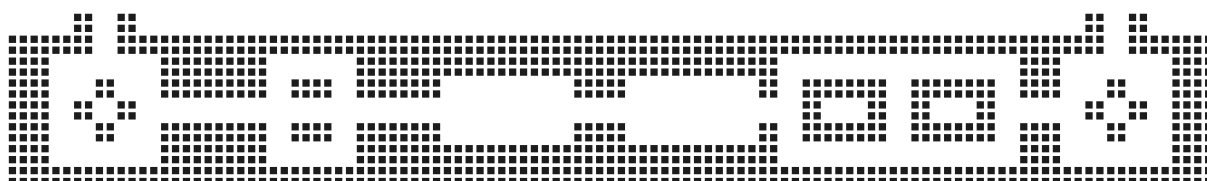


BMW



Blocks Master  
v4.0 beta

# *Cuaderno de Apuntes*

**EXPLORANDO EL INTERIOR DE BM**

# CONTENIDO

## Introducción

### PARTE I. Aspectos Básicos

1.1 Ideas Generales	1
1.2 Los Paquetes de Juego	2
1.2.1 Estructura de un archivo ZxPack	2
1.2.2 Estructura de un archivo AxPack	2
1.2.3 Control de flujo de los AxPack	2
1.2.4 La variable AxMode	2
1.2.5 La variable BMDDataType	3
1.3 Componentes de BM	3
1.3.1 Bloques y Eventos	4
1.3.1.1 Los MB, PB, OB y AB	4
1.3.1.2 Los CE	4
1.3.1.3 ¿Cómo se Localizan los Bloques en la Memoria?	5
1.3.1.4 Los Bloques y sus Dimensiones	6
1.3.1.5 Los UE (Eventos Inusuales)	6
1.3.2 Complementos gráficos	6
1.3.2.1 Gráficos	6
Estructura de estos componentes	7
Estructura de las listas: bgGrobList y fgGrobList	7
1.3.2.2 Etiquetas	7
1.3.2.3 Animaciones	7
1.3.3 Los paquetes gráficos	8
1.3.3.1 GLPacks	8
Estructura de un GLPack	8
CBGrobList	9
GLPack no estándar	9
1.3.3.1 aniSecLists (aSL)	9
1.4 Módulos y variables de BM	10
1.5 Los grupos	10

### PARTE II Sistemas de BME

2.1 El Reloj del Juego (contador de ciclos)	12
2.2 El Mapa	12
2.2.1 Cargando el mapa	12
2.2.2 El tamaño del mapa en Gbuff	13
2.3 Sistema de Desplazamiento	14
2.3.1 Los saltos	14
2.3.2 Regla para calcular coordenadas	14

2.3.3 Retardo en primer movimiento de Eli	14
2.3.4 La zona estática	15
2.3.5 Sobre el módulo M1	15
2.3.6 Limites del mapa	16
2.4 Sistema Bomb	16
2.5 Sistema M5	17
2.6 Sistema M6	17
2.7 Sistema de control de los MB	17
2.8 Gestor de animaciones (M3)	17
2.9 Mecanismo de carga del juego	18
2.10 Módulo principal (M1)	19
2.11 Sistema replay	19
2.11.1 Módulos relacionados	19
2.11.2 Variables relacionadas	19
2.11.3 Puntos donde concluye grabación de Rx	20
2.12 Sistema de salida	20
2.12.1 Módulos relacionados	20
2.12.2 La variable M1Off?	20
2.13 Sistema de salvado	20
2.14 Sistema T&M	20
2.14.1 Módulos relacionados	20
2.14.2 Variables relacionadas	20
2.15 Inventario	21
2.15.1 Variables relacionadas	21
2.16 Gestión de aventuras	22
2.17 Gestión de paquetes gráficos	23
2.18 Sistemas de entrada de texto	23
2.18.1 TxInp	23
2.18.1.1 Módulos relacionados	23
2.18.1.2 Variables relacionadas	23
2.18.2 CodeInp	24
2.18.2.1 Módulos relacionados	24
2.19 Sistema de acciones por choques	24
2.19.1 Variables relacionadas	24
2.20 Sistema de localización de componentes	25
2.20.1 Módulo SearchMBGetIndex	25
2.20.2 Módulo SearchPBGetIndex	25
2.20.3 Módulo SearchOBGetIndex	25
2.20.4 Módulo SearchABGetIndex	25

## PARTE III *Sistemas de BMB*

3.1 Sistemas para construcción	27
3.1.1 CBBUILDER	27
3.1.2 MBUILDER	28
3.1.3 PBBUILDER	28
3.1.4 OBBUILDER	29
3.1.5 ABBUILDER	29
3.1.6 CEBUILDER	30
3.1.7 GrbBuilder	30
3.1.8 LabelBuilder	31
3.1.9 AnmBuilder	31
3.1.10 MsgBBUILDER	31
3.1.11 MsgOBUILDER	31
3.1.12 MsgHBUILDER	32
3.1.10 GroupsBuilder	32
3.1.11 YWBuilder	32
3.2 Sistemas para configuración	33
3.2.1 EdZxName	33
3.2.2 EdZxMaster	33
3.2.3 GetMP	33
3.2.4 ConfInv	33
3.2.5 ConfSB	34
3.2.6 ConfKb	34
3.2.7 FxConf	34
3.2.8 BoxGrobSelection	35
3.2.9 ConfM1	35
3.3 Otros sistemas	35
3.3.1 Coords	35
3.3.2 GetSBInfo	36
3.3.3 GData	36

## PARTE IV *Código Fuente*

4.1 Código fuente de BM 4.0	39
4.2 Estructura de un archivo UpMs	39
4.3 Crear nuevos módulos de BM	39
4.4 Reinstalando BM	39
4.5 Ventajas de usar la librería Link	40
4.6 Compatibilidad entre BMB y BME	40
4.7. Al modificar BM...	40

## PARTE V *Notas Varias*

# ANEXOS

- Anexo 1. Módulos y Submódulo
- Anexo 2. Clasificación de los Módulos
- Anexo 3. Variables Lambda
- Anexo 4. Variables Locales
- Anexo 5. Estructura de los Componentes
- Anexo 6. Definiciones de Variables

# Introducción

Hola ☺

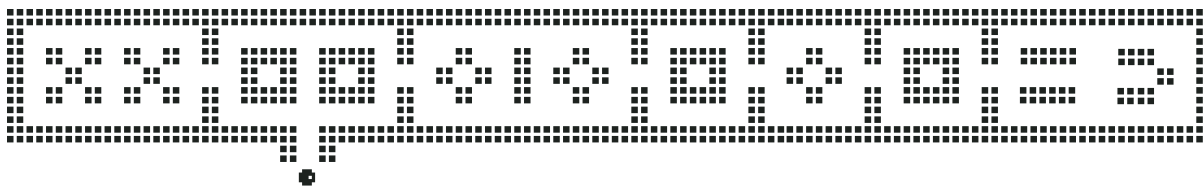
Gracias por leer mi Cuaderno de Apuntes de BM.

Este documento es básicamente una recopilación de anotaciones que he realizado durante la creación del sistema BM. Claro que aquí las he puesto en forma más organizada y presentable que en el cuaderno original. Además he agregado muchas cosas nuevas.

La idea es intentar explicar diversos aspectos del diseño interno y el funcionamiento de BM, tanto del Engine como del Builder. La mayor parte de la información aquí mostrada no se encuentra en ninguno de los otros documentos publicados anteriormente. También se explica el manejo de la librería BML, la cual utilizo durante la compilación del código fuente.

A esta recopilación de apuntes me gusta llamarla Manual [A], por ser un manual de [A]puntes, sobre la [A]rquitectura de BM y por tratar asuntos más [A]vanzados que en los otros documentos de BM.

Espero que este manual ofrezca ideas y reflexiones útiles para los proyectos emprendidos por el lector.



## PARTE I

# *Aspectos Básicos*



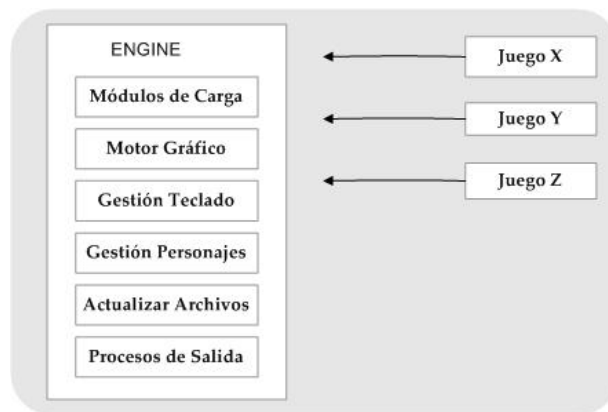


## 1.1 Ideas Generales

BM es un sistema para crear y ejecutar determinados tipos de juegos 2D en calculadoras hp50g. Aunque estos juegos son sencillos, pueden llegar a ser muy entretenidos. Pero lo más importante es que son muy fáciles de crear, jugar y compartir.

Para crear juegos con BM no se necesita conocimientos sobre programación, ya que los componentes son elegidos y configurados a través de menús, y el código se genera automáticamente.

Un aspecto fundamental del diseño de BM es que los juegos se encuentran separados del programa que los ejecuta. Esto es lo que permite poder reutilizar dicho programa con distintos juegos.



El sistema BM consta de las siguientes librerías:

- BME: Librería para correr/ejecutar juegos.
- BMB: Librería para crear juegos.
- BML: Librería para compilar BMB y BME

La documentación del sistema BM es la siguiente:

- BME: Manual de la librería BME.
- BMB: Manual de la librería BMB.
- BML: Manual de la librería BML.
- BMA: Cuaderno de Apuntes.

## 1.2 Los Paquetes de Juego

En BM existe dos tipos de archivos o paquetes de juego: los de Zona o ZxPack, y los de Aventura o AxPack.

Cada ZxPack constituye una Zona de juego, con su mapa, enemigos, objetivos, etc.

Los AxPack no son más que un grupo de ZxPack en un solo paquete. Al inicio de la Aventura solo tienes acceso a la primera Zona y a medida que avanzas obtienes acceso a las siguientes.

### 1.2.1 Estructura de un archivo ZxPack

{	
\$BMZxPack	(Tipo de paquete)
\$BMver	(Versión de BM)
\$NombredePack	(Nombre de la Zona)
#TamañoZx	(Tamaño al descomprimir en bytes)
{ Zx.BZ }	(archivo de juego comprimidos)
{ Sx.BZ }	(Salvados de juego comprimidos)
{ Rx.BZ }	(Replay de juego comprimidos)
}	

### 1.2.2 Estructura de un archivo AxPack

{	
\$BMAxPack	(Tipo de paquete)
\$BMver	(Versión de BM)
\$NombredePack	(Nombre de la Aventura)
#NumerodeMundos	
{ NombreZx1 NombreZx2 ... NombreZxN }	(Nombres de mundos)
{ flag1 flag2 ... flagN }	(Permisos de acceso a mundos. )
{ ZxMundo1.BZ ZxMundo2.BZ ... ZxMundoN.BZ }	(Lista de archivos de juego comprimidos)
{ SxMundo1.BZ SxMundo2.BZ ... SxMundoN.BZ }	(Lista de salvados comprimidos)
{ RxMundo1.BZ RxMundo2.BZ ... RxMundoN.BZ }	(Lista de replays comprimidos)
}	

El máximo de Mundos.BZ es 16, por tanto, el máximo de elementos de un Ax es 20.

Mientras que el mínimo de elementos de un Zx, Sx o Rx es MBi que siempre será mayor que 20

Esto lo utiliza Inst para saber si lo que va a instalar es un Ax o no.

EL USUARIO DEL BUILDER DEBE RECORDAR QUE DEBE INSTALAR COMPRESOR BZ2 PARA CREAR AVENTURAS.

### 1.2.3 Control de flujo de los AxPack

Si se sale de un juego por haberlo ganado se deja un TRUE en la pila en otro caso se deja un FALSE.

Este indicador (flag) lo utiliza El sistema Ax para controlar flujo al siguiente nivel.

Los replays no dejan indicador (flags) en la pila.

### 1.2.4 La variable AxMode

Es una variable creada por el Engine en el directorio oculto que le dice a los módulos si se está jugando una Aventura o un Zx o Sx. Es creada y actualizada por AxLevels, ZxLevels y SxLevels.

### 1.2.5 La variable BMDaType

Esta variable indica el tipo de archivo que se esta ejecutando:

#A indica un Zx  
#B indica un Sx  
#C indica un Rx

El módulo CkGoalc lo utiliza para saber si gestiona el flag de control de Ax o si lo deja al módulo M1, esto último solo para el caso de los Rx

## 1.3 Componentes de BM

Todo juego de BM está formado por la combinación de los siguientes componentes:

**Bloques Comunes o CB:** son los que aparecen con mayor frecuencia, forman los muros o paredes. Son los que ocupan menos memoria, así que puedes crear todos los que desees.

