

FACULTAD DE INGENIERÍA

Escuela Académico Profesional de Ingeniería de Minas

Tesis

**Automatización del modelo matemático Holmberg para
el cálculo y diseño de mallas de perforación en frentes
de desarrollo**

Jimenez Casimiro Alexander Gonzalo

Para optar el Título Profesional de
Ingeniero de Minas

Huancayo, 2021

Repositorio Institucional Continental
Tesis digital



Esta obra está bajo una Licencia "Creative Commons Atribución 4.0 Internacional" .

Dedicado a mi madre Estela Casimiro Salcedo, por su lucha y dedicación, ella es la razón de que todo esto exista.

AGRADECIMIENTOS

A mi madre y hermana por su apoyo económico y sentimental. He tenido la suerte de tener una buena cohesión familiar y doy gracias a Dios por eso.

Al doctor Randy Austin por las fantásticas clases de programación con un nivel altísimo que enriquece el desarrollo profesional.

A mi asesor, el Ingeniero Benjamín Ramos Aranda por su alta excelencia académica y profesional y principalmente por su enfoque a la mejora continua. El deseo de hacer investigación es posible si se trabaja con profesionales de calidad.

INDICE DE CONTENIDO

AGRADECIMIENTOS.....	III
INDICE DE CONTENIDO.....	IV
INDICE DE TABLAS.....	IX
INDICE DE FIGURAS.....	X
INDICE DE RUTINAS.....	XII
RESUMEN.....	XIV
ABSTRACT.....	XV
INTRODUCCION	16
CAPITULO I.....	18
PLANTEAMIENTO DEL ESTUDIO	18
1.1 Planteamiento y Formulación del Problema.....	18
1.2 Formulación del Problema	19
1.2.1 Problema General	19
1.2.2 Problema Especifico.....	19
1.3 Objetivos.....	19
1.3.1 Objetivo General.....	19
1.3.2 Objetivo Especifico.....	19
1.4 Justificación e Importancia.....	20
1.3.1 Teórica.....	20
1.3.2 Práctica.....	20
1.5 Alcance	20
1.6 Hipótesis y Descripción de Variables.....	21
1.6.1 Hipótesis General.....	21

1.6.2	Hipótesis Específica	21
1.6.3	Operacionalización de las Variables.....	21
CAPITULO II.....		22
MARCO TEORICO.....		22
2.1	Antecedentes	22
2.1.1	Entorno Internacional.....	22
2.1.2	Entorno Nacional.....	23
2.2	Generalidades.....	23
2.2.1	Ubicación y Acceso del Prospecto	23
2.2.2	Geología General.....	25
2.3	Propiedades Físicas y Mecánicas de los Materiales Rocosos.....	25
2.3.1	Densidad.....	25
2.3.2	Resistencia a la Compresión Simple.....	26
2.4	Clasificación Geomecánica.....	26
2.4.1	Rock Mass Rating (RMR)	27
2.4.2	Designación de la Calidad de la Roca (RQD).....	27
2.4.3	Caracterización de las Discontinuidades.....	28
2.5	Modelo Matemático Holmberg.....	29
2.5.1	Esquemas de Corte.....	30
2.5.2	Avance por Disparo	31
2.5.3	Factor de Carga.....	32
2.5.4	Burden Máximo & Burden Práctico.....	33
2.5.5	Sumario de Fórmulas para cada Sección	35
2.6	Microsoft Visual Basic.....	38
2.6.1	Paradigmas de Programación	38
2.7	Entorno de Visual Basic para Aplicaciones.....	39

2.7.1	Contenido de Carpetas	39
2.7.2	Jerarquía de Objetos	41
2.7.3	Variables en un Proyecto.....	42
2.7.4	Estructuras y Funciones más Utilizados	43
2.8	Interacción entre Visual Basic y AutoCAD	46
2.8.1	Modelo de Objetos de AutoCAD	46
2.8.2	Objeto Application	46
2.8.3	Objeto Document.....	47
2.8.4	Objetos Gráficos y no Gráficos.....	47
2.8.5	ActiveX/VBA de AutoCAD.....	47
2.8.6	Proyectos VBA de AutoCAD	48
2.8.7	Entidades.....	48
2.8.8	Conceptos Básicos Sobre Propiedades y Métodos.....	49
2.8.9	Referencia a Objetos	49
2.9	Definición de Términos	50
CAPITULO III.....		51
METODOLOGIA.....		51
3.1	Métodos, y Alcance de la Investigación.....	51
3.1.1	Método de Investigación	51
3.1.2	Tipo de Investigación.....	51
3.1.3	Nivel de Investigación.....	52
3.2	Diseño de Investigación.....	52
3.3	Población	52
3.4	Muestra.....	52
3.5	Técnica para la Recolección de datos	52
3.6	Validación de la Hipótesis	52

3.7	Instrumentos para la recolección de datos.....	53
3.8	Evaluación Geomecánica	53
3.8.1	Observaciones Geológicas	53
3.8.2	Determinación del Peso Específico	53
3.8.3	Cálculo de la Tensión de Rotura	55
3.9	Clasificación Geomecánica del Macizo Rocoso.....	57
3.10	Diseño de Programación.....	58
3.10.1	Consolidación de la Base de Datos.....	58
3.10.2	Creación de Carpetas y Sub Carpetas	58
3.10.3	Apertura de un Nuevo Archivo	58
3.10.4	Primera Base de Datos (Explosivos).....	59
3.10.5	Segunda Base de Datos (Equipos)	60
3.10.6	Tercera Base de Datos (Tipos de Arranque).....	61
3.11	Programación Orientada a Objetos	61
3.11.1	Preparación de un Nuevo Proyecto VBA.....	61
3.11.2	Proceso de Abstracción y Materialización - First UserForm.....	62
3.11.3	Proceso de Abstracción y Materialización - Second UserForm.....	72
3.12	Diseño de Malla.....	90
3.12.1	Acceso a la Jerarquía de Objetos y Esquema de Desarrollo.....	90
CAPITULO IV.....		113
PROCESAMIENTO ANÁLISIS E INTERPRETACIÓN DE RESULTADOS		113
Caso Práctico.....		113
4.1	Ingreso y Validación de Parámetros Generales	113
4.1.1	Parámetros de Perforación y Voladura.....	114
4.1.2	Parámetros Geológicos-Geomecánicos	115
4.2	Procesamiento y Análisis de Cálculo de Secciones (Cutting)	116

4.3	Procesamiento y Análisis de Cálculo de Secciones (ContourD)	117
4.4	Procesamiento y Análisis de Cálculo de Secciones (Lifter-StopingB-StopingC-Walls) 118	
4.5	Reporte General.....	120
4.6	Diseño de Malla.....	121
4.7	Discusión	123
	CONCLUSIONES.....	124
	RECOMENDACIONES.....	125
	BIBLIOGRAFIA	126
	ANEXOS.....	128

INDICE DE TABLAS

Tabla 2-1. Coordenadas del Prospecto	23
Tabla 2-2. Distancias por cada tramo de ruta	24
Tabla 2-3: Interpretación de valores de RQD (Deere 1964)	28
Tabla 2-4. Longitud de carga en columna, fondo y taco para cada sección	32
Tabla 2-5. Ecuaciones correspondientes a la sección Corona	36
Tabla 2-6. Ecuaciones en Lifter, Stopping y Walls	36
Tabla 2-7. Tipos de datos incorporados en VBA.....	42
Tabla 2-8. Constante más utilizados dentro de la ventana MsgBox.....	45
Tabla 3-1. Diámetro, Longitud, Masa, Volumen y Densidad de cada espécimen.....	54
Tabla 3-2. Resultados de deformación y tensión sobre el testigo número 3	57
Tabla 3-3. Base de datos del tipo de explosivo.....	60
Tabla 3-4 Base de datos del tipo de Equipo	60
Tabla 3-5. Base de datos sobre los tipos de arranque o cuele	61
Tabla 3-6. Nombre Interno y Externo por cada control del primer formulario	63
Tabla 3-7. Nombre externo e interno de los controles Geomecánicos del primer formulario	66
Tabla 3-8. Nombre externo e interno de los controles de voladura del primer formulario	68
Tabla 3-9. Nombre interno y externo respecto a los parámetros CE y C	71
Tabla 3-10. Controles de la sección Corona (ContourD)	73
Tabla 3-11. Controles de Arrastre-Ayuda Hastial-Ayuda Corona y Hastiales	74
Tabla 3-12. Controles de la sección Arranque (Cutting)	75
Tabla 3-13. Secuencia de validación para establecer el burden práctico en Arrastre, ayuda corona, hastial y ayuda hastial	88
Tabla 4-1. Leyenda del diseño de malla	122

INDICE DE FIGURAS

Figure 1-1. Operacionalización de las Variables	21
Figure 2-1. Mapa de Ubicación del Prospecto Minero.....	24
Figure 2-2. Proceso de Elaboración RMR89	27
Figure 2-3. División de un frente	29
Figure 2-4. Tipos de corte quemado	30
Figure 2-5 Corte de cuatro secciones.....	31
Figure 2-6. Características del editor de Visual Basic	40
Figure 2-7. Jerarquía de objetos.....	41
Figure 2-8. Características de la ventana MsgBox.....	44
Figure 2-9. Objeto Application.....	46
Figure 2-10. Ejemplo de Referencia a Objetos 1	49
Figure 2-11. Ejemplo de referencia a objetos 2	49
Figure 3-1. Proceso de corte y lavado.....	54
Figure 3-2. Pesos Unitarios Típicos de algunas Rocas (19).....	55
Figure 3-3. Ensayo tipo UCS sobre el testigo número 3.....	56
Figure 3-4. Datos del equipo de compresión en tiempo real.....	56
Figure 3-5. Ruta de acceso de carpetas y sub carpetas	58
Figure 3-6. Protocolo de activación de la pestaña programador	59
Figure 3-7. Esquema de edición VBA.....	61
Figure 3-8 Ventana principal o de arranque.....	62
Figure 3-9. Acceso al esquema de procesos del UserForm.....	64
Figure 3-10. Definición de rangos dentro del administrador de nombres	67
Figure 3-11. Flujograma de programación sobre el evento click del Option Button 1,2 y 3... ..	69
Figure 3-12. Formulario “frmCalculo” dividido en 7 pestañas	72
Figure 3-13. Resultado de la rutina 7.8	76
Figure 3-14. Pestaña Corona.....	83
Figure 3-15. Diseño del segundo cuadrante	93
Figure 3-16. Búsqueda del punto medio sobre la base del segundo cuadrante	95
Figure 3-17. Recorrido desde el punto medio y diseño del tercer cuadrante (C3).....	96
Figure 3-18. Búsqueda del punto medio a partir del círculo base y referencia	97
Figure 3-19. Recorrido y diseño del cuarto cuadrante (C4)	98

Figure 3-20. Ubicación de partida – Ayuda corona	99
Figure 3-21. Diseño del Stoping C si el número de taladros es 3	100
Figure 3-22. Diseño del Stoping C si el número de taladros es 2	100
Figure 3-23. Diseño del Stoping C si el número de taladros es 4	102
Figure 3-24. Diseño del Stoping C si el número de taladros es 5	102
Figure 3-25. Punto de partida del Stoping B.....	103
Figure 3-26. Diseño del Stoping B.....	104
Figure 3-27. Declaración de variables para el desarrollo de la sección corona.	105
Figure 3-28. Punto de partida para el diseño de la corona.....	106
Figure 3-29. Diseño de arco-ContourD.....	108
Figure 3-30. Diseño del Arrastre	109
Figure 3-31. Diseño del Hastial derecho.....	110
Figure 3-32. Diseño del Hastial izquierdo	111
Figure 4-1. Arranque del formulario “Diseño de Malla”	113
Figure 4-2. Selección de los diámetros de perforación y llenado de los parámetros de diseño	114
Figure 4-3. Selección del tipo de explosivo por cada sección.....	115
Figure 4-4. Ingreso de los valores RMR, RQD y Densidad de la roca	115
Figure 4-5. Selección del tipo de Arranque.....	116
Figure 4-6. Resultado del cálculo de secciones.....	117
Figure 4-7. Selección del explosivo y uso del reductor de carga	117
Figure 4-8. Resultado del modelo - sección Corona	118
Figure 4-9. Resultados del modelo - sección Arrastre.....	118
Figure 4-10. Resultados del modelo - Stoping B.....	119
Figure 4-11. Resultados del Stoping C.....	119
Figure 4-12. Resultados del modelo - Hastiales	120
Figure 4-13. Reporte de Explosivos y Accesorios.....	121
Figure 4-14. Punto de partida X, Y.....	121
Figure 4-15. Resultado final - Diseño de malla 2.5x2.5m.....	122

INDICE DE RUTINAS

Rutina 3-1. Ingreso de valores sobre los Combos 4, 5 y 6	64
Rutina 3-2. Atributos más relevantes del ListBox-First UserForm.....	65
Rutina 3-3. Acceso al control ListBox y uso del evento click para cargar y actualizar los Combos 4 y 5.....	65
Rutina 3-4. Instrucciones sobre el evento click del Option Button 1.....	70
Rutina 3-5. Primer esquema de desarrollo sobre la fuente del Combo Box 1	70
Rutina 3-6. Segundo esquema de desarrollo sobre la fuente del Combo Box 1.....	71
Rutina 3-7. Rutina para el cálculo del GSI, CE y C	71
Rutina 3-8. Notificación, de pertenencia List Box-Arranque	76
Rutina 3-9. Selección y cálculo de los parámetros Dequiv, Hteórico y Error de perforación. 78	
Rutina 3-10. Esquema básico para el cálculo de Lcf, Lcc y Taco dentro del módulo 2.....	78
Rutina 3-11. Rutina dentro del Spin Button1 para definir valores de H practico, Lcf, Lcc y Taco	79
Rutina 3-12. Evento Initialize del formulario "frmCalculo" para establecer atributos propios del List Box y constantes de voladura.....	80
Rutina 3-13. Representación de ecuaciones sobre el primer cuadrante	81
Rutina 3-14. Representación de ecuaciones sobre el resto de cuadrantes.....	82
Rutina 3-15. Rutina dentro del objeto MultiPage1	84
Rutina 3-16. Reductor del factor de carga en columna - Spin Button 2.....	85
Rutina 3-17. Peso, Diámetro y Densidad en función a la selección del Option Button 1 y 2 - Corona.....	85
Rutina 3-18. Desarrollo de la lista de ecuaciones dentro del Command B3 de la sección Corona.....	86
Rutina 3-19. Desarrollo del Burden, Constante de roca corregida, B máximo y B práctico en Arrastre, ayuda corona, hastial y ayuda hastial.	87
Rutina 3-20. Desarrollo del N° Taladros, Carga de fondo, Carga columna, carga por taladro y número de cartuchos por taladro en la sección Arrastre	88
Rutina 3-21. Desarrollo de las distancias y número de taladros para Ayuda Hastial, Ayuda Corona y Hastial	89
Rutina 3-22. Espaciamiento del Arrastre, centro y esquinas	89
Rutina 3-23. Desarrollo del Círculo Base como punto de partida	91

Rutina 3-24. Esquema para el diseño del primer cuadrante – desarrollo del segundo círculo base.....	91
Rutina 3-25. Esquema de cotación para el primer cuadrante.....	92
Rutina 3-26. Desarrollo del primer cuadrante con el método Array.....	93
Rutina 3-27. Desarrollo del segundo cuadrante con los métodos PolarPoint, AddCircle y Arraypolar	94
Rutina 3-28. Desarrollo del Tercer cuadrante.....	96
Rutina 3-29. Desarrollo del cuarto cuadrante	98
Rutina 3-30. Desarrollo del punto de partida – Ayuda corona	99
Rutina 3-31. Desarrollo del Stopping C para dos y tres taladros	101
Rutina 3-32. Desarrollo del Stopping C para cuatro, cinco y seis taladros	103
Rutina 3-33. Desarrollo del Punto de partida del Stopping B.....	104
Rutina 3-34. Desarrollo del Stopping B	105
Rutina 3-35. Desarrollo del punto de partida de la sección corona.....	107
Rutina 3-36. Transformación de arco en círculos equidistantes	108
Rutina 3-37. Desarrollo del Arrastre	110
Rutina 3-38. Desarrollo del Hastial derecho	111
Rutina 3-39. Desarrollo del Hastial izquierdo.....	112

RESUMEN

La presente tesis tiene como objetivo automatizar el modelo matemático Holmberg para el cálculo y diseño de mallas mediante lenguaje de programación VBA tomando como caso práctico la galería 710 SE del prospecto Monserrat.

En primera instancia se realizó el respectivo mapeo geomecánico y toma de muestras seguido de los ensayos de mecánica de rocas en donde se obtuvieron resultados satisfactorios debido a la calidad de las muestras; con una densidad promedio de 2.7 t/m³. En total se utilizaron tres probetas, de las cuales la tercera fue dispuesta para el ensayo de compresión uniaxial obteniendo valores de esfuerzo máximo y mínimo iguales a 45 y 0 Mpa respectivamente.

Posteriormente se determinó la calidad del macizo rocoso en base a los resultados del mapeo geomecánico, definiendo de esta manera un RMR del tipo III y un RQD de 75.4.

En el tema de programación se trabajó en tres etapas principales. La primera que establece la consolidación de la base de datos. La segunda que parte de una programación orientada a objetos estableciendo el esquema de desarrollo con los diversos parámetros y variables operacionales del modelo matemático Holmberg, obteniendo resultados cuantitativos de carga, número de taladros, burden y espaciamiento. La tercera etapa genera resultados a nivel de diseño utilizando la interfaz ActiveX de AutoCAD y VBA la cual define y dibuja la distribución final de los taladros a escala en formato DWG.

Palabras Clave: Visual Basic, Diseño de Malla, AutoCAD, Programación, Automatización, Holmberg

ABSTRACT

The present thesis aims to automate the model mathematical Holmberg for the calculation and design of meshes using the VBA programming language taking a particular case the gallery 710SE in the Monserrat mining prospect.

Firstly increases, the respective geomechanical mapping and sampling got carried out, followed by the rock mechanics tests, where satisfactory results got obtained because of quality samples; with an average density of 2.7 t / m³. In total three specimens got used, of which the third one was used for the uniaxial compression test, obtaining maximum and minimum stress values equal to 45 and 0 MPa respectively.

Subsequently, the quality rock mass was determined based on the results of the geomechanical mapping, defining this way a rock class III in RMR and a value of 75.4 in RQD.

As for programming subject, it got worked in three main stages. The first one that set up the consolidation of database. The second one that starts from an object oriented programming setting up the development scheme with the parameters and operational variables of the Holmberg mathematical model through the VBA programming language, obtaining quantitative results of load, number of holes, burden and spacing. The third stage generates results at the design level using the ActiveX interface of AutoCAD and VBA which defines and draws the final distribution of the scaled holes in DWG file.

Keywords: Visual Basic, Mesh Design, AutoCAD, Programming, Automation, Holmberg

INTRODUCCION

Actualmente el sector minero se ha venido desarrollando con nuevas propuestas de programación, nuevos softwares y metodologías para simplificar tareas de diseño y cálculo referente a los procesos de perforación y voladura en labores frontales, existen diversos lenguajes de programación los cuales contribuyen al desarrollo del proyecto consiguiendo resultados a corto y mediano plazo.

La problemática muestra que no existe un sistema que genere automáticamente un diseño de malla en tiempo real y que existe un desconocimiento sobre las herramientas que nos ofrece Microsoft Office y AutoCAD para materializar modelos matemáticos. Esto conlleva a generar diseños lentos y en ciertos casos sin un sustento teórico técnico.

Es así que surge la pregunta. ¿Como automatizar el modelo matemático Holmberg utilizando un lenguaje de programación? La hipótesis central sugiere que esto permitirá diseñar diferentes tipos de malla en formato DWG, sujeta a distintos escenarios en función de las variables controlables y no controlables.

Por lo expuesto, el objetivo de la presente tesis es desarrollar un sistema automatizado para la elaboración de diseños de malla mediante lenguaje de programación VBA que englobe de manera integral los procesos de perforación y voladura partiendo del presente caso de estudio y utilizando una base teórica sólida como lo es el modelo matemático Holmberg.

Por lo cual la presente investigación desarrolla los siguientes ítems. En el capítulo I se da a conocer el planteamiento del problema: Objetivos, Justificación, Alcance e Hipótesis del proyecto.

El capítulo II describe los aspectos generales de programación y evaluación geomecánica del proyecto, incluido los aspectos de cálculo y diseño referentes al modelo matemático de Holmberg.

El capítulo III hace referencia a la metodología empleada en donde se tiene como línea de estudio una investigación de nivel Aplicativo.

Seguidamente se muestra en detalle el proceso de evaluación geomecánica partiendo de las observaciones geológicas y dar paso a los distintos ensayos de laboratorio para el cálculo de

las propiedades físicas y mecánicas más relevantes, RQD y la clasificación geomecánica del macizo rocoso.

Posteriormente se detalla todo el diseño de programación empleado para la representación gráfica del sistema propiamente dicho estableciendo en primera instancia una base de datos sólida seguido del desarrollo del esquema de programación para la estructura del modelo matemático empleado y su respectivo diseño en formato DWG.

El capítulo IV hace referencia al procesamiento, análisis e interpretación de resultados mostrando un caso práctico en donde se valida el sistema propuesto mediante el procesamiento de las variables controlables y no controlables, analizando e interpretando los resultados finales en relación a las hipótesis planteadas.

CAPITULO I

PLANTEAMIENTO DEL ESTUDIO

1.1 Planteamiento y Formulación del Problema

Según DE LA CRUZ MERCADO. Ingeniero de planeamiento AESA. Actualmente se viene generando nuevos sistemas de contratación para la preparación y desarrollo de labores frontales, pero no se han enfocado en un análisis previo al desarrollo del proyecto y a cómo debería ser el diseño de excavación a mediano y corto plazo ya que conforme se va desarrollando, las características del terreno pueden cambiar y es allí donde debemos saber qué tipo de malla debemos utilizar aplicando sistemas automatizados de diseño (1).

Según PÉREZ MONTERO et al, en su artículo **“La Programación Orientada a Objetos Facilidad para Crear”** sostiene que: “Teniendo en cuenta que la programación permite encontrar diversas maneras de abordar problemas y de plantear soluciones, [...] ¿Por qué no enseñar a que estas nuevas generaciones construyan sus propias herramientas con el modelo de programación orientado a objetos para dar solución a los problemas? Este contexto es, por lo tanto, un importante desafío donde se hace necesario implementar la programación orientada a objetos” (2).

Sabemos muy bien que no todas las empresas hacen uso de estas herramientas. De los 750M de usuarios que puedan tener Microsoft Excel solamente el 60% de estos usuarios hacen un uso adecuado de las funciones y características que ofrece M. Excel y solo el 30% de ellos han accedido al conocimiento de las herramientas que ofrece Visual Basic para Aplicaciones.

El lenguaje de programación es una herramienta fundamental para la solución de un problema de este tipo, no solo se aplica al campo de la ingeniería, es universal.

En ese sentido resulta necesario contar con una herramienta automatizada que nos permita crear un diseño de malla en tiempo real considerando todos los parámetros controlables y no controlables, con escenarios múltiples y una demostración práctica con parámetros reales.

1.2 Formulación del Problema

1.2.1 Problema General

¿Cómo automatizar el modelo matemático Holmberg para el cálculo y diseño de mallas?

1.2.2 Problema Especifico

- ¿Como optimizar el proceso de diseño mediante la automatización del modelo matemático Holmberg reduciendo el tiempo empleado en esta actividad?
- ¿Como destacar la similitud a nivel de carga y número de taladros mediante la validación de los resultados del programa automatizado?
- ¿Como obtener resultados conservadores a nivel de diseño manteniendo una distribución de taladros progresiva?

1.3 Objetivos

1.3.1 Objetivo General

Automatizar el modelo matemático Holmberg mediante lenguaje de programación VBA para el cálculo y diseño de mallas.

1.3.2 Objetivo Específico

- Optimizar el proceso de diseño mediante la automatización del modelo matemático Holmberg reduciendo el tiempo empleado en esta actividad
- Destacar la similitud a nivel de carga y número de taladros mediante la validación de los resultados del programa automatizado.
- Obtener resultados conservadores a nivel de diseño manteniendo una distribución de taladros exponencial.

1.4 Justificación e Importancia

1.3.1 Teórica

Un buen diseño de malla se adapta a un buen modelo matemático dicho esto las empresas de construcción utilizan este criterio para obtener resultados factibles al momento de realizar sus diseños. Es así que se lleva a cabo este estudio para automatizar el diseño de malla utilizando como base el modelo matemático Holmberg por su gran aceptación de optimizar los procesos de perforación y voladura.

El valor añadido en este proyecto de investigación es el lenguaje de programación VBA de AutoCAD el cual permite la ejecución simultánea de AutoCAD y el entorno VBA proporcionando un control de programación mediante la interfaz de ActiveX. Esta interacción entre AutoCAD, ActiveX y VBA constituye una interfaz muy avanzada para crear modelos de este tipo.

1.3.2 Práctica

El beneficio es universal ya que al tratarse de un sistema automatizado que mejora los procesos del modelo Matemático Holmberg se puede llegar a evitar los errores de cálculos manuales y se tendría una representación gráfica del diseño de malla en tiempo real, a escala y con detalles explícitos de carga, distancia y secuencia.

1.5 Alcance

- Características del sistema operativo de 64bits, Windows 7 e instalación de AutoCAD 2020
- No se consideran todos los explosivos presentes en el mercado, los tipos de broca como los tipos de arranque son limitadas.
- Se consideran arranques de un taladro de alivio, cortes quemados de cinco y sus variantes.
- Valor de flecha de arco igual a 0.5 metros.
- Diseño de malla en formato DWG enfatizando la cotación, número de taladros, escala, color, etiquetas y distribución.
- Resumen general automatizado con resultados de carga, secuencia y número de taladros.

1.6 Hipótesis y Descripción de Variables

1.6.1 Hipótesis General

La automatización del modelo matemático Holmberg permite el cálculo y diseño de mallas .

1.6.2 Hipótesis Específica

- La automatización del modelo matemático Holmberg optimiza el proceso de diseño y reduce el tiempo empleado en esta actividad
- La validación de los resultados del programa automatizado destaca la similitud a nivel de carga y número de taladros.
- Los resultados a nivel de diseño mantienen una distribución de taladros exponencial.

1.6.3 Operacionalización de las Variables

Variable	Definición Conceptual	Dimensiones	Indicadores	Instrumento
Variable Independiente X: Automatización del Modelo Matemático Holmberg	Es el desarrollo de un conjunto de algoritmos sujetos a un lenguaje de programación con la cual generan un esquema de desarrollo sobre la base del modelo matemático en mención lo que permite dinamizar los criterios de cálculo y diseño	Entorno de Visual Basic	Carpetas, Formularios y Módulos Métodos y Propiedades Controles	Lenguaje de Programación VBA
		Autocad VBA	Entidades de Dibujo	ActiveX
Variable Dependiente Y: Cálculo y Diseño de Mallas de perforación	Conjunto de operaciones matemáticas necesarias para averiguar el resultado de algo y representarlo de forma gráfica. Ese algo se define como un número de agujeros en cantidades y longitudes distintas distribuidas a lo largo del área transversal de una labor frontal.	Evaluación Geomecánica	Resistencia a la Compresión RMR, Q BARTON, RQD, Caracterización Geomecánica	Equipos de corte y medición Tablas y Ávacos Formato de Mapeo Geomecánico
		Modelo Matemático Holmberg	Avance por Disparo Factor de Carga Burden Máximo y Burden Práctico	Fórmulas

Figure 1-1. Operacionalización de las Variables

CAPITULO II

MARCO TEORICO

2.1 Antecedentes

2.1.1 Entorno Internacional

RAMIREZ RUIZ en su tesis “**Desarrollo De Una Aplicación En Programación Orientada A Objetos VBA En Excel, Para El Diseño De Plantas Potabilizadoras De Filtración Directa**”. Desarrolló un sistema de cálculo automatizado con el objetivo de determinar el número de filtros, dimensión de tuberías, volumen y tiempos de recirculación de agua tratada para un mejor procesamiento de la planta potabilizadora, esto se hizo mediante un lenguaje de programación orientada a objetos en complemento con la aplicación Custom UI Editor la cual vincula la macro a la cinta de opciones, consiguiendo de esa manera acelerar el esquema de cálculo. (3)

SINGH & RAWAT en su artículo “**Integration of CAD model with Calculation Sheet using Excel VBA**”. Sostiene el mismo propósito de la presente investigación el cual es automatizar los procesos de cálculo y diseño mejorando la productividad del sistema CAD el cual genera diseños en 2D y 3D referentes a los componentes Cotter y Spigot. Esto se hizo posible mediante el lenguaje de programación de Visual Basic obteniendo una sincronización de esta con el programa CAD sin la necesidad del uso de sistemas complejos. (4)

2.1.2 Entorno Nacional

MATEO MENDOZA et al en su artículo “**Modelo Matemático de Pearse y Holmberg para Reducir la Zona de Daños en Labores Horizontales de Minería Subterránea**”, parte del objetivo de mejorar la distribución de los taladros y el consumo de explosivos utilizando en conjunto los modelos matemáticos en mención, obteniendo como resultado un burden de arranque óptimo el cual reduce significativamente el número de taladros y segundo; los explosivos ideales, en base a una selección condicional minimizando la zona de daños. (5)

MAYTA FLORES, en su tesis “**Implementación de un Programa para los Procesos de Análisis, Diseño y Dibujo de Zapatas**” valida la hipótesis de optimizar los procesos en mención mediante la utilización de algoritmos propios de la interacción entre AutoCAD y Visual Basic.net obteniendo resultados conservadores de cálculo y un diseño automatizado. Cabe precisar que no se hace énfasis en el uso adecuado del programa por lo que resulta difícil entender su aplicación. (6)

2.2 Generalidades

2.2.1 Ubicación y Acceso del Prospecto

Según la plataforma GEOCATMIN las coordenadas UTM del presente caso de estudio corresponden al prospecto minero Monserrat, siendo estas las siguientes. (7)

Tabla 2-1. Coordenadas del Prospecto

VERT	NORTE	ESTE
1	8713000	461000
2	8712000	461000
3	8712000	460000
4	8713000	460000

El prospecto minero se encuentra en el anexo de Quero, distrito de Molinos, provincia de Jauja. Se detallan a continuación las respectivas distancias aproximadas según los tramos.

Tabla 2-2. Distancias por cada tramo de ruta

TRAMO	DISTANCIA	VIA	TIEMPO
Lima – Jauja	265 km	Asfaltado	6 h 32 min
Jauja – Molinos	8.6 km	Asfaltado	13 min
Molinos – Quero	8.7km	Trocha	16 min
Quero - UEA Monserrat	4.8 km	Trocha	9 min
TOTAL	287.1 km		7 h 10 min

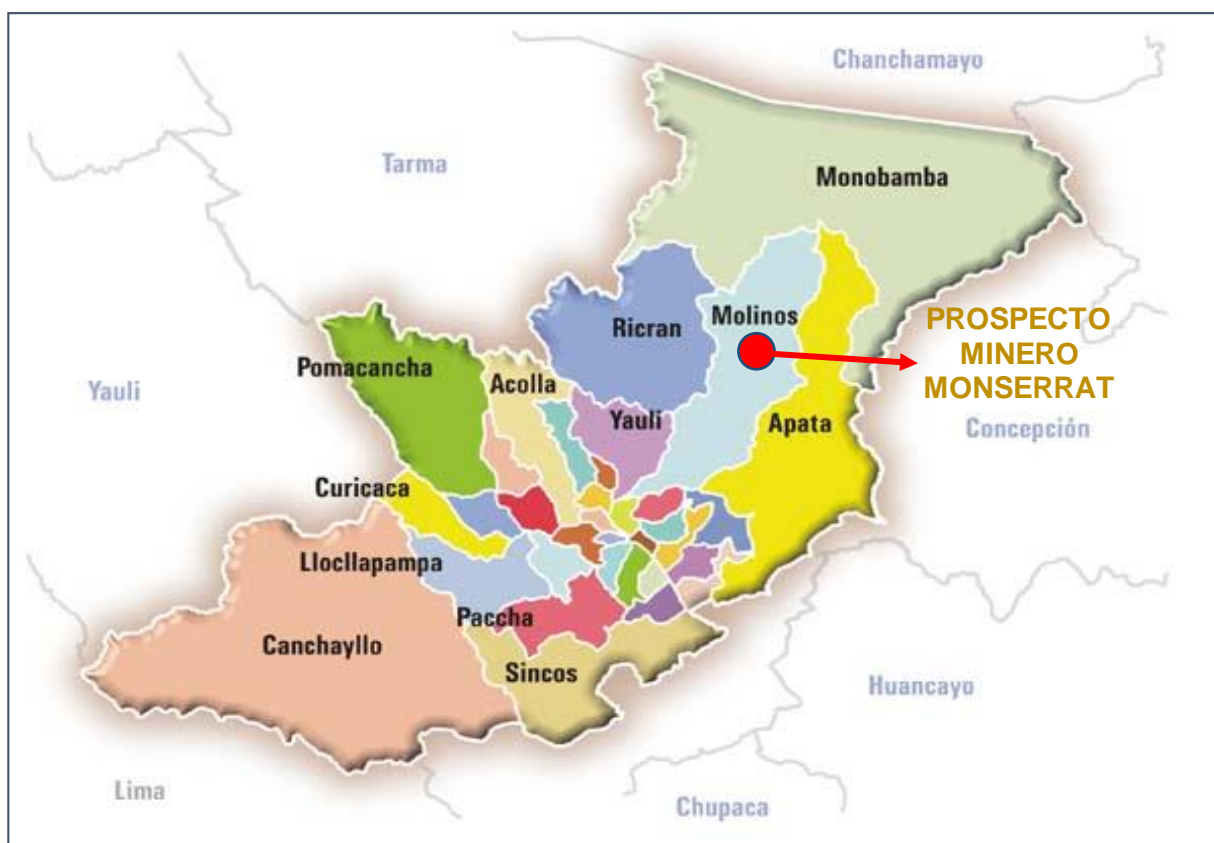


Figure 2-1. Mapa de Ubicación del Prospecto Minero

2.2.2 Geología General

Localmente, el principal material que está conformando el subsuelo del área del proyecto es de tipo fluvio-aluvial, constituido por capas horizontales de conglomerado de grava mal gradada, alternando con capas de arenisca de arena fina con limo poco arcilloso.

