro 0, 0, 63, 63
End If
End If

' Se dibujan los nuevos colores del ratón en la parte inferior
' del área de estado
Dibujar_Colores_del_Ratón
End Sub

Sub Amplificar_Registro (X1, Y1, X2, Y2)
' Hay varias situaciones en las que todo o parte de un registro necesita ser amplificado y desplegado
' en el área de edición. Esta rutina se utiliza para realizar la ampliación mediante la rutina API
' StretchBlt().
' Nos aseguramos de que las coordenadas X1 y Y1 se refieran a la esquina superior izquierda y
' de que X2 y Y2 se refieran a la esquina inferior derecha del área que se amplificará.
If X1 < X2 Then Canjear_Valores X1, X2
If Y1 < Y2 Then Canjear_Valores Y1, Y2

' El área que se amplificará no deberá tener ningún punto fuera del área
' del registro, entonces se debe verificar esta situación y ajustar los valores
' si fuera necesario.
If X1 < 0 Then X1 = 0
If X2 > 63 Then X2 = 63
If Y1 < 0 Then Y1 = 0
If Y2 > 63 Then Y2 = 63

' Se calcula la altura y la anchura del área a amplificar.
Ancho_Area = X2 - X1 + 1
Altura_Area = Y2 - Y1 + 1

' Se calcula la anchura, la altura y la esquina superior izquierda
' del área amplificada.
X_Ampl = X1 * Tamaño_del_Píxel
Y_Ampl = Y1 * Tamaño_del_Píxel
Ancho_Ampl = Ancho_Area * Tamaño_del_Píxel
Altura_Ampl = Altura_Area * Tamaño_del_Píxel
' Se amplifica el registro. Se copia la imagen del registro del área
' de estado y se coloca en el área de edición, manteniendo siempre el
' tamaño en el área de edición un múltiplo de 64 (tamaño de un registro).

```



H TamañoImagen = 1024 * 1024

R StretchBlt(Editor.Pic_Edit.hDC, X_Ampl, Y_Ampl, Ancho_Ampl, Altura_Ampl,
Editor.Registros(Registro_Actual).hDC, X1, Y1, Ancho_Area, Altura_Area, SRCCOPY)

* Se redibuja la malla en el área amplificada de la opción de ver malla
* Esta selección a:

R Editor.Selección_Menu_VER(MID_MALLA).Checked = 1 * Redibujar Malla
(Editor.Pic_Edit.hDC), X1, Y1, X2, Y2

H

R BitBlt(Editor.Pic_Edit.hDC, X_Ampl, Y_Ampl, Ancho_Ampl, Altura_Ampl,
Editor.Pic_EditTemp.hDC, X_Ampl, Y_Ampl, SRCCOPY)

H

* Se verifica si ha cambiado alguna de las variables de selección de esta
* Se debe redibujar el contenido de la selección

H Selección en Movimiento **H**

* Se calcula la anchura y la altura del mapa de bits contenido en la
* selección, manteniéndose siempre los valores globales

* X1 de selección - Y1 de selección - Y1 de selección y Y2 de selección

Ancho_Area = X2 de selección - X1 de selección

Altura_Area = Y2 de selección - Y1 de selección

* Se calcula la anchura y la altura del área amplificada.

Ancho_Ampl = Ancho_Area * Tamaño del Pixel

Altura_Ampl = Altura_Area * Tamaño del Pixel

* Se determina el modo de selección, opaco o no Opaco.

H Opaco **H**

* Selección opaca: Se amplifica la selección incluyendo los atributos de fondo o inverso.

R StretchBlt(Editor.Pic_Edit.hDC, X1Región, Y1Región, Ancho_Ampl, Altura_Ampl,
Editor.Pic_Work.hDC, X1 de selección, Y1 de selección, Ancho_Area, Altura_Area, SRCCOPY)

H

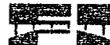
* Selección no opaca: Se amplifica la selección sin incluir los atributos de fondo o inverso.

R StretchBlt(Editor.Pic_Edit.hDC, X1Región, Y1Región, Ancho_Ampl, Altura_Ampl,
Editor.Pic_TempMask.hDC, X1 de selección, Y1 de selección, Ancho_Area, Altura_Area,
SRCAND)

R StretchBlt(Editor.Pic_Edit.hDC, X1Región, Y1Región, Ancho_Ampl, Altura_Ampl,
Editor.Pic_TempImage.hDC, X1 de selección, Y1 de selección, Ancho_Area, Altura_Area,
SRCINVERT)

H **H**

H **H**



```

' Si hay una seleccion hecha, se redibujó el rectángulo de seleccion.
f Seleccionando : . Dibujar Rectángulo de Selección
' Editor.Selección Menú VER(MIB_LINEA_MEDIJA).Checked f=on
' Editor.Pic Edit. : 32 Tamaño del Pixel, 0-32 Tamaño del Pixel, 64
Tamaño del Pixel : ANILRILLO
' Editor.Pic Edit. : 32 Tamaño del Pixel 1, 0-32 Tamaño del Pixel 1, 64
Tamaño del Pixel : NEGRO
' Editor.Pic Edit. : 32 Tamaño del Pixel 1, 0-32 Tamaño del Pixel 1, 64
Tamaño del Pixel : NEGRO
' Fin de
End Sub

```

```

' Se Archivar Registro PathCompleto : NombreArchivo
' Archiva el Registro y actualiza el menú registro y el encabezado del Editor
' con el Nombre del Archivo del registro.
' Se registra el Nombre del Archivo y el Path
InfoRegis(Registro_Actual).NombreArchivo = NombreArchivo
InfoRegis(Registro_Actual).PathCompleto = PathCompleto
InfoRegis(Registro_Actual).Modificado = on
' Se despliegan los Archivos de Archivos y el Path en el menú registro del Editor
' Editor.Selección Menú Registros(Registro_Actual).Caption = "&" + NombreRegistro_Actual
1 " - |" PathCompleto " |" A TAB NombreArchivo
' Se despliega el Nombre del Archivo en el encabezado del Editor
Editor.Caption = "Clausur" + Registro_Actual 1 " - " NombreArchivo
' Se archiva el registro en un directorio particular especifico.
f RptRr, PathCompleto, 1 " " PathCompleto - PathCompleto " "
Navegacion : Editor.Registros(Registro_Actual).Picture, PathCompleto - NombreArchivo

```

End Sub

Sub Canjear Valores Param1, Param2.

```

' Debido a que no hay una función Swap en Visual Basic, esta rutina
' se encarga de realizar el canjeo de valores.

```