**Bloques Móviles o MB:** son aquellos que tienen la capacidad de moverse por si solos, algunos en una ruta fija y otros para perseguir.

**Bloques Push o PB:** son aquellos que se mueven o se modifican al ser empujados.

**Bloques Objetos OB:** son los que reaccionan al ser recogidos, es decir al colocarte sobre ellos y presionar la tecla T.

**Bloques Active o AB:** generalmente sirven como obstáculos y puertas que desaparecen/aparecen al activar un interruptor.

**Eventos Comunes o CE:** son los componentes más potentes del juego y los que ocupan más memoria, por lo que deben usarse con discreción.

**Mensaje de Ayuda:** son informaciones o sugerencias que el creador del juego proporciona el jugador. Se accede a esta presionando la tecla H durante la partida.

**Mensaje de Bienvenida:** es un texto que se muestra al jugador al inicio de la partida y que sirve como presentación o introducción al juego.

**Gráficos:** esta opción permite insertar a tu juego los gráficos creados con la herramienta del Builder (opción **Manejar Gráficos** en el menú).

**Animaciones:** esta opción permite insertar a tu juego las secuencias gráficos creados con la herramienta del Builder (opción **Manejar Animaciones** en el menú).

**Evento de Salida:** es el que dice cómo se gana el juego. Si usas por ejemplo “Llegar hasta aquí” el jugador ganara tu juego cuando logre llegar a ese lugar.

**Paquete gráfico de componentes (GLPacks):** lista de gráficos utilizados por BM para representar los componentes en la pantalla.

**Paquete de secuencias gráficas (aSL):** lista de varias secuencias gráficas utilizadas por BM.



0	CEtype
1	CEstatus
2	CEtest
3	CEprogT
4	CEprogF
5	CEx5pix
6	CEy5pix
7	CEx4pix
8	CEy4pix
9	CEx3pix
BINT10	CEy3pix
BINT11	CEx2pix
BINT12	CEy2pix
BINT13	CEx1pix
BINT14	CEy1pix
BINT15	CEx0pix
BINT16	CEy0pix
BINT17	CEL2
BINT18	CEL1
BINT19	CEL0
BINT20	CEc2
BINT21	CEc1
BINT22	CEc0
BINT23	CEf2
BINT24	CEf1
BINT25	CEf0
BINT26	CEg2
BINT27	CEg1
BINT28	CEg0
BINT29	CEk2
BINT30	CEk1
BINT31	CEk0

### 1.3.1.3 ¿Cómo se Localizan los Bloques en la Memoria?

Para referirse a cualquier MB, PB, OB, AB o CE se utilizan como punteros las variables lambda de BM llamadas actuales, que son: MBActual, PBActual, OBActual, ABActual y CEActual.

NÚMERO		NOMBRE	TIPO	DESCRIPCIÓN
DEC	HEX			
6	6	MBActual	#	Número de la primera variable lambda del MB con que se está trabajando en un momento determinado.
7	7	PBActual	#	Número de la primera variable lambda del PB con que se está trabajando en un momento determinado.
8	8	OBActual	#	Número de la primera variable lambda del OB con que se está trabajando en un momento determinado.
9	9	ABActual	#	Número de la primera variable lambda del AB con que se está trabajando en un momento determinado.

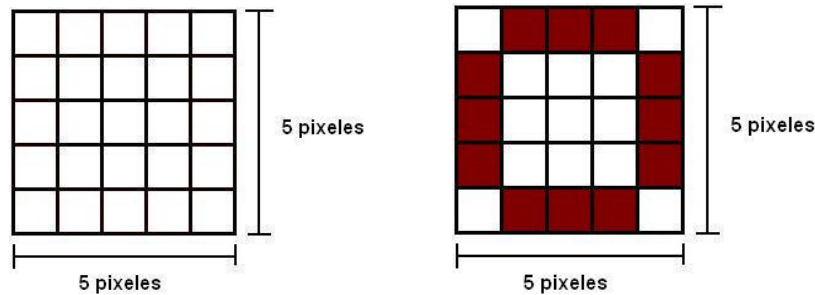
Para trabajar con un objeto (bloque o evento) determinado lo primero es hacer que la **Actual** correspondiente apunte hacia él. O sea, se coloca la número de la primera variable de dicho objeto en la **Actual** que le corresponde, y luego se hará referencia los distintos componentes del objeto incrementando su **Actual** en la cantidad necesaria.

### 1.3.1.4 Los Bloques y sus Dimensiones

Todos los bloques deben tener las mismas dimensiones. Dichas dimensiones serán las mismas del gráfico de CBGrobList que M0 utilice para generar mapa en GBUFF:

BoxGrobH x BoxGrobW

Estas dimensiones pueden variar, pero se toma como tamaños estándar los bloques de 5x5, 7x7 y 9x9 píxeles.



El tamaño de los gráficos de Eli y sus animaciones también debe ser igual que el de los bloques.

El espacio de separación entre bloques está contenido en la variable InterBox y tiene un valor por defecto de 1 píxel. Algunos valores de esta variable no funcionan correctamente, ya que hacen que al llegar al final del mapa Eli se salga de la pantalla.

Tómese en cuenta que el gráfico de la Bomba también debe ser igual al de los bloques.

### 1.3.1.5 Los UE (Eventos Inusuales)

Este nuevo tipo de componente permite evaluar cíclicamente código de eventos que normalmente no son evaluados así. Por ejemplo, permite crear OBs y PBs que contengan código que se evalúe en cada ciclo del módulo M1. El OB “Interruptor Temporizado de grupo” aprovecha este sistema.

## 1.3.2 Complementos gráficos

### 1.3.2.1 Gráficos

Los gráficos complementarios se dividen en: gráficos de fondo y gráficos de sobrefondo.

Los de fondo se colocan antes que los bloques y por tanto el g0 de cada bloque es una parte del fondo. Esto implica que al mover el bloque no afecta el fondo.

El módulo encargado de colocarlos es bgIni al ser llamado por Load o EdZx.

Los gráficos de fondo serán todos los que estén en bgGrobList. Ejemplo: Los gráficos de Telepuertos.

Los de sobrefondo se colocan después de los bloques y por tanto sobrescriben los bloques en GBUFF.

El módulo encargado de colocarlos es fgIni, al ser llamado por M0.

Los gráficos de fondo serán todos los que estén en fgGrobList.

#### **Estructura de estos componentes:**

{gn xn yn fn}

donde:

gn: es el gráfico

xn: coordenada x en abuff

yn: coordenada y en abuff

fn: flag para activar y desactivar carga del gráfico

#### **Estructura de las listas: bgGrobList y fgGrobList**

Poseen la misma estructura:

{ {g1 x1 y1 f1} {g2 x2 y2 f2} {g3 x3 y3 f3} ... {gn xn yn fn} }

El módulo común de carga es cgLoader. Utilizado tanto por bgIni como por fgIni.

Los gráficos de fondo son manejados en el Builder por: Grb (Crear, Instalar, Recuperar, Eliminar) y son agregados a un juego y configurados por GrbBuilder.

Para ser utilizados por el Builder, los grob deben ser instalados, lo que implica ser guardados en el directorio oculto y agregar su nombre a la lista BMGrbList.

### **1.3.2.2 Etiquetas**

Estas no constituyen un nuevo tipo de componente, ya que son en esencia un gráfico de fondo. Para crear una etiqueta el módulo LabelBuilder toma una cadena de caracteres y la convierte en un grob. Luego agrega este grob a la lista fgGrobList.

### **1.3.2.3 Animaciones**

Existen dos tipos de animaciones: Las que se ejecutan en un ciclo y las que se ejecutan en varios ciclos.

Las animaciones de un solo ciclo son manejadas por el módulo M3.

Las animaciones de varios ciclos son un tipo especial de CE.

Las secuencias gráficas que utiliza son manejadas en el Builder por Anm (Crear, Instalar, Recuperar, Eliminar).

Las animaciones son agregadas a un juego y configuradas por AnmBuilder.

En su ejecución utiliza los módulos siguientes:

AnmRunner:	Corre un ciclo de la secuencia grafica
AnmBGroundUpdater:	Cactualiza gráficos de fondo
AnmL1Runner:	Corre lista L1 de acciones a ejecutar

### 1.3.3 Los paquetes gráficos

#### 1.3.3.1 GLPacks

Contiene los gráficos de los bloques y de Eli, e información sobre el tamaño de dichos gráficos.

Son manejadas en el Builder por GLpack (Crear, Instalar, Recuperar, Eliminar y Editar).

Los GLPack son asignados a un juego por el módulo del Builder GLPackSelection

Los GLPack son cargados por el módulo del Engine GLPackLoader

Para Instalar un GLPack, este debe estar comprimido con BZ2

El Creador y el Editor de GLPack utiliza:

GLPackIni  
GLPackInpName  
GLPackExtractor

El GLPackSelection utiliza:

GLPackStdOn  
GLPackStdOff

El módulo StdGClear utiliza:

GLPackCleaner:      hace lo opuesto de GLPackLoader, eliminar del archivo Zx los GrobList que son estándar.

El módulo StdGLoader utiliza:

GLPackLoader:      hace lo opuesto de GLPackCleaner, agregar al archivo Zx los GrobList que son estándar.

Los GLPack y otros archivos son comunes para muchos juegos y por tanto, para reducir el consumo de memoria, estos archivos comunes se han dejado en el Engine en lugar de tenerlos repetidas veces en los distintos juegos.

La función de StdGClear es retirar del archivo de juego aquellas cosas que estén en el Engine, reduciendo así su tamaño.

La función de StdGLoader es restaurar las cosas que fueron retiradas del archivo de juego por StdGClear.

#### Estructura de un GLPack:

```
{
  { "BMGLPack" "NombredePaquete" #Dimensiones #NumerodeColumnas #NumerodeFilas }
  {CBGrobList}
  {MBGrobList}
  {PBGrobList}
  {OBGrobList}
  {ABGrobList}
  {eNList sin dividir}      (Es dividida por el módulo ListSplitter en GLPackLoader)
}
```



**CBGrobList:**

CBtype:

#1	BoxGrob	diseño 1 (Gráfico del bloque común utilizado)
#2	BoxGrob	diseño 2 (Gráfico del bloque común utilizado)
#3	BoxGrob	diseño 3 (Gráfico del bloque común utilizado)
#4	BoxGrob	diseño 4 (Gráfico del bloque común utilizado)
#5	BoxGrob	diseño 5 (Gráfico del bloque común utilizado)
#6	BoxGrob	diseño 6 (Gráfico del bloque común utilizado)
#7	BoxGrob	diseño 7 (Gráfico del bloque común utilizado)
#8	BoxGrob	diseño 8 (Gráfico del bloque común utilizado)
#9	IndGrob	
#A	BombGrob	(Gráfico de fondo para bombas)
#B	AfterBombGrob	(Gráfico de puntitos colocado luego de explosión de bombas)

**GLPack no estándar**

Cuando se crea un mundo con un GLPack que no es estándar, es decir que no viene incluido en el Engine, este GLPack se incluíra en el archivo del juego, es decir en el Zx o Sx, pero comprimido. Se ha creado la variable GLPackIncluded? para indicar esta situación y se tiene la variable GLPackBZ para guardar el GLPack comprimido.

En el Builder, el responsable de agregarlo al paquete es BMBBackup, a través de StdGClear.

En el Engine, el responsable de extraerlo del paquete es Load, a través de StdGLoader.

EL USUARIO DEL BUILDER DEBE RECORDAR QUE DEBE COMPRIMIR LOS GLPacks CON BZ2 ANTES DE INTENTAR INSTALARLOS.

**1.3.3.1 aniSecLists (aSL)**

Es una lista que contiene varias secuencias gráficas utilizadas por diferentes módulos del Engine y por varios componentes.

El estándar 5x5 esta en el archivo aSLStd1. El archivo aSLStd1 esta comprimido y contiene 5 secuencias:

1- Secuencia telepuerto:	13 diapositivas 382 bytes	7x19 pixeles
	Usado por: CE tipo 4 (ver módulo CEBuilder)	
2- Secuencia de onda de ataque corta	2 diapositivas 395 bytes	37x37 pixeles
	Usado por: MB tipos 1, 2, 3, 4 (ver módulo MBBuilder)	
4- Secuencia de desaparición de Eli	7 diapositivas 124 bytes	7x7 pixeles
	Usado por CE tipo F1, F2, F3, F4, F5 (ver módulo YWBuilder)	

## 1.4 Módulos y variables de BM

La versión 4.0 de BMB consta de 287 módulos, de los cuales 177 son de BME. Algunos módulos ejecutan su trabajo de manera independiente, pero la mayoría se auxilia de otros. Los anexos 1 y 2 contienen información sobre los módulos de BM.