Según el plano N°1 del anexo 5. En la galería 710SE existe presencia de cuarzo lechoso como diseminados y vetillas, puntos de molibdeno con óxidos de fierro, cobre y carbonatos de cobre. Alteraciones del tipo silicificación y limonitización. El tipo de roca caja corresponde a una Diorita.

2.3 Propiedades Físicas y Mecánicas de los Materiales Rocosos

Existen una serie de parámetros que se emplean para la identificación y descripción cuantitativa de las propiedades elementales de las rocas que permitan establecer una primera clasificación geotécnica. Estas propiedades, denominadas propiedades índice serán las que determinan en primera instancia, las propiedades y el comportamiento mecánico de la matriz rocosa.

Las propiedades físicas o propiedades índice se determinan en laboratorio; las más importantes a nivel de influencia en el comportamiento mecánico son la porosidad, peso específico, resistencia a la compresión, deformabilidad, resistencia a la tracción y la velocidad de propagación de las ondas.

2.3.1 Densidad

En función de cómo se encuentre la roca, se puede distinguir la densidad natural y la densidad seca, para efectos prácticos se hace uso de la misma ecuación.

$$\rho = \frac{Wm}{Vm}$$

- ρ : Densidad natural o húmeda (Kg/m³)
- Wm : Peso de la muestra (Kg)
- Vm : Volumen de la muestra (m³)

2.3.2 Resistencia a la Compresión Simple

También llamada resistencia uniaxial, es el máximo esfuerzo que soporta la roca sometida a compresión uniaxial, determinada sobre la base del ensayo de compresión utilizando una probeta cilíndrica sin confinar, la cual está representada por la fórmula:

$$\sigma_c = \frac{F_c}{A}$$

- F_c: Fuerza compresiva
- A: Área de aplicación
- σ_c: resistencia o esfuerzo de compresión

También se puede determinar de forma aproximada a partir de índices obtenidos en sencillos ensayos de campo, como el ensayo de carga puntual o el martillo Schmit.

2.4 Clasificación Geomecánica

Las clasificaciones geomecánicas tienen como objetivo asignarle un valor de calidad al macizo rocoso a partir de ensayos simples y la caracterización de las discontinuidades.

Para definir los sistemas de clasificación geomecánica OSINERGMIN, afirma lo siguiente:

“Los sistemas de clasificación del macizo rocoso sustituyen un enfoque para estimar las propiedades del macizo rocoso a gran escala. En la industria minera, el índice GSI y los sistemas RMR y Q, son parámetros de entrada para muchos métodos de diseño, así como parámetro de entrada de muchos programas de modelamiento numérico. Los sistemas de clasificación geomecánica son un intento de representar con un solo valor las propiedades de resistencia de un macizo rocoso. El macizo rocoso es usualmente un material altamente anisotrópico y puede ser representado por más de un sistema de clasificación”. (8)

2.4.1 Rock Mass Rating (RMR)

El sistema Rock Mass Rating es una referencia mundial para clasificar geomecánicamente el macizo rocoso desarrollado inicialmente por Bieniawski en 1973, posteriormente actualizado en 1989, fue nuevamente actualizado en 2014, modificándose los criterios de valoración e incorporando el efecto de la alterabilidad de la matriz rocosa por el agua, el cual ofrece un enfoque más acertado para calificar al macizo rocoso. (8)

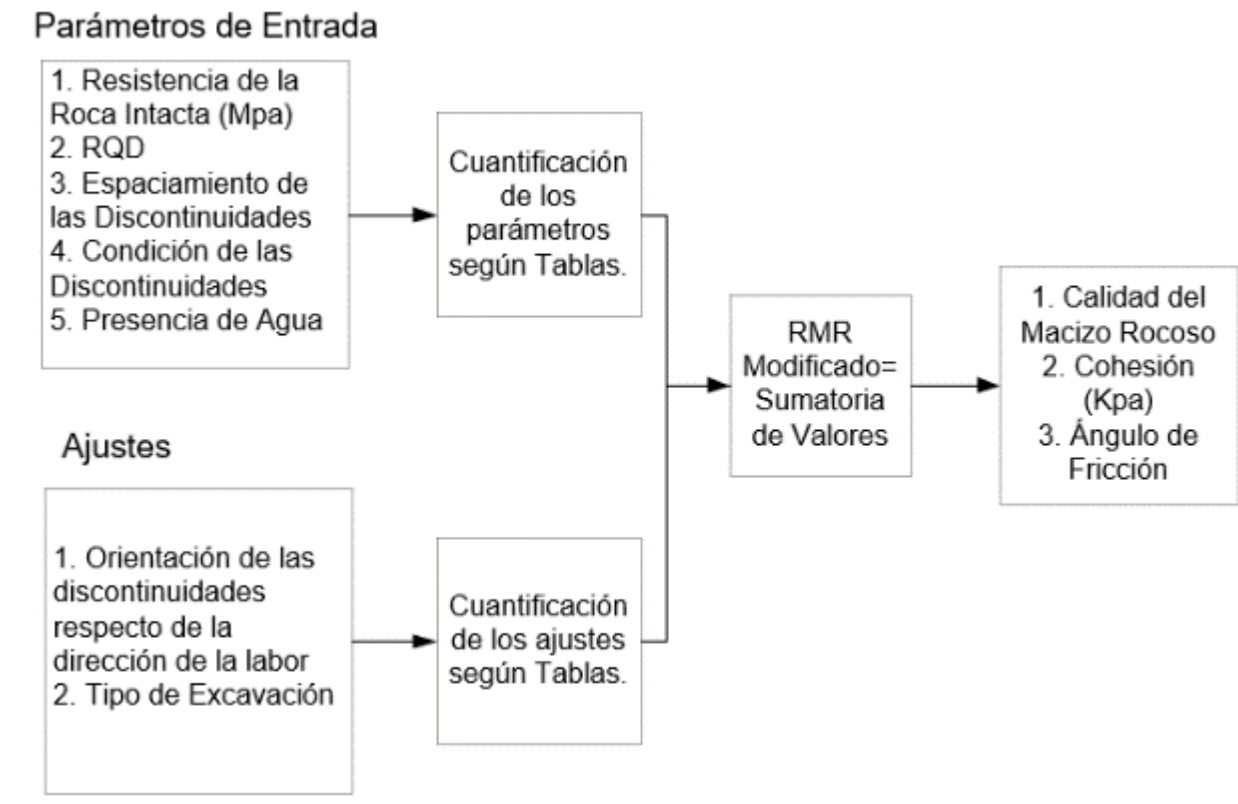


Figure 2-2. Proceso de Elaboración RMR89

2.4.2 Designación de la Calidad de la Roca (RQD)

El índice de calidad de roca (Rock Quality Designation. Deere, 1964), es uno de los sistemas más empleados para caracterizar la competencia del macizo rocoso. El RQD asigna un porcentaje de calificación al macizo rocoso, de 100% (el más competente) a 0% (el menos competente). (8)

La interpretación del valor del RQD se muestra en la tabla siguiente:

Tabla 2-3: Interpretación de valores de RQD (Deere 1964)

DESCRIPCION	VALOR (%)
Muy mala	<25
Mala	25-50
Regular	50-75
Buena	75-90
Excelente	90-100

Fuente: OSINERGMIN, pág. 21

El RQD es el porcentaje de fragmentos de longitud superior a 10 cm, sobre la longitud total del testigo. Su expresión se refleja en la fórmula siguiente.

$$RQD(\%) = \frac{\sum Long. > 10cm}{L. total} * 100$$

También es posible determinar este valor mediante un mapeo geomecánico el cual consiste en dibujar líneas de detalle sobre las áreas en estudio, es así que la fórmula para determinar el RQD viene dada por:

$$RQD = 100 * e^{-0.1*\lambda} * (0.1\lambda + 1) \quad \lambda = \frac{N^{\circ} Fracturas}{metro lineal}$$

En resumen, al tratarse de líneas de detalle y no muestras de testigo las condiciones para su tratamiento es distinta ya que no se requerirán de muestras de testigo para determinar este valor, se puede hacer uso de una línea imaginaria dibujada a lo largo de una sección con diferentes discontinuidades que intersectan a la misma.

2.4.3 Caracterización de las Discontinuidades

Según el Anexo 4 La condición de las juntas, así como la presencia de agua y orientación, mantienen el mismo criterio de funcionalidad, estas representan un valor de calidad en función de la selección que se haga dentro de los rangos de clasificación que se muestran ya sea por tipo de persistencia, abertura, rugosidad, relleno, alteración, espaciamiento y presencia de agua. Para el caso de la orientación el valor por defecto tiende a ser negativo, esto con el fin de obtener el valor más desfavorable que represente al macizo rocoso.

2.5 Modelo Matemático Holmberg

Según MENDOZA et al. La aplicación del modelo matemático de Roger Holmberg consiste en diseñar la malla de perforación de una labor frontal, como se aprecia en la figura 2.3, considerando parámetros controlables como burden, dimensión de los taladros, tipo de arranque, tipo de explosivos, dimensiones de la labor y parámetros no controlables como peso específico, RMR, GSI y RQD optimizando el proceso post voladura. (5)

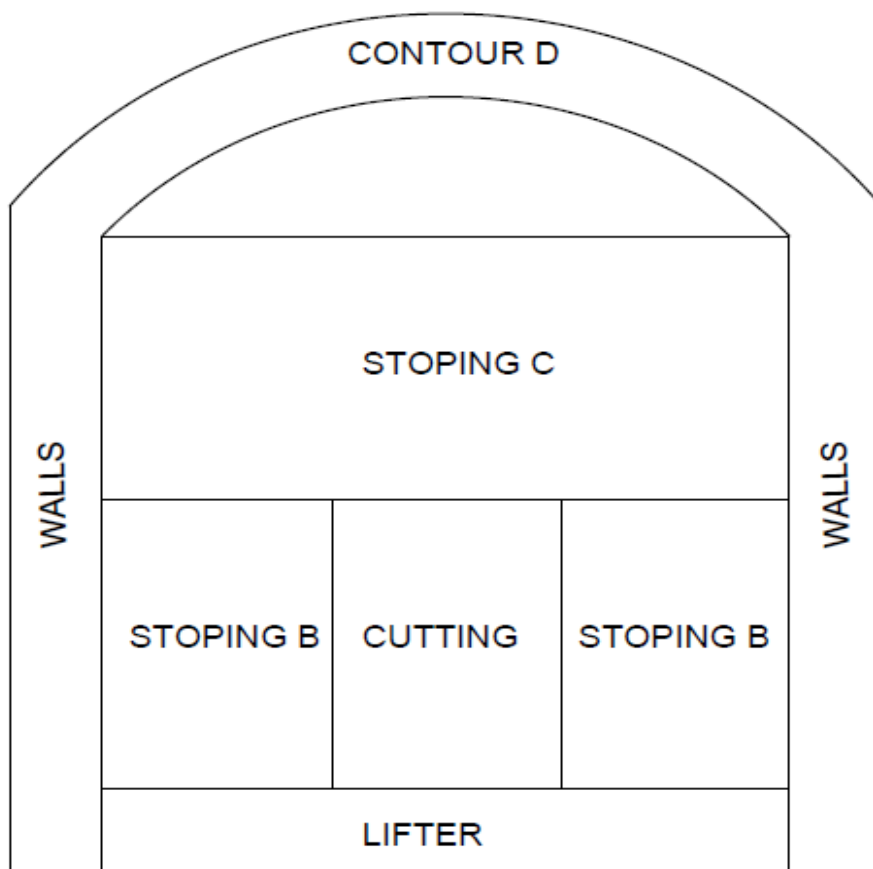


Figure 2-3. División de un frente

2.5.1 Esquemas de Corte

2.5.1.1 Corte Quemado

Comprende un grupo de taladros de igual diámetro perforados cercanamente entre sí con distintos trazos o figuras de distribución, algunos de los cuales no contienen carga explosiva de modo que sus espacios vacíos actúan como caras libres para la acción de los taladros con carga explosiva cuando detonan.

El diseño más simple es de un rombo con cinco taladros cargados (cuatro vacíos en los vértices y uno cargado al centro). Para ciertas condiciones de roca el esquema se invierte con el taladro central vacío y los cuatro vértices cargados, en la figura 2.6 se puede apreciar todas las variantes del corte quemado. (9)

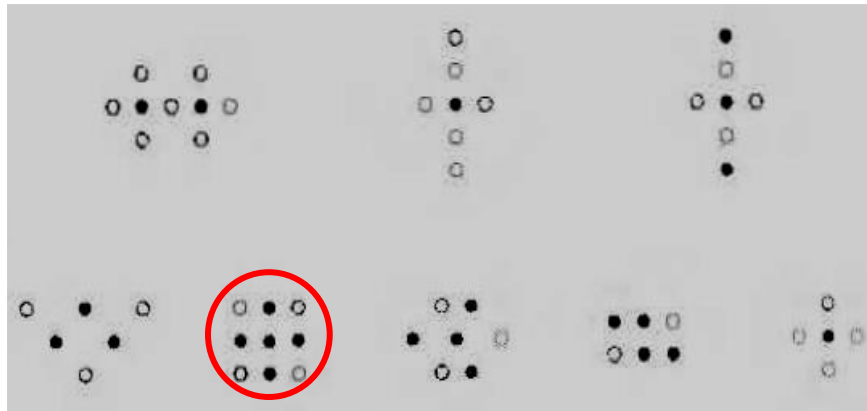


Figure 2-4. Tipos de corte quemado

2.5.1.2 Corte de Cuatro Secciones

El esquema es muy parecido a un corte cilíndrico, la diferencia radica en la secuencia de arranque, burden y espaciamiento. Si queremos una buena fragmentación y salida del material, la distancia entre el taladro vacío y el taladro cargado no debe ser superior a $1.7D_2$. Si la concentración de carga es elevada se producirá la sinterización de la roca fragmentada y el fallo del cuele. (9)

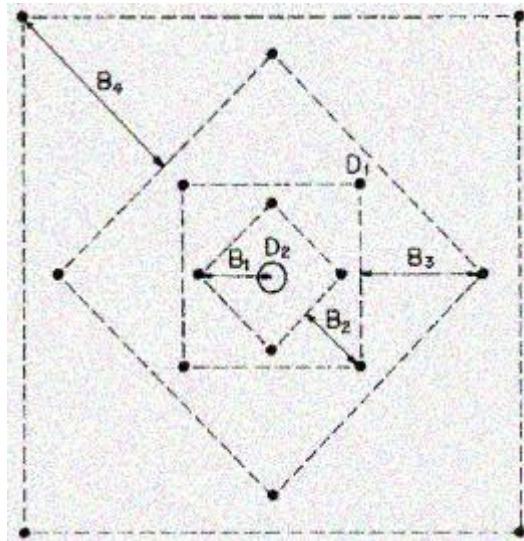


Figure 2-5 Corte de cuatro secciones

Donde:

- D1: Diámetro del taladro cargado
- D2: Diámetro del taladro vacío
- B1: Búrden práctico del primer cuadrante

2.5.2 Avance por Disparo

Existe una diferencia entre profundidad teórica y profundidad práctica. En los arranques de cuatro secciones la profundidad teórica se puede estimar usando la siguiente formula:

$$H = 0.15 + 34.1Dequiv - 39.4 * Dequiv^2$$

Donde:

- H: Profundidad teórica
- Dequiv: Diámetro equivalente

La profundidad práctica es la distancia real que se desea perforar independiente del diámetro equivalente. Es así que para efectos de cálculo ya sea de burden o espaciamento se utiliza únicamente la profundidad teórica.

Cuando se utilizan cueles de dos o más taladros vacíos debemos establecer un diámetro equivalente como un valor que engloba a todas estas. (5)

$$Dequiv = D2 * \sqrt{n}$$

Donde:

- n: Número de taladros vacíos
- D2: Diámetro de los taladros vacíos

El avance por disparo es la longitud real de perforación o H práctico multiplicada por un factor de incidencia, dicho factor generalmente toma valores por debajo del 95%. La longitud de perforación puede dividirse en tres partes según el tipo de carga que se le aplique, cada una de estas longitudes se estiman en base al siguiente cuadro de fórmulas.

Tabla 2-4. Longitud de carga en columna, fondo y taco para cada sección

Tipo	Arranque	Arrastre	Corona	Ayuda Hastial	Ayuda Corona	Hastiales
Lcc	Hp-(Lcf+T)	Hp-(Lcf+T)	Hp-(Lcf+T)	Hp-(Lcf+T)	Hp-(Lcf+T)	Hp-(Lcf+T)
Lcf	Hp/3	Hp/3	Hp/6	Hp/3	Hp/3	Hp/6
T	10*D1	10*D1	10*D1	10*D1	10*D1	10*D1

Donde:

- Lcc: Longitud de Carga en Columna
- Lcf: Longitud de Carga en Fondo
- T: Taco o Retacado
- Hp: Profundidad práctica

2.5.3 Factor de Carga

El factor de carga expresa la idea de lo que necesitamos en términos de voladura, supongamos que queremos realizar un avance de 3 m. Lo que necesitaríamos saber es que cantidad de explosivo necesito para dicho avance. En efecto se establece una relación que puede estar expresada en Kg/m, kg/Área, Kg/Ton, etc. De aquí se deriva los diferentes tipos de carga tales como: Factor de Carga Lineal (Kg/m) y Factor de Potencia (Kg/Ton)

Existen muchas fórmulas que demuestran su grado de validez pero que no son 100% validas, además su aplicación va de la mano con variables que dependiendo del tipo de voladura resultan ser satisfactorias.

Según LÓPEZ JIMENO. “La densidad de un explosivo es un factor importante para el cálculo de la cantidad de carga necesaria para una voladura. Por regla general, en el fondo de los barrenos, que es donde se necesita mayor concentración de energía para el arranque de la roca, se utilizan explosivos más densos como son los gelatinosos e hidrogeles, mientras que en las cargas de columna se requieren explosivos menos densos, como son los pulverulentos y los de base ANFO”. (9)

En base a este hecho la concentración lineal de carga se puede calcular en base a la siguiente fórmula:

$$q = \frac{7.854 * \rho_{exp} * \phi_{exp}^2}{10000}$$

Donde:

- q: Factor de carga lineal (Kg/m)
- ρ_{exp} : Densidad del explosivo (g/cm³)
- ϕ_{exp} : Diámetro del explosivo (mm)

El factor de carga lineal como valor unitario puede ser determinado según los tipos de explosivos a utilizar, si se tiene dos explosivos de selección para un disparo en frente entonces se tendrán dos factores de carga lineal (q1 y q2) donde q1 representa al explosivo con mayor potencia.

2.5.4 Burden Máximo & Burden Práctico

2.5.4.1 Enfoque de Arranque

Partiendo de un arranque de cuatro secciones, el burden máximo es la distancia de separación que existe entre el taladro de alivio y el taladro cargado, pero es una distancia teórica donde no se considera los errores de perforación. El burden práctico es el mismo valor solo que en adición a esta se incluye los errores de perforación.

El burden máximo se calcula sobre la base de 1.7Dequiv. Cuando la desviación de perforación es superior al 1% el burden práctico se calcula sobre la base de:

$$B_{pr\acute{a}ctico} = 1.7Dequiv - Ep = 1.7Dequiv - (\alpha * L + e')$$

Donde:

- α : Desviación Angular (m/m)
- L: Profundidad Teórica (m)
- e' : Error de Emboquille (m)
- Ep: Error de Perforación(m)

El Ancho de abertura inicial generado “Ah” se calcula sobre la base de:

$$(B_{max} - Ep) * \sqrt{2}$$

De allí en adelante para calcular el resto de cuadrantes, se considera que ya existen unos huecos rectangulares “Ah” y que se conocen las concentraciones lineales de carga “q”, el valor del burden máximo se calcula a partir de:

$$B_{max} = 8.8 * 10^{-2} * \sqrt{\frac{Ah * q1 * PRPanfo}{D1 * c}}$$

Donde:

- Ah: Ancho de abertura
- q1: Factor de carga lineal
- PRPanfo: Potencia Relativa en Peso
- D1: Diámetro del Taladro Cargado
- c: Constante de Roca

Para la constante de roca se utilizó la fórmula propuesta por Langefors.

$$C = 0.8784 * CE + 0.0052$$

Siendo CE el consumo específico del explosivo, el cual es calculado mediante la fórmula propuesta por Ashby:

$$CE = \frac{0.56 * \rho r * \tan\left(\frac{GSI + 15}{2}\right)}{3\sqrt{\frac{115 - RQD}{3.3}}}$$

Donde:

- CE: Consumo Específico del Explosivo
- GSI: Geological Strength Index
- RQD: Rock Quality Designation
- ρr : Densidad de la Roca

Finalmente, el ancho generado para el resto de los cuadrantes se calcula sobre la base de:

$$B_{practico'} + \left(\frac{Ah'}{2}\right) - Ep * \sqrt{2}$$

Donde:

- Ah': Ancho generado del cuadrante anterior
- Ep: Error de Perforación
- B practico': Burden Práctico del cuadrante anterior

2.5.5 Sumario de Fórmulas para cada Sección

Se muestra un resumen de las fórmulas antes mencionadas incluyendo otras propias de cada sección que difieren del Arranque pero que mantienen una coherencia propia por cada sección del modelo matemático.

El esquema de funciones de la sección corona es muy diferente al resto de secciones esto debido a las diferencias en B máximo, B práctico, Espaciamiento y número de taladros. También es importante enfatizar las características de esta sección ya que se asemeja a la forma de un arco la cual es generada según las dimensiones de flecha y cuerda.

Tabla 2-5. Ecuaciones correspondientes a la sección Corona

CRITERIO	CORONA	UNIDADES
B máximo	$\frac{\text{Espaciamiento}}{S/B}$	M
B práctico	$B_{max} - H_{Teorico} * \text{sen}(\gamma) - E_p$	M
Arco ¹	$Arco = 2 * \left(\frac{Fl}{2} + \frac{Cu^2}{8Fl} \right) * \sin^{-1} \left(\frac{4Fl * Cu}{4Fl^2 + Cu^2} \right) * \frac{\pi}{180}$	M
Espaciamiento	$K * D3$	M
Nº Talad/Corona	$\frac{Arco}{\text{Espaciamiento}} + 1$	Unid
Nº Cart/Talad	Ct/P_{expl}	Unid
Carga Fondo	$q3 * L_{cf}$	Kg/Talad
Carga Columna	$Reductor * C_f$	Kg/Talad
Carga/Taladro Ct	$C_f + C_c$	Kg/Talad
Carga Total	$Ct * N^\circ \text{ Talad}$	Kg
q mínima	$90 * D3^2$	Kg/m
q3	$\frac{7.85 * D_{expl}^2 * \rho_{expl}}{10000}$	Kg/m
K	16	Adimensional
S/B	0.8	Adimensional

Donde:

- γ : Ángulo de Talados de Contorno (Rad); para un avance de 3m se considera un realce de 3º que en radianes es 0.052.
- Fl, Cu: Flecha de arco (0.5) y Cuerda de arco (ancho de la labor)
- Reductor: Es un valor que va entre 0.5 y 0.9
- D3: Diámetro de los taladros de precorte
- K & S/B: Constante y Relación Espaciamiento Burden

Tabla 2-6. Ecuaciones en Lifter, Stopping y Walls

¹ La fórmula propuesta es de carácter particular propia del autor, esto debido a la necesidad de conseguir una distribución equidistante de los taladros de contorno, esto se realizó mediante el análisis de diferentes ecuaciones geométricas y trigonométricas obteniendo una ecuación en función de la cuerda y flecha de un arco, su validez se demuestra en el resultado del diseño.

CRITERIO	ARRASTRE	AYUDA HASTIAL	AYUDA CORONA	HASTIALES	UNID
B	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C * f * S/B}}$	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C * f * S/B}}$	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C * f * S/B}}$	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C * f * S/B}}$	M
Bmax	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C' * f * S/B}}$	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C' * f * S/B}}$	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C' * f * S/B}}$	$0.9 * \sqrt{\frac{q2 * PRPAnfo}{C' * f * S/B}}$	M
Bpráctico	$Bmax - HTeorico * \text{sen}(\gamma) - Ep$	$Bmax - Ep$	$Bmax - Ep$	$Bmax - Ep$	M
Espaciamiento	$\frac{ATunel + (2 * HTeorico * \text{sen}(\gamma))}{N^{\circ} Talad - 1}$	$S/B * Bmax$	$S/B * Bmax$	$S/B * Bmax$	M
Espaciamiento Esquinas	$Espaciamiento - HTeorico * \text{sen}(\gamma)$	Nothing	Nothing	Nothing	M
Nº Talad/sección	$\frac{ATunel + 2HTeorico * \text{sen}(\gamma)}{Bmax} + 2$	$\frac{Distancia}{Bmax * S/B} + 2$	$\frac{Distancia}{Bmax * S/B} + 2$	$\frac{Distancia}{Bmax * S/B} + 1$	Unid
Nº Cartuchos/Tal.	$Ct/Pexpl$	$Ct/Pexpl$	$Ct/Pexpl$	$Ct/Pexpl$	Cart/Talad
C'	$C + 0.07/B$	$C + 0.07/B$	$C + 0.07/B$	$C + 0.07/B$	Adimensional
Carga Fondo (Cf)	$q2 * Lcf$	$q2 * Lcf$	$q2 * Lcf$	$q2 * Lcf$	Kg/Talad
Carga Columna (Cc)	$q2 * Lcc$	$q2 * Lcc$	$q2 * Lcc$	$q2 * Lcc$	Kg/Talad
Carga/Taladro (Ct)	$Cf + Cc$	$Cf + Cc$	$Cf + Cc$	$Cf + Cc$	Kg/Talad
Carga Total	$Ct * N^{\circ} Talad$	$Ct * N^{\circ} Talad$	$Ct * N^{\circ} Talad$	$Ct * N^{\circ} Talad$	Kg
F	1.45	1.45	1.20	1.20	Adimensional
S/B	0.8	1.25	1.25	1.25	Adimensional
Distancia	Nothing	$Ancho Tunel - Ahc - Bph * 2$	$Altura Tunel - Ahc - Bc - Bz$	$Ancho Tunel - Bz - Bc * 2$	M

Donde:

- F: Factor de fijación
- C': Constante de Roca Corregida
- Pexpl: Peso del explosivo (Kg)
- Cf: Carga Fondo
- Cc: Carga Columna
- Bph: Burden Practico Hastial
- Ahc: Ancho generado de la cuarta sección
- Bc: Burden practico Corona
- Bz: Burden practico Zapatera
- B: Burden máximo sin replanteo
- Bmax: Burden Máximo (Replanteado)
- At: Ancho Túnel

2.6 Microsoft Visual Basic

Es un lenguaje de programación que nos permite crear programas para el sistema operativo Windows. A partir de 1993 Microsoft implemento Visual Basic para Aplicaciones (VBA) en base a Visual Basic que nos permitiera crear automatizaciones en cada una de las aplicaciones que comprende la suite ofimática.

Cuando se instala MO ya viene por defecto VBA sin embargo al abrir cualquiera de las aplicaciones no encontraremos en muchos casos activado la pestaña programador, por lo que tiene que ser activado desde la cinta de opciones de la pestaña archivo y activar la pestaña programador.

2.6.1 Paradigmas de Programación

Son las diferentes maneras de darle solución a un problema de programación mediante los estilos documentados como buenas prácticas en el mundo del desarrollo. Se pueden clasificar en dos grupos.

2.6.1.1 Paradigma Imperativo

Los programas consisten en una sucesión de instrucciones o conjunto de sentencias, como si el programador diera órdenes concretas. El desarrollador describe en el código paso por paso todo lo que hará su programa. (10)

Enfoques subordinados al paradigma de programación imperativa:

→ **Programación Estructurada:** La programación estructurada es un tipo de programación imperativa donde el flujo de control se define mediante bucles anidados, condicionales y subrutinas.

→ **Programación Procedimental:** Este paradigma de programación consiste en basarse en un número muy bajo de expresiones repetidas, englobarlas todas en un procedimiento o función y llamarlo cada vez que tenga que ejecutarse.

→ **Programación Orientada a Objetos:** En este modelo de paradigma se construyen modelos de objetos que representan elementos (objetos) del problema a resolver, que tienen características y funciones. Permite separar los diferentes componentes de un programa, simplificando así su creación, depuración y posteriores mejoras. La programación orientada a objetos disminuye los errores y promueve la reutilización del código. Es una manera especial de programar, que se acerca de alguna manera a cómo expresaríamos las cosas en la vida real tales. (10)

2.6.1.2 Paradigma Declarativo

Este paradigma no necesita definir algoritmos puesto que describe el problema en lugar de encontrar una solución al mismo. Este paradigma utiliza el principio del razonamiento lógico para responder a las preguntas o cuestiones consultadas tales como SQL Server.

2.7 Entorno de Visual Basic para Aplicaciones

2.7.1 Contenido de Carpetas

2.7.1.1 Carpeta Microsoft Excel Objetos

Contiene un módulo de clase asociado al libro del proyecto (llamado por defecto "ThisWorkbook") y un módulo de clase por cada una de las hojas de cálculo u hojas de

gráfico del libro. En particular, en estos módulos de clase se encuentran los procedimientos de eventos asociados al libro y a las hojas.

2.7.1.2 **Carpeta Formularios**

Contiene los formularios o cuadros de diálogo personalizado del proyecto y el código VBA asociado. Cada formulario que se diseña incluye los botones Maximizar, Minimizar y Cerrar de forma automática. Estos botones ya están implementados.

Para añadir un código al control, basta con hacer doble clic en él cuando se haya colocado en el formulario. De esta forma se abre la ventana de código del control.

2.7.1.3 **Carpeta Módulos de Clase**

Contiene los módulos de clase usados para la creación de nuevas clases de objetos. Los módulos de clase se utilizan especialmente para la escritura de los procedimientos de eventos asociados a los objetos Application y Chart (tema que no está en discusión en este proyecto).

2.7.1.4 **Carpeta Módulos**

Agrupar los diferentes módulos estándares (compuestos por uno o más procedimientos) que pueden ser llamados desde cualquier procedimiento del proyecto.

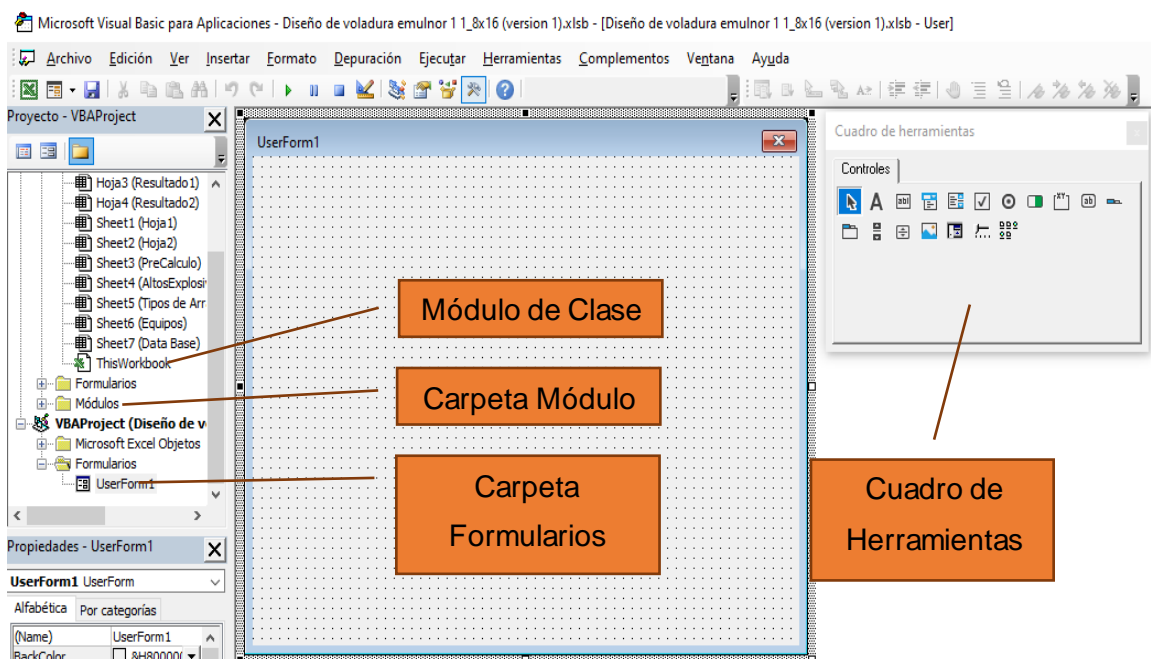



Figure 2-6. Características del editor de Visual Basic

- Para insertar un nuevo módulo en el entorno VBA, se usa la opción Módulo del menú Insertar, o haciendo clic en el icono 
- Si el explorador de proyectos no está visible, elija la opción **Explorador de proyectos** del menú **Ver** o pulse el método abreviado **Ctrl + R**.
- Para ver el código asociado a un módulo, haga doble clic en el nombre del módulo.
- Para pasar de un módulo a otro, en la ventana Proyecto, haga doble clic en el nombre del módulo que desea activar.
- Para eliminar un módulo, en la ventana Proyecto, haga **clic derecho** en el nombre del módulo que desea eliminar y elija **Quitar**
- Para dar nombre a un módulo, active el módulo y cambie la propiedad Nombre en la ventana de Propiedades.