```

temp = Param1
Param1 = Param2
Param2 = temp

```

End Sub



Forma 10: Desea descartar los cambios

* Esta subrutina determina si un registro fue modificado desde la última vez que se guardó, y advierte al usuario si fue así.

Texto ""

Desea descartar los cambios

* Verifica si el registro fue Modificado desde la última vez que se guardó.

InfoRegis(Registro_Actual).Modificado

* Establece las propiedades necesarias para abrir el dispositivo MIC.

Editor.MIMControl.Wait

Editor.MIMControl.Wait

Editor.MIMControl.Shareable

Editor.MIMControl.DeviceType = "Wave Audio"

Editor.MIMControl.FileName = App.Path + "Oclusa1.WAV"

* Habilita el dispositivo de audio MIC.

Editor.MIMControl.Command = "Open"

Editor.MIMControl.Command = "Play"

* Informa al usuario que el registro ha sido modificado.

Texto Texto "Registro" A TAB ""

Texto = Texto "Nombre" A TAB InfoRegis(Registro_Actual).Nombre.Archivo CRLF

Texto = Texto "Path" A TAB InfoRegis(Registro_Actual).PathCompleto CRLF

CRLF

Texto = Texto "¿Desea descartar los cambios ?"

Desea descartar los cambios

End If

Editor.MIMControl.Command = "Close"

End Form

Sub Dibujar Colores del Ratón

* Cuando se selecciona un color por cualquiera de los botones del ratón, o cuando el área de estado necesita ser redibujada, se requiere esta subrutina para dibujar los 4 cuadros de color en la parte inferior del menú de estado. Estos cuadros son rellenados con los colores seleccionados por los botones del ratón.

* Calcula el centro en la anchura del área de estado.

Middle = Editor.Area de Estado.ScaleWidth / 2

* Se dibujan los 4 cuadros de color.

For I = 0 To 3

* Los cuadros son centrados en las mitades izquierda y derecha del área de estado, la altura y



* anchura de estos se hacen igual a la altura de los botones de opción usados para seleccionar los colores. Igualmente Derecha y Fondo Inverso, con esta información se calcula la posición de las esquinas de los cuadros de color.

X1 = (I * W) / 2 + Middle - (Middle - Editor.Opt_Mouse(1 - 2), Height) - 2

X2 = X1 + Editor.Opt_Mouse(1 - 2), Height

Y1 = Editor.Opt_Mouse(1 - 2), Top

Y2 = Y1 + Editor.Opt_Mouse(1 - 2), Height

* Dibuja el cuadro de color.

Editor.Area_de_estado, I = X1, Y1 - X2, Y2, MouseColores(I), BF

* Se dibuja un cuadro para cada color.

Editor.Area_de_estado, I = X1, Y1 - X2, Y2, NEGRO, B

Next I

* Establece el valor de cada I en el área de estado para localizar el lugar

* en que las coordenadas del cuadro se desplazarán.

Editor.Area_de_estado.CurrentY = Editor.Registros(5), Top + Editor.Registros(5), Height

RECUADRO 1

End Sub

Sub Dibujar_Paleta_de_Colores (Paleta_de_Colores) Control

* Esta rutina es invocada cuando la paleta de colores necesita ser dibujada o redibujada. La paleta de colores consiste de 3 filas de 16 colores. Para que el despliegue y la determinación de un color seleccionado haciendo clic en la paleta de colores sean fáciles de realizar, se establece una escala en la paleta de colores que corresponda al número de renglones y columnas.

Paleta_de_Colores, I = (0, 0)-(16, 3)

* Desplegando la paleta de colores de columna en columna

For I = 0 To 15

* Desplegando una columna de colores

Paleta_de_Colores, I = (I, 0)-(I + 1, 1), Colores(I), BF

Paleta_de_Colores, I = (I, 1)-(I + 1, 2), Colores(I + 16), BF

Paleta_de_Colores, I = (I, 2)-(I + 1, 3), Colores(I + 32), BF

* Se dibuja una línea vertical a la izquierda de la columna para

* visualizar una división entre las columnas, excepto en la primera columna,

* ya que aquí no es necesario debido al recuadro de la paleta de colores.

BF I = Paleta_de_Colores, I = (I, 0)-(I, 3)

Next I

* Se dibujan 2 líneas horizontales para y surtir una división entre renglones.

Paleta_de_Colores, I = (0, 1)-(16, 1)

Paleta_de_Colores, I = (0, 2)-(16, 2)

End Sub



Sub Dibujar Rectángulo de Selección :

• Cuando se ha hecho una selección, se está haciendo o una selección se está moviendo, o cuando el área de Edición necesita redibujarse mientras una selección está activa, se recurre a esta subrutina para dibujar o redibujar un rectángulo alrededor de la selección.

• Se establece el modo de dibujo (INVERSO) para dibujar sobre lo desplegado en el área de Edición y así se pueda apreciar el rectángulo de selección.

```
Editor.Pic Edit.DrawMode = INVERSO
```

• Para distinguir entre una selección y una selección en movimiento, se usa una línea punteada para una selección y una línea sólida para una selección en movimiento.

```
Editor.Pic Edit.DrawStyle = SOLIDA Else
```

```
Editor.Pic Edit.DrawStyle = PUNTEADA
```

• Si la selección se incrementa hasta el borde derecho o hasta el borde inferior del área de Edición, se ajusta 1 píxel en la dirección del rectángulo y en la base para asegurar que el rectángulo de selección sea visible completamente.

```
if X2Región = Tamaño del Pixel - 64 then XAjustar = 1
```

```
if Y2Región = Tamaño del Pixel - 64 then YAjustar = 1
```

• Se dibuja el rectángulo:

```
Editor.Pic Edit.Draw (X1Región, Y1Región)-(X2Región + XAjustar, Y2Región + YAjustar), B
```

```
Editor.Pic Edit.DrawStyle = SOLIDA
```

End Sub

Función Encontrar Archivo de Ayuda :

• Busca el archivo de ayuda.

```
Dim Path As String, CurrentDir As String
```

```
Dim Encontrado As Boolean
```

```
CurrentDir = App.Path
```

```
if Right (CurrentDir, 1) = "\" then CurrentDir = CurrentDir + "*"
```

```
Encontrado = Dir$(CurrentDir + "Help.exe") = ""
```

```
if not Encontrado then
```

```
Path = App.Path + "PATH"
```

```
if Path = "" then
```

```
for i = 0 to Path.Length - 1
```

```
SemiColon = Mid$(Path, i, 1)
```

```
if
```

```
CurrentDir = CurrentDir + Path + SemiColon + 1
```

```
if Right (CurrentDir, 1) = "\" then CurrentDir = CurrentDir + "*"
```

```
Encontrado = Dir$(CurrentDir + "Help.exe") = ""
```

```
Path = Path + SemiColon + Path + SemiColon
```

```
SemiColon = Mid$(Path, i, 1)
```



```

    * Comp.Width ((SemiColon = 0) * Val del Encontrado)
  End If
End If
If Encontrado = 1
  Path.Archivo.Ayuda CurrentDir "Help.exe"
  App.HelpFile CurrentDir "Help.exe"
End If
Encontrar Archivo de Ayuda Encontrado
  Comp.Line = 0
End Function

```

Del Enmarcar_Registro_Actual

* Para distinguir el registro seleccionado se le dibuja un marco con esta subrutina.

Del Editor.Selección Menú VER(MIID, ORILLA),Checked Draw

```

  * Dibuja un marco en el registro seleccionado
  X1 = Editor.Registros(Registro_Actual).Left - 1
  X2 = Editor.Registros(Registro_Actual).Left + Editor.Registros(Registro_Actual).Width
  Y1 = Editor.Registros(Registro_Actual).Top - 1
  Y2 = Editor.Registros(Registro_Actual).Top + Editor.Registros(Registro_Actual).Height
  Editor.Area de Estado, (X1, Y1)-(X2, Y2), NEGRO, B
End If
  * Establece el valor actual de la barra de desplazamiento al igual que se le especifica las coordenadas
  * del mismo
  Editor.Area de Estado.CurrentY = Editor.Registros(5).Top + Editor.Registros(5).Height
RECUADRO = 1
End Sub

```

Sub Escalar_Región_Disminuir, X1, Y1, X2, Y2, VerificarX1Y1

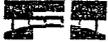
* La Región rectangular que siempre está definida por las variables globales X1Región, Y1Región, X2Región y Y2Región, en la base para la mayoría de las herramientas de edición, es escalada frecuentemente desde la escala del área de Edición a del área del Registro actual y viceversa.
 * Esta rutina realiza la escalación necesaria en cualquier dirección.
 * Determina si se va a disminuir o aumentar.

Del Disminuir Draw

```

  * Se escalan las variables Globales al tamaño de un Registro
  X1 = X1Región / Tamaño del Pixel
  Y1 = Y1Región / Tamaño del Pixel

```



```
X2 = X2Región * Tamaño del Pixel  
Y2 = Y2Región * Tamaño del Pixel
```

* Si se requiere, se asegura de que X1 y Y1 se refieren a la esquina superior izquierda
* y de que X2 y Y2 a la esquina inferior derecha de la Región.