El número de variables lambda creadas por BM para un juego va a depender de la cantidad de componentes que tenga dicho juego. Las primeras 281 variables están presentes en todos los juegos de BM. A estas se les suma 16 por cada MB, 16 por cada PB, 16 por cada OB, 5 por cada AB y 32 por cada CE, que el juego incluya. El anexo 3 contiene información detallada sobre todas las variables lambda de BM.

## 1.5 Los grupos

Una de las características más importantes de BM es que permite organizar los componentes de un juego en grupos. Esto hace posible ejecutar operaciones sobre varios componentes a la vez.

Los Grupos son manejados mediante el módulo GroupsBuilder del Builder, que cuenta con las siguientes opciones:

Agregar Componente:	Sirve para agregar un componente a un grupo.
Eliminar Componente:	Sirve para eliminar un componente de un grupo.
Ver Componentes:	Sirve para ver cuales componente pertenecen a un grupo.
Limpiar Grupo:	Sirve para eliminar todos los componentes de un grupo.
Agregar Grupo:	Sirve para agregar un grupo a otro grupo (suma de grupos).
Activar/Desactivar:	Sirve para activar/desactivar todos los componentes de un grupo.

Entre los nuevos componentes de grupo que se pueden agregar a un juego están:

PB: Interruptores de grupo.  
 PB: Interruptores de grupo por Impacto.  
 OB: Interruptor de grupo.  
 CE: Interruptor de grupo.

Un módulo muy importante aquí es TongleGroups, que sirve para invierte el estado de un grupo de componentes, es decir, si están activados los desactiva y si están desactivados los activa.

Si se quiere activar/desactivar varios grupos se debe:

Primero - Crear la variable para TongleGroups:

ID BMBuilderc1: una lista de indices  
 { #GrupoMB #GrupoPB #GrupoOB #GrupoAB #GrupoCE #GrupoFx }

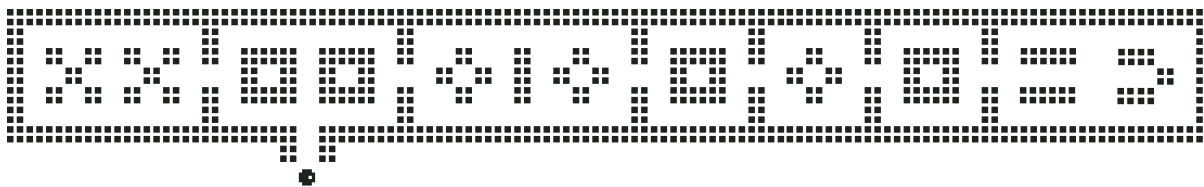
ID BMBuilderf1: una lista de flags  
 { flagMB flagPB flagOB flagAB flagCE flagFx }

Segundo - Llamar TongleGroups en modo 6, es decir ejecutar el código: #6 ID TongleGroups

La lista de flags indica a TongleGroups cuales tipos de grupos se activan/desactivan.

La lista índices indican a TongleGroups el numero del grupo a activar/desactivar.

Las variables ListListXBtoRun contienen los grupos de componentes.

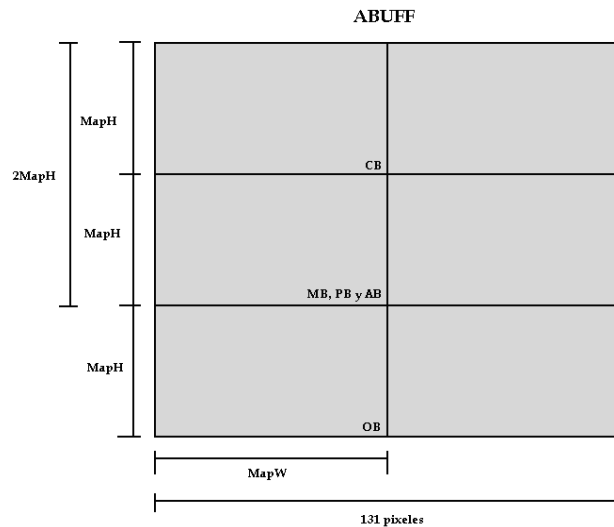


PARTE II

# *Sistemas de BME*

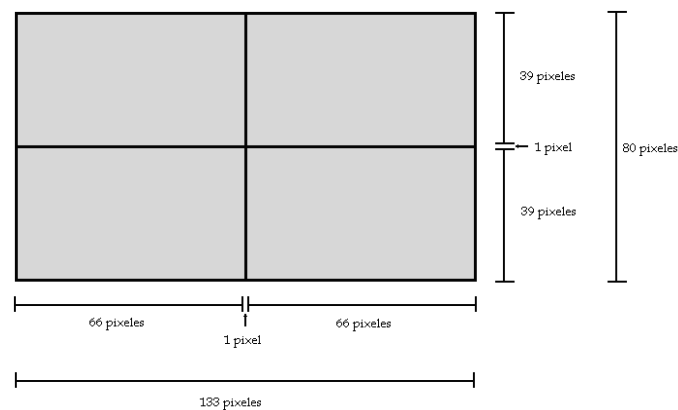






El mapa nunca puedes tener más de 131 píxeles de ancho, pero la altura solo esta limitada por la memoria de la calculadora. Sin embargo, se recomienda un tamaño estándar de 39x65 píxeles, ya que cuando el mapa es muy grande el consumo de memoria es demasiado alto y toma mucho tiempo para cargarlo.

En una ventana normal (80x131) tiene cuatro veces el tamaño el tamaño estándar 39x65.



El Módulo M0 pasa luego a la etapa que consume más tiempo de todo el proceso de carga del juego: Crea una ampliación del mapa en Gbuff.

Este proceso consiste en escanear el mapa (Abuff) píxel por píxel y colocar un bloque común por cada píxel activado en su lugar correspondiente. Las dimensiones del Gbuff fueron determinadas previamente multiplicando las del mapa por las de un bloque común.

### 2.2.2 El tamaño del mapa en Gbuff

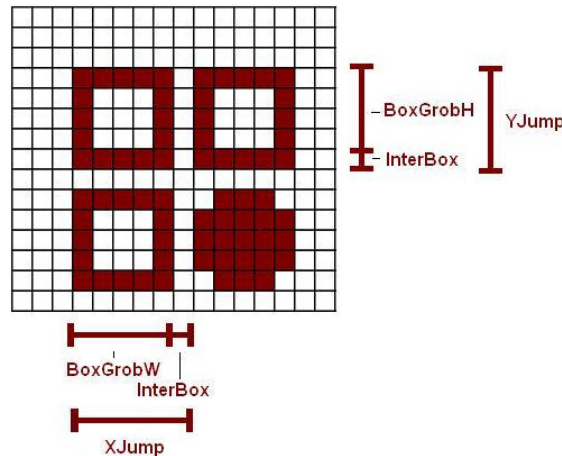
$$W_{gbuffinpix} = MapW \times XJump$$

$$H_{gbuffinpix} = MapH \times YJump$$

## 2.3 Sistema de Desplazamiento

### 2.3.1 Los saltos

Cada vez que Eli se mueve realiza un salto. Esto es: se borra el gráfico de Eli de las coordenadas en que se encuentra y se coloca en las coordenadas de destino.



Xpix y Ypix son las coordenadas de Eli en Abuff.

Xeli y Yeli son las coordenadas de Eli en Gbuff, o sea, las coordenadas del píxel superior izquierdo del gráfico de Eli en Gbuff. Estas se calculan de la siguiente forma:

$$X_{eli} = X_{pix} * X_{Jump}$$

$$Y_{eli} = Y_{pix} * Y_{Jump}$$

### 2.3.2 Regla para calcular coordenadas

$$Coord = Pixel - 1$$

Ejemplos:

Si Eli está en el segundo píxel de Abuff, su coordenada es  $2 - 1 = 1$

Si Eli está en el tercer píxel de Abuff, su coordenada es  $3 - 1 = 2$

Esto se explica por el hecho de que el primer píxel tiene coordenada cero.

### 2.3.3 Retardo en primer movimiento de Eli

Cuando queremos que Eli se desplace dejamos presionada una de las teclas de dirección, si al hacer esto observamos con cuidado veremos que luego de dar el primer salto Eli se detiene durante una fracción de segundo, para luego continuar su marcha con cierta velocidad. Este pequeño retardo es necesario para permitir un mayor control al jugador.

Las siguientes variables están vinculadas a este evento:

FMDelayOn?: Indica si el retardo está activado. Este es el flag principal. Este permite crear zonas del mapa en la que el jugador tenga menos control del movimiento de Eli, algo así como moverse sobre un piso resbaladizo.

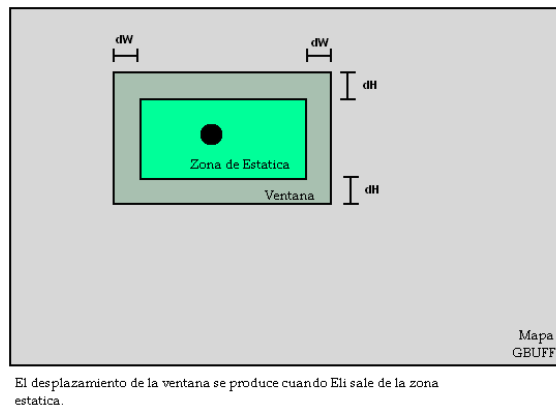
FMDelay?: Indica si el retardo debe ocurrir. Este interruptor se revisa solo si flag principal está activado.

DelayFM: contiene la cantidad de segundos del retardo.

### 2.3.4 La zona estática

Existe una zona de la pantalla en la que Eli se puede mover sin que se produzca desplazamiento de la ventana, esta es llamada la zona estática de la ventana. Dicha zona queda determinada por las constantes dW y dH.

La ventana solo se desplazará cuando Eli trate de salir de la zona estática.



Los valores por defecto son:

$$dW = 3 \times XJump$$

$$dH = 3 \times YJump$$

### 2.3.5 Sobre el módulo M1

Al presionar la tecla  $\odot$  el Módulo M1 efectúa dos pruebas para saber si se debe desplazar la ventana:

- Prueba 1: Que la ventana no esté en el límite del mapa:  
RIGHTCOL Wgbuffinpix #1- #<>
- Prueba 2: Que Eli se encuentra fuera de la zona estática:  
Xeli RIGHTCOL eW+dW #- #>

Si ambas pruebas son afirmativas, M1 procede a mover la ventana.

Verificaciones análogas se realizan al presionar la tecla  $\nabla$

Para ayudar a que M1 funcione mas rapido se ha creado algunas constantes especiales:

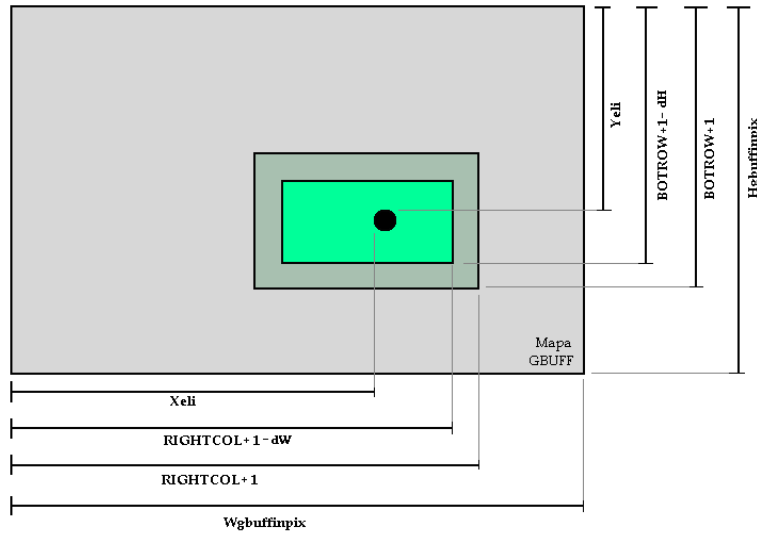
$$XeliB = Wgbuffinpix - XJump$$

$$YeliB = Hgbuffinpix - Yjump$$

$$EW+dW = EW + dW$$

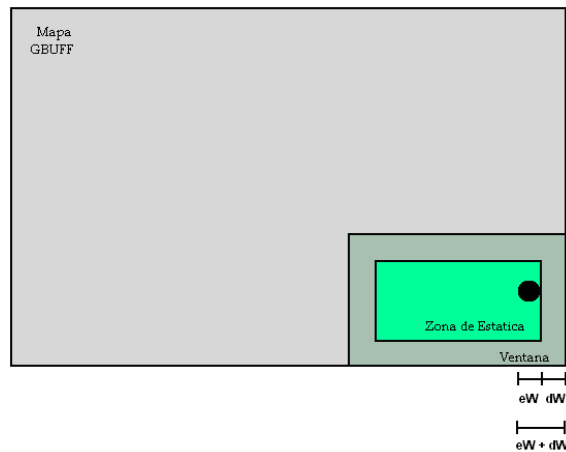
$$EH+dH = EH + dH$$

Esto evita tener que efectuar su cálculo en cada ciclo del Módulo M1.