2.7.2 Jerarquía de Objetos

La estructura de objetos es jerárquica, unos están contenidos en otros. El objeto Application es el de mayor rango, contiene a todos los demás. Representa al propio programa Excel y su uso proporciona toda la información referida a la aplicación que está en uso. Contiene valores y opciones de toda la aplicación y, aunque en ocasiones puede ser necesario declarar este objeto explícitamente, lo normal es no tener que hacerlo. (11)

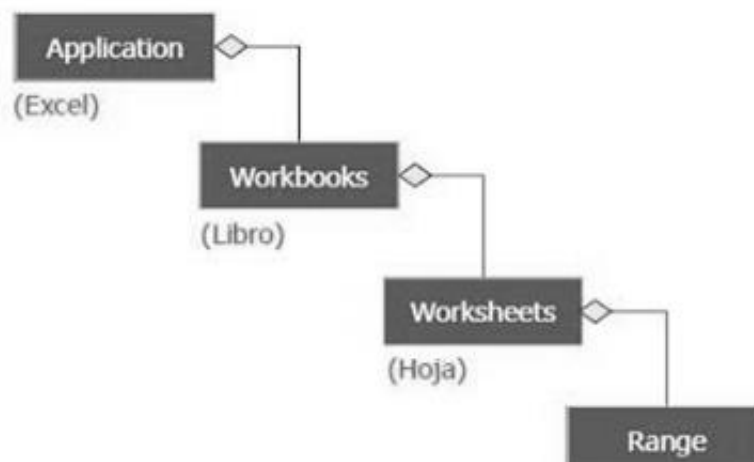


Figure 2-7. Jerarquía de objetos

2.7.3 Variables en un Proyecto

Una variable es un espacio en memoria donde almacenamos valores las cuales irán cambiando su valor conforme se ejecute una macro. Las variables se identifican por un **Nombre** que permite hacer referencia al valor que contienen y un **Tipo** que determina la naturaleza de los datos que pueden almacenar. (12)

Tabla 2-7. Tipos de datos incorporados en VBA

Tipo de Dato	Bytes	Rango de Valores
Boolean	2 bytes	Verdadero o falso
Integer	2 bytes	32,768 a 32,767
Long	4 bytes	2,147,483,648 a 2,147,483,647
Single (Decimal simple)	4 bytes	Números reales de 32 bits y 7 decimales. 3.402823E-38 a 1.401298E-45 (para valores negativos); 1.401298E-45 a 3.402823E-38 (valores positivos)
Double (decimal doble)	8 bytes	Números de 64 bits y 16 decimales. 1.79769313486232E-308 a 4.94065645841247E-324 (valores negativos); 4.94065645841247E-324 a 1.79769313486232E308 (para valores positivos)
String (variable length)	10 bytes + 1 byte por cada carácter	0 a aproximadamente 2 billones de caracteres
String (longitud fija)	Tamaño de la cadena	1 a 65,535 caracteres
Variant (with numbers)	16 bytes	Cualquier valor numérico hasta el rango de un tipo de datos doble. También puede contener valores especiales, como Empty, Error, Nothing y Null.

2.7.4 Estructuras y Funciones más Utilizados

2.7.4.1 Estructura For_Next

Permite ejecutar una instrucción una serie de veces mientras la variable toma los valores definidos dentro de un intervalo, es decir, es una estructura de bucle. Después, continúa con la siguiente instrucción hasta terminar el número de ciclos.

Sintaxis

- *For* “variable” = valor inicial To valor final
- *Steps*: Aquí le indicamos las tareas en las que se encuentra implícita la variable
- *Next*: Fin del bucle con retorno al siguiente valor de la variable.

2.7.4.2 Estructura End_If

Sintaxis

- *Condition If*: Es la condición que evaluará, podemos poner caracteres de comparación como “=”, “<”, “>”, “<>”.
- *Steps*: Aquí le podemos indicar que, si se cumple alguna condición especial, podemos ejecutar la instrucción deseada.
- *End If*: Punto final del recorrido sin retorno

2.7.4.3 Input Box Function

Podemos usar la función Input Box en Excel VBA para coaccionar al usuario de ingresar un valor ya sea del tipo texto, numérico, etc.

Sintaxis

“Application.InputBox(*Prompt, Title, Default, Left, Top, HelpFile, HelpContextID, Type*)”

- *Prompt*: Es el texto que se mostrará en el pop-up. Es el único valor obligatorio
- *Title*: Es el título del pop-up que aparecerá en la parte superior.
- *Default*: Es el valor que mostrará por defecto el Inputbox.
- *Type*: Es el tipo de valor que podemos usar que puede ser de tipo numérico (1), texto (2), verdadero/falso (4), referencia a una celda (8), una matriz (64), etc.

2.7.4.4 **Format Function**

Es una función que toma una expresión, una fecha, símbolos de moneda o un valor numérico y lo convierte a una variable del tipo String de acuerdo al formato especificado, en pocas palabras es una función que nos permite asignar diferentes tipos de formato ya sea a celdas o controles del tipo "TextBox". Se puede usar el comando "Msgbox" después de cada expresión que se quiera evaluar.

Sintaxis

Application.WorksheetFunction.Format (*expression, format*)

- *expression*: Es el valor de referencia el cual se verá afectado por el tipo de formato
- *format*: Es el tipo de formato a elegir el cual tiene que ir entre comillas.

2.7.4.5 **MsgBox**

La función MsgBox o también llamada variable de retorno permite devolver una cadena de valores ya sean del tipo numéricos o textos dentro de una ventana emergente, esto facilita mucho si queremos corroborar un resultado. El primer texto corresponde al mensaje a utilizar, luego se encuentra una constante que permite identificar que botón se mostrará, y como tercer dato tenemos el título a ser empleado.

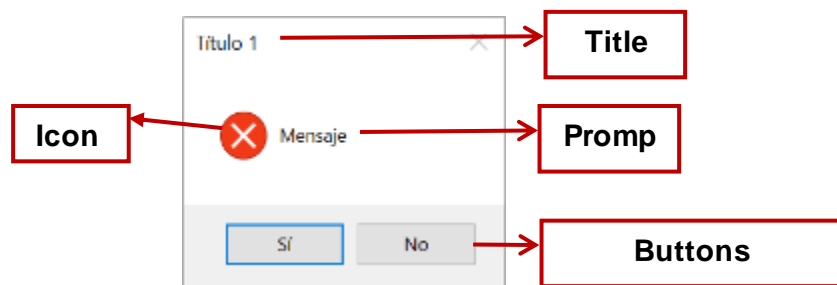


Figure 2-8. Características de la ventana MsgBox

Cabe precisar que no es necesario completar todos los requerimientos de campo de una función solo las más resaltantes.

Sintaxis

`MsgBox (Prompt, Buttons as VbMsgBoxStyle = vbOkOnly, Title, HelpFile, Context) As VbMsgBoxResult.`

- *Prompt*: Corresponde al texto o descripción a mostrarse en el MsgBox, se puede utilizar hasta 1024 caracteres, aunque es poco probable que se llegue a necesitar tal cantidad.
- *Buttons*: El segundo dato corresponde al botón a ser mostrado, en la tabla 2.15 se encuentran los valores que pueden ser usados.
- *Title*: Es el título que aparece en la parte superior izquierda del MsgBox
- *Context*: es un número que permite identificar un tema dentro del archivo de ayuda, al igual que el cuarto parámetro del MsgBox tampoco es usado con regularidad.

Tabla 2-8. Constante más utilizados dentro de la ventana MsgBox

Constante	Valor	Función
vbOKOnly	0	Botón Ok
vbOKCancel	1	Botón Ok y Cancel
vbYesNoCancel	3	Botones Yes, No y Cancel
vbYesNo	4	Botones Yes, No
vbQuestion	32	Símbolo de interrogación
vbExclamation	38	Símbolo de exclamación
vbInformation	64	Símbolo de información

2.8 Interacción entre Visual Basic y AutoCAD

2.8.1 Modelo de Objetos de AutoCAD

Según AUTODESK en su guía “*AutoCad 2007: ActiveX and VBA Developer's Guide*”. Los objetos constituyen el bloque constructor principal de la interfaz de ActiveX de AutoCAD. Cada objeto expuesto representa un componente concreto de AutoCAD. Existen multitud de tipos de objetos diferentes en la interfaz de ActiveX de AutoCAD. Por ejemplo:

- Los elementos gráficos como las líneas, los arcos, el texto y las cotas son objetos.
- Los parámetros de estilo como el tipo de línea y el estilo de cota son objetos.
- Las estructuras de organización como las capas, los grupos y los bloques son objetos.
- Se consideran objetos incluso el dibujo y la aplicación AutoCAD.

Los objetos se estructuran de forma jerárquica, siendo la raíz el objeto Application. A la presentación de esta estructura jerárquica se le denomina Modelo de objetos. El Modelo de objetos permite ver el objeto que proporciona acceso al siguiente nivel de objetos. (13)

2.8.2 Objeto Application

El objeto Application es la raíz del Modelo de objetos de ActiveX Automation de AutoCAD. Desde él puede obtener acceso a cualquiera de los demás objetos o a los métodos y propiedades que tengan asignados. (13)

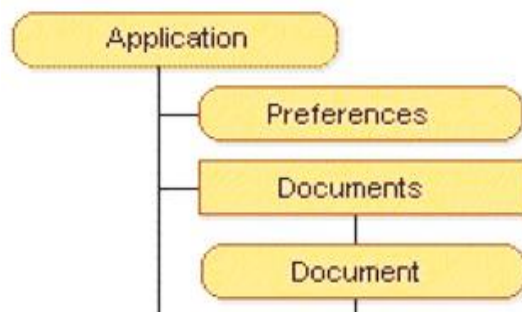


Figure 2-9. Objeto Application

2.8.3 Objeto Document

El objeto Document, que en realidad es un dibujo de AutoCAD, se encuentra en la colección Documents y proporciona acceso a todos los objetos de AutoCAD gráficos y a la mayoría de los que no son gráficos. El acceso a los objetos gráficos (líneas, círculos, arcos, etc.) se realiza a través de las colecciones ModelSpace y PaperSpace. El objeto Document también proporciona acceso a los objetos Plot y Utility. (13)

2.8.4 Objetos Gráficos y no Gráficos

Los objetos gráficos, también conocidos como entidades, son los objetos visibles (líneas, círculos, imágenes ráster, etc.) que componen un dibujo. Para crearlos, se utiliza el método Add seguido de la entidad apropiada. Los objetos gráficos tienen métodos que permiten que una aplicación ejecute la mayoría de los comandos de edición de AutoCAD como Copiar, Borrar, Desplazar, Simetría, etc. Los objetos gráficos tienen propiedades básicas como Layer, Linetype, Color y Handle. Los objetos no gráficos son los objetos invisibles (informativos) que forman parte de un dibujo, como Layers, Linetypes, DimStyles, SelectionSets, etc. (13)

2.8.5 ActiveX/VBA de AutoCAD

Existen tres elementos fundamentales que definen la programación ActiveX y VBA en AutoCAD. El primero es el propio AutoCAD, que incluye un amplio conjunto de objetos que engloba entidades, datos y comandos de AutoCAD.

El segundo elemento es la interfaz ActiveX Automation de AutoCAD, que establece mensajes (comunicación) con los objetos de AutoCAD. La programación en VBA requiere un conocimiento fundamental de ActiveX Automation.

El tercer elemento es el entorno de programación VBA, que cuenta con su propio conjunto de objetos, palabras clave, constantes, etc., los cuales facilitan el flujo de los programas y su control, depuración y ejecución. (13)

2.8.6 Proyectos VBA de AutoCAD

Un proyecto VBA de AutoCAD consiste en un conjunto de módulos de código, módulos de clase y formularios que se combinan para realizar una función determinada. Los proyectos pueden almacenarse en un dibujo de AutoCAD o en un archivo independiente.

Puede utilizar el Administrador de VBA para ver todos los proyectos VBA que estén cargados en la sesión actual de AutoCAD. El Administrador de VBA es una herramienta de AutoCAD que permite cargar, descargar, guardar, crear, incrustar y extraer proyectos VBA. (13)

2.8.7 Entidades

Para hacer un uso eficaz de ActiveX Automation de AutoCAD, debemos estar familiarizados con los objetos, entidades y funciones de AutoCAD relativos al tipo de aplicación que esté desarrollando. Cuanto más conocimiento se tenga de las propiedades gráficas y no gráficas de un objeto, más fácil será manipular los objetos en ActiveX Automation de AutoCAD. (13)

2.8.7.1 *Line Object*

Sintaxis: *AddLine (StartPoint, EndPoint) As AcadLine*

- *AddLine*: Es el método que representa las colecciones Model Space.
- *StartPoint*: Parámetro inicial que representa al array en el eje X del tipo Double.
- *EndPoint*: Parámetro final que representa al array en el eje Y del tipo Double.

2.8.7.2 *Circle Object*

Sintaxis: *AddCircle (Center, Radius as Double) As AcadCircle*

- *AddCircle*: Es el método que representa las colecciones Model Space.
- *Center*: Parámetro inicial que representa al array en los tres ejes del tipo Double
- *Radius*: Es el radio del círculo que queremos crear

2.8.8 Conceptos Básicos Sobre Propiedades y Métodos

Cada objeto lleva asociados sus propios métodos y propiedades. Las propiedades describen aspectos individuales del objeto; los métodos son acciones que pueden realizarse con el objeto concreto. Una vez creado el objeto, lo puede consultar y modificar a través de sus propiedades y métodos.

Por ejemplo, un objeto Circle tiene la propiedad Center. Esta propiedad representa las coordenadas 3D del SCU en el centro del círculo. Para cambiar el centro del círculo, solo tiene que definir esta propiedad con unas nuevas coordenadas. Para ver una lista de todos los métodos y propiedades del objeto Circle y en general, véase el objeto Circle en http://entercad.ru/acad_aag.en/.

2.8.9 Referencia a Objetos

Se puede hacer referencia a los objetos directamente o a través de una variable definida por el usuario. Para utilizar una referencia directa a un objeto, inclúyalo en la jerarquía de la llamada. Por ejemplo, la siguiente instrucción añade una línea en el espacio modelo. Observe que la jerarquía comienza con ThisDrawing, va al objeto ModelSpace y procede a llamar al método *AddLine*:

```
Dim startPoint(0 To 2) As Double, endPoint(0 To 2) As Double
Dim LineObj as AcadLine
startPoint(0) = 0: startPoint(1) = 0: startPoint(2) = 0
endPoint(0) = 30: endPoint(1) = 20: endPoint(2) = 0
Set LineObj = ThisDrawing.ModelSpace.AddLine(startPoint,endPoint)
```

Figure 2-10. Ejemplo de Referencia a Objetos 1

Para hacer referencia a los objetos mediante una variable definida por el usuario, defina la variable con el tipo deseado y, a continuación, establézcala como el objeto adecuado. Por ejemplo, el código siguiente se define una variable “moSpace” de tipo AcadModelSpace y la establece como igual al espacio modelo actual (13)

```
Dim moSpace As AcadModelSpace
Set moSpace = ThisDrawing.ModelSpace
Dim startPoint(0 To 2) As Double, endPoint(0 To 2) As Double
Dim LineObj as AcadLine
startPoint(0) = 0: startPoint(1) = 0: startPoint(2) = 0
endPoint(0) = 30: endPoint(1) = 20: endPoint(2) = 0
Set LineObj = moSpace.AddLine(startPoint,endPoint)
```

Figure 2-11. Ejemplo de referencia a objetos 2

2.9 Definición de Términos

Macizo rocoso

Es el conjunto de los bloques de la matriz rocosa y de las discontinuidades de diversos tipos que afectan al medio rocoso. Mecánicamente son medios discontinuos, anisotrópicos y heterogéneos. Prácticamente presentan una resistencia a la tracción nula. (14)

Matriz rocosa

También llamado roca matriz o roca intacta, es el material rocoso no afectado por las discontinuidades, o los bloques de roca intacta que quedan entre las mismas, se caracterizan principalmente por su densidad, deformabilidad y resistencia. (15)

Mecánica de rocas

La Mecánica de Rocas o de las Rocas puede considerarse como aquella parte de la Geotecnia que abarca todos los estudios teóricos y experimentales destinados a conocer el comportamiento mecánico e hidromecánico de las rocas, al ser sometidas a cambios en sus estados tensionales y en sus condiciones hidráulicas. (16)

Objeto

Un objeto es un elemento provisto de propiedades (datos) y funciones (métodos) que se relaciona con otros objetos de una manera concreta. Todas las acciones o instrucciones que contiene el programa están incluidas en un objeto. (11)

Abstracción

Es el proceso de reflexión o conceptualización en donde se busca la mejor manera de integrar los objetos esto con el objetivo de conseguir una perfecta sincronización de las variables y objetos.

Desarrollo

En términos de programación hace referencia al tipo de paradigma usado o que se pretende utilizar para la representación de un determinado sistema. Es una acción o proceso realizada por el usuario para transformar modelos matemáticos a esquemas de programación más tangibles y entendibles con un procesamiento de datos más rápido y dinámico.

CAPITULO III

METODOLOGIA

3.1 Métodos, y Alcance de la Investigación

Según el Dr. SCHWARZ en su guía *“Guía de referencia para la elaboración de una investigación aplicada”* la línea de investigación conlleva a considerar el método, tipo, nivel y diseño para el desarrollo de la misma. (17)

3.1.1 Método de Investigación

El método empleado en la presente investigación corresponde al método **Inductivo** puesto que parte de las características intrínsecas de casos particulares para luego ser utilizados en casos generales es así que se utiliza experiencias propias o lecciones aprendidas para poder comprender o mejorar una idea general.

3.1.2 Tipo de Investigación

Según el objetivo de esta, la presente tesis, corresponde a una investigación del tipo **Aplicada**; puesto que se centra en resolver un determinado problema utilizando la investigación básica.

Según el tipo de datos a emplear correspondería a un enfoque **Cuantitativo** en el sentido de que se parte de una recolección de datos confiable, válida y objetiva y **Cualitativo** en el sentido de que el reporte de resultados de la presente investigación se da de forma narrativa con elementos gráficos.

3.1.3 Nivel de Investigación

También conocido como alcance de investigación, en este caso particular corresponde a una investigación de nivel **Aplicativo** puesto que para el Dr. SCHWARZ “El nivel Aplicativo es el nivel más elevado y no solo requiere alcanzar los niveles previos, sino que también requiere desarrollar una aplicación de la solución al problema de investigación. Este nivel es aplicado principalmente a maestrías y doctorados”. (17)

3.2 Diseño de Investigación

Para el presente caso se emplea un diseño **No Experimental** en donde HERNÁNDEZ y otros señalan lo siguiente.

Podría definirse como la investigación que se realiza sin manipular deliberadamente variables. Es decir, se trata de estudios donde no hacemos variar en forma intencional las variables independientes para ver su efecto sobre otras variables. Lo que se hace en la investigación no experimental es observar fenómenos tal como se dan en su contexto natural, para posteriormente analizarlos. (18)

3.3 Población

La población está representada por la aplicación del modelo en los diferentes frentes de desarrollo de perforación y voladura.

3.4 Muestra

La muestra está representada por la galería 710 SE del prospecto minero Monserrat, considerando un muestreo no probabilístico.

3.5 Técnica para la Recolección de datos

La técnica que se utilizó para la recolección de datos fue la observación.

3.6 Validación de la Hipótesis

La validación de la hipótesis se logrará mediante un análisis cualitativo obteniendo un diseño real a escala en formato DWG sobre la base del modelo Matemático Holmberg incluyendo un resumen general de los resultados.

3.7 Instrumentos para la recolección de datos

- 01 brújula marca Brunton.
- 01 esclerómetro marca Proeti.
- 01 flexómetro milimétrico marca Tools.
- Formatos de registro.
- Ábacos y tablas (Anexo4).
- Libretas de campo.
- Instrumentos y equipos de laboratorio descritos en el Anexo 1

3.8 Evaluación Geomecánica

3.8.1 Observaciones Geológicas

La metodología para el análisis observacional se sustenta en el mapeo por línea de detalle el cual consiste en trazar una línea recta en las zonas con mayor evidencia de discontinuidades por lo general se toma entre 1 y 2 metros. Los resultados se muestran en la ficha de mapeo geomecánico del Anexo 2.

3.8.2 Determinación del Peso Específico

Para determinar el peso específico promedio de la roca se acondicionó el ambiente de trabajo con los equipos apropiados de corte, compresión y medición con el fin de obtener valores confiables sobre la base de los ensayos.

Primeramente, se hizo un lavado de las muestras para facilitar la manipulación de estas, la muestra al tener una forma irregular no facilita su posición dentro de los equipos de compresión y medición por lo que fue imperativo hacer cortes en diferentes lados obteniendo así muestras mucho más uniformes (Anexo1).



Figure 3-1. Proceso de corte y lavado

Una vez que la muestra haya sido cortada se procedió a realizar el ensayo de extracción utilizando el equipo para extracción de testigos (Anexo 1), obteniendo en total tres muestras de testigo. Posteriormente se hizo uso de las herramientas de medición para hallar sus principales propiedades físicas y mecánicas. Estos valores tienen una validez del 99% ya que como se muestra en el Anexo 3 son valores que se tomaron en tiempo real y con la supervisión del encargado de laboratorio.

Tabla 3-1. Diámetro, Longitud, Masa, Volumen y Densidad de cada espécimen.

Specimen	Diameter (mm)	Length (mm)	Weight (g)	Volume (m3)	Density (t/m3)
1	42	43	161.49	5.9574E-05	2.71
2	30	58	109.01	4.09978E-05	2.66
3	41	45	164.9	5.94114E-05	2.78

Con las mediciones hechas en cada testigo, la densidad promedio obtenida fue de 2.7 T/m³ el cual nos indica que el tipo de roca asociado a este valor es una Diorita en referencia a la imagen de pesos unitarios típicos y validada en el plano subterráneo del Anexo 5.

Roca	γ (ton/m ³)	Roca	γ (ton/m ³)
Andesitas	2.5 a 2.8	Anfibolitas	2.7 a 3.1
Areniscas	2.0 a 2.8	Basaltos	2.8 a 3.0
Brechas Igneas	2.7 a 2.9	Calizas	1.8 a 2.9
Carbón	0.7 a 2.0	Creta	1.9 a 2.3
Cuarcitas	2.5 a 2.9	Dacitas	2.5 a 2.8
Diabasas	2.6 a 3.0	Dioritas	2.7 a 3.0
Dolomitas	2.7 a 2.9	Esquistos	2.0 a 2.7
Gabros	2.8 a 3.1	Gneiss	2.6 a 2.9
Granitos	2.5 a 2.9	Granodioritas	2.6 a 2.9
Gravas Cementadas	1.9 a 2.1	Hematitas	4.5 a 5.3
Ignimbrita Cinerítica	1.4 a 1.8	Ignimbrita Cristalina	2.0 a 2.3
Lutitas	2.0 a 2.8	Mármol	2.1 a 2.9
Mica-Esquistos	2.5 a 2.9	Mineral de Hierro	4.0 a 5.5
Pizarras	2.5 a 2.8	Pórfidos	2.4 a 2.8
Riolitas	2.2 a 2.7	Sal	2.0 a 2.0
Tobas	1.5 a 2.0	Yeso	2.2 a 2.4

Figure 3-2. Pesos Unitarios Típicos de algunas Rocas (19)

3.8.3 Cálculo de la Tensión de Rotura

Para determinar el esfuerzo máximo de una probeta se hizo uso de la prensa de compresión el cual fue facilitado por los miembros del laboratorio de mecánica de suelos. Para este ensayo se utilizó el testigo número 3 del Anexo2 teniendo en cuenta las propiedades físicas de la misma, la metodología corresponde al ensayo de compresión uniaxial o UCS el cual consiste en aplicar una fuerza constante axial sin confinamiento sobre la muestra en estudio , la imagen siguiente muestra el resultado post rotura y se demuestra la validez de los datos obtenidos.



Figure 3-3. Ensayo tipo UCS sobre el testigo número 3

El resultado del ensayo genera valores de resistencia proporcionales a la deformación unitaria, estos valores son apreciables según va aumentando la carga axial, en el siguiente gráfico se muestra el comportamiento de la deformación elástica con lecturas iniciales de desplazamiento expresadas en pulgadas con una carga inicial de 5000 kg llegando hasta los 6210 kg (ver tabla 3.2).

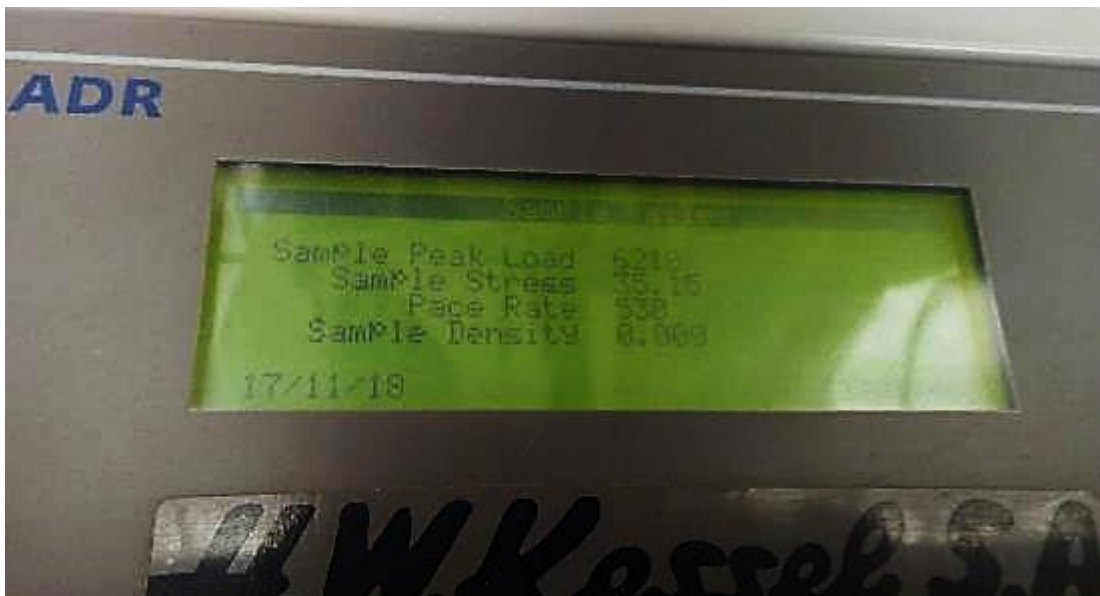


Figure 3-4. Datos del equipo de compresión en tiempo real

Tabla 3-2. Resultados de deformación y tensión sobre el testigo número 3

Lectura Dial (in)	Lectura Carga (Kg)	Deformación unitaria	Factor de corrección	Área corregida (m ²)	Tensión de rotura σ_1 (Kg/m ²)	Tensión de rotura σ_1 (MPa)
0	5000	0.000000	1	0.00132	3787149.15	37.152
0.001	5100	0.000564	0.999436	0.00132	3860711.75	37.874
0.002	5200	0.001129	0.998871	0.00132	3934188.84	38.594
0.003	5300	0.001693	0.998307	0.00132	4007580.42	39.314
0.004	5400	0.002258	0.997742	0.00132	4080886.50	40.033
0.005	5500	0.002822	0.997177778	0.00132	4154107.07	40.752
0.0075	5600	0.004233	0.995766667	0.00133	4223650.91	41.434
0.01	5700	0.005644	0.994355556	0.00133	4292980.99	42.114
0.0125	5800	0.007056	0.992944444	0.00133	4362097.30	42.792
0.015	5900	0.008467	0.991533333	0.00133	4430999.85	43.468
0.02	6000	0.011289	0.988711111	0.00134	4493275.73	44.079
0.025	6100	0.014111	0.985888889	0.00134	4555124.09	44.686
0.03	6200	0.016933333	0.983066667	0.001342996	4616544.91	45.288
0.035	6210	0.019755556	0.980244444	0.001346862	4610716.24	45.231

Se puede apreciar el resultado obtenido en laboratorio consiguiendo una tensión de rotura igual a 4,610,716.24 Kg/m² o 45.231 Mpa.

3.9 Clasificación Geomecánica del Macizo Rocoso

Con los parámetros de resistencia, densidad y caracterización de discontinuidades se procedió a reemplazar los valores dentro de la tabla de funciones del RMR89 (Anexo4). Como resultado final de todos los parámetros calificados se tiene una valoración total de 54 lo cual indica que estamos frente a un macizo rocoso regular de clase III.

3.10 Diseño de Programación

3.10.1 Consolidación de la Base de Datos

Para que el formato funcione correctamente es necesario contar con una base de datos sólida en ella debemos incluir criterios tales como: Explosivos, Tipos de Arranque, Equipos y Brocas a utilizar.

Es importante resaltar que el usuario no está sujeto al mismo tipo de información que se maneja en este proyecto ni mucho menos a los estilos de formato. El usuario está en la libertad de generar su propia base de datos con el estilo que más le satisfaga.

3.10.2 Creación de Carpetas y Sub Carpetas

Primeramente, debemos establecer la carpeta principal o matriz que contendrá las sub carpetas, en este caso se asignó el nombre de “Tesis”, esta será la fuente principal del formato “.xslm”. Posteriormente se crearon dos subcarpetas con el nombre de “CalculosFormatoVBA” y “AutocadMalla”.

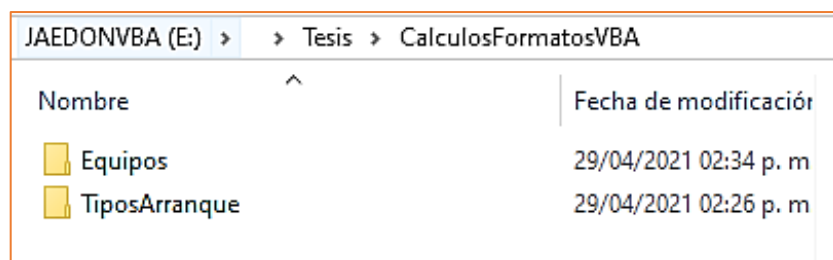


Figure 3-5. Ruta de acceso de carpetas y sub carpetas

Dentro de la carpeta “CalculosFormatosVBA” se crearon otras dos subcarpetas de nombres : “Equipos” y “TiposArranque” las cuales mantendrán la colección de imágenes. Finalmente, la carpeta “AutocadMalla” guardará toda la información correspondiente a los diseños generados en AutoCAD con la extensión DWG.

3.10.3 Apertura de un Nuevo Archivo

Se generó un nuevo archivo Excel el cual contendrá la base de datos y el esquema de desarrollo del presente proyecto. Cuando se instala MO ya viene por defecto VBA sin

embargo al abrir cualquiera de las aplicaciones no encontraremos en muchos casos activado la pestaña programador.

Abrimos un nuevo libro Excel y verificamos que este habilitado el “Programador” en la cinta de opciones, de no estar habilitado nos vamos al menú **Archivo-Opciones- Personalizar la cinta de Opciones** y activamos el **Programador** que se encuentra dentro de la lista **Pestañas principales**.

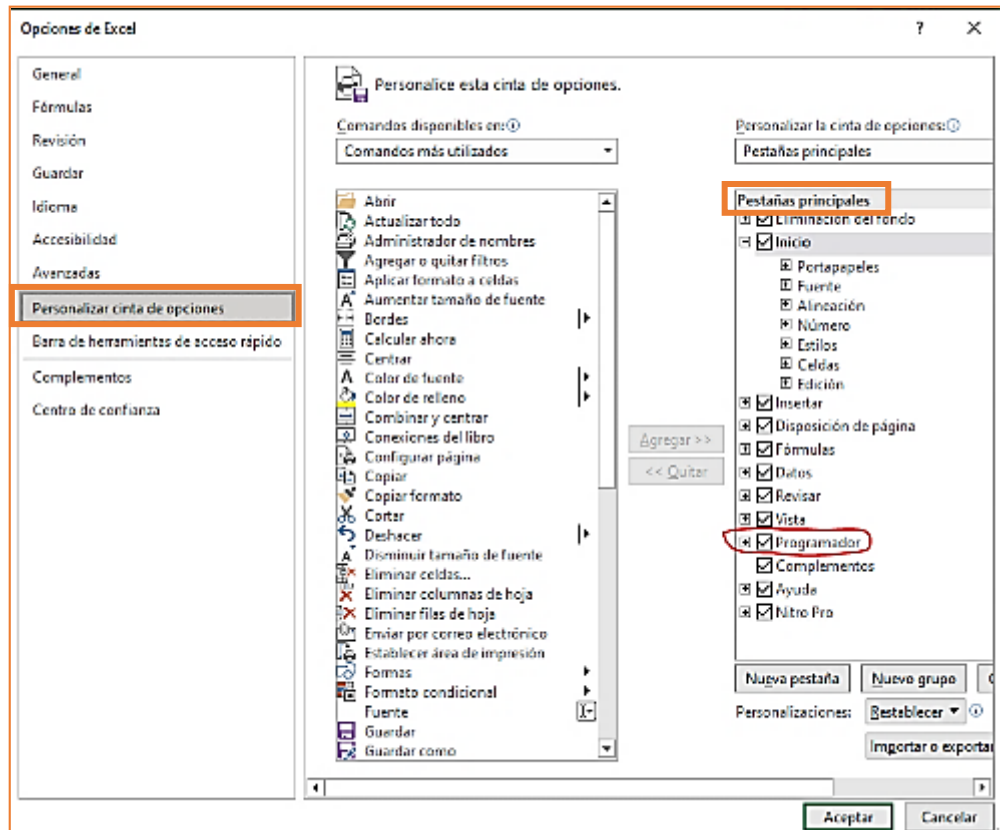


Figure 3-6. Protocolo de activación de la pestaña programador

3.10.4 Primera Base de Datos (Explosivos)

Se consideró únicamente las propiedades más importantes de los explosivos las cuales son: Diámetro, Longitud, Volumen, Densidad, Peso y la Potencia Relativa en Peso.