```
if VerificarX1Y1 then
```

```
    if X1 < X2 then * Cambiar Valores X1, X2
```

```
    if Y1 > Y2 then * Cambiar Valores Y1, Y2
```

```
End if
```

```
else
```

* Se escalan los valores X1, Y1, X2, Y2 al tamaño del área de Edición

* usando los valores de los atributos Global's

```
X1Región = X1 * Tamaño del Pixel
```

```
Y1Región = Y1 * Tamaño del Pixel
```

```
X2Región = X2 * Tamaño del Pixel
```

```
Y2Región = Y2 * Tamaño del Pixel
```

```
End if
```

```
End Sub
```

```
Sub Invert Control (Ctrl = Control)
```

```
Dim Rectangle As RECT
```

* Invíete los valores del rectángulo de control para resaltarlos.

* Se calculan las dimensiones del Rectángulo.

```
Rectangle.Right = Ctrl.ScaleWidth
```

```
Rectangle.bottom = Ctrl.ScaleHeight
```

* Se invierte el rectángulo.

```
R = InvertRect(Ctrl.hDC, Rectangle)
```

```
End Sub
```

```
Sub Obtener Ayuda HelpTopic (Topic As String)
```

* Despliega el tema de ayuda seleccionado del menú Ayuda.

```
if HelpTopic = MID$(USO_DE_AYUDA) then
```

* fue seleccionado "Índice de la Ayuda" entonces se despliega el tema estandar

* de Windows "Índice de Ayuda".

```
R = WinHelp(Editor.hWnd, Dummy, HELP_INDEXHELP, 0)
```

```
else
```

* Se seleccionó un tema diferente a "Índice de la Ayuda".

```
R = WinHelp(Editor.hWnd, PathArchivoAyuda, HELP_CONTEXT, 0) (ngHelpTopic)
```

```
End if
```

```
End Sub
```



Sub Redibujar_Malla (DestHdc, X1, Y1, X2, Y2)

* Esta subrutina se encarga de redibujar una porción de la malla, si la opción de malla visible está activada. Los parámetros X1, Y1, X2, Y2 son las esquinas de la porción de malla que se necesita redibujar. La malla no se redibuja si el registro se presenta en su tamaño normal, así que se debe verificar el valor final de la terna de Zoom.

```

    If Editor.Seri.Zoom.Value < Editor.Seri.Zoom.Min Then
        X_Ampl = X1 / Tamaño del Pixel
        Y_Ampl = Y1 / Tamaño del Pixel
        Ancho_Ampl = (X2 - X1 + 1) / Tamaño del Pixel
        Altura_Ampl = (Y2 - Y1 + 1) / Tamaño del Pixel
        R = BitBlt(DestHdc, X1 / Tamaño del Pixel, Y1 / Tamaño del Pixel, Ancho_Ampl,
        Altura_Ampl, Editor.Pic_Grid,hDC, X_Ampl, Y_Ampl, SRC'AND)
    End If
End Sub

```

Sub Seleccionar_Registro_Nuevo()

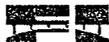
* Esta rutina se usa para realizar los efectos visuales cuando se selecciona un nuevo registro en el área de estado o cuando se selecciona un nuevo registro de Historial.

```

    Seleccionando = 1
    Selección en Movimiento = 1
    * Se dibuja un rectángulo alrededor del registro seleccionado
    Editor.Marco1.Top = Editor.Registros(Registro_Actual).Top - RECUADRO
    Editor.Marco1.Left = Editor.Registros(Registro_Actual).Left - RECUADRO
    Editor.Line1.X1 = Editor.Registros(Registro_Actual).Left - RECUADRO
    Editor.Line1.Y1 = Editor.Registros(Registro_Actual).Top - RECUADRO
    Editor.Line1.X2 = Editor.Registros(Registro_Actual).Left
    Editor.Registros(Registro_Actual).Width - RECUADRO
    Editor.Line1.Y2 = Editor.Registros(Registro_Actual).Top
    Editor.Registros(Registro_Actual).Height - RECUADRO
    Enmarcar_Registro_Actual
    Extraer_Imagen_Y_Mascara Editor.Registros(Registro_Actual)
    * Se actualiza el nombre de archivo de acuerdo al registro seleccionado.
    Actualiza_Registro Editor.Reg_Deshacer
    * Se despliega el historial en el área de estado.
    Amplificar_Registro 0, 0, 63, 63
    * Se despliega el nombre de Archivo del registro seleccionado en el encabezado del Editor.
    Editor.Caption = "Oclusal: " & Editor.Registro_Actual & " - "
InfoRegis(Editor.Registro_Actual).Nombre.Archivo

```

End Sub



5.2.3 OCLUSAL.FRM

Sub-Form Load

* Se determina el modo de color y se establece el tamaño de las imágenes.

* Este valor (TamañoImagen) se usa en las rutinas Actualiza_Registro y

* Extraer_Imagen. * Máquina de OCLUSAL.D3AS

0) GetDeviceCaps(hdc, BI_TSPINEL) * 8 / 1024

TamañoImagen = 1024 Bytes

1)

TamañoImagen = 512 Bytes

End If

* Dependiendo del modo de video usado por Windows, los controladores de las ventanas tendrán

* diferentes dimensiones cuando éstas se abran bajo resoluciones diferentes. Por eso, el Editor

* se basa en un registro de 64x64 píxeles, las imágenes de control usadas para manipular los

* registros deben establecerse de 64x64 también.

* Posición y tamaño de los registros en el área de estado.

Espaciado = (Area_de_estado.ScaleWidth - 192) / 4

Y = 0 / 4

For N = 0 To 2

Registros(N, Y) = Y * Espaciado + N * (64 + Espaciado), Espaciado + Y * (64 +

Espaciado), 64, 64

Next N

Next Y

Pic_Mask, Y = 0, 0, 64, 64

Pic_IMAGE, Y = 0, 0, 64, 64

Pic_TempMask, Y = 0, 0, 64, 64

Pic_tempIMAGE, Y = 0, 0, 64, 64

Pic_Work, Y = 0, 0, 64, 64

Pic_grid, Y = 0, 0, 64, 64

Pic_ToolPalette, Y = 0, 0, 32, 256

Reg_Deshacer, Y = Editor.Registros(4).Top, (Editor.Registros(5).Top -

Editor.Registros(5).Height) / 4 - RECUADRO, 64, 64

* Posición inicial de los botones que identifican al registro actual

Editor.Marco1.Top = Editor.Registros(Registro_Actual).Top - RECUADRO

Editor.Marco1.Left = Editor.Registros(Registro_Actual).Left - RECUADRO

Editor.Marco1.Height = Editor.Registros(Registro_Actual).Height - 2 * RECUADRO

Editor.Marco1.Width = Editor.Registros(Registro_Actual).Width - 2 * RECUADRO

Editor.Line1.X1 = Editor.Registros(Registro_Actual).Left - RECUADRO

Editor.Line1.Y1 = Editor.Registros(Registro_Actual).Top - RECUADRO



```

Editor.Line1.X2 = Editor.Registros(Registro_Actual).Left
Editor.Registros(Registro_Actual).Width = RECUADRO
Editor.Line1.Y2 = Editor.Registros(Registro_Actual).Top
Editor.Registros(Registro_Actual).Height = RECUADRO

```

Obtener parámetros del Archivo INI

*Ajustar el tamaño apropiado del Editor para el modo de Video.