### 2.3.6 Limites del mapa

Cuando Eli se acerca a los límites del mapa, a una distancia  $dW$  o  $dH$ , la ventana deja de desplazarse.



## 2.4 Sistema Bomb

En cada ciclo del módulo M1 la variable CTbomb se incrementa en #1. Cuando CTbomb llega a #2 la variable Tbomb se incrementa en #1 y CTbomb se reinicia. Cuando Tbomb llega a #A se ejecuta la secuencia de detonación de bomba.



## 2.5 Sistema M5

Es el encargado de manejar los PB, evaluando su test y ejecutando el código correspondiente. Este sistema se activa cuando el jugador empuja un bloque. Un punto importante es que M5 no sobrescribe PBActual.

## 2.6 Sistema M6

Es el encargado de manejar los OB, evaluando su test y ejecutando el código correspondiente. Este sistema se activa cuando el jugador recoge un objeto. De forma similar a M5, el módulo M6 no sobrescribe el OBActual.

## 2.7 Sistema de control de los MB

El SCMB es el encargado de manejar los MB. Este sistema se activa en cada ciclo del módulo principal M1, y lo que hace es básicamente evaluar todos aquellos MB cuyos índices están en la lista ListMBtoRun, evaluando su test y ejecutando los programa correspondientes.

## 2.8 Gestor de animaciones (M3)

Es el encargado de ejecutar las animaciones monocíclicas, es decir aquellas cuya secuencia gráfica se presenta en un solo ciclo del modulo M1. (las animaciones policíclicas son manejadas por el SCCE).

El módulo M3 requiere en la pila lo siguiente:

{GROB1 GROB2 GROB3 ... GROBN}	(lista de diapositivas o secuencia gráfica)
#m	(modo de animación: #1 = Gor, #2= Gxor, #3= GROB!)
Flag	(dice si se limpia GBUFF al terminar secuencia)
#xpix	(Coordenada x en ABUFF)
#ypix	(Coordenada y en ABUFF)
%t	(tiempo entre diapositivas)

## 2.9 Mecanismo de carga del juego

Me he inventado este tipo de diagrama, que nos da una idea del flujo de ejecución y nos dice los módulos que intervienen. Aunque no se si ya se le había ocurrido a alguien.

PlayZxPack →	ZxPackIni →	ZxLoader →	CkDescomp (A)		
			GetBMver (A)		
			BMver (A)		
			ShowOnHB2 (A)		
			StdGLoader →	InvBG (A)	
				YouWinMsjStd (A)	
				GLPackLoader →	ShowOnHB2 (A)
					Ubz2 (A)
					ToDMemList (A)
					aSLStd1 (A)
					aSLStd2 (A)
					aSLStd3 (A)
					ListSplitter (A)
			ToDMemList (A)		
			ExpandB →	ShowOnHB2 (A)	
				ToDMemList (A)	
			bgIni	ToDMemList (A)	
				cgLoader (A)	
			M0 →	GMapCBBUILDER →	ShowOnHB2 (A)
				BorderGenerator →	ShowOnHB2 (A)
				SBIIni (A)	
				CEIni →	TeleportLoader (A)
					WormLoader (A)
				fgIni →	ToDMemList (A)
					cgLoader (A)
				InitT&M (A)	
				GBuffIni (A)	
				EliIni →	CopyBox0 (A)
					eNRunner (A)
				M1Rx →	→
				M1ZxSx →	→
			CloseENG		ShowOnHB2 (A)

Aquí tenemos los módulos que se ejecutan al cargar BM. El flujo de ejecución se da de izquierda a derecha y de arriba hacia abajo para los submodulos. Los módulos que no utilizan submodulos tienen una (A) a su derecha. Los otros se indican con una →.

## 2.10 Módulo principal (M1)

Luego de que el juego está cargado se pasa el control al módulo principal M1. Para hacer BM más eficiente se ha creado dos versiones de M1, una para los archivos Zx y Sx, y otra para los Rx.

M1ZxSx →	Shtx (A)
	BEngine
	ShowEliCoords (A)
	M2
	aEli (A)
	SCMB (A)
	SCCE
	SCUE (A)
	Bomb
	T&M
	ACC (A)
	Rc (A)
	CkGoalc
	msg7

M1Rx →	ShowEliCoords (A)
	M2
	aEli (A)
	SCMB (A)
	SCCE (A)
	SCUE (A)
	Bomb
	T&M (A)
	ACC (A)
	CkGoalc
	Shtx (A)

## 2.11 Sistema replay

Los replay permiten reproducir de manera exacta una partida ya jugada. Pudiera dar la impresión de que se trata de un video, pero no lo es. Lo que se guarda es realmente una lista de las **pulsaciones** de teclas que hace el jugador, y del estado de todas las **variables** del juego al momento de iniciar la grabación.

### 2.11.1 Módulos relacionados:

RcStart: módulo que inicia la grabación del replay.

RcStop: módulo que concluye la grabación del replay.

RxErase: módulo que elimina el replay de un archivo ZxPack.

### 2.11.2 Variables relacionadas:

Nombre	Tipo	Descripción
Nvars	#	Cantidad de variables.
VKEY	#	Tecla actual durante reproducción de replay. Actualizada por M1Rx. Utilizada por M2
VKEYc	#	Usada por sistema Rx como contador de teclas presionadas. Tiene <b>restricción posicional</b> por el módulo RcStop.
VKEYList	lista	Usada por sistema Rx para guardar la lista de teclas presionadas. (VKEYList = Virtual Key List) Tiene <b>restricción posicional</b> por el módulo RcStop.
VKEYrec?	flag	Usada por sistema Rx para autorizar iniciar proceso de grabación.
VKEYplay?	flag	Usada para indicar que se debe tomar las pulsaciones de teclas de la lista VKEY y no del teclado. Esto es, que se esta reproduciendo un replay. Usada por M0 para elegir entre M1ZxSx y M1Rx. Usada también por M1Rx Tiene <b>restricción posicional</b> por el módulo RcStop.

### 2.11.3 Puntos donde concluye grabación de Rx:

En estos puntos se utiliza ID RcStop para concluir grabación de replay:

M2: Usando teclas R y ←  
CkGoalc

## 2.12 Sistema de salida

### 2.12.1 Módulos relacionados:

Esc: Presenta el menú de opciones de salida  
Save: Salva la partida. Guardando un archivo Sx de ella.  
SxErase: Elimina el archivo Sx del juego actual.

### 2.12.2 La variable M1Off?

Esta es revisada por M1 en cada ciclo de M1. Para autorizar salir de la partida basta colocar un TRUE en ella.

## 2.13 Sistema de salvado

El módulo Save crea una copia de todas las variables del juego y la guarda en el ZxPack. Para dicha copia invoca el módulo BMBBackup. Luego comprime el archivo obtenido y lo coloca en el lugar correspondiente del ZxPack/AxPack.

## 2.14 Sistema T&M

Creé este sistema para monitorear el consumo de memoria y la velocidad del módulo principal de BM.

### 2.14.1 Módulos relacionados:

SetT&M: Activa/desactiva el sistema, inicializando/limpiando las variables relacionadas.  
T&M: Actualiza los datos.

### 2.14.2 Variables relacionadas:

Nombre	Tipo	Descripción
T&M?	flag	Indica si el sistema T&M está activado. Esto permite que se pueda activar y desactivar con la misma tecla.
T&Mx	#	Coordenada X de T&M. Actualizada por InitT&M. Utilizada por SetT&M
T&My	#	Coordenada Y de T&M. Actualizada por InitT&M. Utilizada por SetT&M
ticks0	#	Conteo de ticks inicial
dticks	#	Variación en conteo de ticks
T&Mg0	grob	Gráfico de fondo de T&M en GBUFF. Actualizada por InitT&M. Utilizada y actualizada por SetT&M
MEMi		Cantidad de memoria libre cuando se arranca Blocks Master.

## 2.15 Inventario

El módulo Inventory recopila información el inventario de Eli, y la presenta en forma organizada al jugador. También gestiona esta función durante la grabación y reproducción de los replay. Durante la reproducción de un replay, en lugar de esperar a que se presione una tecla, espera 0.5 segundos y continúa la ejecución.

### 2.15.1 Variables relacionadas:

Nombre	Tipo	Descripción
InvBGrobStd?	flag	Indica si el gráfico fondo del inventario es estandar (inf no utilizado).
InvBGrob	grob	Gráfico de fondo de ventana de inventario.
InvBGh	#	Altura de ventana de inventario
InvBGw	#	Anchura de ventana de inventario.
InvdX	#	Distancia desde la línea derecha de la ventana actual hasta la del inventario.
InvdY	#	Distancia desde la línea superior de la ventana actual hasta la del inventario.
InvX	#	Coordenada X de ventana de inventario.
InvY	#	Coordenada Y de ventana de inventario.
InvGrob0	grob	Usada para guardar gráfico de fondo de ventana de inventario.
EliPower	#	Cantidad de Energía de Eli.
NBombs	#	Cantidad inicial de bombas o arma 1
NPinch	#	Cantidad de municiones de Pinch o arma 2.
PinchOn?	flag	Indica si se posee el detonador Pinch.
WorldKeys	#	Cantidad de llaves en el inventario.
WorldMap?	flag	Indica si el jugador posee el mapa de la zona. En caso afirmativo se le muestra al presionar las teclas correspondientes.
Shield?	flag	Arma que permite pasar por zonas de reducción de energía sin ser afectado.
ScannerOn?	flag	Indica si Eli posee el Detector de OBs.

## 2.16 Gestión de aventuras

PlayAxPack →	AxPackIni →	AxLoader →	ZxLoader →	CkDescomp (A)			
				GetBMver (A)			
				BMver (A)			
				ShowOnHB2 (A)			
				StdGLoader →	InvBG (A)		
					YouWinMsjStd (A)		
					GLPackLoader →	ShowOnHB2 (A)	
						Ubz2 (A)	
						ToDMemList (A)	
						aSLStd1 (A)	
						aSLStd2 (A)	
						aSLStd3 (A)	
						ListSplitter (A)	
				ToDMemList (A)			
				ExpandB →	ShowOnHB2 (A)		
					ToDMemList (A)		
				bgIni	ToDMemList (A)		
					cglLoader (A)		
				M0 →	GMapCBBUILDER →	ShowOnHB2 (A)	
					BorderGenerator →	ShowOnHB2 (A)	
					SBIIni (A)		
					CEIni →	TeleportLoader (A)	
						WormLoader (A)	
					fgIni →	ToDMemList (A)	
						cglLoader (A)	
					InitT&M (A)		
					GBuffIni (A)		
					EliIni →	CopyBox0 (A)	
						eNRunner (A)	
					M1Rx →	→	
					M1ZxSx →	→	
					CloseENGn	ShowOnHB2 (A)	
			ShowEliPower (A)				

Al iniciar una aventura se pone en TRUE la variable AxMode del directorio oculto.

Esto le dice a Save que debe agregar el archivo salvado a la aventura cuyo nombre está en la variable AxName del directorio oculto

Al ganar un Zx se deja un TRUE en la pila que es usado por el módulo de control de aventuras el cual pone el indicador (flag) de permiso del mundo siguiente en TRUE

## 2.17 Gestión de paquetes gráficos.

Pueden ocurrir tres casos:

- 1- El GLPack es estándar.
- 2- El GLPack no es estándar pero viene comprimido en el archivo de juego.
- 3- El GLPack no es estándar pero debe estar instalado en el directorio oculto.

El módulo StdGLoader es el encargado de esto:

StdGLoader	InvBG (A)	
	YouWinMsjStd (A)	
	GLPackLoader	ShowOnHB2 (A)
		Ubz2 (A)
		ToDMemList (A)
		aSLStd1 (A)
		aSLStd2 (A)
		aSLStd3 (A)
		ListSplitter (A)

La variable GLPStd? le indica si se trata de un paquete estándar.

La variable GLPStdX le indica el nombre del paquete estándar a utilizar.

La variable GLPackIncluded? le indica si el paquete esta incluido en el archivo de juego.

La variable GLPackBZ se usa para contener el paquete no estándar de un juego.

El módulo GLPackLoader toma un GLPack de la pila y lo carga en el juego actual.