Existen muchos explosivos en el mercado entre Altos Explosivos y Agentes de Voladura lo cual nos obliga a enfocarnos únicamente en los explosivos más utilizados, en este proyecto se trabajó con explosivos de la marca FAMESA Y EXSA.

Tabla 3-3. Base de datos del tipo de explosivo

EXPLOSIVO	Φ mm	Long. in	Long. m	Vol. (m ³)	Densidad (g/cc)	Peso (Kg)	S (Anfo)
EMULNOR 3000 1" x 7"	25.4	7	0.178	9.0E-05	1.14	0.1027	0.81
EMULNOR 3000 1" x 8"	25.4	8	0.203	1.0E-04	1.14	0.1174	0.81
EMULNOR 3000 1" x 12"	25.4	12	0.305	1.5E-04	1.14	0.1761	0.81
EMULNOR 3000 1 1/8" x 16"	28.575	16	0.406	2.6E-04	1.14	0.2971	0.81
EMULNOR 5000 1" x 7"	25.4	7	0.178	9.0E-05	1.16	0.1045	0.88
EMULNOR 5000 1" x 8"	25.4	8	0.203	1.0E-04	1.16	0.1194	0.88
EMULNOR 5000 1" x 12"	25.4	12	0.305	1.5E-04	1.16	0.1792	0.88
EMULNOR 5000 1 1/8" x 16"	28.575	16	0.406	2.6E-04	1.16	0.3023	0.88
SEMEXSA 65 1 1/2" x 12"	38.1	12	0.305	3.5E-04	1.12	0.3892	0.77
SEMEXSA 65 7/8" x 7"	22.225	7	0.178	6.9E-05	1.12	0.0773	0.77
SEMEXSA 80 1 1/8" x 8"	28.575	8	0.203	1.3E-04	1.18	0.1538	0.82

3.10.5 Segunda Base de Datos (Equipos)

Para este caso se deberá contar con la ruta de la imagen correspondiente a cada tipo de equipo, el usuario es libre de asignarle el nombre que desea siempre y cuando se mantenga la fuente de la imagen correctamente escrita.

Tabla 3-4 Base de datos del tipo de Equipo

Equipo	Talad Alivio	Diámetro Mínimo	Imagen Ruta
Jumbo	102	45	E:\ \Tesis\CalculosFormatos\VBA\Equipos\Equipo3.jp
Jumbo H781	102	43	E:\ \Tesis\CalculosFormatos\VBA\Equipos\Equipo3.jp
Jumbo	100	45	E:\ \Tesis\CalculosFormatos\VBA\Equipos\Equipo3.jp
SIMBA1250	89	51	E:\ \Tesis\CalculosFormatos\VBA\Equipos\Equipo3.jp
Jumbo	70	38	E:\ \Tesis\CalculosFormatos\VBA\Equipos\Equipo3.jp
Boomer E3C	64	43	E:\ \Tesis\CalculosFormatos\VBA\Equipos\Jack Leg.jp
Boomer E1	64	38	E:\ \Tesis\CalculosFormatos\VBA\Equipos\Equipo3.jp

3.10.6 Tercera Base de Datos (Tipos de Arranque)

Existen muchos tipos de arranque según la necesidad del proyecto, para este caso en concreto se trabajó con un arranque de un solo taladro de alivio de cuatro secciones, existiendo a su vez arranques de 2,3, 4 y 5 taladros de alivio.

Tabla 3-5. Base de datos sobre los tipos de arranque o cuele

Nº Talad Alivio	First Quadrangle	Second Quadrangle	Third Quadrangle	Fourth Quadrangle
1	4	4	4	4
2	6	4	4	4
2	6	4	4	4
3	4	4	4	4
4	5	4	4	4
5	4	4	4	4

3.11 Programación Orientada a Objetos

3.11.1 Preparación de un Nuevo Proyecto VBA

Para aperturarnos en el editor de Visual Basic basta con activar el **programador** de la cinta de opciones y dar click en el **botón Visual Basic** o presionando la combinación de teclas **Alt + F11**.

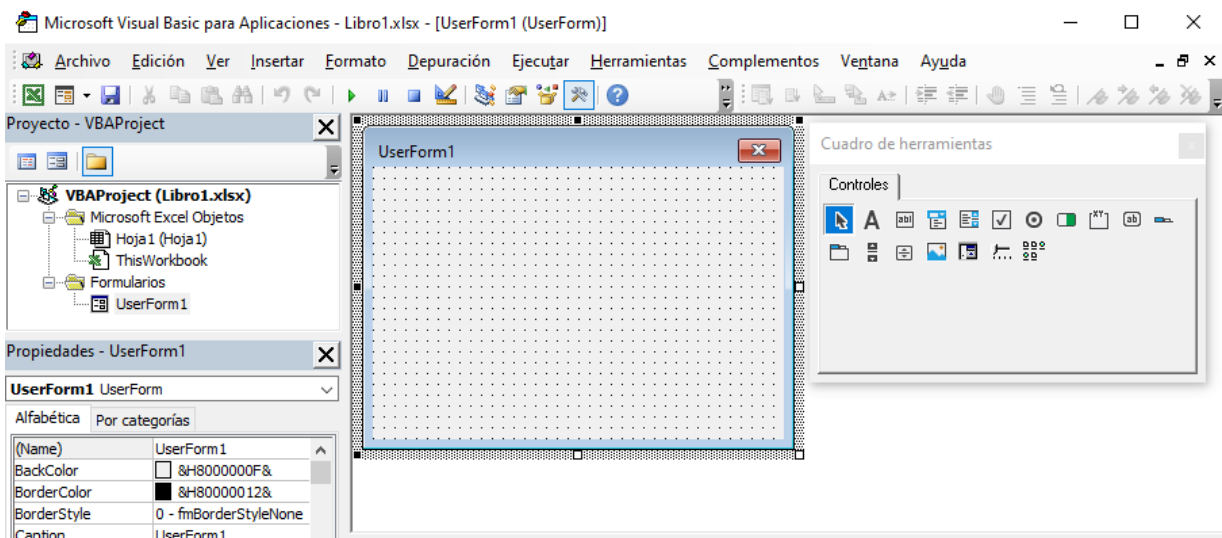


Figure 3-7. Esquema de edición VBA

Presionamos la tecla F4 para activar la ventana de propiedades del UserForm y nos aseguramos de tener activado el “Cuadro de Herramientas” guardando el archivo como libro de Excel habilitado para macros.

3.11.2 Proceso de Abstracción y Materialización - First UserForm

La estructura del proyecto está dividida en dos ventanas principales de nombres “frmParametro” y “frmCalculo”. El “frmParametro” como un objeto del tipo UserForm contendrá todas las variables de entrada referentes a la evaluación geomecánica, explosivos y diámetros de perforación mientras que el “frmCalculo” contendrá los parámetros resultantes del modelo matemático Holmberg.

Dicho esto, la estructura “frmParametro” está representado de la siguiente forma.

The screenshot shows a software window titled "Diseño de Malla" with a standard Windows interface. The window is divided into several sections for data entry. On the left, there are two columns of input fields under the heading "Parametros de Diseño". The first column includes fields for "Diametro Taladro Vaco", "Diametro taladro cargado", "Diametro taladro contorno", "Ancho Labor", "Altura Labor", "Profundidad Talad (H)", "Nombre Labor", "NR Registrar", "Angulo taladro concono", "Desviacion Angular", "Error de Emboquille", and "Error perforacion". The second column has units like "mm", "m", "rad", "m/m", and "m". In the center, there are two large empty rectangular boxes. On the right side, there are radio buttons for "SEMEXSA", "EMULNOR", and "MIXTO". Below these are three dropdown menus labeled "Contour D", "Stopping and Lifter", and "Major Cutting". Further down are fields for "Densidad" (g/cc), "Diametro" (in), "Peso" (Kg), and "S Anfo (RWS)". At the bottom left, there are fields for "Tipo Roca", "RMR", "RQD", "GSI", and "p Roca". At the bottom right, there are fields for "CE" and "C (Ranura)", and a "Calcular" button.

Figure 3-8 Ventana principal o de arranque

Dentro del cuadro de herramientas se utilizó el control Frame el cual permite segmentar mejor los parámetros. Se crearon cuatro Frames para cada tema en particular:

- Parámetros de Diseño
- Parámetros Geológicos, Geomecánicos
- Propiedades del Explosivo
- Consumo Específico y Constante de Roca

3.11.2.1 **Parámetros de Diseño**

Mediante el control Label se definieron los Ítems más importantes los cuales estarán relacionados a cada uno de los Text Boxes (TB), para este caso en particular se asignaron nombres anteponiendo la palabra “txt” y en algunos otros casos dando valores numéricos². También se agregaron controles del tipo ComboBox (CB) los cuales contendrán valores de diámetro, en total se tiene: 3 CB, 9 TB, 1 Imagen y 1 ListBox (LB).

Tabla 3-6. Nombre Interno y Externo por cada control del primer formulario

Control Caption	Control Name
Diámetro del Taladro Vacío	ComboBox4
Diámetro del Taladro Cargado	ComboBox5
Diámetro taladro contorno	ComboBox6
Ancho Labor	txtAncho
Altura Labor	txtAltura
Profundidad Talad (H)	txtH
Nombre Labor	txtNombre
Nº Register	txtRegister
Angulo taladro contorno	txtAngulo
Desviación Angular	TextBox9
Error de Emboquille	TextBox10
F	txtF
Imagen	Image1
ListBox	ListDisplay2

² Asignar valores numéricos de secuencia como nombre interno para cada control facilita mucho al momento de trabajar con bucles ya que permite sintetizar la rutina de programación.

Para cargar la información dentro del ListBox es necesario recapitular la segunda base de datos de la Tabla 3-4 ya que en base a esta información se seleccionará el equipo a utilizar junto con los diámetros disponibles.

Primeramente, se llenó los combos con los valores de tipo de broca esto se hizo desde la base propia del UserForm. Al hacer doble click sobre este; se nos abrirá la ventana que se muestra en la figura 3.9 en donde tendremos los diferentes eventos y controles por defecto.

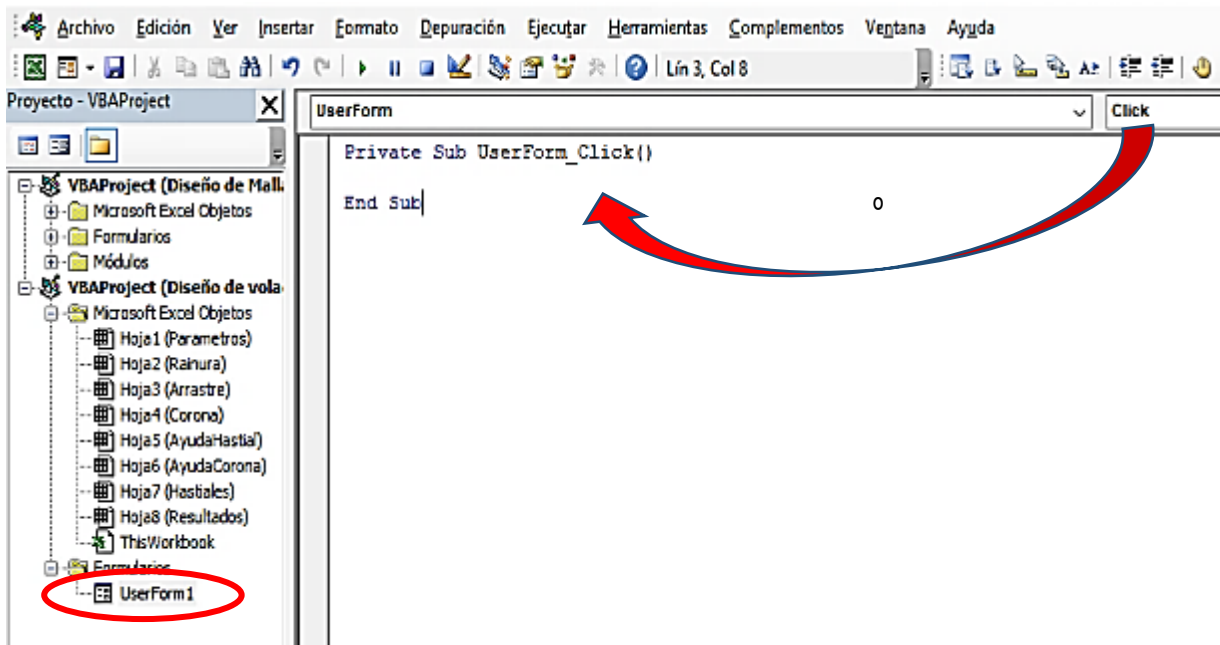
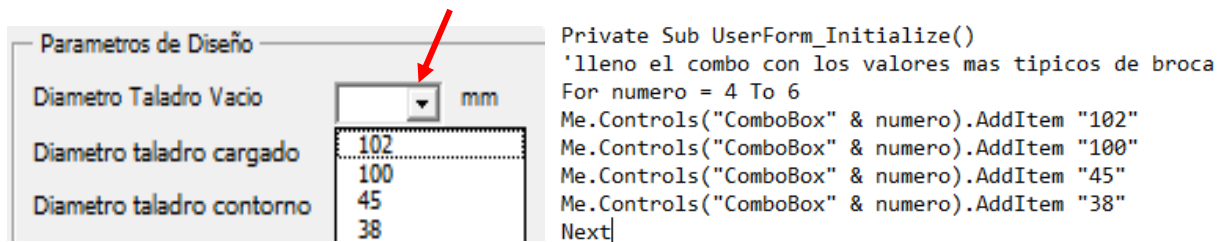


Figure 3-9. Acceso al esquema de procesos del UserForm

Una vez dentro se cambió el evento Click por el evento Initialize y se introdujo la rutina con las funcionalidades propias del CB y el LB. El siguiente código es una muestra de las muchas otras formas de cargar una información, si el usuario desea cambiar de método lo puede hacer siempre y cuando se mantenga el resultado.



Rutina 3-1. Ingreso de valores sobre los Combos 4, 5 y 6

Es importante seleccionar la hoja con la que vamos a trabajar, definir el número de columnas, incluir los encabezados y cargar la fuente de la información. Cada una de estas propiedades la podemos apreciar dentro de la ventana de propiedades, en pocas palabras lo podemos hacer directamente desde su ventana.

```
Me.ListDisplay2.ColumnCount = 4
Me.ListDisplay2.ColumnHeads = True
Me.ListDisplay2.RowSource = Sheet6.Range("D8:G14").Address
Me.ListDisplay2.ColumnWidths = "24 pt;24 pt;24 pt;24 pt"
```

Rutina 3-2. Atributos más relevantes del ListBox-First UserForm

Al momento de seleccionar un ítem dentro del ListBox este nos cargará tanto la imagen como los combos 4 y 5 actualizados. Para ello debemos aperturarnos dentro del control propiamente dicho y utilizar su fuente de eventos, métodos y propiedades para generar dicha acción. En la siguiente imagen se observa el algoritmo utilizado.

```
Private Sub ListDisplay2_Click() |
Dim cuenta As Long
Dim pathpicture As String
cuenta = Me.ListDisplay2.ListCount

For I = 0 To cuenta - 1
    If Me.ListDisplay2.Selected(I) Then

        For numero = 4 To 5
            Me.Controls("ComboBox" & numero).Text = Me.ListDisplay2.List(I, numero - 3)
        Next
        pathpicture = Me.ListDisplay2.List(I, 3)
        Me.Image1.Picture = LoadPicture(pathpicture)
    End If
Next
End Sub
```

Rutina 3-3. Acceso al control ListBox y uso del evento click para cargar y actualizar los Combos 4 y 5.

Como se puede apreciar se generaron dos variables; una variable "cuenta" del tipo Long y una variable "pathpicture" del tipo String. La variable "pathpicture" guardará la ruta de la imagen que voy a utilizar y la variable "cuenta" guardará el total de filas que se tendrá dentro del ListBox de esa manera al momento de seleccionar un ListBox Row la macro comenzara hacer un bucle con el objetivo de identificar esa selección y cargarla automáticamente en los controles "Image1" & "ComboBox4_5".

3.11.2.2 **Parámetros Geológicos, Geomecánicos**

Los controles en esta sección no contienen ningún código de programación el valor de estos se obtiene a partir del análisis de recolección de datos o mediante estimaciones análogas. El siguiente cuadro muestra los nombres asignados a cada uno de los controles correspondientes a este marco.

Tabla 3-7. Nombre externo e interno de los controles Geomecánicos del primer formulario

Caption Control	Name Control
Tipo de Roca	txtRoca
RQD	txtRQD
GSI	txtGSI
Densidad Roca	txtDensidadRoca
RMR	txtRMR
Calidad	txtCalidad

3.11.2.3 **Propiedades del Explosivo**

Las propiedades a tomar en cuenta son: Densidad, Diámetro, Peso y S(anfo). Estos valores se obtendrán según el tipo de explosivo a utilizar, se hizo una segmentación por tipo de explosivo asignando nombre de rangos dentro de la primera base de datos. Para asignarle nombre a los rangos se utilizó las funciones *DESREF*, *CONTARA*, *DIRECCION*, *INDICE E INDIRECTO*³.

³ Hay que tener cuidado con la versión de Excel que estamos usando, si nos encontramos trabajando con una versión en inglés lo correcto sería utilizar la función *OFFSET*, de esa manera no habrá ningún error dentro del administrador de nombres.

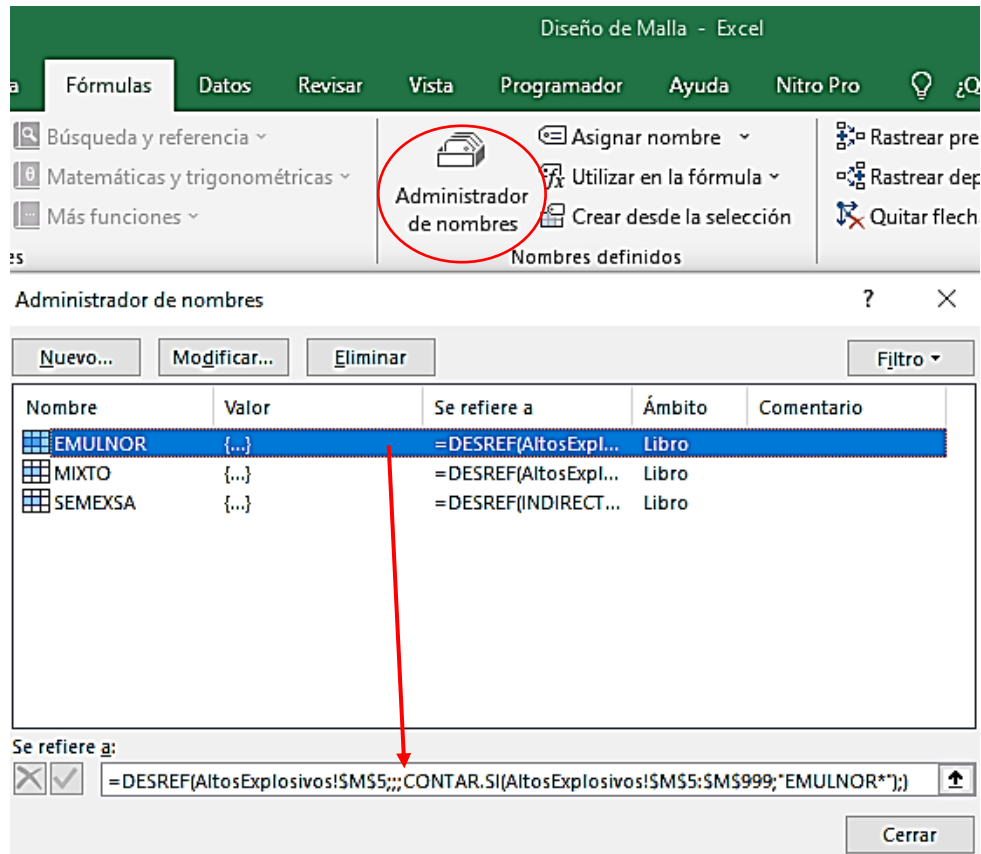


Figure 3-10. Definición de rangos dentro del administrador de nombres

- Emulnor=DESREF(AltosExplosivos!\$M\$5;;;CONTAR.SI(AltosExplosivos!\$M\$5:\$M\$999;"Emulnor*");)
- Mixto=DESREF(AltosExplosivos!\$M\$5;;;CONTARA(AltosExplosivos!\$M\$5:\$M\$999);)
- Semexsa=DESREF(INDIRECTO(DIRECCION(COINCIDIR("SEMEXSA 45*";AltosExplosivos!\$M\$5:\$M\$999;0);13));4;;;CONTAR.SI(AltosExplosivos!\$M\$5:\$M\$999;"SEMEXSA*");)

Después de asignarle nombre a los rangos se procedió a establecer el esquema para la obtención de las propiedades básicas del explosivo. Se insertaron tres controles del tipo Option Button, tres controles del tipo Combo Box y doce controles del tipo Text Box (ver Figure 3-8), quedando de la siguiente manera.

Tabla 3-8. Nombre externo e interno de los controles de voladura del primer formulario

Caption Control	Name Control
Semexsa	OptionButton1
Emulnor	OptionButton2
Mixto	OptionButton3
Contour D	ComboBox1
Stoping	ComboBox2
Major Cut	ComboBox3
Densidad	txt18
Diámetro	txt19
Peso	txt20
S Anfo	txt21
Densidad	txt22
Diámetro	txt23
Peso	txt24
S Anfo	txt25
Densidad	txt26
Diámetro	txt27
Peso	txt28
S Anfo	txt29

El procedimiento para establecer los valores dentro de cada Combo se resume en el siguiente flujograma, esta secuencia se aplica tanto para el Command Button uno, dos y tres; el resultado se reflejará según la opción que nosotros elijamos al momento de seleccionar un Option Button.

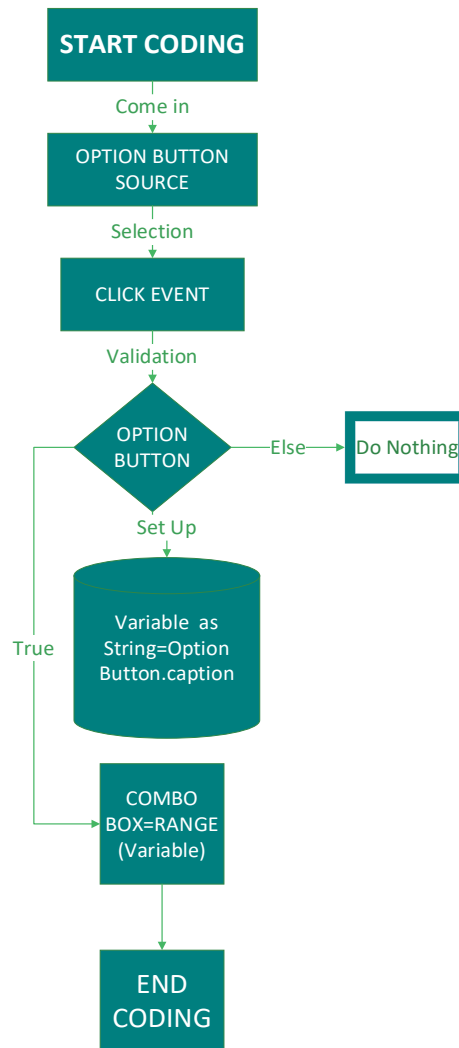


Figure 3-11. Flujograma de programación sobre el evento click del Option Button 1,2 y 3

Como se aprecia en el flujograma la variable que guarda el nombre del tipo de explosivo se limita a las clases (Emulnor, Semexsa, etc.) a diferencia de los Combos que pueden llegar a tener diversas categorías de acuerdo a los atributos del explosivo dependiendo de la cantidad de información que tengamos en nuestra primera base de datos, cuanto mayor sea la cantidad de explosivos a utilizar mayor será la lista.

En la siguiente imagen se muestra el código que representa al presente flujograma.

```
Private Sub OptionButton1_Click() 'SEMEXSA
If Me.Controls("OptionButton" & 1).Value = True Then
explosivo = Me.Controls("OptionButton" & 1).Caption
Me.Controls("ComboBox" & 1).RowSource = Sheet4.Range(explosivo).Address
Me.Controls("ComboBox" & 2).RowSource = Sheet4.Range(explosivo).Address
Me.Controls("ComboBox" & 3).RowSource = Sheet4.Range(explosivo).Address
End If
End Sub
```

Rutina 3-4. Instrucciones sobre el evento click del Option Button 1

A continuación, el explosivo a seleccionar según la opción marcada, debe estar condicionada a un nuevo evento, para ello debemos ingresar a la fuente del Combo Box 1, 2 y 3 y establecer el evento Change de esa manera cualquier cambio que se realice en el Combo dará como resultado la ejecución de una macro.

Se declararon las variables “founditem” y “rango”, ambas del tipo Range; seguido de una variable “explosivo” del tipo String. La variable “founditem” contendrá un rango en específico el cual resulta de una búsqueda usando el método Find; la variable “explosivo” contendrá el nombre externo del Option Button que hemos seleccionado y finalmente la variable “rango” contendrá únicamente el rango en general según el valor de la variable “explosivo”. Se estableció un bucle inicial el cual recorrerá cada uno de los Option Button, obteniendo el “Caption” del botón que este seleccionado (variable “rango”) esto con el fin de darle un valor a la variable “explosivo”, ya que la variable “filaemulnor” depende de esta. (R01).

```
Dim founditem As Range
Dim rango As Range
For numero = 1 To 3
  If Me.Controls("OptionButton" & numero).Value = True Then
    explosivo = Me.Controls("OptionButton" & numero).Caption
  End If
Next
```

(R01)

Rutina 3-5. Primer esquema de desarrollo sobre la fuente del Combo Box 1

Una vez definido las variables y haber seleccionado un tipo de explosivo, la variable “filaemulnor” contendrá la posición en la que se encuentra el contenido del Combo dentro de nuestra base de datos inicial (WorkSheet), de esa manera podremos cargar las propiedades que tenemos en esa hoja de ese explosivo en particular hacia los controles (Densidad, Diámetro, Peso y PRPanfo) (R02).

```
'EL NOMBRE RANGO ASIGNADO ES "EMULNOR"
Set founditem = Sheet4.Range(explosivo).Find(Me.ComboBox1.Value)
If Not founditem Is Nothing Then
    Filaemulnor = founditem.Row

    Me.txt18.Text = Sheet4.Range("Y" & Filaemulnor).Value
    Me.txt19.Text = Sheet4.Range("U" & Filaemulnor).Value
    Me.txt20.Text = Sheet4.Range("Z" & Filaemulnor).Value
    Me.txt21.Text = Sheet4.Range("AA" & Filaemulnor).Value
End If
```

(R02)

Rutina 3-6. Segundo esquema de desarrollo sobre la fuente del Combo Box 1

3.11.2.4 Consumo Especifico y Constante de Roca

Según el cuadro de fórmulas se procede a determinar los valores de GSI, CE y C los cuales tendrán incidencia dentro de los parámetros de Burden máximo y Burden práctico.

Tabla 3-9. Nombre interno y externo respecto a los parámetros CE y C

Control Caption	Control Name
Consumo Especifico	TxtCE
Constante de Roca	TxtC
GSI	TxtGSI

```
radianes = Application.WorksheetFunction.Radians((Val(Me.txtGSI.Value) + 15) / 2)
tangente = Tan(radianes)

Me.txtCE.Text = (0.56 * Me.txtDensidadRoca.Value * tangente) / _
Application.WorksheetFunction.Power(((115 - Val(Me.txtRQD.Value)) / 3.3), 1 / 3)
Me.txtC.Text = (0.8784 * Me.txtCE.Text) + 0.0052
```

Rutina 3-7. Rutina para el cálculo del GSI, CE y C

Una vez completado todos los parámetros de precálculo se agregó un Command Button el cual manda llamar al formulario “frmCalculo” utilizando el método Show.

3.11.3 Proceso de Abstracción y Materialización - Second UserForm

Debemos tener en cuenta el nombre de las divisiones de un frente ya que en base a estos términos se irá desarrollando las rutinas de programación.

El formulario “frmCalculo” contendrá todas las variables y parámetros necesarios para la cuantificación del modelo matemático Holmberg. La siguiente imagen es una representación de cómo está estructurado este formulario, dividiéndose en 7 páginas o pestañas.

The screenshot shows a Windows-style form titled "Division del Frente". At the top, there is a tabbed interface with seven tabs: CUT, CONTOUR D, LIFTER, STOPING B, STOPING C, WALLS, and RESULTADOS. Red arrows point to each of these tabs. The main area of the form is divided into several sections:

- Frame 1:** Contains input fields for "Eficiencia Voladura", "Nº Tald Alivio", "Diámetro Equivalente" (with a unit 'm'), and "Avance por Disparo" (with a unit 'm').
- Frame 2:** Contains input fields for "H Practico" (with a unit 'ft'), "Long Carga Fondo" (with a unit 'm'), "Taco" (with a unit 'm'), and "Columna" (with a unit 'm').
- H Teorico:** A single input field with a unit 'ft'.
- Frame 3:** A table with 8 columns: "B Max (m)", "B Practico (m)", "Ah (m)", "N Tladros", "q (Kg/m)", "Kg/Talad", "Kg", and "Cart/Talad". It has four rows labeled "Primer Cuadrante", "Segundo Cuadrante", "Tercer Cuadrante", and "Cuarto Cuadrante".
- Other elements:** A "SORT OF CUT" label, a large empty rectangular area, and two buttons at the bottom right: "Calcular" and "Close".

Figure 3-12. Formulario “frmCalculo” dividido en 7 pestañas

- CUT (Arranque)
- CONTOUR D (Corona)
- LIFTER (Arrastre)
- STOPING B (Ayuda Hastial)
- STOPING C (Ayuda Corona)
- WALLS (Hastiales)
- RESULTADOS

Del cuadro de herramientas se eligió la opción MultiPage, con la cual se pudo segmentar las secciones de manera ordenada. Es necesario conocer los controles del esquema de desarrollo de cada sección, el siguiente cuadro muestra un resumen de dichos controles.

Tabla 3-10. Controles de la sección Corona (ContourD)

Caption Control	Name Control
Eficiencia de Voladura	Similar1
Diámetro Taladro	txtTaladCorona
Avance por Disparo	Similar2
K	txtKCorona
S/B	txtSBCorona
Long Carga Fondo	Similar3
Taco	Similar4
Columna	Similar5
B Max	txt47
B Practico	txt48
Espaciamiento	txt49
Nº Taladro	txt50
Nº Cartuchos/Taladro	txt51
Carga Fondo	txt52
Reductor	txtReductor
Carga Columna	txt53
Carga/Taladro	txt54
Carga Total	txt55
Nombre del Explosivo ⁴	Option Button1
Nombre del Explosivo	Option Button2
Peso	txt80
Diámetro	txt81
Densidad	txtDensidadCorona
Spin Button	SpinButton2
Calcular	Command Button 3

⁴ El nombre del explosivo hace referencia al nombre que va tener tanto el Option Button 1 y 2, de esa manera se sabrá qué tipo de explosivo sería el más adecuado para esta sección enfatizando el hecho de que pueden ser iguales dependiendo de la selección previa que hagamos en el primer formulario la cual se replicará en la pestaña Corona sobre los botones ya mencionados.