```

If screen.Height < ALTURA_EGA Then

```

```

    'Modo VGA

```

```

    AnchoNuevo = Width

```

```

    AlturaNueva = screen.Height

```

```

    TopeNuevo = 0

```

```

End If

```

```

'Modo VGA e superior.

```

```

AnchoNuevo = screen.Width - .75

```

```

AlturaNueva = screen.Height - .75

```

```

TopeNuevo = (screen.Height - AlturaNueva) - 2

```

```

End If

```

```

Move (screen.Width - AnchoNuevo) - 2, TopeNuevo, AnchoNuevo, AlturaNueva

```

*El registro se le aplica un tamaño aumentado al máximo

```

Serf.Zoom.Value = Serf.Zoom.Max

```

*Se Crean los mapas de bits Mono compatible y el Hdc que se usan para sobreponer la máscara (máscara compatible de los registros).

```

MonoHdc = CreateCompatibleDC(hDC)

```

```

MonoHbm = CreateCompatibleBitmap(MonoHdc, 64, 64)

```

```

OldObject = SelectObject(MonoHdc, MonoHbm)

```

*Se establece el color de fondo del área de edición.

```

Pie_Edit.BackColor = BLANCO

```

*Se selecciona la herramienta actual y el tamaño de Brocha.

```

Herra_Actual = HERR_LAPIZ

```

```

Tamaño de Brocha = 1

```

*Se Inicializan los Colores del Raton

```

MOUSEColores(0) = Negro 'Izquierdo

```

```

MOUSEColores(1) = Rojo 'Derecho

```



MOUSEColores(2) BLANCO Fondo

MOUSEColores(3) Negro Inverso

* Se inicializa el Nombre de Archivo de los 6 Registros con SIN_NOMBRE.

1 1 0 1 5

InfoRegis(I).Nombre.Archivo = SIN_NOMBRE

2 2 1

Editor.Abierto = 0

EndProc

300 Activar Malla

* Se usa esta rutina para activar o desactivar la opción de Ver Malla.

* base cambia la opción VER_MALLA, entonces se crea ésta.

Selección_Menú VER(MID_MALLA).Checked = Not

Selección Menú VER(MID_MALLA).Checked

Selección Menú VER(MID_MALLA_LINEAL).Enabled =

Selección Menú VER(MID_MALLA).Checked

Selección Menú VER(MID_MALLA_PUNTEADA).Enabled =

Selección Menú VER(MID_MALLA).Checked

Selección Menú VER(MID_MALLA).Checked 1 1 Crear Malla

Amplificar Registro 0, 0, 63, 63

EndProc

500 Ajustar Área de Estado

* Cuando el editor cambia de tamaño, también debe cambiarse el tamaño del área de estado.

* o es cuando se es necesario.

* Se determina si el tamaño del editor es suficiente para que quepa el área de estado.

1 ((ScaleHeight (Registros(5).Top Registros(5).Height RECUADRO)) < (ScaleWidth

2 * Area.de.Estado.Width)) Selección Menú VER(MID AREA DE ESTADO).Checked

Then

Selección Menú VER(MID AREA DE ESTADO).Checked = 1

Selección Menú VER(MID AREA DE ESTADO).Enabled = 1

Else

Selección Menú VER(MID AREA DE ESTADO).Enabled = 0

Selección Menú VER(MID AREA DE ESTADO).Checked = 0

OldHeight = Area.de.Estado.Height

Area.de.Estado.Width = ScaleWidth - Area.de.Estado.Width - 1, Area.de.Estado.Top,

Area.de.Estado.Width ScaleHeight - 2

TopoNuevo = Area.de.Estado.ScaleHeight - 4,25 Opt_Mouse(0).Height

Opt_Mouse(0).Top TopoNuevo - 1,25 Opt_Mouse(0).Height

Opt_Mouse(1).Top TopoNuevo - 3 Opt_Mouse(0).Height

Visibility = TopoNuevo (Registros(5).Top Registros(5).Height - 2 RECUADRO - 4)



```

Opt_Mouse(0).Visible = Visibility
Opt_Mouse(1).Visible = Visibility
* Se determina si hay espacio para allegar el registro de deshacer.
1) Opt_Mouse(0).Visible = True
   Reg_Deshacer.Visible = TopoNuevo = (Reg_Deshacer.Top Reg_Deshacer.Height)
   Reg_Deshacer.Visible = (Reg_Deshacer.Top Reg_Deshacer.Height)
Area_de_Estado.ScaleHeight
   TopoH
   * Si su tamaño es demasiado pequeño se vuelve a redimensionar.
   OldHeight = Area_de_Estado.Height + 100, Area_de_Estado.Width =
   TopoH
   End If
End Sub

Sub Ajustar Paleta de Colores
* Determina si el editor es lo suficientemente grande para dibujar la Paleta de Colores
   (ScaleHeight (Paleta de Colores.Height 34)) And
Selección Menú VER(MID PALETA DE COLORES).Checked = True
Selección Menú VER(MID PALETA DE COLORES).Checked = True
Selección Menú VER(MID PALETA DE COLORES).Enabled = False
Else
Selección Menú VER(MID PALETA DE COLORES).Enabled = True
* Verifica si la selección de la Paleta de Colores está activada
1) Selección Menú VER(MID PALETA DE COLORES).Checked = True
   TopoNuevo = ScaleHeight Paleta de Colores.Height
   AnchoNuevo = ScaleWidth 70
   * Se verifica si el ancho de la paleta es suficiente para que quepa en el área de estado
   Selección Menú VER(MID AREA DE ESTADO).Checked = True, AnchoNuevo =
AnchoNuevo = Area_de_Estado.Width - 1
   * Se reposiciona la Paleta de Colores en su posición original.
   Paleta de Colores.Left = Paleta de Colores.Left, TopoNuevo, AnchoNuevo
   Cargar, True, Paleta de Colores.Width, TopoNuevo, 70
   Dinámico, True, Paleta de Colores.Width, ScaleHeight Dinámico.Height, 70
Else
   Dinámico, True, 0, ScaleHeight Dinámico.Height, 32
   Cargar, True, 0, ScaleHeight Paleta de Colores.Height, 32
End If
End Sub
End Sub

```



Sub_Ajustar_ScrollBar

* Esta rutina se utiliza para cambiar el tamaño de la barra de Zoom cuando sea necesario.

```
    B Seleccion Menú VER(MID HERRAMIENTAS).Checked = 1:01 NewLeft
Pic_ToolPalette.Width = NewLeft - 1
    Altura Nueva = ScaleHeight / 2
Seleccion Menú VER(MID PALETA DE COLORES).Checked = Paleta de Colores.Height
NewMax = ScaleWidth / (NewLeft - Seleccion Menú VER(MID ZOOM).Checked
Serl_Zoom.Width) / 3 (Seleccion Menú VER(MID AREA DE ESTADO).Checked
Area de Estado.Width)
    * Se establece NewLeft como el mayor grado de división de la altura de X y NewMax
    * NewMax = Altura Nueva / NewMax = Altura Nueva
    * Se debe ajustar el tamaño de las barras para que sea múltiplo de 64
    Total = (NewMax - 2) / 64
    NewMax = NewMax + 1
Width
Serl_Zoom.Left = NewLeft, Serl_Zoom.Top, Serl_Zoom.Width, Altura Nueva
Serl_Zoom.Max = NewMax
    * Se determina si el tamaño del área de edición es mayor que NewMax.
    B Pic Edit.Height > Serl_Zoom.Max then
        Serl_Zoom.Value = Serl_Zoom.Max
    End B
End Sub
```

Sub_Ajustar_Tool_Palette

* Cuando el editor cambia de tamaño, la paleta de herramientas tal vez debería esconderse.