El módulo GLPackCleaner elimina el paquete estándar de un juego antes de hacer un backup de él.


## 2.18 Sistemas de entrada de texto

Estos sistemas son muy importante, ya que permite que el usuario de BM pueda introducir texto para su posterior utilización. Existen dos: TxInp y CodeInp

### 2.18.1 TxInp

TxInp	TxBGList (A)
	Ubz2 (A)
	ShowOnHB2 (A)
	TxUpd (A)
	TxBSpC (A)

#### 2.18.1.1 Módulos relacionados:

TxBGList:	Lista de gráficos de barras de fondo de la ventana TxInp.
TxUpd:	Actualiza información cada vez que se introduce un caracter.
TxBSpC:	Elimina ultimo caracter. Se ejecuta al presionar  .

#### 2.18.1.2 Variables relacionadas:

Nombre	Tipo	Descripción
TxMode	#	Usada para indicarle a TxInp el modo en que debe operar.
TxBG	grob	Usada por TxInp para guardar barra de fondo del texto.
TxNChars	#	Usada por TxInp para guardar el número de caracteres por línea.
InpX	#	Usada por TxInp para guardar coordenada X relativa del texto.

InpY	#	Usada por TxInp para guardar coordenada Y relativa del texto.
InpK	#	Usada por TxInp para guardar el número de la tecla actual presionada durante la entrada de texto.
InpText	\$	Usado por TxInp para guardar texto introducido.
Inpc	#	Usado por TxInp como contador de caracteres.
Inpg0	grob	Usado por TxInp para guardar gráfico de fondo.
Inppl	#	Usado por TxInp para guardar el plano de la tecla presionada.
InpFinish?	flag	Usada por TxInp para indicarle a su bucle interno terminar entrada de caracteres.
InpXpix	#	Usada por TxInp para guardar coordenada X absoluta del texto.
InpYpix	#	Usada por TxInp para guardar coordenada Y absoluta del texto.
TxBGH	#	Usado por TxInp para guardar altura de barra de texto.
TxBGW	#	Usado por TxInp para guardar anchura de barra de texto.
TxFlag	flag	Usada por TxInp para indicarle al próximo programa si se presiono ENTER o CANCEL.
TxBBGx	#	Usada por TxInp para guardar coord relativa x de fondo negro ABUFF.
TxBBGy	#	Usada por TxInp para guardar coord relativa y de fondo negro ABUFF.
TxBBGh	#	Usada por TxInp para guardar altura de fondo negro.
TxBBGw	#	Usada por TxInp para guardar anchura de fondo negro.
TxBBGxpix	#	Usada por TxInp para guardar coord relativa x de fondo negro GBUFF.
TxBBGypix	#	Usada por TxInp para guardar coord relativa y de fondo negro GBUFF.
TxInfl	lista	Usada por TxInp para guardar el texto de ayuda.

Si se presiona ENTER sin haber introducido algún texto se colocará como texto introducido la cadena de caracteres "-----".

## 2.18.2 CodeInp

Este sistema es en realidad una variante de txInp. Utiliza incluso las mismas variables. La diferencia fundamental es que el texto introducido se presenta en el área de menú.

### 2.18.2.1 Módulos relacionados:

ClearMenuLabel:	Limpia el área de menú que se utilizará.
CodeInpCompKeys:	Actualiza varios de los caracteres introducido.

Si se presiona ENTER sin haber introducido algún texto se colocará como texto introducido la cadena de caracteres "00000000".

## 2.19 Sistema de acciones por choques:

Permite crear componentes que reaccionen cuando choquen con otros.

### 2.19.1 Variables relacionadas:

>MBClash?	(Ha ocurrido un choque?	)
>HitEli?	(Con Eli?	)
>MBClash3?	(Con OB?	)
>MBClash1?	(Con CB?	)
>MBClash2?	(Con MB o PB o AB?	)
>MBClashY	(Coord Y del objeto con que choco, ABUFF )	
>MBClashX	(Coord X del objeto con que choco, ABUFF )	
>PBClash?	(Ha ocurrido un choque?	)
>HitEli?	(Con Eli?	)



---

>PBCLash3?	(Con OB?	)
>PBCLash1?	(Con CB?	)
>PBCLash2?	(Con MB o PB o AB?	)
>PBCLashY	(Coord Y del objeto con que choco, ABUFF	)
>PBCLashX	(Coord X del objeto con que choco, ABUFF	)

## 2.20 Sistema de localización de componentes:

### 2.20.1 Módulo SearchMBGetIndex

- >Variables que usa: XSearch, YSearch, MB?, MBindex, MBActual y MBActualA.
- >Sirve para busca un MB en las coordenadas dadas en XSearch y YSearch.
- >Si lo encuentra devuelve true en MB?, en caso contrario devuelve false.
- >Si encuentra un MB lo deja como MBActual, en caso contrario lo deja como estaba.
- >Pero en todo caso el MB original se deja en MBActualA.
- >Si encuentra un MB deja su índice en MBindex.

### 2.20.2 Módulo SearchPBGetIndex

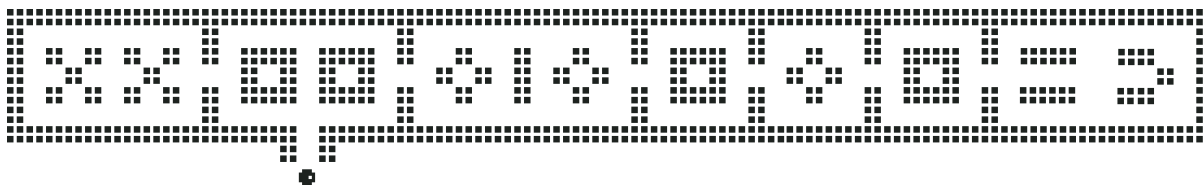
- >Variables que usa: XSearch, YSearch, PB?, PBindex, PBActual y PBActualA.
- >Sirve para busca un PB en las coordenadas dadas en XSearch y YSearch.
- >Si lo encuentra devuelve true en PB?, en caso contrario devuelve false.
- >Si encuentra un PB lo deja como PBActual, en caso contrario lo deja como estaba.
- >Pero en todo caso el PB original se deja en PBActualA.
- >Si encuentra un PB deja su índice en PBindex.
- >Es usado por módulo M5.

### 2.20.3 Módulo SearchOBGetIndex

- >Variables que usa: XSearch, YSearch, OB?, OBindex, OBActual y OBActualA.
- >Sirve para busca un OB en las coordenadas dadas en XSearch y YSearch.
- >Si lo encuentra devuelve true en OB?, en caso contrario devuelve false.
- >Si encuentra un OB lo deja como OBActual, en caso contrario lo deja como estaba.
- >Pero en todo caso el OB original se deja en OBActualA.
- >Si encuentra un OB deja su índice en OBindex.
- >Es usado por módulo M6.

### 2.20.4 Módulo SearchABGetIndex

- >Variables que usa: XSearch, YSearch, AB?, ABindex, ABActual y ABActualA.
- >Sirve para busca un AB en las coordenadas dadas en XSearch y YSearch.
- >Si lo encuentra devuelve true en AB?, en caso contrario devuelve false.
- >Si encuentra un AB lo deja como ABActual, en caso contrario lo deja como estaba.
- >Pero en todo caso el AB original se deja en ABActualA.
- >Si encuentra un AB deja su índice en ABindex.



### PARTE III

## *Sistemas de BMB*



El módulo principal de la librería BMB es BEngine

BEngine	ShowEliCoords (A)
	Shtx (A)
	CBBUILDER (A)
	CopyBox0 (A)
	MBBUILDER
	msg16
	PBBUILDER
	OBBUILDER
	CEBUILDER
	ABBUILDER
	YWBUILDER
	CBMode
	CGMenu
	SBMenu
	GData
	GetSBInfo
	ZxCONFIG
	ConfInv
	FxMenu
	GroupsBuilder
	CoordsMenu
	Map1
	Map2
	Map3
	TongleEliCoords
	ExitBLDR
	MenuHBLDR
	GetAbuffDA3
	ZxCFA (A)
	ZxCFP
	PutAbuffDA3 (A)
	ExitENG
	BMrev (A)
	Ubz2 (A)
	Inst

### 3.1 Sistemas para construcción:

#### 3.1.1 CBBUILDER

Es el encargado de la construcción y gestión de los CB.

Módulos relacionados:

CBBUILDER  
 CBMode  
 TurnCBBUILDEROn  
 TurnCBBUILDEROff  
 TurnCBOFFCopyBG  
 GetCBLISTToInv

Variables relacionadas (locales):

CB+?  
 CBBUILDER?

### 3.1.2 MBBuilder

Es el encargado de la construcción y gestión de los MB

Módulos relacionados:

MBBuilder  
 IntroRutaMB  
 getMBSpeed  
 getMBGunStatus  
 getMBGunRange  
 MoveMBtoRuta  
 AttackCodeMB  
 getMBRadarRange  
 GetMBDistanceEli  
 MoveMBtoEli  
 MBIntModeSelect  
 MBStop  
 MBStart

Variables relacionadas:

Nombre	Tipo	Descripción
MBActual	#	Número de la primera variable lambda del MB con que se está trabajando en un momento determinado.
MBi	#	Número de la primera variable lambda de los MB.
MBf	#	Número de la última variable lambda de los MB. Usada por BMBBackup.
MBActualA	#	Usada por algunos módulos para guardar una copia de MBActual.
MBNum	#	Cantidad de MB del juego.
MBGrobList	lista	Lista de gráficos para los MB.
ListMBtoRun	lista	Usada por SCMB (Sistema de control de los MB). Contiene la lista de MBs que serán evaluados en cada ciclo de M1ZxSx o M1Rx.
SizeListMBtoRun	#	Contiene la cantidad de Mbss de la lista ListMBtoRun
RunListMBtoRun?	flag	Usado por SCMB. Sirve para desactivar dicho sistema indicándole si debe o no evaluar los MB de la lista ListMBtoRun.
ListListMBtoRun	lista	Grupos de MBs. Lista que contiene 16 sublistas para los 16 grupos de MB.
MBStep	#	Indica al módulo MBMove en que dirección ha de mover el MB.

### 3.1.3 PBBuilder

Es el encargado de la construcción y gestión de los PB.

Módulos relacionados:

PBBuilder  
 PBMov  
 PBoff  
 PBOon  
 CodeInp  
 IfPBClashMBTurnMBOff

Variables relacionadas:

Nombre	Tipo	Descripción
PBActual	#	Número de la primera variable lambda del PB con que se está

		trabajando en un momento determinado.
PBi	#	Número de la primera variable lambda de los PB.
PBf	#	Número de la última variable lambda de los PB. Usada por BMBBackup.
PBActualA	#	Usada por algunos módulos para guardar una copia de PBActual.
PBNum	#	Cantidad de PB del juego.
PBGrobList	lista	Lista de gráficos para los PB.
ListListPBtoRun	lista	Grupos de PBs. Lista que contiene 16 sublistas para los 16 grupos de PB

### 3.1.4 OBBuilder

Es el encargado de la construcción y gestión de los OB.

Módulos relacionados:

OBBuilder  
WorldKeysInc  
OBCLr  
NBombsInc  
NPinchInc  
EliPowerInc  
WorldMapInc  
GoalcInc  
msg18

Variables relacionadas:

Nombre	Tipo	Descripción
OBAActual	#	Número de la primera variable lambda del OB con que se está trabajando en un momento determinado.
OBi	#	Número de la primera variable lambda de los OB.
OBf	#	Número de la última variable lambda de los OB. Usada por BMBBackup.
OBAActualA	#	Usada por algunos módulos para guardar una copia de OBAActual.
OBNum	#	Cantidad de OB del juego.
OBGrobList	lista	Lista de gráficos para los OB.
ListListOBtoRun	lista	Grupos de OBs. Lista que contiene 16 sublistas para los 16 grupos de OB

### 3.1.5 ABBuilder

Es el encargado de la construcción y gestión de los OB.

Módulos relacionados:

ABBuilder  
ABOff  
ABOn

Variables relacionadas:

Nombre	Tipo	Descripción
ABAActual	#	Número de la primera variable lambda del AB con que se está trabajando en un momento determinado.
ABi	#	Número de la primera variable lambda de los AB.
ABf	#	Número de la última variable lambda de los AB. Usada por BMBBackup.
ABNum	#	Cantidad de AB del juego.

ABActualA	#	Usada por algunos módulos para guardar una copia de ABActual.
ABGrobList	lista	Lista de gráficos para los AB.
ListListABtoRun	lista	Grupos de ABs. Lista que contiene 16 sublistas para los 16 grupos de AB

### 3.1.6 CEBuilder

Es el encargado de la construcción y gestión de los CE.