Tabla 3-11. Controles de Arrastre-Ayuda Hastial-Ayuda Corona y Hastiales

Name Control / Caption Control	Arrastre	Ayuda hastial	Ayuda corona	Hastiales
Eficiencia de Voladura	Similar6	Similar11	Similar16	Similar21
Diámetro Taladro	TextBox62	TextBox82	TextBox101	TextBox101
Avance por Disparo	Similar7	Similar12	Similar17	Similar22
Factor f	Factor1	Factor2	Factor3	Factor4
S/B	SB1	SB2	SB3	SB4
Factor Roca Corregida(c')	C1	C2	C3	C4
Long Carga Fondo	Similar8	Similar13	Similar18	Similar23
Taco	Similar9	Similar14	Similar19	Similar24
Columna	Similar10	Similar15	Similar20	Similar25
B	Burden1	Burden2	Burden3	Burden4
Espaciamiento Esquinas	txtEsquinas Arranque	Empty	Empty	Empty
B Max	valor1	valor10	valor19	valor28
B Practico	valor2	valor11	valor20	valor29
Espaciamiento	valor3	valor12	valor21	valor30
Nº Taladro	valor4	valor13	valor22	valor31
Nº Cartuchos/Taladro	valor5	valor14	valor23	valor32
Carga Fondo	valor6	valor15	valor24	valor33
Carga Columna	valor7	valor16	valor25	valor34
Carga/Taladro	valor8	valor17	valor26	valor35
Carga Total	valor9	valor18	valor27	valor36
Ah Ultimo Cuadrante (Rainura)	Empty	TextBox1201	TextBox1203	TextBox1205
Burden Practico (Hastial)	Empty	TextBox1202	Empty	TextBox1206
Burden Practico (Corona)	Empty	Empty	TextBox1204	Empty
Burden Practico (Zapatera)	Empty	Empty	TextBox1207	Empty
Distancia	Empty	Distancia2	Distancia3	Distancia4
Calcular	Command Button 4	Empty	Empty	Empty

3.11.3.1 Pestaña Arranque

En esta etapa del proyecto se usó diferentes tipos de algoritmos las cuales permitieron un mejor funcionamiento y desenvolvimiento del código de programación, sintetizando todas las operaciones matemáticas con la mejor combinación de métodos y propiedades dando lugar al modelo deseado. El siguiente cuadro muestra la lista de controles que forman parte de esta primera sección al igual que el resto de secciones (Tabla 3-10 y Tabla 3-11), para un mayor entendimiento se consideró renombrar las etiquetas en base al idioma nativo, pero es libertad del usuario si desea hacerlo en otro idioma.

Tabla 3-12. Controles de la sección Arranque (Cutting)

Caption Control	Name Control
Eficiencia de Voladura	TextBox4
Nº Taladros Alivio	TextBox75
Diámetro Equivalente	TextBox5
Avance por Disparo	TextBox6
H Práctico	TextBox76
Long Carga Fondo	TextBox1
Taco	TextBox2
Columna	TextBox3
H Teórico	txtHTeorico
B Max	txt7, txt8, txt9, txt10
B Practico	txt11, txt12, txt13, txt14
Ah	txt15, txt16, txt17, txt18
Nº Taladro	txt19, txt20, txt21, txt22
q (q1-q2)	txt23, txt24, txt25, txt26
Kg/Taladro	txt27, tx28, tx29, txt30
Kg	txt31, txt32, txt33, txt34
Cartuchos/Taladro	txt35, txt36, txt37, txt38
SpinButton	SpinButton1
ListBox	ListDisplay1
Image	Image1
Calcular	Command Button 1

3.11.3.1.1 Rutina dentro del List Box

Se aplicó el mismo criterio tal y como se hizo en el primer formulario de la sección “Parámetros de Diseño”; por cuestiones de seguridad fue necesario establecer ciertas condiciones antes de ejecutar la rutina ya que al iniciar accidentalmente un comando; este no nos genere un error o saltos de página no deseados, para seguir un correcto orden en la ejecución del proyecto se incluyeron algunos mensajes de seguridad del tipo Boolean los cuales permiten dilucidar las secuelas que tendrá la toma de decisiones una vez ejecutado el Command Button 1 “Calcular”.

Para empezar, se establecieron dos variables principales “pathpicture” y “cuenta”. La variable “pathpicture” del tipo String mantiene la ruta de la imagen que queremos cargar mientras que la variable cuenta mantiene el número de filas o listas que disponemos para la selección; esto dependerá de cuanta información tengamos en la tercera base de datos de la Tabla 3-5. Si los controles de Eficiencia de Voladura o H practico están vacíos entonces el mensaje inicial tendrá la siguiente forma (R03).

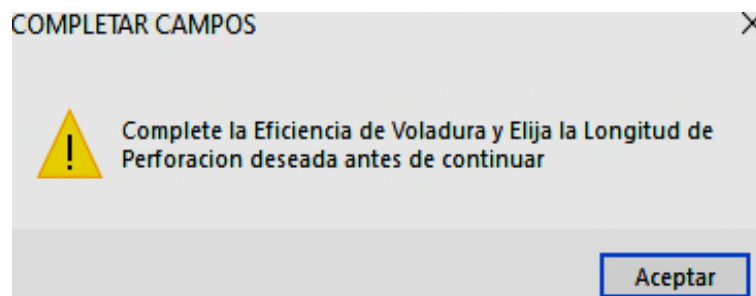


Figure 3-13. Resultado de la rutina 7.8

```
Private Sub ListDisplay1_Click() 'seleccion del tipo de arranque  
Dim pathpicture As String, pathpicture2 As String  
cuenta = Me.ListDisplay1.ListCount  
If Me.TextBox4.Value = Empty Or Me.TextBox76.Value = Empty Then  
MsgBox "Complete la Eficiencia de Voladura y Elija la Longitud _  
de Perforacion deseada antes de continuar", vbExclamation, "COMPLETAR CAMPOS"  
Exit Sub  
End If
```

(R03)

Rutina 3-8. Notificación, de pertenencia List Box-Arranque

Se realizó un primer bucle que afecta únicamente a la lista en su conjunto es decir al momento de arrancar la rutina de selección, este hará una búsqueda de la fila que hallamos elegido, en el momento que lo encuentre se ejecutará otro bucle con el objetivo de sintetizar la rutina que me ayudará a cargar el número de taladros (vacíos y cargados) incluyendo las imágenes. Por ejemplo, al asignarle nombres de la forma ("txt" & numero) se hace mucho más fácil la ejecución del código; como el bucle necesita de un punto de inicio y uno final le asignamos los valores numéricos de los controles (txt19, txt20, txt21 y txt22); esto va de la mano con la fila seleccionada del cuadro de lista. Con el método "selected" del cuadro de lista, se valida la selección hecha y por ende la ejecución de la macro post selección (R04).

Seguidamente se desarrolló el esquema de programación para el cálculo del diámetro equivalente el cual obedece únicamente al tipo de selección del List Box, si el index de selección es igual a cero es decir a la primera fila (Arranque de 1) entonces se tendrá un diámetro equivalente basado en el valor que tengamos en el control ComboBox4 (D2). Si el index de selección es igual a 5 (Arranque de 5) entonces se tendrá un Dequiv basado en el valor del control ComboBox5 (D1) (R04).

Posteriormente se desarrolló la rutina para el cálculo del H teórico. Esto nos dará un valor teórico el cual nos ayudará a determinar valores de Burden y Espaciamiento. La variable potencia guardará el método "Power" el cual dará valores resultantes en base a una raíz "x" que en este caso sería 2 y el control txtH utilizará esa variable para obtener su propio valor (R04).

Por último, para hallar el valor de txtF se utilizó la misma metodología del txtH. Todas las fórmulas utilizadas en este proyecto están descritas y definidas en el marco teórico de la sección Avance por Disparo por lo que no se entrara en detalle (R04).

```

For i = 0 To cuenta - 1
If Me.ListDisplay1.Selected(i) Then
pathpicture = frmCalculo.ListDisplay1.List(i, 1)
frmCalculo.Image1.Picture = LoadPicture(pathpicture) 'cargamos la imagen
For numero = 19 To 22 'cargamos numero de taladros cargados segun tipo de arranque
Me.Controls("txt" & numero).Text = Me.ListDisplay1.List(i, numero - 17)
Next
Me.TextBox75.Text = Me.ListDisplay1.List(i, 0) 'cargamos el numero talad vacios

'calculo del diametro equivalente
If Me.ListDisplay1.ListIndex = 0 Then
diametrovacio = Application.WorksheetFunction.Power(frmCalculo.TextBox75.Value, 1 / 2)
frmCalculo.TextBox5.Text = frmParametro.ComboBox4.Text * diametrovacio
End If
If Me.ListDisplay1.ListIndex = 5 Then
diametrovacio = Application.WorksheetFunction.Power(frmCalculo.TextBox75.Value, 1 / 2)
frmCalculo.TextBox5.Text = frmParametro.ComboBox5.Text * diametrovacio
End If
'calculo del H TEORICO
On Error Resume Next
potencia = Val(Application.WorksheetFunction.Power(frmCalculo.TextBox5.Text / 1000, 2))
frmParametro.txtH.Text = Val(0.15) + Val((34.1 * frmCalculo.TextBox5.Text / 1000)) - Val(39.4 * potencia) |
'recalculo F,ERRORDE PERFORACION
frmParametro.txtF.Text = (frmParametro.txtH.Value * frmParametro.TextBox9.Value) + Val(frmParametro.TextBox10.Value)
frmParametro.txtF.Text = Format(frmParametro.txtF.Text, "0.000")
'H Teorico
Me.txtHTeorico.Text = frmParametro.txtH / 0.3048 'en pies
'calculo del avance por disparo
Me.TextBox6.Text = Me.TextBox76.Value * 0.3048 * Me.TextBox4.Value
End If
Next
End Sub

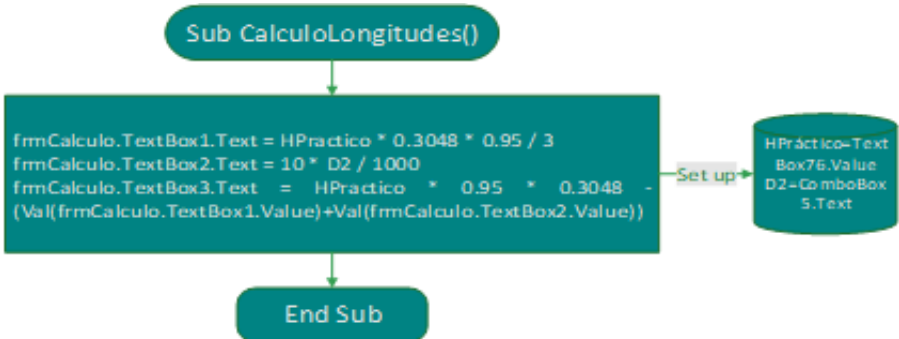
```

(R04)

Rutina 3-9. Selección y cálculo de los parámetros Dequiv, Hteórico y Error de perforación

3.11.3.1.2 Rutina Dentro del Modulo2 (Longitud de Carga de Fondo, Columna y Taco)

Como se explicó anteriormente no todas las secciones de arranque van a tener el mismo esquema matemático, la estructura en algunos casos es diferente incluido las constantes de voladura. En este caso particular las variables de Lcf, Lcc y Taco, al tener la misma estructura de fórmula, esto se puede representar mediante el flujograma siguiente.



Rutina 3-10. Esquema básico para el cálculo de Lcf, Lcc y Taco dentro del módulo 2

La instrucción que se muestra corresponde únicamente al Arranque, Ayuda de Hastial y Ayuda Corona, y se encuentra ligado al control Spin Button 1 (Rutina 3-11), para aplicar este mismo criterio al resto de las secciones de la labor debemos sustituir las variables por las que les corresponden según la Tabla 2-4.

3.11.3.1.3 Rutina dentro del Spin Button 1

La idea del Spin Button es generar un ajuste en la elección del “Hpractico”, debemos elegir el valor más cercano mayor al “Hteorico” y en base a este valor podremos calcular las longitudes de carga.

Se estableció el control de referencia que va tomar el valor del Spin que en este caso sería el Hpractico; los límites menor y mayor hacen referencia a la longitud de la barra de perforación; se consideró una longitud mínima de 6 pies y una longitud máxima de 14 pies; esto no es necesariamente preciso ya que como sabemos las perforaciones en frente pueden superar los 14 ft de longitud si queremos cambiar el límite máximo bastaría con sustituir el valor de 14 por uno de mayor longitud (R07).

La constante incremental toma el valor de 2 ya que generalmente se trabaja con barrenos de longitudes pares. Por último, aprovechamos el evento Change para mandar a llamar el procedimiento “CalculoLongitudes”⁵(R08)

```
Private Sub SpinButton1_Change()  
Me.TextBox76.Value = Me.SpinButton1.Value  
Me.SpinButton1.Max = 14  
Me.SpinButton1.Min = 6 (R07)  
Me.SpinButton1.SmallChange = 2 (R08)  
Call CalculoLongitudes2  
Call CalculoLongitudes|  
End Sub
```

Rutina 3-11. Rutina dentro del Spin Button1 para definir valores de H practico, Lcf, Lcc y Taco

3.11.3.1.4 Evento Initialize

⁵El procedimiento “CalculoLongitudes2” contiene las variables de Lcf, Lcc y T correspondientes a la Corona y Hastiales; mientras que “CalculoLongitudes” contiene las mismas variables respecto al Arrastre y demás secciones.

Para mantener valores predefinidos se utilizó eventos propios del formulario, entre ellas el más importante, el evento Initialize; se predefinieron 14 parámetros relevantes: las 5 primeras correspondientes a las propiedades del List Box y 9 constantes para el cálculo del burden máximo y espaciamiento por cada sección de la labor.

```
Private Sub UserForm_Initialize()  
Me.MultiPage1.Value = 0 'al iniciarse el formulario se posicionara en la primera pagina  
'cargamos el listbox de los tipos de arranque  
Sheet5.Select  
Me.ListDisplay1.ColumnCount = 6  
Me.ListDisplay1.ColumnHeads = True  
Me.ListDisplay1.RowSource = Sheet5.Range("D7:I12").Address  
Me.ListDisplay1.ColumnWidths = "25 pt;25 pt;25 pt;25 pt;25 pt;25 pt"  
'arrastre  
Me.Factor1.Text = 1.45  
Me.SB1.Text = 0.8  
'corona  
Me.txtKCorona.Text = 16  
Me.txtSBCorona.Text = 0.8  
'ayuda hastial  
Me.Factor2.Text = 1.45  
Me.SB2.Text = 1.25  
'ayuda corona  
Me.Factor3.Text = 1.25  
Me.SB3.Text = 1.25  
'Hastial  
Me.Factor4.Text = 1.2  
Me.SB4.Text = 1.25  
End Sub
```

Rutina 3-12. Evento Initialize del formulario "frmCalculo" para establecer atributos propios del List Box y constantes de voladura

3.11.3.2 Command Button 1 - Arranque

3.11.3.2.1 Primer Cuadrante

Se procedió a calcular el resto de valores tales como Burden, Factor de carga lineal, Ancho generado, etc. cumpliendo con cada una de sus ecuaciones. En el caso del factor de carga lineal se determina los dos primeros valores uno correspondiente al arranque y el segundo al Stopping (q1 y q2), en el tercer y cuarto cuadrante se copiarán los mismos valores del q2 declarando las variables “potenciaANFO”, “potenciaANFO2” y “potenciaANFO3”, las cuales van de mayor a menor potencia (R10).

```
frmParametro.txt26.Text / 10000 'factor de carga lineal 1  
Me.txt24.Text = 7.85 * Application.WorksheetFunction.Power(frmParametro.txt23.Text, 2) * (R10)  
frmParametro.txt22.Text / 10000 'factor de carga lineal 1  
Me.txt25.Text = Me.txt24.Text  
Me.txt26.Text = Me.txt24.Text  
potenciaANFO = frmParametro.txt29.Text '5000  
potenciaANFO2 = frmParametro.txt25.Text '3000  
potenciaANFO3 = frmParametro.txt21.Text '1000
```

Rutina 3-13. Representación de ecuaciones sobre el primer cuadrante

3.11.3.2.2 Segundo, Tercer y Cuarto Cuadrante

Para completar los valores de ancho generado, burden máximo y burden práctico se hizo uso de la instrucción “For Next” partiendo del segundo cuadrante con un numero de eventos igual a tres. Seguidamente se aplicó el mismo criterio referente a los kilogramos por taladro partiendo del primer cuadrante con numero de eventos igual a cuatro, en el caso de los cartuchos por taladro se utilizó dos criterios por separado, la primera instrucción que afecta únicamente al primer cuadrante y la segunda instrucción que afecta al resto de cuadrantes mediante un recorrido de tres eventos (R11).

Es importante resaltar que para tener una mejor visualización de los resultados fue necesario trabajar en base a dos decimales como máximo esto se logró mediante la función Format recorriendo los cuatro cuadrantes (R12).


```

For numero = 8 To 10 'burden maximo restante
    Me.Controls("txt" & numero + 8).Text = (Me.Controls("txt" & numero + 3).Text + _ (R11)
(Me.Controls("txt" & numero + 7).Text / 2) - frmParametro.txtF.Text) * _
Application.WorksheetFunction.Power(2, 1 / 2) 'Ah restante
    If numero = 9 Or numero = 10 Then
        potenciaANFO = potenciaANFO2
    End If
    'burden maximo restante
    Me.Controls("txt" & numero).Text = (8.8 / 100) * Application.WorksheetFunction.Power _
(((Me.Controls("txt" & numero + 8).Value * Me.txt24.Text * Val(potenciaANFO)) / _
((frmParametro.ComboBox5.Text / 1000) * frmParametro.txtC.Text))), 1 / 2)
    Me.Controls("txt" & numero + 4).Text = Me.Controls("txt" & numero).Text - _
    frmParametro.txtF.Text 'burden practico restante
Next
For numero = 27 To 30 'kilogramos por taladro
Me.Controls("txt" & numero).Text = Me.Controls("txt" & numero - 4).Text * _
(Me.TextBox76.Text * 0.3048 - Me.TextBox2.Text)
Next
For numero = 31 To 34 'kilogramos
Me.Controls("txt" & numero).Text = Me.Controls("txt" & numero - 4).Text * _
Me.Controls("txt" & numero - 12).Text
Next
'cartuchos por taladr primer cuadrante
Me.txt35.Text = Me.txt27.Text / (frmParametro.txt28.Text)
Me.txt35.Text = Format(Me.txt35.Value, "0")
For numero = 36 To 38 'cartuchos por taladro segundo c
Me.Controls("txt" & numero).Text = Me.Controls("txt" & numero - 8) / (frmParametro.txt24.Text)
Me.Controls("txt" & numero).Text = Format(Me.Controls("txt" & numero).Value, "0")
Next
'FORMATO DE CELDAS
For numero = 31 To 34 'kilogramos (R12)
Me.Controls("txt" & numero).Text = Format(Me.Controls("txt" & numero).Value, "0.00")
Next
End Sub

```

Rutina 3-14. Representación de ecuaciones sobre el resto de cuadrantes

3.11.3.3 Pestaña Corona

El código que se implantó corresponde al evento Change del Multipage1, dentro de este evento se desarrolló lo siguiente:

En cada cambio de página se ejecutará una macro la cual permitirá copiar los valores de Eficiencia de Voladura, Avance, D3, Lcf, Lcc y Taco sobre las demás secciones. Del mismo modo se tendrá como alternativa de selección dos tipos de explosivos una relacionada a la corona propiamente dicha (q3) y la otra relacionada al Stopping (q2). Para obtener el nombre de los mismos se utilizó la propiedad "Caption" de los Combos 1 y 2 del formulario "frmParametro". También se hizo una copia de su respectivo diámetro de taladro el cual se encuentra dentro del primer formulario "frmParametro" (R13).

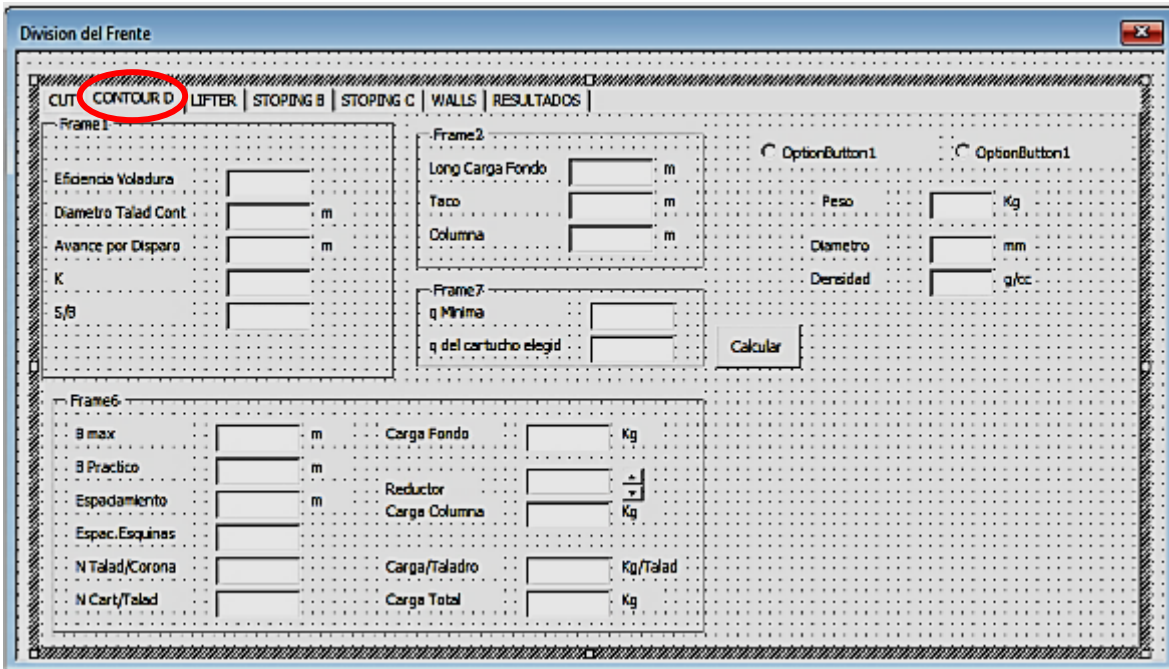


Figure 3-14. Pestaña Corona

La rutina empieza con la variable “numero” del tipo Long la cual mantendrá la secuencia del bucle recorriendo desde un punto inicial 1 hasta un punto final 5 mientras que la variable “valor” del tipo Single contendrá los valores de las variables secundarias que queremos que se copien. Las variables secundarias a su vez de nombre “Similares” mantienen los valores de cada TB, en total tendremos cinco TB a copiar (Eficiencia de voladura, Avance, Lcf, Lcc y Taco). La primera condicional dentro del bucle afecta únicamente a la Corona considerando los valores de eficiencia de voladura y avance, seguidamente se continua con el recorrido original afectando al Arrastre y considerando todos los valores a copiar. Por último, una condición final que afecta únicamente al Hastial copiando los valores de eficiencia de voladura y avance. (R14).

En Hastiales y Corona las longitudes se definieron con anticipación dentro del control Spin Button 1 según la Rutina 3-11.


```
Private Sub MultiPage1_Change() 'eleccion del explosivo para la corona
Me.OptionButton1.Caption = frmParametro.ComboBox1.Text
Me.OptionButton2.Caption = frmParametro.ComboBox2.Text
Me.txtTaladCorona.Text = frmParametro.ComboBox6.Text
```

(R13)

```
'repetimos los valores ya hallados en rainura
Dim numero As Long
Dim valor As Single

Similares1 = Me.TextBox4.Value 'eficiencia voaldura
Similares2 = Me.TextBox6.Value 'avance
Similares3 = Me.TextBox1.Value 'Long fondo|
Similares4 = Me.TextBox2.Value 'taco
Similares5 = Me.TextBox3.Value 'columna

For numero = 1 To 5
valor = Me.Controls("Similares" & numero).Value
'corona, eficiencia voladura y avance, las longitudes ya se definieron anteriormente
If numero >= 1 And numero <= 2 Then
Me.Controls("Similar" & numero).Value = valor
End If
'para arrastre y ayudas se consideran las longitudes iguales
Me.Controls("Similar" & numero + 5).Text = valor
Me.Controls("Similar" & numero + 10).Text = valor
Me.Controls("Similar" & numero + 15).Text = valor
'Hastiales, eficiencia voladura y avance, las longitudes ya se definieron anteriormente
If numero >= 1 And numero <= 2 Then
Me.Controls("Similar" & numero + 20).Text = valor
End If
Next
End Sub
```

(R14)

Rutina 3-15. Rutina dentro del objeto MultiPage1

3.11.3.3.1 Rutina dentro del Spin Button 2

Al igual que el Spin anterior para el ajuste del H práctico fue necesario contar con un reductor que establezca el valor de Cc.

Se estableció el control de referencia que en este caso sería el txt53; los límites menor y mayor delimitan la constante "Reductor" (ver Tabla 2-5); se consideró una constante mínima de 0.5 y una constante máxima de 0.9.

La diferencia con el Spin Button 1 es que este trabaja con valores decimales y como sabemos dentro de la sintaxis del SB solo permite trabajar con valores enteros positivos; dicho esto para obtener valores de 0.5 a 0.9 se cambiaron los valores decimales a valores

enteros de 50 y 90 respectivamente con un valor incremental de 10. Finalmente, el txtReductor tomará los valores predefinidos divididos entre 100.

```
Private Sub SpinButton2_Change()  
'asignamos la secuencia que va tener el spin de 0.5 a 0.9  
para reducir el factor de carga en columna  
Me.txtReductor.Text = Me.SpinButton2.Value / 100  
Me.txtReductor.ControlTipText = vbDecimal  
Me.SpinButton2.Max = 90  
Me.SpinButton2.Min = 50  
Me.SpinButton2.SmallChange = 10  
End Sub
```

Rutina 3-16. Reductor del factor de carga en columna - Spin Button 2

3.11.3.3.2 Option Button

La rutina del Option Button es muy simple, se valida el estado del control si esta activado o no; de estar activado se dispara la subrutina que indica copiar los valores de Peso, Diámetro y Densidad las cuales fueron previamente establecidas dentro del primer formulario. Ya sea si elijamos el OB 1 o 2 la rutina tendrá la misma estructura.

```
Private Sub OptionButton1_Click() |  
If Me.OptionButton1.Value = True Then  
Me.txt80.Text = frmParametro.txt20.Text  
Me.txt81.Text = frmParametro.txt19.Text  
Me.txtDensidadCorona.Text = frmParametro.txt18.Text  
End If  
End Sub
```

Rutina 3-17. Peso, Diámetro y Densidad en función a la selección del Option Button 1 y 2 - Corona

3.11.3.3.3 Command Button 3 - Corona

Para el desarrollo de la rutina final se hizo una validación en primera instancia, la cual permite ejecutar la macro sin problemas, esta validación se basa en constatar que efectivamente se haya completado los criterios de Peso, diámetro y Reductor. De esta manera si uno de ellos se encuentra vacío se desplegará una ventana indicando o informando acerca del error.

Posteriormente se procedió a codificar las ecuaciones del cuadro de fórmulas de la Tabla 2-5, es importante mantener un orden ya que existe dependencia de valores es por eso que se partió por desarrollar el Espaciamiento, Burden Máximo, Burden Práctico, N° Taladros⁶, q mínima, Factor de carga Lineal, Carga Fondo, Carga Columna, Carga/Taladro, Carga Total y N° Cartuchos por Taladro.

```
Private Sub CommandButton3_Click() 'calculo de corona en base al explosivo elegido
If Me.txt80.Value = Empty Or Me.txt81.Value = Empty Or Me.txtReductor.Value = Empty Then
MsgBox "Make sure to complete all variables", vbInformation, "Some Advice"
Exit Sub
End If
Dim flecha As Double, cuerda As Double
flecha = 0.5
cuerda = frmParametro.txtAncho.Text
Me.txt49.Text = Me.txtKCorona.Text * Me.txtTaladCorona.Text / 1000 'espaciamiento
Me.txt49.Text = Format(Me.txt49.Value, "0.0") 'espaciam
'calculo del arco utilizando formula personal
Dim arcoprueba As Double
Dim valornumerico As Double
valornumerico = (4 * flecha * cuerda) / ((4 * Application.WorksheetFunction.Power(flecha, 2)) + _
Application.WorksheetFunction.Power(cuerda, 2))
arcoprueba = WorksheetFunction.Asin(valornumerico) * 180 / 3.14
Dim arco As Double
arco = 2 * ((flecha / 2) + Application.WorksheetFunction.Power(cuerda, 2) / (8 * flecha)) * arcoprueba * 3.14 / 180
Me.txt47.Text = Me.txt49.Text / Me.txtSBCorona.Text 'burden maximo
'burden practico
Me.txt48.Text = Me.txt47.Text - (frmParametro.txtH.Text * Sin(frmParametro.txtAngulo.Text)) - _
frmParametro.txtF.Text
Me.txt50.Text = (arco / Me.txt49.Text) + 1 'numero taladros
Me.txtqMin.Text = 90 * Application.WorksheetFunction.Power(Me.txtTaladCorona.Text / 1000, 2) 'q minima
Me.txtqCorona.Text = 7.85 * Application.WorksheetFunction.Power(Me.txt81.Text, 2) * Me.txtDensidadCorona.Value / 10000
Me.txt52.Text = Similar3.Text * Me.txtqCorona.Text 'carga fondo
Me.txt53.Text = Format(Me.txtReductor.Text, "0.00") * Me.txt52.Text 'carga columna
Me.txt54.Text = val(txt52.Text) + val(txt53.Text) 'carga /taladro
Me.txt55.Text = txt50.Text * txt54.Text 'carga total
Me.txt51.Text = txt54.Text / txt80.Text 'Cart/Talad
```

Rutina 3-18. Desarrollo de la lista de ecuaciones dentro del Command B3 de la sección Corona

⁶ Esto se hizo posible mediante la ecuación arco del cuadro de fórmulas, declarando las variables “valornumerico”, “arcoprueba” y “arco” todas del tipo Double en donde “valornumerico” representa el argumento del arcoseno, la variable “arcoprueba” representa el resultado del arcoseno y la variable “arco” que representa la ecuación general.

3.11.3.4 Arrastre, Ayuda Hastial, Ayuda Corona Y Hastiales

Se inicia el desarrollo del Burden, Constante de Roca Corregida y Burden máximo para ello debemos tener presente dos variables numéricas, la variable “numero” y la variable “secuencia”. La variable “numero” hace referencia a cada una de las secciones por donde hará el recorrido y la variable secuencia es la constante de crecimiento que será igual a una variable secundaria llamada “numero2” multiplicado por la variable “numero”, el valor inicial de la variable secundaria es 9 ya que son 9 los controles que deseamos calcular (Bmax⁷, Bpractico, Espaciamiento, N° Talad/Sección, N° Cart/Tald, Cf, Cc, Carga/Talad y Carga Total) (R15).

El segundo esquema está dado por el B práctico manteniendo la misma secuencia, pero con validaciones distintas (R16).

```
Private Sub CommandButton4_Click() 'calculo del arrastre y demas lados
numero2 = 9
For numero = 1 To 4 'calculo del burden sin corregir
    secuencia = numero2 * numero

    Me.Controls("Burden" & numero).Text = 0.9 * Application.WorksheetFunction.Power(Me.txt24.Text * _
    frmParametro.txt25.Text / (frmParametro.txtC.Text * Me.Controls("Factor" & numero).Text * _
    Me.Controls("SB" & numero).Text), 1 / 2)
    Me.Controls("C" & numero).Text = frmParametro.txtC.Text + Val((0.07 / Me.Controls("Burden" & _
    numero).Text)) 'constante de roca corregida

    'burden maximo valor1
    Me.Controls("valor" & secuencia - 8).Text = 0.9 * Application.WorksheetFunction.Power(Me.txt24.Text * _
    frmParametro.txt25.Text / (Me.Controls("C" & numero).Text * Me.Controls("Factor" & numero).Text * _
    Me.Controls("SB" & numero).Text), 1 / 2) 'burden maximo
    burdenmaximo = Me.Controls("valor" & secuencia - 8).Text

    'burden practico valor2 arrastre
    If numero = 1 Then
        Me.Controls("valor" & secuencia - 8 + 1).Text = Me.Controls("valor" & secuencia - 8).Text - _
        (frmParametro.txtH * Sin(frmParametro.txtAngulo.Text)) - frmParametro.txtF.Text 'burden practico
    End If

    'burden practico valor2 ayuda corona y hastial
    If numero = 3 Or numero = 4 Then
        Me.Controls("valor" & secuencia - 8 + 1).Text = burdenmaximo - frmParametro.txtF.Text 'burden practico
    End If

    'burden practico valor2 ayuda hastial
    If numero = 2 Then
        Me.Controls("valor" & secuencia - 8 + 1).Text = Me.Controls("Burden" & numero).Text -
        frmParametro.txtF.Text 'burden practico
    End If
```

Rutina 3-19. Desarrollo del Burden, Constante de roca corregida, B máximo y B práctico en Arrastre, ayuda corona, hastial y ayuda hastial.

⁷ La idea es tener los valores de burden máximo de las cuatro secciones, cada uno de estos controles están identificados con los nombres de “valor1, 2, 3 y 4” según el cuadro de controles de la Tabla 3-11

Tabla 3-13. Secuencia de validación para establecer el burden práctico en Arrastre, ayuda corona, hastial y ayuda hastial

Validación	Sección	Condición de Validación
Validacion1	Arrastre	numero=1
Validacion2	Ayuda Corona-Hastial	numero=3 v numero=4
Validacion3	Ayuda Hastial	numero=2

El tercer esquema corresponde al N° Taladros en donde se aplica únicamente a la sección Arrastre (R17).