```
    B ((ScaleHeight / (Pic_ToolPalette.Height - Paleta de Colores.Height)) And
Seleccion Menú VER(MID PALETA DE COLORES).Checked) = 1: ((ScaleHeight
Pic_ToolPalette.Height) / 2) = 1:01
Seleccion Menú VER(MID PALETA DE COLORES).Checked) then
    Seleccion Menú VER(MID HERRAMIENTAS).Checked = 1:01
    Seleccion Menú VER(MID HERRAMIENTAS).Enabled = 1:01
    End B
    Seleccion Menú VER(MID HERRAMIENTAS).Enabled = 1:01
End Sub
```

End Sub

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**



Sub Ajustar_X_y_Y (X = Tamaño del Pixel, Y = Tamaño del Pixel, NewX, NewY, Tipo de Ajuste)

* Para hacer que un píxel amplificado aparente ser realmente un solo píxel, se usa esta rutina para ajustar el píxel a su tamaño al centro y a la orilla del píxel amplificado.

! Seleccionar Tipo de Ajuste

Case CENTRAR

NewX = (X - Tamaño del Pixel) / Tamaño del Pixel + (Tamaño del Pixel / 2)

NewY = (Y - Tamaño del Pixel) / Tamaño del Pixel + (Tamaño del Pixel / 2)

Case ORILLAR

Xpix = X - Tamaño del Pixel

Ypix = Y - Tamaño del Pixel

H = (X - Tamaño del Pixel) / (Tamaño del Pixel / 2) + 1 + Xpix * Xpix + 1

V = (Y - Tamaño del Pixel) / (Tamaño del Pixel / 2) + 1 + Ypix * Ypix + 1

NewX = Xpix / Tamaño del Pixel

NewY = Ypix / Tamaño del Pixel

End Case

End Routine

Sub Borrar la Selección

* Esta rutina se invoca cuando se selecciona el ítem Borrar del menú Editar.

Escalar_Región (X1, Y1, X2, Y2, 100)

Pic_IMAGE = (X1, Y1)-(X2 + 1, Y2 + 1), BLANCO, BF

Pic_Mask = (X1, Y1)-(X2 + 1, Y2 + 1), BLANCO, BF

Actualizar_Registro Registros(Registro_Actual)

End Sub

Sub Colocar Selección ReinicializarBanderas

* Esta rutina se invoca para colocar una selección en el registro.

! Se borra el cuadro que marca la selección.

! Seleccionando el ítem Dibujar Rectángulo de Selección

Escalar_Región (X1, Y1, X2, Y2, 100)

! Selección Menú VER(MID MALLA) Checked (100) Redibujar_Malla (Pic_Edit.hDC), X1, Y1, X2, Y2

W = X2 - X1

H = Y2 - Y1

! Opaco (100)

! Se hace una copia exacta de la Mascará y la imagen de la Selección

R = BitBlt(Pic_IMAGE.hDC, X1, Y1, W, H, Pic_TempIMAGE.hDC, X1_de_selección, Y1_de_selección, SRCCOPY)

R = BitBlt(Pic_Mask.hDC, X1, Y1, W, H, Pic_TempMask.hDC, X1_de_selección, Y1_de_selección, SRCCOPY)



```
End Sub
' Se copian solo los colores de superficie.
R = BitBlt(Pic_IMAGE.hDC, X1, Y1, W, H, Pic_TempMask.hDC, X1 de selección,
Y1 de selección, SRCAND)
R = BitBlt(Pic_IMAGE.hDC, X1, Y1, W, H, Pic_tempIMAGE.hDC, X1 de selección,
Y1 de selección, SRCINVERT)
R = BitBlt(Pic_Mask.hDC, X1, Y1, W, H, Pic_TempMask.hDC, X1 de selección,
Y1 de selección, SRCAND)
End Sub
# ReinitializarBanderas
Seleccionando
Selección en Movimiento
End Sub
' Se actualiza el registro de Mask e Image
Actualiza_Registro Registros(Registro_Actual)
End Sub
```

Sub Copiar Selección al Portapapeles (X1, Y1, X2, Y2)

' Esta rutina se utiliza para copiar la imagen del editor al portapapeles y cuando se corta ésta.

```
Escalar_Región = (X1, Y1, X2, Y2, 1)
Pic_Work.Picture = (X1, Y1, X2, Y2)
Pic_Work.Left = Pic_Work.Left, Pic_Work.Top, X2 - X1, Y2 - Y1
R = BitBlt(Pic_Work.hDC, 0, 0, Pic_Work.Width, Pic_Work.Height,
Registros(Registro_Actual).hDC, X1, Y1, SRCCOPY)
Clipboard.SetData(Pic_Work.Image)
End Sub
```

Sub Cortar Copiar o Borrar Selección Index

' Esta rutina se utiliza cuando se hace una selección de Cortar, Copiar o Borrar en el menú Editar.

```
# Index = MID BORRAR
Escalar_Región = (X1, Y1, X2, Y2, 1)
# Copiar Selección al Portapapeles X1, Y1, X2, Y2
End Sub
# Index = MID COPIAR
Borrar la Selección
Seleccionando
Amplificar_Registro X1, Y1, X2, Y2
End Sub
End Sub
```



Subs Crear_Malla (*)

* La malla se dibuja en Negro en un cuadro separado de control con fondo BLANCO. Esta se dibuja completa solamente una vez. Cuando se necesita actualizar la malla en una región determinada, se utiliza la función Fillfill, la cual toma una porción del cuadro de malla y lo encierra en la región.

Pic_grid:=(0, 0, Serl.Zoom.Value, Serl.Zoom.Value)

Pic_grid:=0

If Selección_Menú_VER(MID MALLA LINEAL).Checked == 1

 Pic_Grid:=63

 Pic_grid:=(0,1,Tamaño_del_Pixel)-(Pic_grid.ScaleWidth,1-Tamaño_del_Pixel)

 Pic_grid:=(1-Tamaño_del_Pixel,0)-(1-Tamaño_del_Pixel,Pic_grid.ScaleHeight)

 Pic_Grid:=1

End If

 For Y = 0 To 63

 For X = 0 To 63

 Pic_grid:=(X-Tamaño_del_Pixel)-(Tamaño_del_Pixel-2), (Y-T

Tamaño_del_Pixel)-(Tamaño_del_Pixel-2))

 Pic_Grid:=X

 Next Y

 Pic_Grid

End Subs

Subs Dibujar Línea o Rectángulo (*)

* Esta rutina se utiliza para dibujar la línea o rectángulo que requieren las herramientas.

If Herra_Actual = HERR LINEA Then

 Pic_Edit:=(X1Región, Y1Región)-(X2Región, Y2Región)

 Pic

 Pic_Edit:=(X1Región, Y1Región)-(X2Región, Y2Región), , B

 Pic_Edit

End Subs

Subs Guardar_Registro_Index

* Esta rutina se utiliza para realizar las operaciones de Archivar y archivar como... del menú Archivo.

* Si la operación es Archivar se requiere el nombre Archivo para guardar el registro.

* Si la operación es Archivar como... siempre se invocará un cuadro de diálogo.

* también la operación Archivar lo invocará si el registro es uno SIN_NOMBRE.

 On Error Goto Error

 Dim LongName As String

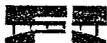
 Dim Title As String

 Dim CDPATH As String

 If (Index = MID GUARDAR COMO) Then (InfoRegis(Registro_Actual).Nombre.Archivo = SIN_NOMBRE) Then

 Se_establece_el_filtro_de_default

 Editor.CMDialog1.Filter = "Archivos de Registros (*.pyf)*.pyf"