Módulos relacionados:

CEBuilder  
CE01  
CE02A  
CE02B  
CE03  
CE04A  
CE04B  
CE05  
CE06  
CE07  
CEg0Restore  
CEIni  
CEStart  
CEStop

Variables relacionadas:

Nombre	Tipo	Descripción
CEActual	#	Número de la primera variable lambda del CE con que se está trabajando en un momento determinado.
CEi	#	Número de la primera variable lambda de los CE.
CEf	#	Número de la última variable lambda de los CE. Usada por BMBBackup.
CENum	#	Cantidad de CE del juego.
ListCEtoRun	lista	Usada por SCCE (Sistema de control de los CE). Contiene la lista de CEs que serán evaluados en cada ciclo de M1ZxSx o M1Rx.
SizeListCEtoRun	#	Contiene la cantidad de CEs de la lista ListCEtoRun.
ListListCEtoRun	lista	Grupos de CEs. Lista que contiene 16 sublistas para los 16 grupos de CE.
RunListCEtoRun?	flag	Usado por SCCE. Sirve para desactivar dicho sistema indicándole si debe o no evaluar los CE de la lista ListCEtoRun.

### 3.1.7 GrbBuilder

Encargado de la creación y gestión de gráficos complementarios.

Módulos relacionados:

Grb  
GrbBuilder

Variables relacionadas:

GRBx1 (global)  
GRBy1 (global)  
GRBGLPack (global)  
GRBw (global)  
GRBh (global)

NombreGrb (global)  
BMGrbList (en directorio oculto)  
fgGrobList (lambda)  
FBstatus (global)

### 3.1.8 LabelBuilder

Encargado de la creación y gestión de etiquetas.

Módulos relacionados:  
InpText  
LabelBuilder

Variables relacionadas:  
fgGrobList

### 3.1.9 AnmBuilder

Encargado de la creación y gestión de animaciones.

Módulos relacionados:  
Anm  
AnmBuilder  
AnmT1Executer  
AnmBGroundUpdater  
AnmT2Executer  
AnmBuilderDatSECSUST  
AnmT3Executer

Variables relacionadas:  
BMAnmList (en directorio oculto)  
ANMx1 (global)  
ANMy1 (global)  
ANMGLPack (global)  
ANMw (global)  
ANMh (global)  
ANMn (global)  
NombreAnm (en directorio oculto)

### 3.1.10 MsgBBuilder

Encargado de crear y gestionar el mensaje de bienvenida.

Módulos relacionados:  
MsgBBuilder  
TxInp

Variables relacionadas:  
TxList?  
TxList

### 3.1.11 MsgOBuilder

Encargado de crear y gestionar el mensaje de objetivos.

Módulos relacionados:

MsgOBuilder  
TxInp

Variables relacionadas:

TxObj?  
TxObj

### **3.1.12 MsgHBuilder**

Encargado de crear y gestionar el mensaje de ayuda.

Módulos relacionados:

MsgHBuilder  
TxInp

Variables relacionadas:

TxHelp?  
TxHelp

### **3.1.10 GroupsBuilder**

Encargado de la creación y gestión de grupos.

Módulos relacionados:

ListaGroupsBuilder  
ListaIndices  
ListaGrupos  
TongleGroups

Variables relacionadas:

GList (global)  
SBindice (global)  
Gindice (global)  
GrupodeTrabajo (global)  
GrupodeTrabajoLEN (global)  
SBindicePOS (global)  
SBindicePOS (global)  
Gindice2 (global)  
GrupodeTrabajo2 (global)  
BMBuilderc1 (global)  
BMBuilderf1 (global)

### **3.1.11 YWBuilder**

Encargada de crear y gestionar los mecanismos que evalúan cuándo se ha ganado la partida.

Módulos relacionados:

YWBuilder  
YW01  
YW02B  
YW02D  
YW03



YW04  
YouWinMsjStd

Variables relacionadas:  
Goal?  
Goaln  
Goalc  
YouWinStd?

## **3.2 Sistemas para configuración**

### **3.2.1 EdZxName**

Encargado de la actualización del nombre de la zona.

Módulos relacionados:  
EdZxName  
TxInp

Variables relacionadas:  
ZxName

### **3.2.2 EdZxMaster**

Encargado de la actualización del nombre de la zona.

Módulos relacionados:  
EdZxMaster  
TxInp

Variables relacionadas:  
ZxMaster

### **3.2.3 GetMP**

Encargado de asignar clave de acceso para poder editar un ZxPack. Por razones obvias no voy a dar mucha información sobre esta parte. Pero diré que el password se guarda encriptado en el archivo ZxPack.

Módulos relacionados:  
GetMP  
TxInp

Variables relacionadas:  
MasterPassword  
MasterPassword?

### **3.2.4 ConfInv**

Encargado de configurar el inventario inicial de Eli.

Módulos relacionados:  
ConfInv

Variables relacionadas:  
WorldMap?

EliPowerIni  
 EliPower  
 EliPowDown?  
 WorldKeys  
 NBombs  
 Shield?  
 NPinch  
 PinchOn?  
 ScannerOn?  
 Goalc

### 3.2.5 ConfSB

Ayuda a modificar el estado de componentes ya creados.

Módulos relacionados:

ConfSB

MBOOn	PBOOn	OBOOn	ABOn	CEStart
MBOff	PBOff	OBOff	ABOff	CEStop
MBi	PBi	OBi	ABi	CEi

Variables relacionadas:

CONFGtipo (global)  
 CONFGindice (global)  
 CONFGaccion (global)  
 CONFGBIGLIST (global)

### 3.2.6 ConfKb

Permite configurar el teclado.

Módulos relacionados:

ConfKb

RestoreKBStd

Variables relacionadas:

KUp,KDown,KLeft,KRight  
 KMap1,KMap2,KMap3  
 KGun1,KGun2,KGun3  
 KInventory  
 KTakeOb  
 KHelp  
 KSpeedUp,KSpeedDw  
 KExit  
 KeyboardStd?

### 3.2.7 FxConf

Permite modificar el estado de los interruptores Fx de BM.

Módulos relacionados:

FxMenu

FxGetState

FxOn  
FxOff

Variables relacionadas:

Nombre	Tipo	Descripción
F1	flag	Estado del interruptor F1.
F2	flag	Estado del interruptor F2.
F3	flag	Estado del interruptor F3.
F4	flag	Estado del interruptor F4.
F5	flag	Estado del interruptor F5.
F6	flag	Estado del interruptor F6.
F7	flag	Estado del interruptor F7.
F8	flag	Estado del interruptor F8.
F9	flag	Estado del interruptor F9.
FA	flag	Estado del interruptor FA.
FB	flag	Estado del interruptor FB.
FC	flag	Estado del interruptor FC.
FD	flag	Estado del interruptor FD.
FE	flag	Estado del interruptor FE.
FF	flag	Estado del interruptor FF.

### 3.2.8 BoxGrobSelection

Permite cambiar el gráficos de los CB.

Módulos relacionados:

BoxGrobSelection  
GMapCBBuilder

Variables relacionadas:

CBGrobList  
BoxGrobindex  
BoxGrob

### 3.2.9 ConfM1

Permite configurar varias características del módulo principal de BME, M1.

Módulos relacionados:

ConfM1

Variables relacionadas:

Accion?  
AnimateEli?  
DelayFM  
ShowCiclos?  
Nciclos

## 3.3 Otros sistemas

### 3.3.1 Coords

Algunos componentes requieren que se les indique varias coordenadas del mapa que serán tomadas en cuenta durante su funcionamiento. Es aquí donde interviene este sistema.

Módulos relacionados:

CoordsMenu  
 CoordsSaver  
 CoordsViewer  
 TongleEliCoords

Variables relacionadas:  
 CoordsList (global)

### 3.3.2 GetSBInfo

Permite obtener información sobre cualquier MB, PB, OB o AB creado.

Módulos relacionados:  
 GetSBInfo  
 SearchMBGetIndex  
 SearchPBGetIndex  
 SearchOBGetIndex  
 SearchABGetIndex  
 ShowResumen

Variables relacionadas:  
 XSearch  
 YSearch  
 MB?  
 PB?  
 OB?  
 AB?

### 3.3.3 GData

Ofrece diversas informaciones sobre el juego en construcción, como cantidad de componentes de cada tipo, memoria utilizada, valores de algunas variables importantes, etc.

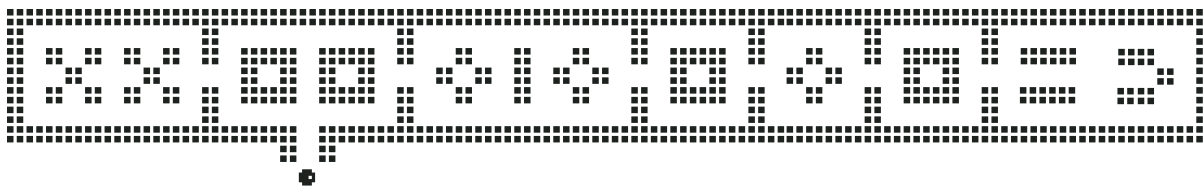
Módulos relacionados:  
 GData  
 RndRApndRTToStr  
 Shtx

Variables relacionadas:

ZxName	MBNum	MEMi	MBi
ZxMaster	PBNum	DMemList	PBi
Xpix	OBNum		OBi
Xeli	ABNum		ABi
Ypix	CENum		CEi
Yeli	Nvars		
LEFTCOL			
TOPROW			
MapW			
MapH			

La variable DMemList  
 Contiene datos sobre consumo de memoria

```
{
%      (contenido: tamaño del GLPack, cargado por: GLPackLoader)
%      (contenido: tamaño del aSLStd, cargado por: GLPackLoader)
%      (contenido: tamaño del Gbuff, cargado por: ExpandB)
%      (contenido: tamaño del Abuff, cargado por: ExpandB)
%      (contenido: tamaño de bgGrobList, cargado por: bgIni)
%      (contenido: tamaño de fgGrobList, cargado por: fgIni)
%      (contenido: consumo máximo para cargar juego, cargado por:M0)
%      (contenido: tamaño del archivo de juego savxx, cargado por:Load)
}
```



## PARTE IV

# *Código Fuente*



#### 4.1 Código fuente de BM 4.0

He decidido publicar el código fuente de BM 4.0, y la herramienta que utilizo para compilarlo.

En la carpeta del código fuente encontrarán 287 archivos. Algunos son archivos de texto y otros son archivos específicamente de la hp.

La mejor forma de **compilar BM** es hacerlo en el Emulador Emu48 utilizando la librería BML. En el manual de BML se explica los pasos a seguir.

Lo que hace UpMs de BML es:

- 1) Convertir los archivos al formato MASD.
- 2) Agregar las definiciones de variables.
- 3) Compilar el archivo.
- 4) Agregarlo al directorio BMD.

Lo que hace Link de BML es:

- 1) Crear las variables \$ROMID, \$TITLE, \$CONFIG, \$VISIBLE y \$HIDDEN.
- 2) Ordenar los archivos.
- 3) Convertir el directorio BMD en librería.

#### 4.2 Estructura de un archivo UpMs

UpMs funciona con dos tipos de archivos:

- 1) M1 (Módulo tipo1): Aquellos que deben primero ser compilados.  
Tienen formato "\$<nombre> <fecha> <TDV?> <Comprimir?>CodigoModulo".  
Ejemplos de este tipo de archivos son aSLStd1\_400B, GLPStd1\_400B, L4bz\_400B, etc.
- 2) M2 (Módulo tipo2): Aquellos que no deben ser compilados.  
Tienen formato {"\$nombre" "\$fecha" flag CodigoModulo}.  
Ejemplos de este tipo de archivos son ABBUILDER400B.txt, BLDR\_400B.txt, Inst\_400B, etc.

Nota Importante: el programa operará sobre todos los archivos que encuentre en la pila, por tanto solo debe haber en ella archivos validos de BM.

#### 4.3 Crear nuevos módulos de BM

Para agregar nuevos módulos a BM es necesario modificar las variables

\$V.ENGN: \$VISIBLE de la librería BME.  
\$V.BLDR: \$VISIBLE de la librería BMB.  
\$H.ENGN: \$HIDDEN de la librería BME.  
\$H.BLDR: \$HIDDEN de la librería BMB.

Para agregar un archivo al código fuente que no esté en formato adecuado debe convertirlo:

- > Use la opción **File --> BMLfile** del menú **OOpc** para convertirlo a formato tipo M2.
- > Use el editor "Bloc de Notas" de Windows para convertirlo a formato M1.

#### 4.4 Reinstalando BM

Si tienes la librería BM instalada y quieres actualizarla, coloca la versión nueva en la pila y ejecuta UpdB de la librería Link. Lo que hace esta opción es desinstalar BM del puerto 2 e instalar la librería que encuentre en la pila.