El cuarto esquema corresponde a la Carga de Fondo, Carga Columna, Carga por Taladro y Número de Cartuchos por Taladro. Todos estos cálculos se aplican en todas las secciones sin ninguna restricción, cabe precisar, para que exista una secuencia constante los nombres de los controles deben tener la misma lógica con valores numéricos favorables a la rutina que deseamos ejecutar (R18).

```

*numero taladros del arrastre valor4
If numero = 1 Then
Me.Controls("valor" & secuencia - 8 + 3).Text = ((frmParametro.txtAncho.Text + Val((2 * _
frmParametro.txtH.Text * Sin(frmParametro.txtAngulo.Text)))) / (Me.Controls("valor" & secuencia - 8).Text)) + 2
numerotaladros = Me.Controls("valor" & secuencia - 8 + 3).Text
End If
(R17)

*carga fondo valor6
Me.Controls("valor" & secuencia - 8 + 5).Text = Me.txt24.Text * Me.Similar8.Text
cargafondo = Me.Controls("valor" & secuencia - 8 + 5).Text
*carga columna valor7
Me.Controls("valor" & secuencia - 8 + 6).Text = Me.txt24.Text * Me.Similar10.Text
cargacolumna = Me.Controls("valor" & secuencia - 8 + 6).Text
*carga/taladro valor8
Me.Controls("valor" & secuencia - 8 + 7).Text = Val(cargafondo) + Val(cargacolumna)
cargaportaladro = Me.Controls("valor" & secuencia - 8 + 7).Text

*num cartuchos por taladro valor5
Me.Controls("valor" & secuencia - 8 + 4).Text = Val(cargaportaladro) / Val(frmParametro.txt24.Text)

Next
(R18)

```

Rutina 3-20. Desarrollo del N° Taladros, Carga de fondo, Carga columna, carga por taladro y número de cartuchos por taladro en la sección Arrastre

El quinto esquema corresponde al valor de las distancias para las tres últimas secciones, se reemplazan valores y se aplica la función Format para simplificar el número de decimales (R19). Seguidamente calculamos el número de taladros con las distancias halladas y el espaciamiento de las tres últimas secciones según la Tabla 2-6 (R20)

```

constante = 9
For numero = 2 To 4
secuencia = constante * numero
'cuarta seccion menos el burden del hastial
Distancia2 = Format(frmParametro.txtAncho.Text - lastAh - (Me.valor29.Text * 2), "0.00")
'cuarta seccion menos el burden de al corona
Distancia3 = Format(frmParametro.txtAltura.Text - lastAh - Me.txt48.Text - Me.valor2.Text, "0.00")
'burden arrastre menos el burden de la corona
Distancia4 = Format(frmParametro.txtAncho.Text - Me.valor2.Text - (Me.txt48.Text * 2), "0.00")

```

(R19)

```

'calculamos el numero de taladros con la distancia hallada
burdenmaximo = Me.Controls("valor" & secuencia - 8).Text
Me.Controls("valor" & secuencia - 8 + 3).Text = val((Me.Controls("Distancia" & numero).Text / _
(burdenmaximo * Me.Controls("SB" & numero).Text))) + val(2)
numerotlaladros = Me.Controls("valor" & secuencia - 8 + 3).Text
'espaciamiento del resto
If numero = 2 Then 'hastial
Me.Controls("valor" & secuencia - 6).Text = 1.25 * burdenmaximo
End If
If numero = 3 Or numero = 4 Then
Me.Controls("valor" & secuencia - 6).Text = 1.25 * burdenmaximo
End If
Next

```

(R20)

Rutina 3-21. Desarrollo de las distancias y número de taladros para Ayuda Hastial, Ayuda Corona y Hastial

El cálculo del espaciamiento del Arrastre se realiza tanto en el centro como en las esquinas (R21).

```

'espaciamiento del arrastre
Me.Controls("valor" & 3).Text = (frmParametro.txtAncho.Text + Val((2 * frmParametro.txtH.Text * _
Sin(frmParametro.txtAngulo.Text)))) / (Application.WorksheetFunction.Round(Me.valor4.Value, 0) - 1)
'espaciamiento de las esquinas del arrastre
Me.txtEsquinasArranque.Text = Me.valor3.Text - (frmParametro.txtH.Text * Sin(frmParametro.txtAngulo.Text))

```

(R21)

Rutina 3-22. Espaciamiento del Arrastre, centro y esquinas

3.11.3.5 Pestaña Resultados

La idea general del Command Button “Generate Result” es generar un reporte de todos los resultados obtenidos, al ser un código de programación complementario no se mostrará ni se entrará en detalle acerca de su contenido, pero si se tendrá una visualización de esta en el caso práctico expuesto de (Figure 4-13).

3.12 Diseño de Malla

Una vez preparado la rutina para la obtención del burden y espaciamiento de todas las secciones se procedió a estructurar la rutina que nos ayudará a diseñar de forma gráfica el tipo de malla que deseamos visualizar.

3.12.1 Acceso a la Jerarquía de Objetos y Esquema de Desarrollo

Para obtener un vínculo con el dibujo activo se parte de una cabecera que en este caso sería el “autocad” propiamente dicho seguido del objeto “Application” y el documento actual activo “ActiveDocument”, en ciertos casos toda esta secuencia puede resumirse con la palabra “ThisDrawing” pero va depender de la versión de AutoCAD con la que estemos trabajando, en una versión AutoCAD 2020 no se aplica dicho termino.

Se utilizó el método “Add” el cual añade un nuevo dibujo y lo añade a la colección “Documents”. Para iniciar un nuevo procedimiento se agregó un nuevo módulo de nombre “**Diseño**” la cual contendrá toda la estructura base de procesos e interacción con otros módulos.

Cabe precisar que no se entrará en detalle sobre la rutina de programación referente a los atributos y rutinas complementarias, solo se explicará el desarrollo estructural de la malla.

3.12.1.1.1 Punto de Partida

Para iniciar con el diseño de malla se estableció un punto de partida, es necesario trabajar en base a un sistema de coordenadas que el usuario disponga dentro del espacio modelo. Para ello se declararon dos variables “X” y “Y” del tipo Public, esto debido a que iremos usando estos valores en los diferentes módulos de trabajo. Las variables de dibujo en esta primera etapa tienen el nombre de “circulo”, “pc (0 to 2)” y “Diametro”, la variable “circulo” del tipo AcadCircle, la variable “Diametro” del tipo Double y la variable “pc (0 to 2)” del tipo Double, esta última es un array de tres elementos que hacen referencia a las coordenadas

(x, y, z). Las variables “X” y “Y” contendrán el resultado de la función InputBox en donde se tiene que ingresar valores numéricos referentes al punto inicial del diseño dentro del espacio modelo mientras que la variable “Diametro” contendrá el valor del diámetro del taladro de alivio (R22).

Seguidamente se dibuja el círculo en base a las coordenadas ingresadas utilizando el método AddCircle con referencia a la variable “circulo” la cual al ser un objeto debe estar antecedida de la palabra “Set”. (R23).

```
'dibujamos
Dim Diametro As Double, pc(0 To 2) As Double, linea(0 To 5) As Double, _
linea2(0 To 5) As Double, circulo2x As Double, circulo2y As Double
Diametro = frmParametro.ComboBox4.Value / 1000
Dim circulo As AcadCircle, circulo2 As AcadCircle, referencia As AcadCircle, circulo3 As AcadCircle
Dim pcAlivio(0 To 5) As Double

X = InputBox("Defina el punto de partida en X", "START POINT X", Default:=True)
Y = InputBox("Defina el punto de partida en Y", "START POINT Y", Default:=True)
'x          'y          'z
pc(0) = X: pc(1) = Y: pc(2) = 0
```

(R22)

```
'dibujamos el círculo inicial |
Set circulo = autocad.Application.ActiveDocument.ModelSpace.AddCircle(pc, Diametro / 2)
```

(R23)

Rutina 3-23. Desarrollo del Círculo Base como punto de partida

3.12.1.1.2 Diseño del Primer Cuadrante

Dibujamos el segundo círculo en base al burden práctico inicial para ello se declararon las variables “diametro2” del tipo Double y la variable “circulo2” del tipo AcadCircle. Le asignamos a la variable “diametro2” el valor que le corresponde que en este caso sería el diámetro de los taladros cargados, la variable “pc(0)” tomará el valor del punto inicial más el burden práctico manteniendo la variable “pc(1)” en su estado inicial. Con las variables declaradas y definidas se procede a dibujar el círculo de partida del primer cuadrante (R24).

```
'PRIMER CUADRANTE
'dibujamos el segundo círculo,le damos formato y array
Dim diametro2 As Double
diametro2 = frmParametro.ComboBox5.Value / 1000
pc(0) = X + Val(frmCalculo.txt11.Value): pc(1) = Y
Set circulo2 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(pc, diametro2 / 2)
```

(R24)

Rutina 3-24. Esquema para el diseño del primer cuadrante – desarrollo del segundo círculo base

Seguidamente se procede a dibujar la cota teniendo como referencia ambos círculos, para ello es necesario partir de un módulo adicional la cual crearemos dentro de la carpeta módulos con el nombre "COTAS", dentro de este nuevo módulo se creó al rutina de cotación que engloba a todos los objetos presentes en el espacio modelo, conforme se avanza en el diseño de todas las secciones se irá acotando haciendo uso de la misma rutina mediante la llamada de la misma de esa manera se tendrá una estructura de codificación más sintetizada y dinámica.

Primeramente, se definieron las variables públicas con las que trabajará el módulo "COTAS", se declararon las variables "cota1x", "cota1y", "cota2x", "cota2y" y "posicionlineaExtension" todas del tipo Double. Adicionalmente se agregaron variables secundarias "circulo2x" y "circulo2y". Las variables "cota1x" y "cota1y" mantendrán los valores X y Y respectivamente, las variables "circulo2x" y "circulo2y" mantendrán las coordenadas del "circulo2" en los ejes (x, y) respectivamente, las variables "cota2x" y "cota2y" a su vez mantendrán los valores de las variables "circulo2x" y "circulo2y" y la variable "posicionlineaExtension" hace referencia al valor de separación de la línea de extensión de la cota. Finalmente hacemos el llamado del módulo "COTAS" con el procedimiento de nombre "generatecota" (R25).

La macro "generatecota" al ser un complemento no afecta la estructura base del diseño de malla, por lo que no se entrará en detalle, el código estará presente dentro del disco que engloba todo el proyecto (R25).

```
circulo2x = circulo2.center(0)
circulo2y = circulo2.center(1)
'COTA
posicionlineaExtension = 0.7 (R25)
cota1x = X: cota1y = Y
cota2x = circulo2x: cota2y = circulo2y
Call generatecota
```

Rutina 3-25. Esquema de cotación para el primer cuadrante

Una vez tenido el círculo de partida del primer cuadrante se procederá a utilizar el método "Arraypolar", para ello se declararon las variables: "nofobjects" del tipo Integer, "angletofill" del tipo Double, "basepoint (0 to 2)" del tipo Double, "arraypolar" del tipo Variant, "Each Circle" del tipo AcadCircle, "nombrearray" del tipo String y la variable "intCount" del tipo Integer. La

variable “nofobjects” contendrá el número de taladros del primer cuadrante incluido el círculo de partida, la variable “angletofill” contendrá el valor establecido por defecto de 360° expresado en radianes, la variable “basepoint ()” contendrá las coordenadas del punto de referencia o centro del array resultante y la variable “arraypolar” que representa el resultado del método arraypolar como un objeto del tipo Variant (R26).

```

*array
..... (R26)
Dim nofobjects As Integer
Dim angletofill As Double
Dim basepoint(0 To 2) As Double
Dim arraypolar As Variant
Dim EachCircule As AcadCircle
Dim nombreakarray As String
Dim intCount As Integer
nofobjects = frmCalculo.txt19.Value + 1
angletofill = 3.14 * 2 * 360 degrees
basepoint(0) = X: basepoint(1) = Y
arraypolar = circulo2.arraypolar(nofobjects, angletofill, basepoint)

```

Rutina 3-26. Desarrollo del primer cuadrante con el método Array

3.12.1.1.3 Diseño del Segundo Cuadrante

Para generar el área del segundo cuadrante (C2) se hicieron algunos cálculos geométricos en base al tipo de arranque que se encuentra en tratamiento, al tratarse de un arranque de un solo taladro de alivio se tendrá ángulos de 45 grados en todas las intersecciones del primer cuadrante, la idea es posicionarnos a una distancia perpendicular (burden del segundo cuadrante) a uno de los lados del primer cuadrante tal y como se muestra en la siguiente figura.

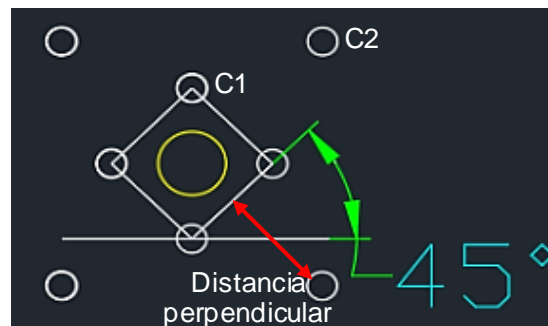


Figure 3-15. Diseño del segundo cuadrante

Se declararon las variables “basePnt()”, “angle” y “distance” todas del tipo Double y las variables “polarPnt()”, “polarPnt1()” y “polarPnt2()” todas del tipo Variant. Estas últimas representan la posición de una coordenada en el espacio modelo en forma de objeto. La variable “basePnt” representa las coordenadas iniciales del método PolarPoint como punto de partida, la variable “angle” representa el ángulo de giro expresada en radianes y la variable “distance” representa la distancia que va recorrer el punto de inicio que será igual al burden práctico del primer cuadrante dividido entre la raíz de dos. Una vez declaradas las variables se procederá a ejecutar el método PolarPoint (R27).

Finalmente aplicamos la misma rutina de desplazamiento con un ángulo de -45° , una distancia igual al burden práctico del segundo cuadrante y el punto de partida (polarPnt). Una vez obtenido el punto final de posicionamiento se dibuja el círculo en base a las coordenadas de ese punto, conservamos las coordenadas del círculo para posteriores usos y aplicamos el arraypolar con número de objetos igual al número de taladros del segundo cuadrante (+1) con las mismas variables “angletofill” y “basepoint” (R28).

```

'SEGUNDO CUADRANTE
' distance and angle from a base point.
|
Dim polarPnt As Variant, polarPnt1 As Variant, polarPnt2 As Variant
Dim basePnt(0 To 2) As Double
Dim angle As Double
Dim distance As Double

basePnt(0) = circulo2x: basePnt(1) = circulo2y: basePnt(2) = 0
angle = -2.355 ' -135 degrees
distance = frmCalculo.txt11.Value / Application.WorksheetFunction.Power(2, 1 / 2)
polarPnt = autocad.Application.ActiveDocument.Utility.PolarPoint(basePnt, angle, distance)

Dim referenciacotax As Variant
Dim referenciacotay As Variant
referenciacotax = polarPnt(0)
referenciacotay = polarPnt(1)

```

(R27)

```

angle = -0.785 ' -45 degrees
distance = frmCalculo.txt12.Value
polarPnt = autocad.Application.ActiveDocument.Utility.PolarPoint(polarPnt, angle, distance)

Set circulo2 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt, diametro2 / 2)
p = circulo2.center(0)
q = circulo2.center(1)

'repetimos los pasos con el arraypolar
nofobjects = frmCalculo.txt20.Value + 1
arraypolar = circulo2.arraypolar(nofobjects, angletofill, basepoint)

```

(R28)

Hace referencia a los parámetros requeridos para el proceso de cotación, no afecta al resto de la rutina

Rutina 3-27. Desarrollo del segundo cuadrante con los métodos PolarPoint, AddCircle y Arraypolar

3.12.1.1.4 Diseño del Tercer Cuadrante

Para el diseño del tercer cuadrante fue necesario tomar como referencia la esquina superior derecha del segundo cuadrante a partir de ese punto se inicia el recorrido para dibujar el círculo base del tercer cuadrante, se hizo un recorrido de objetos o entidades presentes dentro del espacio modelo y se hizo una validación que afecta únicamente a las entidades del tipo círculo, se declaró la variable "XLnCADObject" del tipo AcadObject, al tratarse de la entidad solicitada se igualó a la variable "referencia" y se conservaron los valores (x, y) dentro de las variables secundarias "A" y "b" (R29).

Seguidamente se establecieron las condiciones de selección que cumplan con el rol de encontrar o posicionarse en la esquina superior derecha del segundo cuadrante, una vez cumplido la condición se hizo uso de algunas variables ya declaradas previamente, la variable "angle" que tomará el valor de 90° la cual mantendrá una dirección positiva de abajo hacia arriba partiendo del círculo base del segundo cuadrante, la variable "distance" tomará el valor del punto medio entre las variables "q" y "b" recorriendo una distancia igual a la diferencia entre "distance" y "q", la variable pc() que será igual a la posición del círculo base del segundo cuadrante (p, q) como punto de partida y la variable "polarPnt" que mantendrá el punto final del recorrido, llegando hasta el punto medio. Se conservan los valores (x, y) del "polarPnt" dentro de la variable "referenciacotax(y)" para la generación de cotas (R30).

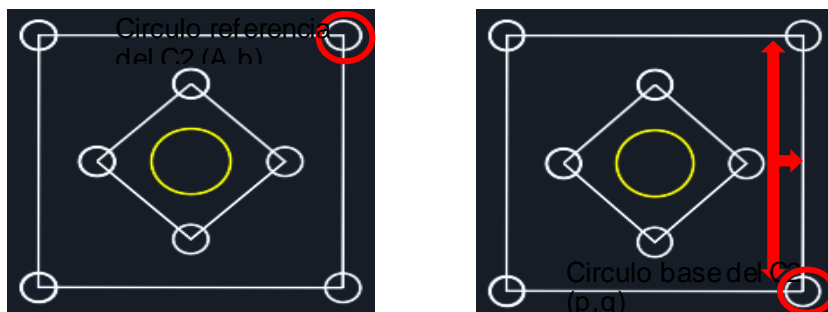


Figure 3-16. Búsqueda del punto medio sobre la base del segundo cuadrante

Repetimos el procedimiento con desplazamiento hacia la derecha definiendo la variable "angle" igual a 0° , la variable "distance" igual al burden práctico del tercer cuadrante y la variable "polarPnt" que mantendrá el punto final del recorrido, seguidamente dibujamos el círculo base del tercer cuadrante y aplicamos el array respectivo con los mismos criterios a excepción de la variable "nofobjects" que tomará el valor del número de taladros cargados

del tercer cuadrante más uno. Se conservan los valores (x, y) del círculo base del tercer cuadrante dentro de las variables "R" y "s" (R31).

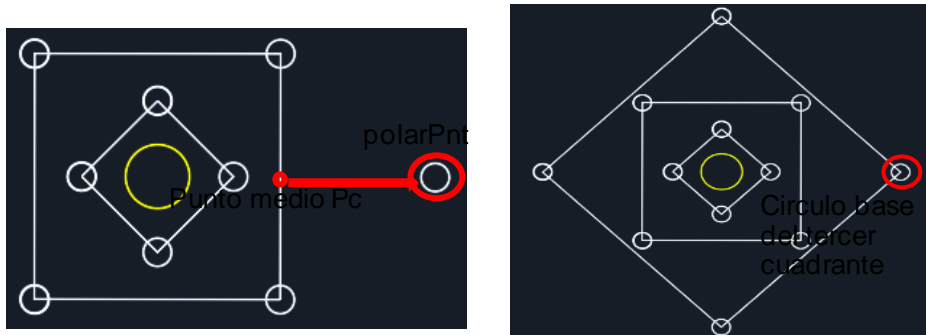


Figure 3-17. Recorrido desde el punto medio y diseño del tercer cuadrante (C3)

```

Dim XLnCADObject As AcadObject
For Each XLnCADObject In autocad.Application.ActiveDocument.ModelSpace
    If XLnCADObject.ObjectName = "AcDbCircle" Then
        Set referencia = XLnCADObject 'circulo de referencia
        A = referencia.center(0)
        b = referencia.center(1)
    End If
    If A > X + 0.1 And A < p + 0.1 And b > Y And b > q Then
        angle = 1.57 '90 degrees 'subimos hacia arriba,debo tener las coordenadas del circulo superior derecho
        distance = (Val(q) + Val(b)) / 2
        distance = distance - q
        pc(0) = p: pc(1) = q
        polarPnt = autocad.Application.ActiveDocument.Utility.PolarPoint(pc, angle, distance)
        .....
        referenciacotax = polarPnt(0)
        referenciacotay = polarPnt(1)
    End If
    'hacia la derecha
    angle = 0 '0 degrees
    distance = frmCalculo.txt13.Value
    polarPnt = autocad.Application.ActiveDocument.Utility.PolarPoint(polarPnt, angle, distance)
    'dibujamos el circulo
    Set circulo2 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt, diametro2 / 2)
    'aplicamos elarray
    nofojects = frmCalculo.txt21.Value + 1
    arraypolar = circulo2.arraypolar(nofojects, angletofill, basepoint)
    R = circulo2.center(0)
    s = circulo2.center(1)
    circulo2.Delete
End If
End If
Next
    
```

Rutina 3-28. Desarrollo del Tercer cuadrante

3.12.1.1.5 Diseño del Cuarto cuadrante

Para el diseño del cuarto cuadrante fue necesario tomar como referencia el círculo base del tercer cuadrante a partir de ese punto se inicia el recorrido para dibujar el círculo base del cuarto cuadrante, se hizo un recorrido de objetos o entidades presentes dentro del espacio modelo y se hizo una validación que afecta únicamente a las entidades del tipo círculo, al tratarse de la entidad solicitada se igualó a la variable “referencia” y se conservaron los valores (x, y) dentro de las variables secundarias “A” y “b”⁸. Seguidamente se establecieron las condiciones de selección que cumplan con el rol de encontrar el círculo inferior del tercer cuadrante, una vez cumplido la condición se calcula el punto medio entre el círculo base del tercer cuadrante y el círculo inferior de la misma manteniendo así las coordenadas del punto medio dentro de las variables “ladoX” y “ladoY” (R32).

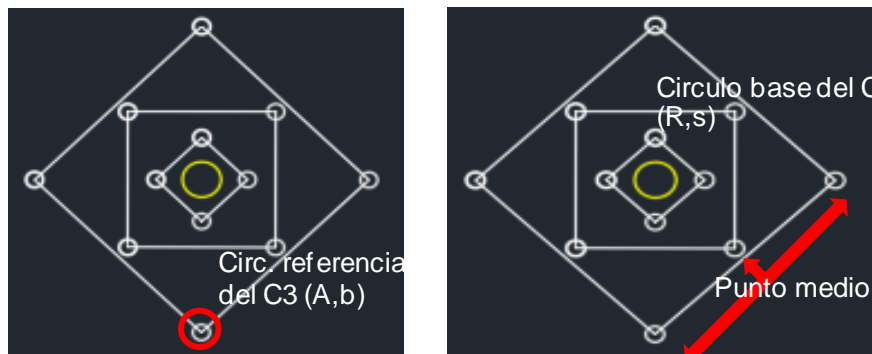


Figure 3-18. Búsqueda del punto medio a partir del círculo base y referencia

Se hizo uso de algunas variables ya declaradas previamente, la variable “angle” que tomará el valor de -45° la cual mantendrá una dirección negativa Sur-Este, la variable “distance” tomará el valor del burden práctico del cuarto cuadrante, la variable “pc()” que será igual a la posición del punto medio (ladoY, ladoX) como punto de partida y la variable “polarPnt” que mantendrá el punto final del recorrido, se procede a dibujar el círculo base del cuarto cuadrante y el array con “nofobjects” igual al número de taladros cargados del cuarto cuadrante más uno. Se conservan los valores (x, y) del círculo base del cuarto cuadrante dentro de las variables “T” y “u” (R33)

⁸ Se utilizó las mismas variables del segundo cuadrante por razones de herencia, ya que estos pueden adoptar nuevos valores una vez hayan cumplido su objetivo. Lo mismo sucede con el objeto PolarPnt y Pc

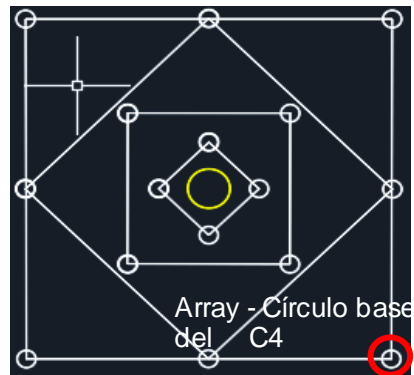
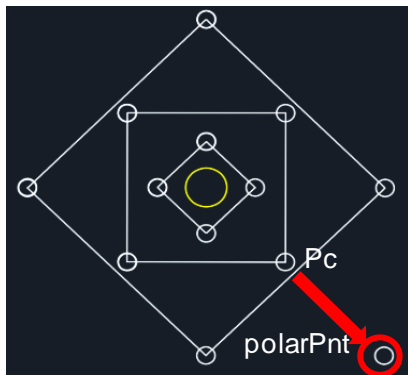


Figure 3-19. Recorrido y diseño del cuarto cuadrante (C4)

```
'CUARTO CUADRANTE
For Each XLnCADObject In autocad.Application.ActiveDocument.ModelSpace
  If XLnCADObject.ObjectName = "AcDbCircle" Then

    Set referencia = XLnCADObject 'circulo de referencia
    A = referencia.center(0)
    b = referencia.center(1)
    'usamos el q del segundo cuadrante ya que se encuentra posicionado mas abajo
    If A > X - 0.1 And A < R And b < q - 0.1 Then
      ladoY = (Val(s) + Val(b)) / 2
      ladoX = (Val(R) + Val(A)) / 2
```

(R32)

```
pc(0) = ladoX: pc(1) = ladoY
angle = -0.785 '-45 degrees
distance = frmCalculo.txt14.Value
polarPnt = autocad.Application.ActiveDocument.Utility.PolarPoint(pc, angle, distance)

'dibujamos el circulo
Set circulo2 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt, diametro2 / 2)

'aplicamos el array
nofobjects = frmCalculo.txt22.Value + 1
arraypolar = circulo2.arraypolar(nofobjects, angletofill, basepoint)

T = circulo2.center(0)
u = circulo2.center(1)
circulo2.Delete
Exit For
End If
End If
Next
```

(R33)

Rutina 3-29. Desarrollo del cuarto cuadrante

3.12.1.1.6 Diseño del Stopping C

Primeramente se ubica el punto de referencia mediante la instrucción “For Each” utilizando el mismo criterio de búsqueda (tercer y cuarto cuadrante), la idea es posicionarse en el círculo superior del tercer cuadrante, a partir de ese punto, se hará el recorrido de puntos utilizando las variables: “angle” igual a 90°, la variable “distance” igual al burden práctico de la sección StoppingC, la variable “polarPnt1” declarada previamente mantendrá el resultado del punto final del recorrido, seguidamente se procede a ejecutar la misma instrucción con valores de ángulo igual a 180° (hacia la izquierda), una distancia equivalente a la mitad del espaciamiento del StoppingC y la variable polarPnt2 como resultado del desplazamiento (R34)

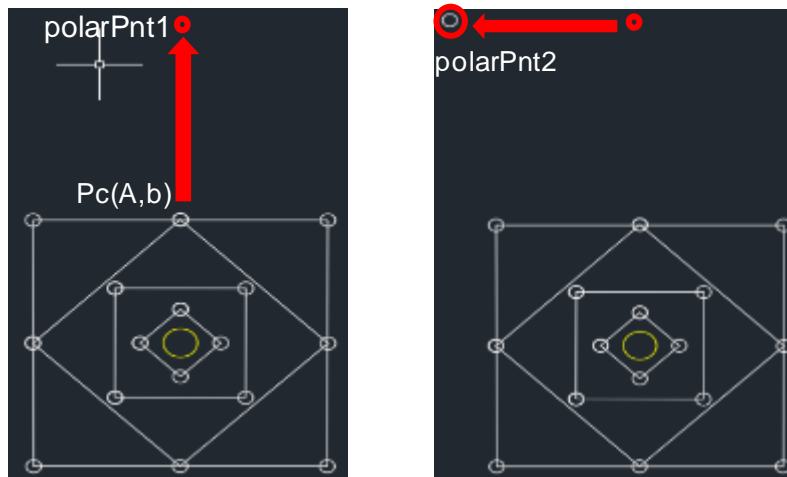


Figure 3-20. Ubicación de partida – Ayuda corona

```
'AYUDA CORONA
'ubico mi punto de referencia
For Each XLnCADObject In autocad.Application.ActiveDocument.ModelSpace
  If XLnCADObject.ObjectName = "AcDbCircle" Then

    Set referencia = XLnCADObject 'circulo de referencia
    A = referencia.center(0)
    b = referencia.center(1)

    If A > X - 0.1 And A < X + 0.1 And b > Y + Val(frmCalculo.txt11.Value) + Val(frmCalculo.txt13.Value) Then
      pc(0) = A: pc(1) = b
      angle = 1.57 '90 hacia arriba
      distance = frmCalculo.valor20.Value
      polarPnt1 = autocad.Application.ActiveDocument.Utility.PolarPoint(pc, angle, distance)

      angle = 3.14 '0 hacia izquierda
      distance = frmCalculo.valor21.Value / 2
      polarPnt2 = autocad.Application.ActiveDocument.Utility.PolarPoint(polarPnt1, angle, distance)
    End If
  End If
Next
```

(R34)

Rutina 3-30. Desarrollo del punto de partida – Ayuda corona

Una vez ubicado nuestro punto de partida debemos establecer ciertas condiciones para el número de taladros que se quiera dibujar. Si el número de taladros es dos simplemente se dibuja el círculo en base al punto de partida polarPnt2 utilizando el método AddCircle con radio igual al diámetro del taladro cargado (D2) dividido entre dos, estableciéndose así el objeto círculo de nombre "circulo3" (R35).

Por último, se utilizó el método Mirror del objeto "circulo3" declarando previamente las variables "point1" y "point2" todas del tipo Double, estas variables tomaran valores que representan un eje de intersección, en función de ese eje se aplicara el método mencionado obteniéndose así una copia perfecta del círculo original manteniendo la simetría deseada. Cabe precisar, cada vez que queramos utilizar algún método de una entidad debemos hacerlo en base a un objeto similar el cual contendrá el resultado de esta es así que se declaró la variable "mirrorObj" del tipo AcadCircle por ser de la misma entidad (R35)

Si el número de taladros es igual a 3 se aplica el mismo criterio de la rutina (R35) y se completa dibujando un nuevo círculo de nombre "circulo3", se considera un ángulo de inclinación de 90°, el punto de partida "polarPnt1" y una distancia igual al burden práctico del StoppingC (R36).