```
* Despliega el dialogo Cerrar como:  
Editor.CMDialog1.FilterIndex = 1  
Editor.CMDialog1.DefaultExt = ".PYF"  
If Right$(InfoRegis(Registro_Actual).PathCompleto, 1) = "." Then  
    Editor.CMDialog1.Filename = InfoRegis(Registro_Actual).PathCompleto & "." &  
InfoRegis(Registro_Actual).Nombre_Archivo  
Else  
    * Si el registro no ha sido analizado automaticamente establece la extension de default ".pyf"  
    InfoRegis(Registro_Actual).Nombre_Archivo = SIN$(NOOMBRE$(InfoRegis(Registro_Actual).Nombre_Archivo), ".pyf")  
    Editor.CMDialog1.Filename = InfoRegis(Registro_Actual).Nombre_Archivo & ".pyf"  
End If  
Editor.CMDialog1.Action = 2  
If C = 32755 Then * Salir de dialogo si el usuario cancela.  
Title = Editor.CMDialog1.FileTitle  
LongName = Editor.CMDialog1.Filename  
InfoRegis(Registro_Actual).PathCompleto = InfoRegis(Registro_Actual).LongName & "." & LongName & ".(Len$(Title))"  
InfoRegis(Registro_Actual).Nombre_Archivo = Editor.CMDialog1.FileTitle  
* Ajustar el directorio  
If C = 3 Then InfoRegis(Registro_Actual).PathCompleto = 3 * InfoRegis(Registro_Actual).PathCompleto  
    CDPPath = InfoRegis(Registro_Actual).PathCompleto  
Else  
    CDPPath = InfoRegis(Registro_Actual).LongName & "." & LongName & ".(Len$(Title))"  
End If  
ChDir CDPPath * Cambiar directorio para el editor y para Historial.  
Historial.Dir.DirectoryList.Path = CDPPath  
End If  
InfoRegis(Registro_Actual).Nombre_Archivo = InfoRegis(Registro_Actual).Nombre_Archivo & InfoRegis(Registro_Actual).PathCompleto,  
InfoRegis(Registro_Actual).Nombre_Archivo  
End Sub  
  
Sub Mostrar_Coordenadas_y_Presion(Xpix, Ypix)  
* Esta rutina se encarga de desplegar las coordenadas del pixel en que se encuentran el  
* apuntador del mouse dentro del area de edicion. Ademas, muestra tambien la magnitud  
* de presion representada por el color de dicho pixel.  
If (Xpix = 0) Or (Xpix = 63) Then  
    Area_de_Estado.CurrentX = Area_de_Estado.ScaleWidth / 25  
    Area_de_Estado.CurrentY = Ypix - 10 * (Area_de_Estado.ScaleWidth / 25) * Ypix;  
Else  
    Area_de_Estado.CurrentX = Xpix  
    Area_de_Estado.CurrentY = Ypix  
End If  
If (Ypix = 0) Or (Ypix = 63) Then  
    Area_de_Estado.CurrentX = Area_de_Estado.ScaleWidth / 75  
    Area_de_Estado.CurrentY = Ypix - 10 * (Area_de_Estado.ScaleWidth / 75) * Ypix;  
Else  
    Area_de_Estado.CurrentX = Xpix  
    Area_de_Estado.CurrentY = Ypix  
End If  
End Sub
```



```

Area de Estado.CurrentY = Area de Estado.CurrentY + 1
Area de Estado.CurrentX = Area de Estado.ScaleWidth - 5
color = 255 - (Pic.IMAGE.Height - (Xpix, Ypix) - 2 - 8) * 0.1 & HFF&
If Pic.IMAGE.Height - (Xpix, Ypix) = -1 Then color = 0
Area de Estado.CurrentX = Area de Estado.CurrentX + color - 100 : Pic.Refresh : color = 100 : Format$(color)
Area de Estado.CurrentY = Area de Estado.CurrentY + 1

```

End Sub

Sub Menu_EditSelection_Click_Index (ByVal Index As Integer)

' Se procesan los botones de menú de edición de texto

Class CEdit_Index

Class MID_DESFAZER

Undo Edits

Class MID_CORTAR, MID_COPIAR, MID_BORRAR

Cortar Copiar o Borrar Selección Index

Class MID_PEGAR, MID_PEGAR_OPACO

Paste Clipboard Contents Index

Class MID_SELECCIONAR_TODO

Select Entire Registro

Class MID_ESPEJO_HORIZONTAL, MID_ESPEJO_VERTICAL

Flip Selection Index

Class MID_ROTAR_DERECHA, MID_ROTAR_IZQUERDA

Rotate Selection Index

Class MID_INVERTIR

Invert Selection

End Class

End Sub

Sub Cargar_Click (ByVal Index As Integer)

' Esta rutina se ejecuta con el programa de adquisición de datos (puertos.exe), mediante la función

' Shell. El programa "puertos.exe" crea un archivo de mapa de bits (TEST1.BMP) el cual contiene,

' en base a colores, la magnitud de la presión ejercida en la matriz transductora. Este archivo se

' coloca sobre el registro actual. Si en el Menú Ver se ha activado el ítem, esta rutina, también realiza

' un pegado para mostrar solo una parte del archivo TEST1.BMP.

' Se verifica si el nuevo registro será sobrepuesto sobre un registro que

' ha sido modificado después de la última vez que se guardó. Si es así,

' entonces se le pregunta al usuario si desea descartar los cambios.

If Desca_descartar_los_cambios() Then

' La ejecución del programa de adquisición de datos toma algunos segundos, es por eso, que se

' cambia el apuntador del ratón a Reloj de arena, mientras se ejecuta dicho programa.

screen.MousePointer = RELOJ de ARENA



* Se invoca al programa de adquisición.
X = 50; F.App.Path = "puertos.exe"

* El archivo "TEST.BMP" arrojado por el programa de adquisición se coloca
* sobre el Registro actual y después se borra.
Registros(Registro_Actual) = "TEST.BMP";
Blt "TEST.BMP" = 1; Borrar "TEST.BMP" del disco.
screen.MousePointer = DEFAULT

* Se copia la imagen en el cuadro de control para facilitar su edición.
Pic_IMAGE.Picture = Registros(Registro_Actual).Picture

* Se verifica el estado de la opción filtro del menú VER
if Selección_Menu_VER(MID, FE, PRO).Checked = 1 then

 * Esta activa la opción filtro, así que se realizará un pegado no
 * opaco para hacer visible solo una parte del mapa de bits.
 Opaco = 0.5;

 Pic_Work.Picture = Pic_Filtro.Picture

 Pic_tempIMAGE.Picture = Pic_Work.Image

 * Para el pegado se debe hacer una máscara del mapa de bits que será pegado, basada
 * en el color de fondo que permitirá ver parte de la imagen. La máscara se crea
 * mediante la rutina BitBlt(1). Todos los píxeles de la máscara que sean de distinto
 * color al del fondo de pantalla cambiarán los píxeles del mapa de bits que se filtra, y los
 * píxeles del mismo color al del fondo permitirán ver los píxeles del mapa de bits.

 Pic_Work.BackColor = Rojo

 R = BitBlt(MonoHdc, 0, 0, Pic_Work.Width, Pic_Work.Height, Pic_Work.hDC, 0, 0,
 SRCCOPY)

 R = BitBlt(Pic_TempMask.hDC, 0, 0, Pic_Work.Width, Pic_Work.Height, MonoHdc, 0, 0,
 SRCCOPY)

 R = BitBlt(Pic_tempIMAGE.hDC, 0, 0, Pic_Work.Width, Pic_Work.Height,
 Pic_TempMask.hDC, 0, 0, SRCAND)

 R = BitBlt(Pic_tempIMAGE.hDC, 0, 0, Pic_Work.Width, Pic_Work.Height, Pic_Work.hDC,
 0, 0, SRCINVERT)

 Escalar_Región = 0, 0, 64, 64, ...

 Escalar_Región = X1_de_selección, Y1_de_selección, X2_de_selección,
 Y2_de_selección, ...

 Colocar_Selección = ...

 Pic_Work.BackColor = MOUSEColores(2)

 goto 11

 Actualiza_Registro Registros(Registro_Actual)

 Amplificar_Registro 0, 0, 63, 63

Else

* Si no se descartan los cambios del registro.

Editor.Registros(Registro_Actual).Click

Amplificar_Registro 0, 0, 63, 63

Fin: If

Fin: Sub

Sub **Dinámico_Click** ()

* Esta rutina es casi igual a la anterior, con la diferencia de que captura seis nuevos

* registros secuencialmente para tener un efecto dinámico.

* Se verifica si el nuevo registro será sobrepuesto sobre un registro que

* ha sido modificado después de la última vez que se guardó, si es así,

* entonces se le pregunta al usuario si desea descartar los cambios.

If **Desca_descartar_los_cambios()** Then

* La ejecución del programa de adquisición de datos toma algunos segundos, es por eso

* que se cambia el apuntador del botón a **Rebot** de arena, mientras se ejecuta dicho programa.

Actual = Registro_Actual

screen.MousePointer = REBOT de **_ARENA**

If **Registro_Actual = 0** Then 5

* Para cada nuevo registro se invoca al programa de adquisición

X = os.system(App.Path & " puertos.exe"

* El archivo "TEST.DAT" arregado por el programa de adquisición se coloca

* sobre el Registro actual y después se borra

Registros(Registro_Actual) = "TEST.DAT"

File.Delete("TEST.BMP") * Borra el archivo "TEST.BMP" del disco.

* Se copia la imagen en el cuadro de control para facilitar su edición

Pic_IMAGE.Picture = Registros(Registro_Actual).Picture

* Se verifica el estado de la opción Filtro del menú VER

If **Selección_Menú_VER(MID_FILTRO).Checked** Then

* Esta activa la opción Filtro, así que se realizara un pegado no

* opaco para hacer visible solo una parte del mapa de bits.

Opaco = 255

Pic_Work.Picture = Pic_Filtro.Picture



Pic_TempIMAGE.Picture = Pic_Work.Image

* Para el pegado se debe hacer una máscara del mapa de bits que será pegado (basado en el color de fondo que permitirá ver parte de la imagen). La máscara se crea mediante la rutina BitBlt(). Todos los píxeles de la máscara que sean de distinto color al del fondo de escritorio ocultarán a los píxeles del mapa de bits que se filtra, y los píxeles del mismo color al de fondo permitirán ver los píxeles del mapa de bits.

Pic_Work.BackColor = Rojo

R = BitBlt(MonoHdc, 0, 0, Pic_Work.Width, Pic_Work.Height, Pic_Work.hDC, 0, 0, SRC'COPY)

R = BitBlt(Pic_TempMask.hDC, 0, 0, Pic_Work.Width, Pic_Work.Height, MonoHdc, 0, 0, SRC'COPY)

R = BitBlt(Pic_TempIMAGE.hDC, 0, 0, Pic_Work.Width, Pic_Work.Height, Pic_TempMask.hDC, 0, 0, SRC'AND)

R = BitBlt(Pic_TempIMAGE.hDC, 0, 0, Pic_Work.Width, Pic_Work.Height, Pic_Work.hDC, 0, 0, SRC'INVERT)

Escalar_Región = 1, 0, 0, 64, 64, 1

Escalar_Región = 1, 0, 0, 64, 64, 1
Y1 de selección, Y1 de selección, X2 de selección, X2 de selección,

Colocar Selección

Pic_Work.BackColor = MouseColors(2)

Inicio:

Actualizar_Registro Registros(Registro_Actual)

Amplificar_Registro 0, 0, 63, 63

Pic = Registro_Actual

Registro_Actual = 5

Selección Menú Registros(Actual).Checked = True

Selección Menú Registros(5).Checked = True

Seleccionar_Registro_Nuevo

screen.MousePointer = DEFAULT

Fin:

* Si no se desmarcan los registros del registro:

Editor.Registros(Registro_Actual).Checked = True

Amplificar_Registro 0, 0, 63, 63

Inicio:

Fin: Fin

Sub Form_Unload Cancel = True

* Termina Fin

* Verifica si todos los registros modificados fueron guardados.

Texto = ""