#### 4.5 Ventajas de usar la librería Link

Lee el nombre de cada archivo desde el mismo código fuente.

Un código fuente simple. La librería Link completará formato MASD.

Todo en un mismo directorio. Link se encarga de tomar lo que necesita (\$H.PROG1, \$H.PROG2....)

#### 4.6 Compatibilidad entre BMB y BME

Puesto que BMB contiene todos los módulos de BME, la clave para hacerlas compatibles está en dar el mismo número a ambas librerías y garantizar que los módulos de BME se encuentren en la misma posición que en BMB.

BML, que es la librería utilizada para compilar tanto BMB como BME, crea en el puerto 1, cuatro archivos especiales, llamados: \$V.ENGN, \$V.BLDR, \$H.ENGN y \$H.BLDR

Para BME:

$\$VISIBLE = \$V.ENGN$

$\$HIDDEN = \$V.BLDR + \$H.ENGN$

Para BMB:

$\$VISIBLE = \$V.ENGN + \$V.BLDR$

$\$HIDDEN = \$H.ENGN + \$H.BLDR$

El hecho de que para BME hagamos  $\$HIDDEN = \$V.BLDR + \$H.ENGN$  es fundamental para garantizar que los archivos creados en BMB sean compatibles con BME, al asegurar que los archivos mantendrán la misma posición en ambas librerías.

#### 4.7. Al modificar BM...

- \* Al modificar un módulo es indispensable saber cuales módulos y variables resultan afectados por esa modificación.
- \* No es conveniente hacer muchos cambios a la vez, ya que si se produce un error será más difícil localizarlo.
- \* Cualquier reducción de tiempo para un bucle es significativa, ya que se multiplica por las veces que dicho bucle se repite.
- \* Es importante no cambiar la configuración del teclado con demasiada frecuencia, ya que esto afecta la curva de aprendizaje a los usuarios.
- \* Para hacer que BM genere más variables lambda solo hay que modificar el módulo MBi.
- \* Todo evento (CE, UE, MB, Bomb, T&M,...) debe dejar la pila limpia luego de ejecutarse.
- \* Cuando se desarrolla un nuevo componente OB se debe agregar código a: OBBuilder, tabla de definiciones de variables, Inventory, GLPStd1, GLPStd2 y GLPStd3, y ConfInv.
- \* Al crear un arma nueva se debe agregar código a los módulos Inventory y a ConfInv.
- \* Es importante que al diseñar un nuevo componente, este no transmita información en base a los variables XBActual, ya que esto generaría errores. La causa de dichos errores es que en el proceso de creación de un ZxPack (en el Builder) estas variables tienen un valor distinto al que tendrán en el juego ya terminado. Esto

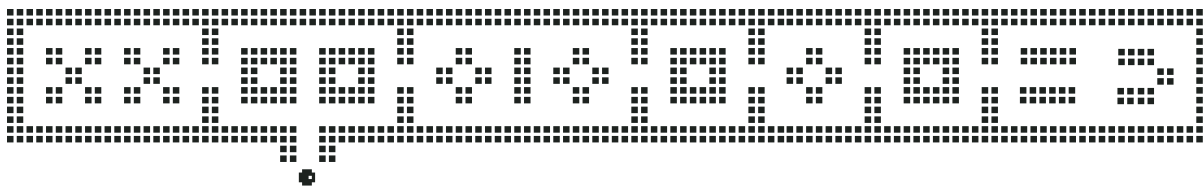


es resultado de la expansión de variables que se efectúa en el Builder para poder crear 16 componentes de cada tipo y que luego se ajusta a los valores reales en el juego terminado. Use siempre los índices de los componentes en lugar de las XBActual.

\* La mejor manera de encontrar fallos en BM es usándolo. Con esto no solo se localizaran fallos, sino que surgirán nuevas ideas.

\* Téngase presente siempre las variables con restricción posicional, es decir, aquellas que no se pueden cambiar de lugar en la tabla de definiciones ya que algún modulo las localiza por su posición y no por su nombre.

\* Ningún módulo de BML debe nombrarse igual que alguno de BMB o BME, ya que esto afectaría los ID de las listas \$V.ENGN, \$V.BLDR, \$H.ENGN o \$H.BLDR, que forman parte de BML. Concretamente se sustituiría durante la compilación de la librería BML el ID de las listas por el PTR del archivo de BML haciendo imposible luego la utilización de las listas.



## PARTE V

# *Notas Varias*



## BM y el Directorio Oculto

Blocks Master crea las siguientes variables en el directorio oculto:

BMZxList:	Contiene los nombres de los archivos Zx instalados.
BMAxList:	Contiene los nombres de los archivos Ax salvados.
BMAnmList:	Contiene los nombres de las secuencias graficas para animaciones que no están en aniSecList.
BMGrbList:	Contiene los nombres de los gráficos complementarios.
BMGLPackList:	Contiene los nombres de los GLPack que no son estándar, es decir que no vienen en el ENGINE.
AxMode	
BMAxID	

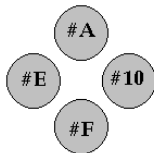
Al agregar o Eliminar uno elemento, su nombre es agregado o eliminado de la lista correspondiente.

## Neutralizando un MB:

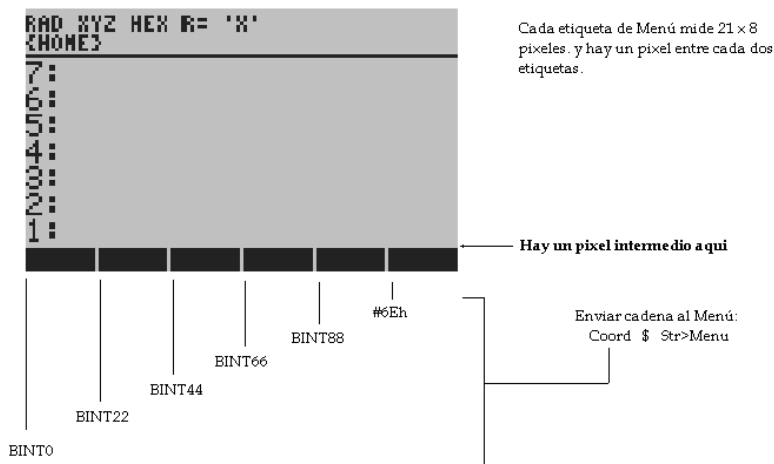
Puedes neutralizar un MB de distintas formas:

- Encerrándolo con un PB.
- Encerrándolo en una habitación.
- Mediante un interruptor para desactiva

## Números de las teclas de dirección:



## Coordenadas de las etiquetas del menú:



## El Modo de Prueba

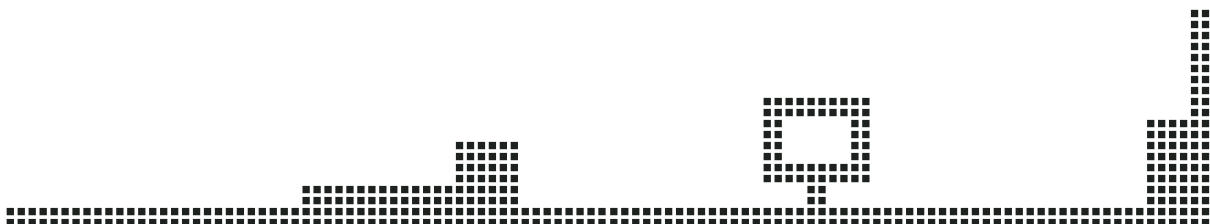
La variable TestMode? se puede utilizar para recopilar datos durante la carga:

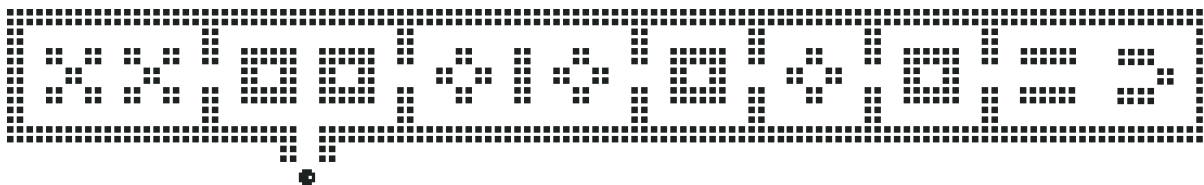
```
TestMode? IT :: #300 #32 DO INDEX@ GETLAM xHALT LOOP ;
```

Esto también se puede hacer durante el juego asignado el código anterior a una tecla

## Puntos diversos:

- ADVERTENCIA: Blocks Master Sobrescribe el GBUFF.
- La copia del mapa grande sirve para utilizarlo como plantilla al diseñar gráficos de fondo, ya que te proporciona las dimensiones exactas.
- Si hay un gráfico en la pila al iniciar el Builder te preguntara si quieres usarlo como mapa.
- Puedes crear tus propios gráficos para componentes usando GLPack (paquete de GrobsList) en el menú CREAR.
- Puedes agregar etiquetas, es decir, textos que aparecen en Gbuff como gráficos de fondo. Para esto usa la opción etiqueta del Menu A.
- Si tienes un mapa ya creado y quieres usarlo para un nuevo juego solo debes colocarlo en la pila antes de iniciar el Builder.
- Puedes guardar copias de los mapas. La del pequeño para reutilizarlo en otras zonas, siguiendo los pasos anteriores.





# *ANEXOS*



# Anexo 1. Módulos y Submódulo

En este anexo presentamos una lista de los módulos y submódulo de BM, organizados en orden alfabético.

En las tablas siguientes cada módulo es representado por su nombre. A la derecha de cada módulo están los módulos que utiliza. Cuando un módulo no utiliza a otros se coloca el símbolo (A) a su derecha.

Los módulos que son utilizados exclusivamente por el Builder se indican con un fondo **amarillo**.  
Los que son usados por el Engine se marcan con un fondo **verde** claro

ABBuilder	Shtx (A)
	GetAbuffDA3 (A)
	SIoNO (A)
	TurnCBOffCopyBG (A)
	AssignToGroup
	AssignStateSB (A)
	PutAbuffDA3 (A)
	TurnCBBuilderOff
	ShowOnHB2 (A)
	ShowResumen
	ShowCancelMsg

ABOff (A)

ABOn (A)

ACC (A)

aEli (A)

AIGMov (A)

Ajustes	InstCompresor (A)
	BzAxPack

Anm	BVCreator
	ZxIni1
	NumListCreator (A)
	TxInp

AnmBGroundUpdater (A)

AnmBuilder	GetAbuffDA3 (A)
	Shtx (A)
	GetCBLISTToInv
	ListaVelocidades (A)
	NumListCreator (A)
	AnmT1Executer
	AnmBGroundUpdater (A)
	AnmT2Executer
	AnmBuilderDatSECSUST
	AnmT3Executer
	AssignToGroup
	AssignStateSB (A)
	TurnCBBuilderOff
	ShowOnHB2 (A)
	ShowResumen
	PutAbuffDA3 (A)
	ShowCancelMsg

AnmBuilderDatSECSUST	Shtx (A)	
AnmCEc1Fixer (A)		
AnmGrobARunner (A)		
AnmRunner	InvCEL2CBLList (A)	
AnmT1Executer	AnmRunner	
	AnmCEc1Fixer (A)	
	AnmGrobARunner (A)	
AnmT2Executer	AnmRunner	
	AnmGrobARunner (A)	
AnmT3Executer	AnmRunner	
aSLStd1 (A)		
aSLStd2 (A)		
aSLStd3 (A)		
AssignStateSB (A)		
AssignToGroup	ListaGrupos	
AtackCodeAIG	GetAIGDistanceEli (A)	
	M3 (A)	
	ShowEliPower (A)	
AtackCodeMB	GetMBDistanceEli (A)	
	ShowEliPower (A)	
AxLoader	ZxLoader	
	ShowEliPower (A)	
AxPackBuilder	BVCreator	
	ZxIni1	
	TxInp	
	NumListCreator	
	TryCompressFile	
	BMver (A)	
AxLoader	ZxLoader	
AxPackIni	AxLoader	ZxLoader

BEngine	ShowEliCoords (A)
	Shtx (A)
	CBBUILDER (A)
	CopyBox0 (A)
	MBBUILDER
	msg16
	PBBUILDER
	OBBUILDER
	CEBUILDER
	ABBUILDER
	YWBUILDER
	CBMode
	CGMenu
	SBMenu
	GData
	GetSBInfo
	ZxCONFIG
	ConfInv
	FxMenu
	GroupsBuilder
	CoordsMenu
	Map1
	Map2
	Map3
	TongleEliCoords
	ExitBLDR
	MenuHBLDR
	GetAbuffDA3
	ZxCFA (A)
	ZxCFP
	PutAbuffDA3 (A)
	ExitENG
	BMrev (A)
	Ubz2 (A)
	Inst

bgIni	ToDMemList (A)
	cgLoader (A)