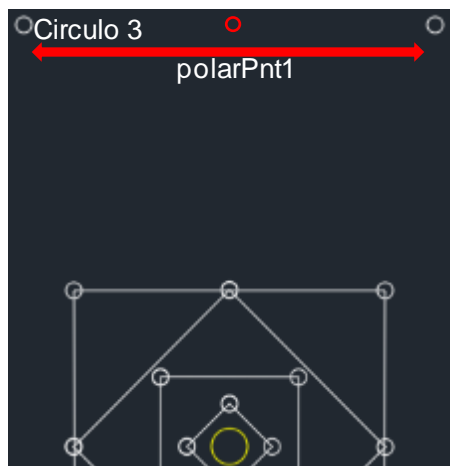


Figure 3-22. Diseño del Stopping C si el número de taladros es 2

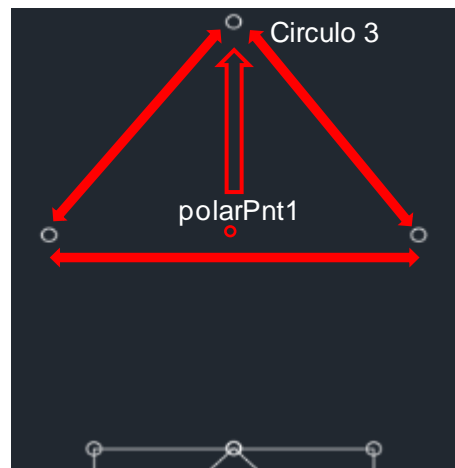


Figure 3-21. Diseño del Stopping C si el número de taladros es 3

```

numero = frmCalculo.valor22.Value
If numero = 2 Then 'si el N°Talad es 2
  Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt2, diametro2 / 2)

  Dim point1(0 To 2) As Double
  Dim point2(0 To 2) As Double
  point1(0) = X: point1(1) = Y: point1(2) = 0
  point2(0) = X: point2(1) = Y + 5: point2(2) = 0
  ' Mirror the polyline
  Dim mirrorObj As AcadCircle
  Set mirrorObj = circulo3.Mirror(point1, point2)

```

(R35)

```

ElseIf numero = 3 Then 'si el N°Talad es 3
  Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt2, diametro2 / 2)

  point1(0) = X: point1(1) = Y: point1(2) = 0
  point2(0) = X: point2(1) = Y + 5: point2(2) = 0
  ' Mirror the polyline
  Set mirrorObj = circulo3.Mirror(point1, point2)

  'agregamos el tercero
  angle = 1.57 '90 hacia arriba
  distance = frmCalculo.valor20.Value 'burden
  polarPnt1 = autocad.Application.ActiveDocument.Utility.PolarPoint(polarPnt1, angle, distance)
  Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt1, diametro2 / 2)

```

(R36)

Rutina 3-31. Desarrollo del Stopping C para dos y tres taladros

Si el número de taladros es cuatro dibujamos la entidad circulo3 en base al punto polarPnt2 con radio igual a la mitad del diámetro del taladro cargado, se declararon las variables “arrayrect” del tipo Variant, la variable “filas” y “columnas” igual a la mitad del número de taladros, se ejecuta el método “arrayrectangular” con las variables predefinidas incluyendo el número de labels igual a uno, una distancia entre filas equivalente al burden practico del StoppingC, una distancia entre columnas equivalente al espaciamiento del StoppingC y una distancia entre labels de 0.06, obteniendo de esta manera un array perfecto de dos por dos (R37)

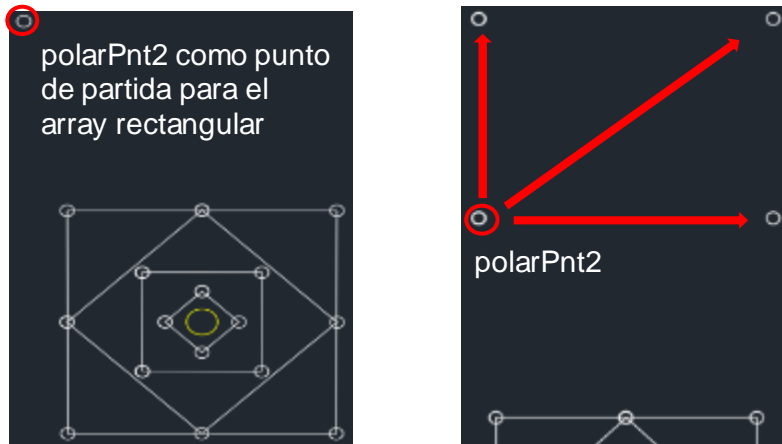


Figure 3-23. Diseño del Stopping C si el número de taladros es 4

Si el número de taladros es cinco dibujamos la entidad circulo3 en base al punto polarPnt2 con radio igual a la mitad del diámetro del taladro cargado, se aplica el mismo criterio de la rutina 18 con numero de filas y columnas igual al número de taladros menos uno dividido entre dos, el resto de criterios se mantiene y se añade el quinto círculo con ángulo de inclinación de 90° , una distancia equivalente a dos veces el burden práctico y el punto de partida polarPnt1 (R38)

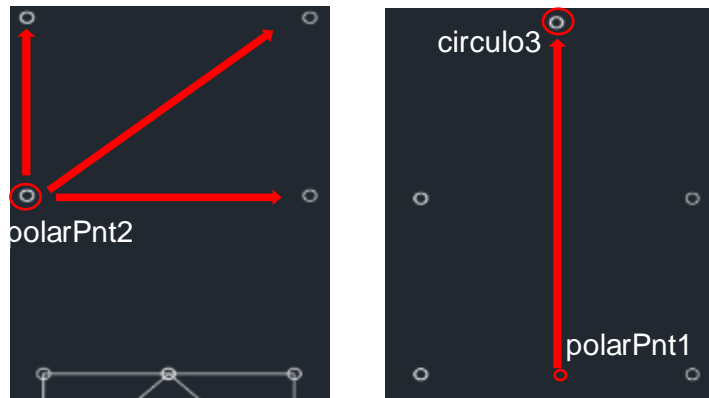


Figure 3-24. Diseño del Stopping C si el número de taladros es 5

Si el número de taladros es 6 se repite la misma instrucción de la rutina número 18 considerando las mismas variables y métodos (R39).

```

ElseIf numero = 4 Then 'si el N°Talad es 4
Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt2, diametro2 / 2) (R37)

'array rectangular
Dim arrayrect As Variant|
filas = numero / 2
Columnas = numero / 2
arrayrect = circulo3.arrayrectangular(filas, Columnas, 1, frmCalculo.valor20.Value, frmCalculo.valor21.Value, 0.06)

```

```

ElseIf numero = 5 Then 'si el N°Talad es 5 (R38)
Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt2, diametro2 / 2)
'array rectangular
filas = (numero - 1) / 2 'frmCalculo.valor13.Value
Columnas = (numero - 1) / 2
arrayrect = circulo3.arrayrectangular(filas, Columnas, 1, frmCalculo.valor20.Value, frmCalculo.valor21.Value, 0.06)

'agregamos el quinto
angle = 1.57 '90 hacia arriba
distance = frmCalculo.valor20.Value * 2
polarPnt1 = autocad.Application.ActiveDocument.Utility.PolarPoint(polarPnt1, angle, distance)
Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt1, diametro2 / 2)

```

```

ElseIf numero = 6 Then 'si el N°Talad es 6 (R39)
Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt2, diametro2 / 2)
'array rectangular

filas = numero / 2
Columnas = numero / 2
arrayrect = circulo3.arrayrectangular(filas, Columnas, 1, frmCalculo.valor20.Value, frmCalculo.valor21.Value, 0.06)

End If

```

Rutina 3-32. Desarrollo del Stopping C para cuatro, cinco y seis taladros

3.12.1.1.7 Diseño del Stopping B

Se eligió como punto de partida el círculo base del tercer cuadrante manteniendo sus coordenadas en la variable “pc()”, en base a este punto se hizo un recorrido con ángulo de inclinación de 0°, una distancia equivalente al burden práctico del StoppingB y la variable polarPnt como resultado del desplazamiento N-E. Seguidamente utilizando el mismo criterio del método PolarPoint se hizo un recorrido de puntos con un ángulo de inclinación de -90°, una distancia equivalente al espaciamiento del StoppingB dividido entre dos y la variable polarPnt como resultado del desplazamiento N-S. Seguidamente se crea la entidad “circulo2” en base al punto polarPnt (R40).

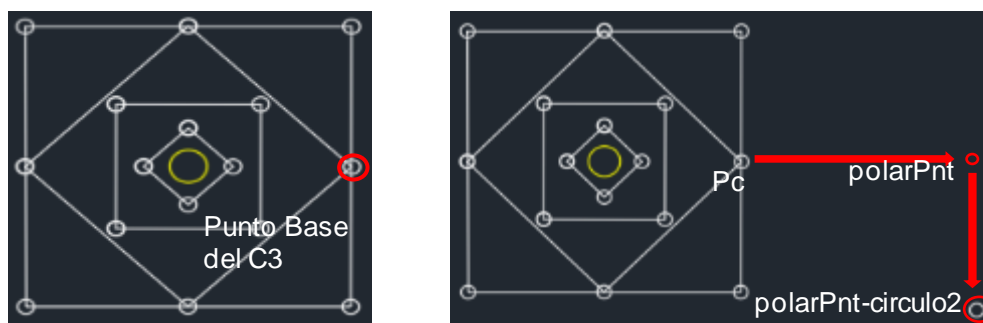


Figure 3-25. Punto de partida del Stopping B


```
'AYUDA HASTIAL
'utilizo las coordenadas r y s
'hacia la derecha
pc(0) = R: pc(1) = s|
angle = 0 '0 degrees
distance = frmCalculo.valor11.Value
polarPnt = autocad.Application.ActiveDocument.Utility.PolarPoint(pc, angle, distance)

'hacia abajo
angle = -1.57 '0 degrees
distance = frmCalculo.valor12.Value / 2
polarPnt = autocad.Application.ActiveDocument.Utility.PolarPoint(polarPnt, angle, distance)

'dibujamos el circulo
Set circulo2 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt, diametro2 / 2)
```

R40

Rutina 3-33. Desarrollo del Punto de partida del Stopping B

En base al punto polarPnt (circulo2) se diseña el array del tipo rectangular con número de filas igual al número de taladros en primera instancia del StoppingB, con número de columnas igual a 1, numero de labels igual a 1, una distancia entre filas equivalente al espaciamiento del StoppingB, una distancia entre columnas igual a 0.06 y una distancia entre labels de 0.06. Finalmente hacemos uso del método “mirror” por cada objeto dentro del espacio modelo, cabe precisa que solo se aplica a las entidades del tipo AcadCircle estableciendo la línea o eje de simetría la cual estará representada por las variables “point1” y “point2” (R41).

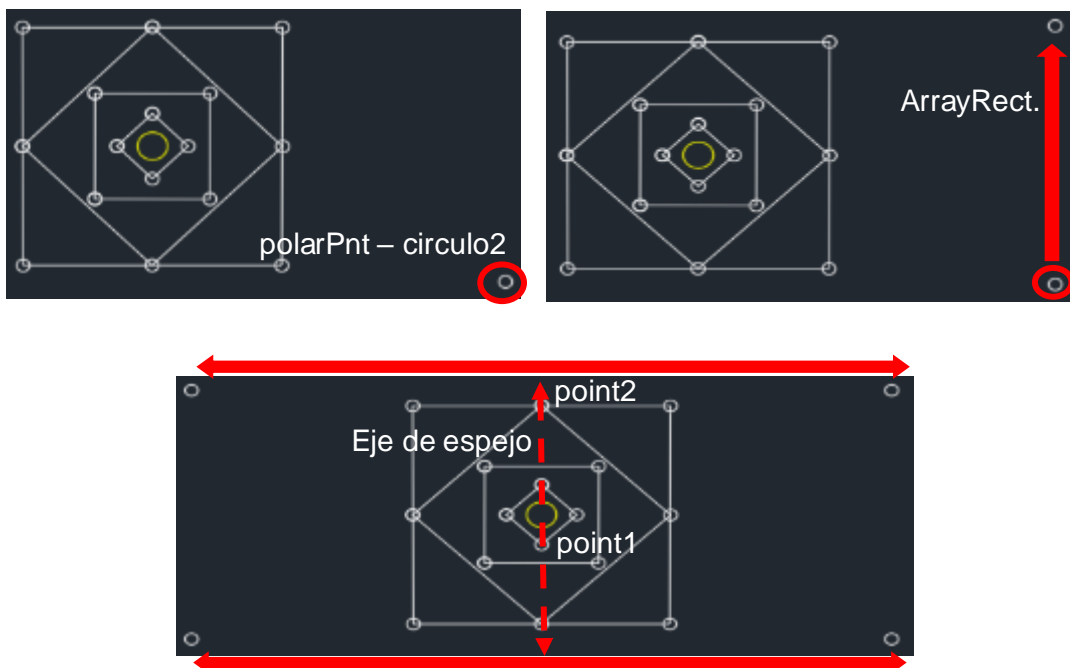


Figure 3-26. Diseño del Stopping B

```

'array rectangular
nofobjects = frmCalculo.valor13.Value
arrayrect = circulo2.arrayrectangular(nofobjects, 1, 1, frmCalculo.valor12.Value, 0.06, 0.06)

'aplicamos el mirror
For Each XLnCADObject In autocad.Application.ActiveDocument.ModelSpace
  If XLnCADObject.ObjectName = "AcDbCircle" Then
    Set referencia = XLnCADObject 'bloque array de referencia
    A = referencia.center(0)
    b = referencia.center(1)
    If A > T + 0.1 Then

      point1(0) = X: point1(1) = Y: point1(2) = 0
      point2(0) = X: point2(1) = Y + 5: point2(2) = 0
      ' Mirror the polyline
      Set mirrorObj = referencia.Mirror(point1, point2)
    End If
  End If
Next

```

R41

Rutina 3-34. Desarrollo del Stopping B

3.12.1.1.8 Diseño del Contour D

Antes de empezar con la rutina es importante resaltar el alcance del diseño que tendrá la periferia en su conjunto, para el caso de la Corona, esta incluye un código complementario que no se discutirá por lo extenso de su nomenclatura, esto no afecta en gran medida el modelo en general.

Se declararon las variables tal y como se muestra en la siguiente figura dentro del procedimiento llamado “periferia”.

```

Sub periferia() 'CORONA,ARRASTRE Y HASTIALES
Dim polarPnt1 As Variant, polarPnt2 As Variant
Dim XLnCADObject As AcadObject
Dim circulo2 As AcadCircle, circulo3 As AcadCircle
Dim pc(0 To 2) As Double
Dim pcc(0 To 5) As Double
Dim referencia As AcadCircle
Dim angle As Double, distance As Double
Dim arcoCentro(0 To 2) As Double
Dim point1(0 To 2) As Double
Dim point2(0 To 2) As Double

```

Figure 3-27. Declaración de variables para el desarrollo de la sección corona.

Para iniciar el diseño del ContourD fue necesario hacer un recorrido de puntos dentro del espacio modelo tomando como punto de partida el taladro inferior del tercer cuadrante, dicha referencia deberá cumplir con la condicional establecida para su ubicación. Una vez hallado nuestro punto de partida, este mantendrá sus coordenadas (x, y) dentro de las variables “A” y “b” y estas a su vez dentro de la variable “pc(0 to 2)”. Se empieza con el recorrido N-S utilizando el método PolarPoint la cual actúa con un ángulo de inclinación de -90° , una distancia equivalente al burden práctico del Lifter y la variable polarPnt1 como objeto puntual del resultado (R42).

Se repite la misma instrucción PolarPoint con ángulo de inclinación de 0° , una distancia igual al ancho de la labor dividido entre dos y la variable PolarPnt2 como objeto resultante del método aplicado. Finalmente se dibuja la entidad círculo de nombre “circulo2” tomando como referencia las coordenadas de la variable polarPnt2 y considerando un radio igual a la mitad del diámetro del taladro cargado (D2). De esta manera nos posicionaremos en la esquina inferior derecha de la labor (R43).

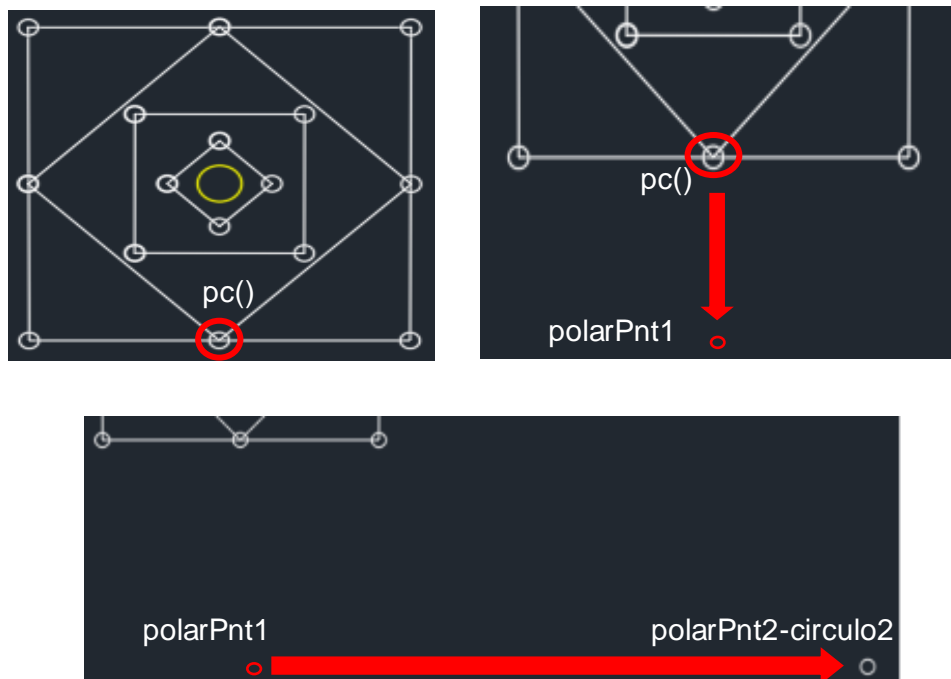


Figure 3-28. Punto de partida para el diseño de la corona

```

Sub periferia() 'CORONA,ARRASTRE Y HASTIALES
|
On Error Resume Next
For Each XLnCADObject In autocad.Application.ActiveDocument.ModelSpace
  If XLnCADObject.ObjectName = "AcDbCircle" Then
    Set referencia = XLnCADObject 'circulo de referencia
    A = referencia.center(0)
    b = referencia.center(1)
    altura = Y - (Val(frmCalculo.txt11.Value) + Val(frmCalculo.txt13.Value))
    If A > X - 0.1 And A < X + 0.1 And b < altura Then 'cambiar los valores por el del userform
      pc(0) = A: pc(1) = b: pc(2) = 0
      angle = -1.57 '90 hacia abajo
      distance = frmCalculo.valor2.Value 'burden practico arrastre
      polarPnt1 = autocad.Application.ActiveDocument.Utility.PolarPoint(pc, angle, distance)

      angle = 0 '0 hacia derecha
      distance = frmParametro.txtAncho.Value / 2
      polarPnt2 = autocad.Application.ActiveDocument.Utility.PolarPoint(polarPnt1, angle, distance)

      diametro2 = frmParametro.ComboBox5.Value
      Set circulo2 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt2, diametro2 / 2000)
      Exit For
    End If
  End If
Next XLnCADObject

```

Rutina 3-35. Desarrollo del punto de partida de la sección corona

Para dibujar el arco de la Corona fue relevante trabajar en base a polilíneas ya que solo estas tienen la particularidad de crear opciones de forma, como punto de partida elegimos el vértice inferior derecho de la labor (circulo2) declarando las variables “c” y “d” las cuales contendrán las coordenadas (x, y) del objeto circulo2, “ancholabor” que tomará el valor del control “txtAncho”, la variable “largolabor” que tomará el valor del control “txtAltura” menos 0.5, la variable “arco” del tipo AcadArc, la variable “rectaYarco” del tipo AcadPolyline la cual representa el resultado del método AddPolyline. (R44)

Para dicho método se tiene la variable “pcc(0 to 5)” la cual contiene las coordenadas de la polilínea que queremos insertar o dibujar desde un vértice superior derecho hacia el vértice superior izquierdo, se define la propiedad “Closed” como verdadero y se aplica el método “SetBulge” con index igual a uno y bulge igual a -0.25, de esa manera tendremos un arco perfecto que representa a la Corona (R44)

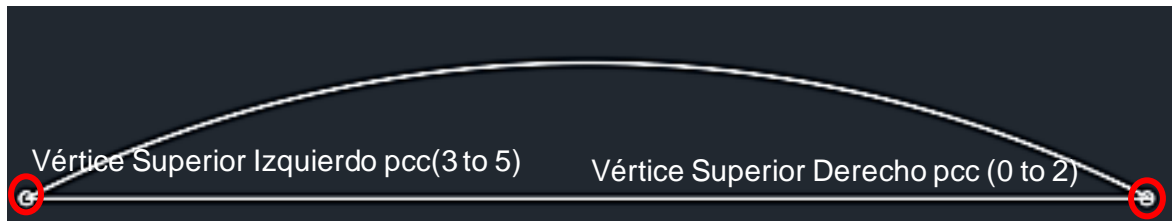


Figure 3-29. Diseño de arco-ContourD

A partir de este punto se aplica una rutina o procedimiento complementario la cual transforma dicho arco en círculos continuos equidistantes las polilíneas creadas se eliminan y se crean nuevas polilíneas las cuales reflejan el contorno de toda la labor, este procedimiento es de carácter universal por lo que no entraremos en detalle sobre su aplicación, el código estará disponible dentro del CD (R45).

```
'dibujamos una recta con el arco
Dim rectaYarco As AcadPolyline
Dim arco As AcadArc
ancholabor = frmParametro.txtAncho.Value
largolabor = frmParametro.txtAltura.Value - 0.5

c = circulo2.center(0)
d = circulo2.center(1)

pcc(0) = c: pcc(1) = d + largolabor 'vertice superior derecho
pcc(3) = c - ancholabor: pcc(4) = d + largolabor 'vertice superior izquierdo

Set rectaYarco = autocad.Application.ActiveDocument.ModelSpace.AddPolyline(pcc)
rectaYarco.Closed = True
rectaYarco.SetBulge 1, -0.25
```

(R44)

```
'dibidimos los segmentos
Dim MatrizObjetos As Variant
MatrizObjetos = rectaYarco.Explode
rectaYarco.Delete
'eliminamos la linea q no sirve
Dim linea As AcadLine
For Each Object In autocad.Application.ActiveDocument.ModelSpace
  If Object.ObjectName = "AcDbLine" Then
    Set linea = Object
    linea.Delete
  End If
Next

'convertimos el arco en polilinea
Call arc2lines
'introducimos los circulos
Call patharray5
```

(R45)

Rutina 3-36. Transformación de arco en círculos equidistantes

3.12.1.1.9 Diseño del Arrastre

Para el caso del arrastre se tomó como punto de referencia el vértice inferior derecho, el objetivo es dibujar los taladros de arrastre de izquierda a derecha es así que se hace una copia de la entidad “circulo2” mediante el método Mirror utilizando como eje de espejo las variables “point1” y “point2” las cuales mantendrán las coordenadas de línea. Se procede a dibujar un nuevo círculo en base al espaciamiento de las esquinas con ángulo de inclinación de 0°, una distancia equivalente al espaciamiento de las esquinas, la variable “diametro2” igual a D2, con punto de partida igual al vértice inferior izquierdo representado como pc(0 to 2) y la variable “polarPnt2” como el resultado del desplazamiento. Seguidamente se dibuja el círculo en base a los criterios mencionados con centro “polarPnt2” y un radio equivalente al D2 expresado en metros. Finalmente se aplica el array del tipo rectangular para multiplicar los objetos equidistantes entre sí en toda la línea del Arrastre (R46)

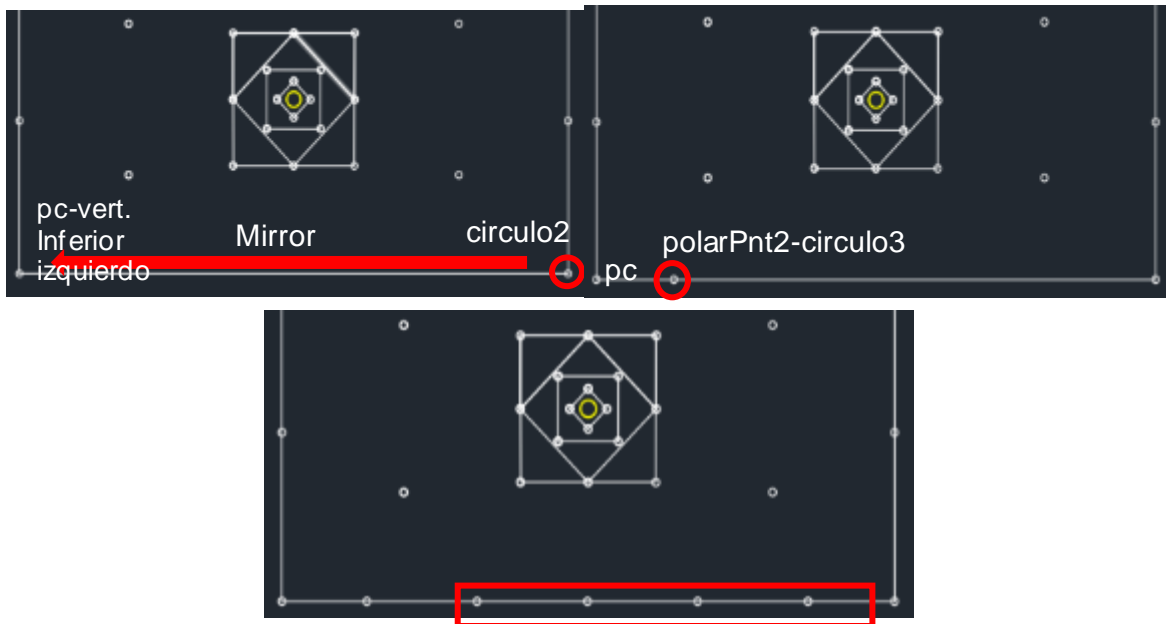


Figure 3-30. Diseño del Arrastre

```
'ARRASTRE
'mirror del circulo inferior derecho
point1(0) = X: point1(1) = Y: point1(2) = 0
point2(0) = X: point2(1) = Y - 6: point2(2) = 0

Set mirrorObj = circulo2.Mirror(point1, point2)

'dibujamos un nuevo circulo
pc(0) = c - ancholabor: pc(1) = d 'VERTICE INFERIOR IZQUIERDO
angle = 0 '0 hacia derecha
distance = frmCalculo.txtEsquinasArranque.Value 'espaciamiento esquinas
polarPnt2 = autocad.Application.ActiveDocument.Utility.PolarPoint(pc, angle, distance)
diametro2 = frmParametro.ComboBox5.Value / 1000
Set circulo3 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(polarPnt2, diametro2 / 2)
'array rectang
Dim arrayrect As Variant
filas = 1
Columnas = frmCalculo.valor4.Value - 2
arrayrect = circulo3.arrayrectangular(filas, Columnas, 1, frmCalculo.valor2.Value, frmCalculo.valor3.Value, 0.06)
```

(R46)

Rutina 3-37. Desarrollo del Arrastre

3.12.1.1.10 Diseño del Hastial

Como rutina final dibujamos los taladros de los Hastiales incluyendo la periferia en su conjunto, se declararon las variables “circuloinicialhastial” del tipo Double la cual mantendrá las coordenadas del vértice inferior derecho en adición al espaciamiento de entre taladros y la variable “circulo4” como objeto del método AddCircle. En base al punto de partida se dibuja un círculo inicial con una distancia igual al espaciamiento entre taladros, seguidamente se utiliza el método Array Rectangular del objeto “circulo4” con numero de filas igual al número de taladros del Hastial, número de columnas igual a uno, número de labels igual a uno, la separación entre filas igual al espaciamiento entre taladros, la separación entre columnas igual al burden práctico y una separación entre labels de 0.06 (R47)



Figure 3-31. Diseño del Hastial derecho

```
'HASTIALES
  'dibujo un primer círculo en base al espaciamiento inicial
  Dim circuloinicialhastial(0 To 2) As Double
  Dim circulo4 As AcadCircle
  circuloinicialhastial(0) = c: circuloinicialhastial(1) = d + frmCalculo.valor30.Value
  Set circulo4 = autocad.Application.ActiveDocument.ModelSpace.AddCircle(circuloinicialhastial, diametro2 / 2)
'array rectangular
filas = frmCalculo.valor31.Value
Columnas = 1
arrayrect = circulo4.arrayrectangular(filas, Columnas, 1, frmCalculo.valor30.Value, frmCalculo.valor29.Value, 0.06)
```

(R47)

Rutina 3-38. Desarrollo del Hastial derecho

Por último, se aplica el método Mirror para hacer una copia del resultado que se tiene del hastial derecho, esto para cada uno de los taladros que se encuentren en la sección Hastial partiendo de la entidad “circulo4”, mediante la instrucción For Each es posible encontrar aquellas entidades del tipo AcadCircle las cuales cumplan con la condición establecida de ubicación, el eje de espejo usará las coordenadas de inicio y fin de las variables “point1” y “point2” respectivamente. Finalmente se dibuja el contorno restante utilizando como coordenadas los vértices, tanto superior derecho, superior izquierdo, inferior derecho e inferior izquierdo, se comienza dibujando la línea del hastial derecho seguido del Arrastre y Hastial izquierdo, el método usado resulta de los mismos criterios usados en la sección Corona (R48).

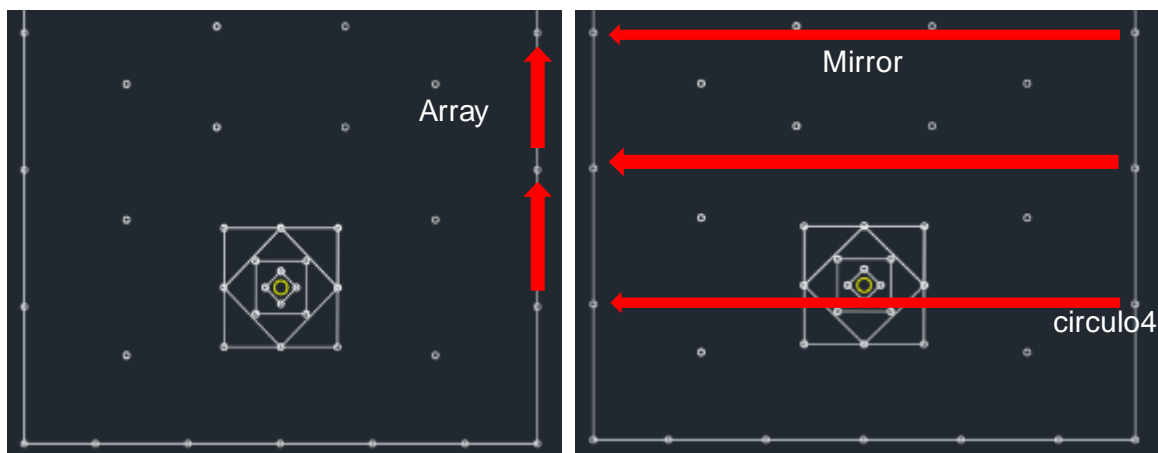


Figure 3-32. Diseño del Hastial izquierdo


```

'mirror
For Each XLnCADObject In autocad.Application.ActiveDocument.ModelSpace
    If XLnCADObject.ObjectName = "AcDbCircle" Then
        Set referencia = XLnCADObject 'circulo de referencia
        A = referencia.center(0)
        b = referencia.center(1)

        If A > X And b > d And b < d + Val(largolabor) Then
            point1(0) = X: point1(1) = Y: point1(2) = 0
            point2(0) = X: point2(1) = Y + 6: point2(2) = 0
            Set mirrorObj = referencia.Mirror(point1, point2)
        End If
    End If
Next

'DIBUJAMOS LAS LINEAS DEL CONTORNO RESTANTE
Dim pccc(0 To 5) As Double
pccc(0) = c: pccc(1) = d 'vertice inferior derecho
pccc(3) = c: pccc(4) = d + largolabor 'vertice superior derecho
Set rectaYarco = autocad.Application.ActiveDocument.ModelSpace.AddPolyline(pccc)
Dim pcccc(0 To 8) As Double
pcccc(0) = c: pcccc(1) = d 'vertice inferior derecho
pcccc(3) = c - ancholabor: pcccc(4) = d 'vertice inferior izquierdo
pcccc(6) = c - ancholabor: pcccc(7) = d + largolabor 'vertice superior izquierdo
Set rectaYarco = autocad.Application.ActiveDocument.ModelSpace.AddPolyline(pcccc)
End Sub

```

(R48)

Rutina 3-39. Desarrollo del Hastial izquierdo

CAPITULO IV

PROCESAMIENTO ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

Caso Práctico

4.1 Ingreso y Validación de Parámetros Generales

Primeramente, activamos el formulario “Diseño de Malla”, para ello debemos hacer click en el botón “Run” el cual nos abrirá la ventana que se muestra en la figura 4.1.

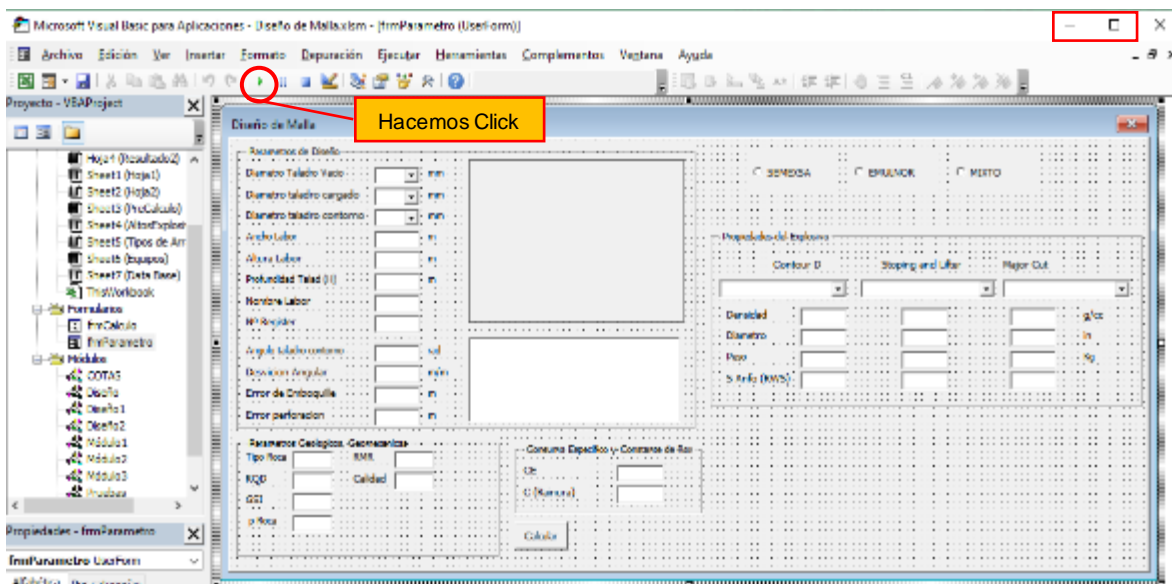
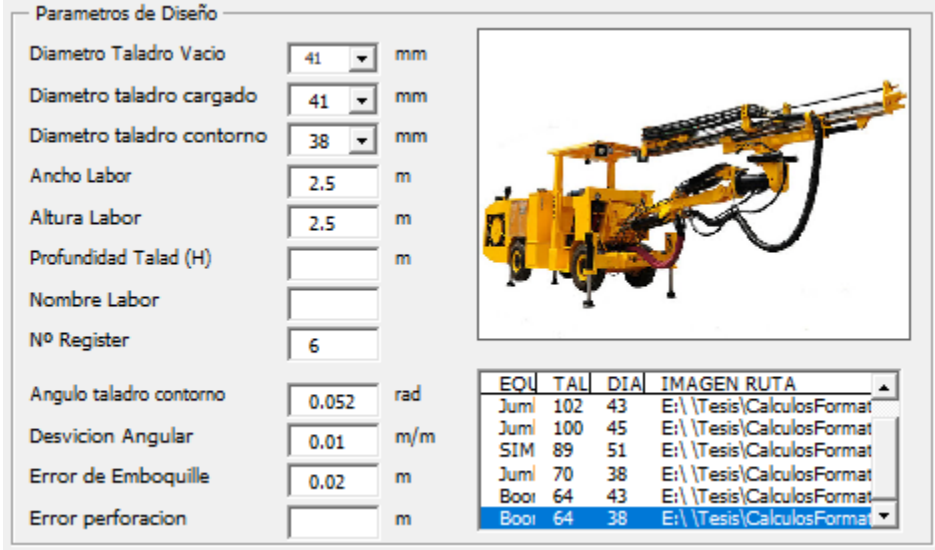


Figure 4-1. Arranque del formulario “Diseño de Malla”

4.1.1 Parámetros de Perforación y Voladura

Previo al cálculo debemos definir los parámetros de diseño comenzando por el diámetro de los taladros a perforar, un diámetro para la corona, un diámetro para el Stopping, y otro para el Arranque. Elegimos dentro del cuadro de lista el diámetro deseado que en este caso sería la última fila obteniendo un diámetro mínimo y máximo según el tipo de equipo seleccionado, para el presente caso se introdujo valores de forma manual, un diámetro del taladro de alivio de 41mm, un diámetro de los taladros cargados de 41mm y un diámetro de contorno de 38mm. Seguidamente se definen las dimensiones de la labor, al tratarse de una veta con potencia 0.70m fue conveniente elegir una dimensión de menor escala que se ajuste a la realidad del prospecto, considerando una desviación de perforación de 0.01m, un ángulo de contorno de 0.052 rad y una desviación de empaque de 0.02m. La longitud del arco de coronación por defecto es de 0.5m.