```

If I = 0 To 5
    If InfoRegis(I).Modificado Then Texto = Texto + "Registro #" + Regis(Selección Menú_Registros(I)).Caption + " en Selección Menú_Registros(I).Caption + " +
    CRLF
    Next I

If Texto = "" Then
    'Uno o más registros fueron modificados sin ser guardados, entonces se notifica al usuario.
    'Establece las propiedades necesarias para abrir MBNO.
    Editor.MIMControl1.Notify = True
    Editor.MIMControl1.Wait = True
    Editor.MIMControl1.Shareable = True
    Editor.MIMControl1.DeviceType = "WaveAudio"
    Editor.MIMControl1.FileName = App.Path + "Oclusal2.WAV"
    'Abre el dispositivo de audio MBNO.
    Editor.MIMControl1.Command = "Open"
    Editor.MIMControl1.Command = "Play"
    Texto = Texto + CRLF + "Los Registros enlistados no han sido guardados desde la última
    modificación. ¿Aún así desea salir?"
    Cancel = Editor.MIMControl1.Command = "Close"
    Editor.MIMControl1.Command = "Close"
    If I = Cancel Then
        'El usuario quiere terminar el Editor, entonces se descarga el programa.
        Editor.Abierto
        Editor.ColorPalette
        R = DeleteDC(NomofIde)
        R = DeleteObject(NomofIbm)
        R = WinHelp(hwnd, dummy, HELP_QUIT, 0)
        Save Settings To INI File

        'Si el Editor fue abierto primero (sin una línea de comando de "h" o "H")
        'entonces este era la ventana principal. Y si Historial está abierto,
        'se debe programar un botón si también se cerrará Historial.
        If (VentanaPrincipal = Editor.de.Oclusal) Then HistorialAbierto Then
            'Establece las propiedades necesarias para abrir MBNO.
            Editor.MIMControl1.Notify = True
            Editor.MIMControl1.Wait = True
            Editor.MIMControl1.Shareable = True
            Editor.MIMControl1.DeviceType = "WaveAudio"
            Editor.MIMControl1.FileName = App.Path + "Oclusal3.WAV"

```