The screenshot shows a software interface for defining design parameters. On the left, there are several input fields with dropdown menus and text boxes. On the right, there is a 3D image of a yellow drilling rig. Below the image is a table with columns for equipment type, diameter, and image path.

Parametro	Valor	Unidad
Diametro Taladro Vacio	41	mm
Diametro taladro cargado	41	mm
Diametro taladro contorno	38	mm
Ancho Labor	2.5	m
Altura Labor	2.5	m
Profundidad Talad (H)		m
Nombre Labor		
Nº Register	6	
Angulo taladro contorno	0.052	rad
Desvicion Angular	0.01	m/m
Error de Emboquille	0.02	m
Error perforacion		m

EQU	TAL	DIA	IMAGEN RUTA
Jum	102	43	E:\Tesis\CalculosFormat
Jum	100	45	E:\Tesis\CalculosFormat
SIM	89	51	E:\Tesis\CalculosFormat
Jum	70	38	E:\Tesis\CalculosFormat
Boor	64	43	E:\Tesis\CalculosFormat
Boor	64	38	E:\Tesis\CalculosFormat

Figure 4-2. Selección de los diámetros de perforación y llenado de los parámetros de diseño

Para la voladura se utilizará emulsión explosiva encartuchada siendo estas: Emulnor 1000 1x7", Emulnor3000 1x7" y Emulnor5000 1x8", simplemente elegimos el tipo de explosivo a utilizar y obtendremos las propiedades más relevantes de cada uno de ellos tanto para Corona, Stopping y Arranque.

SEMEXSA
 EMULNOR
 MIXTO

Propiedades del Explosivo

	Contour D	Stopping and Lifter	Major Cutting	
	EMULNOR 1000 1" x 7"	EMULNOR 3000 1" x 7"	EMULNOR 5000 1" x 8"	
Densidad	1.13	1.14	1.16	g/cc
Diametro	25.4	25.4	25.4	in
Peso	0.10180	0.10270	0.11943	Kg
S Anfo (RWS)	0.72816	0.81452	0.87536	

Figure 4-3. Selección del tipo de explosivo por cada sección

4.1.2 Parámetros Geológicos-Geomecánicos

Con los resultados obtenidos de la evaluación Geomecánica del prospecto minero Monserrat completamos los parámetros Geológicos-Geomecánicos, teniendo así un RMR de 54, un RQD de 75.4 y una densidad de 2.7 T/m3.

Parametros Geologicos, Geomecanicos

Tipo Roca		RMR	54
RQD	75,4	Calidad	
GSI	49		
p Roca	2.7		

Figure 4-4. Ingreso de los valores RMR, RQD y Densidad de la roca

Hacemos click en cualquier área del Formulario y obtendremos los valores de GSI igual a 49, un consumo específico de 0.41 kg/m3 y una constante de roca de 0.367 Seguidamente hacemos click en el botón **Calcular** la cual nos llevará al siguiente formulario.

Frame3	B Max (m)	B Practico (m)	Ah (m)	N Tladros	q (Kg/m)	Kg/Talad	Kg	Cart/Talad
Primer Cuadrante	0.0697	0.0697	4.9073210	4	0.5874826	0.8335204	3.33	7
Segundo Cuadrante	0.1474732	0.1124732	8.3773210	4	0.5773536	0.8191494	3.28	8
Tercer Cuadrante	0.2019317	0.1669317	0.1688003	4	0.5773536	0.8191494	3.28	8
Cuarto Cuadrante	0.2718540	0.2368540	0.3059395	4	0.5773536	0.8191494	3.28	8

Figure 4-6. Resultado del cálculo de secciones

Para terminar con el resto del cálculo procedemos a darle click en el botón “Calcular” obteniendo de esta manera los valores de Burden práctico, factor de carga lineal y el número de cartuchos por taladro.

4.3 Procesamiento y Análisis de Cálculo de Secciones (ContourD)

Una vez que hayamos cambiado de página se nos mostrará un mensaje preventivo el cual nos indicará que completemos todos los criterios de la Corona definiendo previamente el tipo de explosivo a utilizar y el reductor de carga correspondiente al factor de carga lineal en columna, equivalente a 0.7. Al cambiar de página tendremos por defecto los valores preestablecidos de: Eficiencia de voladura, diámetro de los taladros para cada sección, avance por disparo, constantes, relación espaciamento burden, Lcf, Lcc y Taco

CUT | CONTOUR D | LIFTER | STOPING B | STOPING C | WALLS | RESULTADOS

Frame1

Eficiencia Voladura: 0.95

Diámetro Talad Cont: 38 mm

Avance por Disparo: 1.73736 m

K: 16

S/B: 0.8

Frame2

Long Carga Fondo: 0.28956 m

Taco: 0.38 m

Columna: 1.0678 m

EMULNOR 1000 1" x 7" EMULNOR 3000 1" x 7"

Peso: 0.1018 Kg

Diámetro: 25.4 mm

Densidad: 1.13 g/cc

Frame7

q Minima:

q del cartucho elegido:

Frame6

B max: m Carga Fondo: Kg

B Practico: m Reductor: 0.7

Espaciamento: m Carga Columna: Kg

N Talad/Corona: Carga/Taladro: Kg/Talad

N Cart/Talad: Carga Total: Kg

Figure 4-7. Selección del explosivo y uso del reductor de carga

Seguidamente daremos click en “Calcular” y obtendremos el resto de valores , principalmente el “B practico” y “Espaciamiento” los cuales definen el diseño de malla.

The screenshot shows a software interface with several input fields and a 'Calcular' button. The 'LIFTER' tab is selected. The 'Frame 6' section has the following values: B max (0.75 m), B Practico (0.6 m), Espaciamiento (0.6 m), N Talad/Corona (6), and N Cart/Talad (3). The 'Frame 7' section has q Minima (0.12996) and q del cartucho elegido (0.5722891). The 'Calcular' button is highlighted with a red arrow.

Figure 4-8. Resultado del modelo - sección Corona

4.4 Procesamiento y Análisis de Cálculo de Secciones (Lifter-StopingB-StopingC-Walls)

A partir del Arrastre se tendrá resultados de B practico, Espaciamiento y número de cartuchos por taladro, simplemente hacemos click en el botón **Calcular**.

The screenshot shows a software interface with several input fields and a 'Calcular' button. The 'LIFTER' tab is selected. The 'Frame 6' section has the following values: B max (0.8621603 m), B Practico (0.75 m), Espaciamiento (0.66 m), N Talad/Arrastre (5), and N Cart/Talad (7). The 'Frame 2' section has Long Carga Fondo (0.57912 m), Taco (0.41 m), and Columna (0.74824 m). The 'Calcular' button is highlighted with a red arrow, and red arrows point from it to the 'B Practico', 'Espaciamiento', and 'Carga Total' fields in the 'Frame 6' section.

Figure 4-9. Resultados del modelo - sección Arrastre

CUT	CONTOUR D	LIFTER	STOPING B	STOPING C	WALLS	RESULTADOS
Frame 1						
Eficiencia Voladura	0.95					
Diametro Talad			m			
Avance por Disparo	1.73736		m			
Factor f	1.45					
S/B	1.25					
Factor Roca Correg(C')	0.4602892					
B						
		0.7560155				
Frame 2						
Long Carga Fondo	0.57912		m			
Taco	0.41		m			
Columna	0.74824		m			
Frame 6						
B max	0.6757114		m		Carga Fondo	0.3343570 Kg
B Practico	0.64		m		Carga Columna	0.4319991 Kg
Espaciamiento	0.84		m		Carga/Taladro	0.7663561 Kg/Talad
N Talad/AyudaHast	2				Carga Total	1.5327123 Kg
N Cart/Talad	7					

Figure 4-10. Resultados del modelo - Stoping B

CUT	CONTOUR D	LIFTER	STOPING B	STOPING C	WALLS	RESULTADOS
Frame 1						
Eficiencia Voladura	0.95					
Diametro Talad			m			
Avance por Disparo	1.73736		m			
Factor f	1.25					
S/B	1.25					
Factor Roca Correg(C')	0.4536668					
B						
		0.8142536				
Frame 2						
Long Carga Fondo	0.57912		m			
Taco	0.41		m			
Columna	0.74824		m			
Frame 6						
B max	0.7330560		m		Carga Fondo	0.3343570 Kg
B Practico	0.70		m		Carga Columna	0.4319991 Kg
Espaciamiento	0.92		m		Carga/Taladro	0.7663561 Kg/Talad
N Talad/AyudaCoron	2				Carga Total	1.5327123 Kg
N Cart/Talad	7					

Figure 4-11. Resultados del Stoping C

CUT	CONTOUR D	LIFTER	STOPING B	STOPING C	WALLS	RESULTADOS
Frame1				Frame2		
Eficiencia Voladura	0.95		Long Carga Fondo			0.28956 m
Diametro Talad			Taco			0.41 m
Avance por Disparo	1.73736 m		Columna			1.0378 m
Factor f	1.2					
S/B	1.5					
Factor Roca Correg(C)	0.4599693					
B	0.7586360					
Frame6						
B max	0.6782893 m		Carga Fondo	0.3343570		Kg
B Practico	0.64 m		Carga Columna	0.4319991		Kg
Espaciamiento	0.85 m		Carga/Taladro	0.7663561		Kg/Talad
N Talad/Hastiales	2		Carga Total	1.5327123		Kg
N Cart/Talad	7					

Figure 4-12. Resultados del modelo - Hastiales

4.5 Reporte General

Dentro de la pestaña “RESULTADOS” mediante el comando “**Generate Result**” obtendremos el resumen general de explosivos, carga, accesorios, número de taladros y secuencia de arranque. Al momento de hacer click sobre el comando mencionado se nos abrirá una nueva pestaña en la hoja de trabajo del libro actual de nombre “Resultado” seguido del número de registro que hemos ingresado, aquí podremos visualizar el resumen de todos los cálculos realizados.

En cuanto a los accesorios de voladura, la programación genera como resultado un referencia mas no valores concretos esto dependerá del análisis técnico del usuario.

Resultado de la cantidad de Explosivo

SECUENCIA DE SALIDA	DISTRIBUCION	TALADROS	EMULNOR 1000 1" x 7"			EMULNOR 3000 1" x 7"			EMULNOR 5000 1" x 8"		
			Nº Cartuchos/Talad	Nº Cartuchos	Kg Total	Nº Cartuchos/Talad	Nº Cartuchos	Kg Total	Nº Cartuchos/Talad	Nº Cartuchos	Kg Total
1	Arranque I	4							7	28	3.33
2	Arranque II	4				8	32	3.28			
3	Arranque III	4				8	32	3.28			
4	Arranque IV	4				8	32	3.28			
5	Ayuda Hastial	4				7	28	1.5327124			
6	Ayuda Corona	2				7	14	1.5327124			
7	Hastial	4				7	28	1.5327124			
8	Corona	6	3	18	1.58						
9	Arrastre	5				7	35	3.8317809			

ACCESORIOS DE VOLADURA

FANELES	37	pza
PENTACORD	20	m
CARMEK	2	unid
MECHA RAPIDA	0.1	m
CAÑAS DE BAMBU	6	unid
TACO DE ARCILLA	16	unid
TUBOS PVC 1 1/4 X 4m	7	unid

Total Taladros Alivio	1
Total Taladros Cargados	37
Nº Cartuchos EMULNOR 1000 1" x 7"	18
Nº Cartuchos EMULNOR 3000 1" x 7"	201
Nº Cartuchos EMULNOR 5000 1" x 8"	28
Total Kg EMULNOR 1000 1" x 7"	1.58
Total Kg EMULNOR 3000 1" x 7"	18.269918
Total Kg EMULNOR 5000 1" x 8"	3.33

2 PreCalculo AltosExplosivos Tipos de Arranque Equipos Data Base Result1 Resultado1 Resultado2 Resultado5 **Resultado6**

Figure 4-13. Reporte de Explosivos y Accesorios

Dentro de la pestaña "RESULTADOS" al hacer click en el botón "AutocadActiveX" se nos abrirá una ventana preguntando si deseamos generar el modelo en AutoCAD, confirmamos haciendo click en "Si", definimos las coordenadas (x, y) de 20x20 y confirmamos si deseamos guardar el diseño generado, finalmente como resultado final del modelo deseado obtendremos una visualización completa del diseño de malla, de fácil interpretación, a escala y con un nivel de confianza del 99%.

START POINT X

Defina el punto de partida en X

Aceptar Cancelar

20

START POINT Y

Defina el punto de partida en Y

Aceptar Cancelar

20

Figure 4-14. Punto de partida X, Y

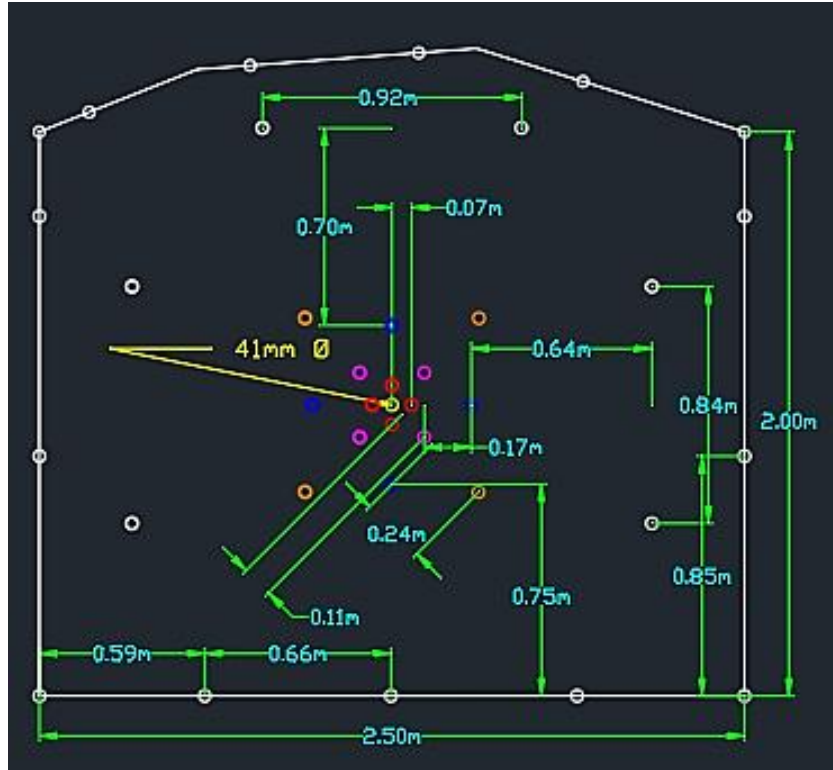


Figure 4-15. Resultado final - Diseño de malla 2.5x2.5m

Tabla 4-1. Leyenda del diseño de malla

Color	Descripción
●	Primer Cuadrante C1
●	Segundo Cuadrante C2
●	Tercer Cuadrante C3
●	Cuarto Cuadrante C4
●	Resto de Secciones
●	Taladros Vacíos
●	Cotas

4.7 Discusión

- I. VBA es un producto final ya que no puede crear compilaciones por separado para poder crear ejecutables y distribuirlos entre clientes y equipos de trabajo, por lo que si se desea distribuir una automatización de office (Word, Power Point, Excel, etc.); se tiene que compartir en si el archivo que contiene dicha programación.
- II. El definir los tipos de datos de las variables, nos permiten hacer un uso adecuado de la memoria en nuestra computadora, El tipo de datos Variant se puede utilizar en la mayoría de tipos de datos, lo que aporta gran flexibilidad, en ciertos casos mantiene la lógica interna de la rutina, pero no es conveniente hacer un uso excesivo de este ya que consume demasiada memoria.
- III. Se hicieron dos ensayos principales sin considerar el ensayo de picota ya que al ser un método organoléptico va depender mucho de las sensaciones que el usuario pueda tener en campo lo cual no ofrece garantía de los resultados a obtener, se optó por métodos más trascendentales.
- IV. La idea de una voladura eficiente radica básicamente en dos criterios: Menor concentración de carga con igual o mayor número de taladros o la reducción del número de taladros con igual concentración lineal de carga

Dicho esto, la teoría de (López Jimerno, 2005 pág. 255) muestra un factor de carga lineal dependiente del diámetro de producción y del burden máximo, esto genera cierta inconsistencia ya que a mayor burden máximo, menor será el número de taladros y el factor de carga lineal será mucho mayor; del mismo modo, a mayor diámetro de producción mayor será el factor de carga lineal, incumpliendo los criterios antes mencionados.

Por esa razón se opta por usar la fórmula propuesta en la sección *Factor de Carga* la cual establece un factor de carga lineal dependiente de las características del explosivo, de esa manera se tendrá una concentración estándar para cada área de voladura en donde la cantidad de explosivos dependerá únicamente de la calidad del macizo rocoso y por ende de la distribución y número de taladros.

CONCLUSIONES

1. Se ha logrado optimizar el proceso de diseño gracias a la automatización del modelo matemático Holmberg reduciendo considerablemente el tiempo empleado en esta actividad, no solo a nivel de diseño sino también a nivel de carga.
2. El diseño de malla automatizado mediante lenguaje de programación ha sido satisfactorio al comprobarse la similitud de los resultados de carga y número de taladros especialmente valores de burden y espaciamiento, con un formato atractivo, a escala, de fácil interpretación y resultados confiables.
3. Como se demuestra la validación de los resultados del programa alcanza una semejanza con diseños de experiencias anteriores es así que a nivel de carga se destaca la similitud, mientras que a nivel de diseño se obtienen resultados conservadores debido a que la hipótesis considera que la distribución de taladros actúa de forma exponencial.

RECOMENDACIONES

- ✓ El modelo matemático Holmberg genera estimaciones de burden y espaciamiento en función del diámetro de los taladros. Es así, que para obtener resultados confiables el usuario deberá validar e interpretar correctamente los resultados finales tanto de cálculo como de diseño considerando aspectos empíricos propios del tema en estudio.
- ✓ El diseño de malla depende fundamentalmente de una buena evaluación geomecánica. Por lo tanto, con el fin de obtener resultados satisfactorios los parámetros de RMR, GSI, RQD y Densidad deben ser confiables y objetivos.
- ✓ Trabajar únicamente con la base de datos propuesta con el fin de evitar cualquier tipo de error en el sistema.
- ✓ Para que el sistema funcione correctamente es necesario tener un sistema operativo de 64 bits en Windows 7/10 y la instalación de AutoCAD 2020.
- ✓ El sistema debe ejecutarse de forma rápida y dinámica. Por lo tanto, La declaración de variables debe estar enfocada al menor consumo de memoria RAM con el fin de que el sistema ocupe el menor espacio posible.

BIBLIOGRAFIA

1. **DE LA CRUZ MERCADO, Plenio.** Horizonte Minero_Revista internacional de minería y energía. *Innovación en minería subterránea*. Fecha de Consulta: 29 de 11 de 2017. Disponible en: <http://www.horizonteminero.com/2017/11/29/innovacion-mineria-subterránea/>.
2. **PÉREZ MONTERO, Eilen Lorena y HERNÁNDEZ PÉREZ, Flor de María.** *La Programación Orientada a Objetos Facilidad para Crear*. Edición 13, Neiva, Colombia : Revista de Investigación de la Corporación Universitaria Comfacauca, 2019, Investigación Tecnología y Ciencia, págs. 56-100. 1909-5775.
3. **RAMIREZ RUIZ, Candido.** *Desarrollo de una Aplicación en Programación Orientada a Objetos VBA en Excel, para el diseño de Plantas Potabilizadoras de Filtración Directa*. Ingeniería, INSTITUTO TECNOLÓGICO DE POCHUTLA. San Pedro Pochutla : s.n., 2016.
4. **SINGH, Shubham y RAWAT, Roopak.** *Integration of CAD model with Calculation Sheet using Excel VBA*. Delhi, India : Journal of Engineering Design and Technology, 2021, International Research Journal of Engineering and Technology (IRJET), Vol. Volumen VIII. 2395-0056.
5. **MATEO MENDOZA, Jhordan, y otros.** *Modelo Matemático de Pearse y Holmberg para Reducir la Zona de Daños en Labores Horizontales de Minería Subterránea*. Lima : LACCEI International Multi-Conference for Engineering, Education, and Technology:, 2020. 978-958-52071-4-1/2414-6390.
6. **MAYTA FLORES, Miguel Angel.** *Implementación de un Programa para los Procesos de Analisis y Diseño de Zapatas*. Juliaca : s.n., 2019.
7. **INGEMMET.** GEOCATMIN. Disponible en: <https://geocatmin.ingemmet.gob.pe/geocatmin/>.
8. **OSINERGMIN.** *Guía de criterios geomecánicos para diseño, construcción, supervisión y cierre de labores*. Lima : Inversiones Jakob S. A. C., 2017. 20562618008.
9. **LÓPEZ JIMERNO, Carlos.** *Manual de Perforación y Voladura de Rocas*. Madrid : Gráficas Arias Montano, S. A., 2005. 8496140156.
10. **MARTINEZ CANELO, Miriam.** ¿Qué son los paradigmas de programación? *profile*. [En línea] 9 de Junio de 2020. Disponible en: <https://www.youtube.com/watch?v=K864xaCk4FE>.

11. **BRIONES GIL, Laura.** Programación de Macros de Excel Utilizando Visual Basic para Aplicaciones. 2017, pág. 52.
12. **AMELOT, Michèle.** *VBA Excel 2016.* s.l. : Editions ENI, 2016. pág. 462.
13. **AUTODESK.** *AutoCad 2007: ActiveX and VBA Developer's Guide.* [ed.] Arthur D. Applegate. San Rafael, CA : Pantone, Inc., 2006.
14. **GONZALES, Luis, y otros.** *Ingeniería Geológica.* Madrid : Pearson Educación S. A., 2002. 9788420531045.
15. **HERNÁNDEZ GUTIÉRREZ, Luis Enrique y SANTAMARTA CEREZAL, Juan Carlos.** *Ingeniería Geológica en Terrenos Volcánicos.* Madrid : s.n., 2015. 978-84-608-5072-4.
16. **RODRIGUEZ ORTIZ, J. M., y otros.** *Manual de Ingeniería Geológica.* Madrid : Pearson Educación SA, 2002. 84-205-3104-9.
17. **SCHWARZ DIAZ, Max.** *Guía de Referencia para la Elaboración de una Investigación Aplicada.* Lima : Universidad de Lima, Facultad de Ciencias Empresariales y Económicas, 2017.
18. **HERNANDEZ SAMPIERI, Roberto, FERNÁNDEZ COLLADO, Carlos y BAPTISTA LUCIO, María del Pilar.** *Metodología de la Investigación.* Mexico : McGRAW-HILL, 2010. 978-607-15-0291-9.
19. **documentation.HELP.** Modelo de objetos de AutoCAD. Disponible en: <https://documentation.help/AutoCAD-ActiveX-es/WS1a9193826455f5ff1a32d8d10ebc6b7ccc-6cb4.htm>.
20. **CÓRDOVA ROJAS, Nestor David.** *Geomecánica en el Minado Subterráneo.* Lima : s.n., 2008.
21. **MUNICIPALIDAD PROVINCIAL DE JAUJA.** MAPAS DE JAUJA. *JAUJA MI PERU.* [En línea] [Citado el: 15 de 09 de 2021.] Disponible en: <https://www.jaujamiperu.com/>.

ANEXOS

ANEXO 1

EQUIPOS DE LABORATORIO

Cortador de Muestras



Cortador de Bloques



Equipo para Extracción de Testigos



Vista General del Laboratorio



ANEXO 2

MAPEO GEOMECANICO

MAPEO GEOMECANICO RMR (MAPEO LINEA DE DETALLE)										SOCIEDAD MINERA		PROSPECTO MONSERRAT					
										RESPONSABLE		BACHILLER JIMENEZ CASIMIRO ALEXANDER					
ZONA										ESTACION		ESTE	NORTE	ELEVACION			
NIVEL		4433								1		460220	8712680	4433			
LABOR										2		460205	8712710	4433			
FECHA		18/11/2018								3							
Distancia linea (m)	TIPO DE ESTRUCTURA	DIP DIR	DIP	Quantity	#Fracturas por metro lineal	TIPO DE ROCA	GEOMETRIA	JUNTA		1		2		3			
								NOMBRE		Estación 680 NE		Galería 710 SE					
								RC	RQD	Espaciamiento	Quantity	CONDICION DISCONTINUIDADES					
										Persistencia (m)	Apertura (mm)	Rugosidad	Relleno	Alteracion	Agua		
J1	D	189	40	1	9	Diorita	p	R3	75.4	0.15	2	<1m	cerrada	Ligeramente rugoso	N	Manchas oxido	H
	D	177	25	3	9	Diorita	p	R3	75.4	0.14	4	<1m	cerrada	Ligeramente rugoso	N	Manchas Oxido	H
	D	217	29	1	9	Diorita	p	R3	75.4	0.16	2	<1m	cerrada	Ligeramente rugoso	N	Manchas Oxido	H
	D	202	30	1	9	Diorita	p	R3	75.4	0.11	1	<1m	cerrada	Ligeramente rugoso	N	Manchas Oxido	H
	D	221	29	2	9	Diorita	p	R3	75.4	0.09	2	<1m	cerrada	Ligeramente rugoso	N	Manchas Oxido	H
	D	239	41	1	9	Diorita	p	R3	75.4	0.13	2	<1m	cerrada	Ligeramente rugoso	N	Manchas Oxido	H
	D	225	41	2	9	Diorita	p	R3	75.4	0.1	1	<1m	cerrada	Ligeramente rugoso	N	Manchas Oxido	H
	D	208	34	1	9	Diorita	p	R3	75.4	0.08	1	<1m	cerrada	Ligeramente rugoso	N	Manchas Oxido	H
J2	D	180	17	2	9	Diorita	p	R3	75.4								
	D	194	38	1	9	Diorita	p	R3	75.4								
	D	125	23	1	10	Diorita	p	R3	74	0.1	3	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
	D	116	28	1	10	Diorita	p	R3	74	0.06	1	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
	D	115	31	1	10	Diorita	p	R3	74	0.13	3	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
	D	118	24	5	10	Diorita	p	R3	74	0.12	3	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
	D	119	24	4	10	Diorita	p	R3	74	0.08	4	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
	D	118	33	2	10	Diorita	p	R3	74	0.09	3	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
	D	111	34	1	10	Diorita	p	R3	74	0.05	1	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
	D	113	26	1	10	Diorita	p	R3	74	0.11	2	1-2m	0.5-1m	Rugoso	O	Manchas Oxido	H
D	110	21	2	10	Diorita	p	R3	74									
D	110	22	2	10	Diorita	p	R3	74									

TIPO DE ESTRUCTURA			
D	Diaclasa	C	Contacto litologico
SD	Set diaclasa	ES	Estratificacion
F	Falla	VT	Veta
ZC	Zona de cizalla	DX	Dique

AGUA	
S	Seco
H	Humedo
M	Mojado
G	Goteo suave
F	Flujo constante










RELLENO	
N	Sin relleno
Ca	Calcita
O	Oxidos
A	Arcilla
Si	Silice
Li	Limo

TIPO DE ROCA	
A	Andesita
B	Volcanico
C	Calcita
D	Diorita

GEOMETRIA	
P	Planar
S	Sinuosa
D	Dentado

ALTERACION	
Si	Silisificacion
Pr	Propilizacion
Ar	Argilizacion
Se	Serisificacion
P	Pintizacion
A	Alta
M	Moderada
S	Suave

ANEXO 3 - PRINCIPALES PROPIEDADES FISICAS

Especimen 1	Especimen 2	Especimen 3
		
		
		

ANEXO 4 - FORMATO DE FUNCIONES RMR89 & Q BARTON

SISTEMA RMR 89 (Rock Mass Rating)																
PARAMETROS		RANGO									CLASIFICACION		PUNTAJON			
CONDICION DE JUNTAS																
Picota	R6		R5		R4		R3	R2	R1	R1	CLASIFICACION	PUNTAJON				
Resistencia a la compresion Simple UCS	> 250 Mpa	15	100-250 Mpa	12	50-100 Mpa	7	25-50 Mpa	4	25 M	2	5MP	1	1MP	0	25-50 Mpa	4
RQD (%)	90-100	20	75-90	17	50-75	13	25-50	8	<25			3	75-90	17		
Espaciamiento Discontinuidades	> 2m	20	0.6-2m	15	200-600mm	10	60-200mm	8	<60mm			5	60-200mm	8		
Persistencia	"< 1m"	6	"1-3m"	4	3-10m	2	10-20m	1	> 20m			0	"1-3m"	4		
Abertura	"Ninguna"	6	< 0.1mm	5	0.1-1mm	4	1-5mm	1	> 5mm			0	1-5mm	1		
Rugosidad	"Muy Rugosa"	6	Rugosa	5	Ligeramente Rugosa	3	Lisa	1	Superficies Pulidas			0	Ligeramente Rugosa	3		
Relleno	"Ninguna"	6	Duro <5mm	4	Duro >5mm	2	Blando <5mm	2	Blando > 5mm			0	Duro <5mm	4		
Alteracion	"Inalterada"	6	Ligeram. Meteorizad	5	Moderadam. met.	3	Altamente met.	1	Descompuesta			0	Ligeram. Meteorizad	5		
AGUA SUBTERRANEA (CONDICIONES GENERALES)																
Flujo por cada 10m de longitud del tunel (Lt/min)	"Seco"	15	"Humedo"	10	"Mojado"	7	"Goteo"	4	"Flujo"			0	"Humedo"	10		
Presion de agua en la diacla. / Tension principal r	Ninguna	15	< 10	10	10 a 25	7	25 a 125	4	> 125			0	"Humedo"	10		
	0	15	< 0.1	10	0.1 a 0.2	7	0.2 a 0.5	4	> 0.5			0	"Humedo"	10		
AJUSTE DE LA ORIENTACION DE LAS DISCONTINUIDADES EN TUNELES Y GALERIAS																
Efecto de la orientacion del rumbo y buza	Rumbo perpendicular al eje del tunel_direccion en el sentido buz. 20°-45°						Interpretacion de la Orientacion del rumbo				"Favorable"	-2				
DESCRIPCION DE LA VALORACION																
RMR (Corregido)	100-81		80-61		60-41		40-21		< 21			54				
Clase	I		II		III		IV		V			III				
Descripcion	Muy Buena		Buena		Regular		Mala		Muy mala			Regular				
Tiempo medio de sostenimiento/L	20 años, claro de 15m		1 año, claro de 10m		1 semana, claro de 5m		10 horas, claro de 2.5m		30 minutos, claro de 1m			1 semana, claro de 5m				
Cohesion Macizo Rocosos	> 400		300-400		200-300		100-200		< 100			200-300				
Angulo de friccion del MR	> 45		35-45		25-35		15-25		< 15			25-35				
SISTEMA DE CLASIFICACION "Q" BARTON																
PARAMETROS		SIMBOLOGIA					CLASIFICACION					VALOR				
Rock Quality Designation		RQD%										75				
Joint Set Number (numero de familias)		Jn					Dos familias de diaclasas					4				
Joint Roughness Number		Jr					Rugosas o irregulares pero planas					1.5				
Joint Alteration Number (Ja)		a) Contacto entre los planos de la discontinuidad (sin minerales de b) Contacto entre los planos de la discontinuidad ante un des aplaz c) No se produce contacto entre los planos de la discontinuidad ant					Planos de discontinuidad ligeramenteadaptadas.					25-30				
Joint Water Reduction Factor		Jw					Excavaciones secas o pequeñas afluencias de agua inferiores a 5l					1				
Stress Reduction Factor (SRF)		a) Las zonas débiles intersecan a la excavación, pudiendo producirse desprendimientos de roca a medida que la excavación del túnel va avanzando					Zonas débiles aisladas, conteniendo arcilla o roca desintegrada químicamente (profundidad de la excavación > 50m)					2.5				
$Q = \frac{RQD\%}{J_n} * \frac{J_r}{J_a} * \frac{J_w}{SRF}$		$RMR = 9 \ln Q + 44$ Para tuneles de ingeniería					Q					6.91	5.66			
DESCRIPCION DE LA VALORACION																
Q	400-1000	100-400	40-100	10-40	4-10	1-4	0.1-1	0.01-0.1	0.001-0.01			Media				
	Excepcionalm. Bu	Extremadam. Buena	Muy buena	Buena	Media	Mala	Muy mala	Extremadam. r	Excepcionalm. Mala							
CALCULO DEL DIAMETRO EQUIVALENTE																
Ancho de la excavacion		ESR (Excavation Support Ratio) Relacion de soporte de la excavacion					Tuneles pilotos(exploracion)					1.6	1.875			
Diámetro de la excavacion													2.1875			
Altura de la excavacion													3.5			
De		$De = \frac{\text{Ancho de excavacion, diametro o altura}}{ESR}$											2.031			

ANEXO 5 - PLANO SUBTERRANEO