```
* Abre el dispositivo de audio de NIPL
Editor.NIMControlH.Command = "Open"
Editor.NIMControlH.Command = "Play"
Texto = "¿Salir de Historial también?"
H.Mensaje(Texto, 36, "Oclusal", 1, MBSI)
VentanaPrincipal = Ventana de Historial
Editor.NIMControlH.Command = "Close"
End If
End If
NIMControlH.Command = "Close"
End Sub
```

RESULTADOS

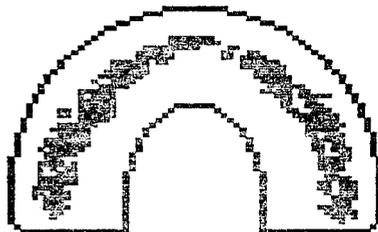
Las gráficas que representan a los registros capturados por el sistema, tienen formato BMP, con color de 24 bits y miden 64x64 pixeles. Estas gráficas, pueden ser editadas y pegadas a otros paquetes de Windows. En el menú principal, pueden escogerse las opciones de mostrar línea media de referencia y arco de referencia (filtro), las cuales ayudan al médico a localizar físicamente los puntos de contacto oclusal. La resolución para medir las presiones en un registro, está confinada a la resolución del convertidor analógico -digital (8 bits). Así, se tienen hasta 255 tonos de color (dependiendo de la memoria de video del equipo), que representan dichas presiones; estos colores van desde el blanco hasta el magenta, pasando por el azul. Entonces, las presiones más altas se representan en magenta. El rango de presiones que se pueden medir con este sistema depende de las características del transductor (compósito piezorresistor y plantillas); de este modo, el sistema puede ser sensible para varios rangos de presión, desde 20 Kg hasta 400 Kg. A continuación se describen brevemente los resultados de tres relaciones oclusales que se obtuvieron con el sistema:



Figura R.1



Figura R.2

**Figura R.3**

En la figura R.1 se muestra un registro con pocos contactos oclusales, en este registro se utilizó un compósito muy sensible con un rango de hasta 20 Kg. En la figura R.2, el compósito utilizado fue uno sensible a un rango amplio de presiones (hasta 300 Kg), en este registro puede observarse que a pesar de que hay más puntos de contacto en el lado derecho, los puntos de mayor presión se encuentran del lado izquierdo de la arcada. En el registro de la figura R.3 el paciente mordió con toda su fuerza. Aquí se observa que la mayor carga se localiza del lado derecho de la arcada.

El diseño del sistema cumple con los requerimientos establecidos inicialmente. Además, se añadieron nuevas opciones como la de captura dinámica, que sirve para capturar 6 registros consecutivos y poder visualizar un efecto dinámico.

CONCLUSIONES

Se analizaron las propiedades físicas de diversos transductores comerciales, en general, éstos resultaron ser demasiado grandes para satisfacer nuestros requerimientos de tener un espesor máximo de 2 mm y buena flexibilidad. Así, se llegó a la conclusión de que era necesario diseñar un transductor nuevo, a fin de obtener mayor flexibilidad y homogeneidad, además de un costo mucho más bajo que el de los transductores comerciales. Este transductor se elaboró en el Instituto de Física mediante el uso de negro de humo [21].

- La etapa transductora consiste de una matriz de transducción conformada por el campósito piezorresistor elaborado en el Instituto de Física y por dos plantillas conductoras que permiten seleccionar una celda específica de la matriz y de este modo, determinar la presión ejercida en ésta celda. Para la elaboración de las plantillas conductoras fue necesario realizar una investigación sobre posibles materiales conductores, flexibles y delgados, que permitieran que estas plantillas fueran desechables o en su defecto esterilizables para evitar posibles contagios, debido a que éstas se introducen en la boca de los pacientes. El material que cumplió estos requerimientos es uno económico e higiénico que se utiliza para el empaque de alimentos, formado por nylon y aluminio.

- Fue necesario el diseño y fabricación de una tarjeta de adquisición de datos que tuviera un canal analógico de entrada y dos canales digitales de 8 bits de entrada / salida, para facilitar el muestreo y la adquisición de los datos. Esta tarjeta puede ser utilizada en otras aplicaciones para el control y la adquisición de datos.

CONCLUSIONES



- El diseño del dispositivo permite que éste pueda ser utilizado en cualquier computadora que cuente con ambiente Windows y con una ranura de expansión (PC, ISA o EISA) libre en la que pueda insertarse la tarjeta de adquisición de datos diseñada. El programa de computadora se diseño para ser ejecutado en ambiente Windows, lo que permite aprovechar las ventajas que ofrece este sistema, como son la interacción amigable con el usuario, la interacción con otros programa y la posibilidad de utilizar dispositivos multimedios.

- El costo de elaboración de este dispositivo, conformado por una tarjeta de adquisición de datos, una tarjeta de muestreo, una matriz de transducción y el software de control, es de aproximadamente 800 pesos. Este costo es muy bajo con respecto al de otros sistemas cuantitativos de diagnóstico (como Foto-oclusión), lo que permite que el dispositivo diseñado por nosotros sea más accesible para los Odontólogos.

- El dispositivo ha comenzado a ser usado en la medición de presiones oclusales dentales [22, 24]. Las perspectivas de aplicación incluyen el registro de huellas plantares en la Podología [23].

GLOSARIO

Los siguientes términos se utilizan para describir los diferentes tipos de relaciones oclusales.

OCLUSIÓN CÉNTRICA: es la posición o relación de máxima intercuspidad entre los dientes maxilares y mandibulares.

RELACIÓN CÉNTRICA: es la posición más retruida de la mandíbula en la fosa glenoidea, desde donde un individuo puede hacer movimientos laterales. Esta es una posición no forzada y el movimiento ocurre alrededor del eje terminal de bisagra.

PROTRUSIÓN: es el movimiento anterior de la mandíbula.

CONTACTOS EN TRABAJO: son los contactos de los dientes en el lado de oclusión hacia el cual la mandíbula ha sido movida.

CONTACTO EN BALANCE: es el contacto entre los dientes superiores e inferiores, en el lado opuesto al lado de trabajo.

LADO DE BALANCE: es el lado opuesto al lado de trabajo, o el lado contrario al cual se mueva la mandíbula.

BIBLIOGRAFÍA

1. Enciclopedia Salvat Diccionario. Ed. Salvat ,1983.
2. The 1996 Grolier Multimedia Encyclopedia. Ed. Sun Moon Star, 1996.
3. KROGH - POULSEN, W. G and Olson, A. Management of the occlusion of the teeth. Parts I and II: In Schwartz, L, and Chages, C.M., editor: Facial pain and Mandibular Dysfunction. Philadelphia, 1968, W.B. Saunders Co.
4. JACOB , E. Intercuspal contacts of the natural dentition in centric occlusion. Journal of Prosthetic Dentistry, april 1981,45 (4), 419 - 421.
5. PERRY, H., Lammie . G. Occlusion in stress situation. J. Am Dent Assoc. 1960, 60: 626.
6. BERRY, P., Singh, B. Daily variations in occlusal contacts. Journal of Prosthetic Dentistry, september 1983, 50(3), 386 - 391.
7. DAWSON, P. Attaining harmonic occlusion through visualized strain analysis. Journal of Prosthetic Dentistry, december 1981, 46(6), 615 - 622.
8. Gran Enciclopedia Universal QUID. Ed. Promexa. 1983. Vol 9. 66 p.p.
9. NASSER, R. Oclusión básica para estudiantes de Odontología. Programa del libro de texto universitario, 1984, 13 - 16, 61 - 79
10. GAZIT, E. Liberman. The intercuspal surface contact area registration - An additional tool for evaluation of normal occlusion. Angle Orthod. 1973, 43(96).
11. GAZIT, E.,Fitzig . S., Lieberman. Reproducibility of occlusal marking techniques. Journal of Prosthetic Dentistry, april 1986, 55(4), 505 - 509.
12. WAYNE L., Royce, A. Computerized occlusal analysis: An evaluation of the sensors. Journal of Prosthetic Dentistry., january 1991, 65(1), 89 - 92.
13. HOLMAN, J. Métodos experimentales para Ingenieros. Ed. Mc Graw Hill, 1988, 123-125, 177-198.

BIBLIOGRAFÍA



14. SCHILLING, D., Belove, Ch. Circuitos electrónicos. Ed. Alfaomega, 1991, 707-725.
15. COUGHLIN, R. Amplificadores operacionales y circuitos integrados lineales. Ed. Prentice hall, 1993, 406-441.
16. Data Acquisition Databook. National Semiconductor, 1993, 2.22 - 2.53, 8.29.
17. Products Data Catalog. Ed. INTEL, 1979., 11.43-11.63.
18. High - Speed CMOS Logic Databook. Texas Instruments. 1989, 2.597 - 2.599.
19. DOWDEN, Tony. Inside the EISA computers. Ed. Addison - Wesley, 1990, 20 - 33.
20. Borland C++. Ed. Borland, 1994.
21. RANGEL, José Luis. Tesis profesional, Facultad de Química, UNAM (en proceso, 1996).
22. MUÑOZ LEDO, Cristina. Tesis de especialidad, Facultad de Odontología, UNAM (en proceso, 1996).
23. RANGEL, Domingo. Tesis profesional, Facultad de Ingeniería, UNAM, 1996.
24. FERNANDEZ, Arturo. Tesis de doctorado, Facultad de Odontología, UNAM (en proceso, 1996).