BLDR	BMver (A)	
	Crear	ZxPackBuilder
		AxPackBuilder
		Anm [1]
		Grb [1]
		GLPack [1]
	Modificar	EdZxPack
		GLPack [5]
	Instalar	Inst
		Anm [2]
		Grb [2]
		GLPack [2]
	Recuperar	Rcl
		Anm [3]
		Grb [3]
		GLPack [3]
	Eliminar	Del
		Anm [4]
		Grb [4]
		GLPack [4]



BMBackup	ShowOnHB2 (A)
	StdGClear
	StdGLoader
BorderGenerator	ShowOnHB2 (A)
BMrev (A)	
BMver (A)	
Bomb	EliPowerDec
BorderGenerator	ShowOnHB2 (A)
BoxGrobSelection	GetAbuffDA3 (A)
	NumListCreator (A)
	GMapCBBUILDER
	PutAbuffDA3 (A)
BVCreator	ShowOnHB2 (A)
	MBi (A)
BZAxPack	BMver (A)
	Ubz2 (A)
CBBUILDER (A)	
CBMode	TurnCBBUILDEROff
	Shtx (A)
	TurnCBBUILDEROn (A)
CE01	GetGroupTTCE
	CEg0Restore
	TongleGroups
	IndTest
CE02A	GetGroupTTCE
	TongleGroups
CE02B	GetCEZoneCorners
	GetGroupTTCE
	Shtx (A)
	TestEliInZone (A)
	TongleGroups
CE03	NumListCreator (A)
	Shtx (A)
	ListaTiposdeGrupo (A)
	ListaGrupos
	TongleGroups
CE04A	NumListCreator (A)
	TxInp
	Shtx (A)

CE04B	NumListCreator (A)
	TxInp
	Shtx (A)
CE05	Shtx (A)
	TeleportLoader (A)
	CopyBox0 (A)
	M3 (A)
	CEStop
	CESstart (A)
	ListaIndices
CE06	Shtx (A)
	WormLoader (A)
	GetAIGDistanceEli (A)
	AIGMov (A)
	AtackCodeAIG
CE07	GetCEZoneCorners
	ListaVelocidades (A)
	TestEliInZone (A)
	EliPowerDecFast (A)
CEBuilder	GetAbuffDA3 (A)
	CE01
	CE02A
	CE02B
	CE03
	CE04A
	CE04B
	CE05
	CE06
	CE07
	AssignToGroup
	AssignStateSB (A)
	TurnCBBuilderOff
	ShowOnHB2 (A)
	ShowResumen
	PutAbuffDA3 (A)
	ShowCancelMsg
CEg0Restore	SearchMBGetIndex (A)
	SearchPBGetIndex (A)
	SearchABGetIndex (A)
CEIni	TeleportLoader (A)
	WormLoader (A)
CESstart (A)	
CESstop	ListSplitterE (A)
cgLoader (A)	

CGMenu	GetAbuffDA3 (A)
	AnmBuilder
	GrbBuilder
	LabelBuilder
	MsgMenu
	BoxGrobSelection
	MapMenu
	PutAbuffDA3 (A)

ChangeEliAnimation (A)
------------------------

CkDescomp (A)
---------------

CkGoalc	RcStop
	Rc

CkIsGLPack (A)
----------------

ClearMenuLabel (A)
--------------------

CloseENGN	ShowOnHB2 (A)
-----------	---------------

CodeInp	ClearMenuLabel (A)
	CodeInpCompKeys

CodeInpCompKeys	ClearMenuLabel (A)
-----------------	--------------------

ConfInv	GetAbuffDA3 (A)
	NumListCreator (A)
	PutAbuffDA3 (A)

ConfKb	GetAbuffDA3 (A)
	Shtx (A)
	RestoreKBStd (A)
	PutAbuffDA3 (A)

ConfM1	GetAbuffDA3 (A)
	ListaDelayFM
	PutAbuffDA3 (A)

ConfSB	GetAbuffDA3 (A)
	NumListCreator (A)
	Shtx (A)
	MBOOn (A)
	MBOff (A)
	PBOn (A)
	PBOff (A)
	OBOOn (A)
	OBOff (A)
	ABOOn (A)
	ABOff (A)
	CESstart (A)
	CESstop
	PutAbuffDA3 (A)

CoordsMenu	GetAbuffDA3 (A)
	CoordsSaver
	CoordsViewer
	TongleEliCoords
	PutAbuffDA3 (A)

CoodsSaver	GetAbuffDA3 (A)
	NumListCreator (A)
	ShTx (A)
	PutAbuffDA3 (A)

CoodsViewer	GetAbuffDA3 (A)
	NumListCreator (A)
	ShTx (A)
	PutAbuffDA3 (A)

CopyBox0 (A)
--------------

Creat	ZxPackBuilder
	AxPackBuilder
	Anm [1]
	Grb [1]
	GLPack [1]

Del	DelZxPack (A)
	DelAxPack (A)

DelAxPack (A)
---------------

DelayFMDwENGn (A)
-------------------

DelayFMUpENGn (A)
-------------------

DelZxPack (A)
---------------

EdZxMaster	TxInp
------------	-------

EdZxName	TxInp
----------	-------

EdZxPack	GetZxPack (A)
	CkDescomp (A)
	GetBMver (A)
	BMver (A)
	ShowOnHB2 (A)
	BVCreator
	ZxLoadV1
	StdGLoader
	SBnFixer
	ZxLoadV2 (A)
	ExpandB
	bgIni
	TestMP
	M0

EliIni	CopyBox0 (A)
	eNRunner (A)

Eliminar	Del
	Anm [4]
	Grb [4]
	GLPack [4]

EliPowerDec (A)
-----------------

EliPowerDecFast (A)
---------------------

EliPowerInc (A)
-----------------

ENG	BMver (A)	
	Play	PlayZxPack
		PlayAxPack
	Inst	BMver (A)
	Rcl	RclZxPack (A)
		RclAxPack (A)
	Del	DelZxPack (A)
		DelAxPack (A)
	Ajustes	InstCompresor (A)
		BzAxPack
	InfE	BMver (A)
		BMrev (A)
		Ubz2 (A)

eNRunner (A)
--------------

Esc	GetAbuffDA3 (A)
	Save
	SxErase (A)
	PutAbuffDA3 (A)

ExitBLDR	GetAbuffDA3 (A)
	PutAbuffDA3 (A)

ExitENG	GetAbuffDA3 (A)
	PutAbuffDA3 (A)
	ZxToZxPack
	RcStop
	Esc

ExpandB	ShowOnHB2 (A)
	ToDMemList (A)

fgIni	ToDMemList (A)
	cgLoader (A)

FxGetState (A)
----------------

FxMenu	GetAbuffDA3 (A)
	NumListCreator (A)
	FxGetState (A)
	Shtx (A)
	FxOn (A)
	FxOff (A)
	PutAbuffDA3 (A)

FxOff (A)
-----------

FxOn (A)	
GBuffIni (A)	
GData	Shtx (A) RndRApndRToStr (A)
GetAbuffDA3 (A)	
GetAIGDistanceEli (A)	
GetBMver (A)	
GetCBListToInv	NumListCreator (A) Shtx (A)
GetCEZoneCorners	NumListCreator (A) Shtx (A)
GetGroupTTCE	ListaTiposdeGrupo (A) ListaGrupos
GetGroupTTOB	ListaTiposdeGrupo (A) ListaGrupos
GetGroupTTPB	ListaTiposdeGrupo (A) ListaGrupos
GetMBDistanceEli (A)	
GetMBGunRange	ListaAlcancesArma (A)
getMBGunStatus (A)	
getMBRadarRange	ListaPotenciasRadar (A)
GetMBSpeed	ListaVelocidades (A)
GetMP	TxInp Shtx (A)
GetSBInfo	ShowResumen
GetZxPack (A)	
GLPack	BVCreator ZxIni1 GLPackIni (A) GLPackInpName GLPackExtractor (A) Ubz2 (A) CkIsGLPack (A)
GLPackCleaner (A)	
GLPackExtractor (A)	

GLPackIni (A)	
GLPackInpName	TxInp
GLPackLoader	ShowOnHB2 (A)
	Ubz2 (A)
	ToDMemList (A)
	aSLStd1 (A)
	aSLStd2 (A)
	aSLStd3 (A)
	ListSplitter (A)
GLPackSelection	GLPStd3 (A)
	GLPStd2 (A)
	GLPStd1 (A)
	GLPackStdOn (A)
	GLPackStdOff (A)
	GLPackLoader
GLPackStdOff (A)	
GLPackStdOn (A)	
GLPStd1 (A)	
GLPStd2 (A)	
GLPStd3 (A)	
GMapCBBUILDER	ShowOnHB2 (A)
GoalcInc (A)	
Grb	BVCreator
	ZxIni1
	NumListCreator (A)
	TxInp
GrbBuilder	GetAbuffDA3 (A)
	CopyBox0 (A)
	ShowResumen
	PutAbuffDA3 (A)
GroupsBuilder	GetAbuffDA3 (A)
	ListaGroupsBuilder (A)
	ListaIndices
	Shtx (A)
	ListaGrupos
	PutAbuffDA3 (A)
IfPBCLashMBTurnMBOff	SearchMBGetIndex (A)
	MBOff (A)
	M3 (A)
IndTest	Shtx (A)

InfE	BMver (A)
	BMrev (A)
	Ubz2 (A)
InitT&M (A)	
Inst	BMver (A)
Instalar	Inst
	Anm [2]
	Grb [2]
	GLPack [2]
InstCompresor (A)	
IntroRutaMB	Shtx (A)
InvBG (A)	
InvCEL2CBList (A)	
Inventory (A)	
L4bz (A)	
L5bz (A)	
L7bz (A)	
LabelBuilder	TxInp
	CopyBox0 (A)
ListaAlcancesArma (A)	
ListaCSM (A)	
ListaDelayFM	L7bz (A)
	Ubz2 (A)
ListaGroupsBuilder (A)	
ListaGrupos	L5bz (A)
	Ubz2 (A)
ListaIndices	L4bz (A)
	Ubz2 (A)
ListaPotenciasRadar (A)	
ListaTiposdeGrupo (A)	
ListaVelocidades (A)	
ListSplitter (A)	
ListSplitterE (A)	



M0	GMapCBBUILDER	ShowOnHB2 (A)
	BorderGenerator	ShowOnHB2 (A)
		GMapCBBUILDER
	SBIIni (A)	
	CEIni	TeleportLoader (A)
		WormLoader (A)
	fgIni	ToDMemList (A)
		cgLoader (A)
	IniT&M (A)	
	GBuffIni (A)	
	EliIni	CopyBox0 (A)
		eNRunner (A)
	M1Rx	ShowEliCoords (A)
		M2
		aEli (A)
		SCMB (A)
		SCCE (A)
		SCUE (A)
		Bomb
		T&M (A)
		ACC (A)
		CkGoalc
		Shtx (A)
	M1ZxSx	Shtx (A)
		BEngine
		ShowEliCoords (A)
		M2
		aEli (A)
		SCMB (A)
		SCCE
		SCUE (A)
		Bomb
		T&M
		ACC (A)
		Rc (A)
		CkGoalc
		msg7
	CloseENGN	ShowOnHB2 (A)

M1Rx	ShowEliCoords (A)
	M2
	aEli (A)
	SCMB (A)
	SCCE (A)
	SCUE (A)
	Bomb
	T&M (A)
	ACC (A)
	CkGoalc
	Shtx (A)

M1ZxSx	Shtx (A)
	BEngine
	ShowEliCoords (A)
	M2
	aEli (A)
	SCMB (A)
	SCCE
	SCUE (A)
	Bomb
	T&M
	ACC (A)
	Rc (A)
	CkGoalc
	msg7

M2	M5
	CopyBox0 (A)
	Shtx (A)
	MenuHENG
	M6
	TxHelpENG
	TxObjENG
	GData
	SetBomb
	Pinch
	Scanner
	Inventory
	RcENG
	SetT&M (A)
	BEngine
	TongleShowCiclos
	TongleAnimateEli
	TongleEliCoords
	DelayFMUpENG (A)
	DelayFMDwENG (A)
	SpeedUpENG
	SpeedDwENG
	Map1
	Map2
	Map3
	ChangeEliAnimation (A)
	ExitENG

M3 (A)

M5      SearchPBGetIndex (A)

M6      SearchOBGetIndex  
Shtx (A)

Map1      Shtx (A)

Map2      Shtx (A)

Map3      Shtx (A)

MapIni (A)

MapMenu	GetAbuffDA3 (A)
	Shtx (A)
	PutAbuffDA3 (A)
MBBuilder	Shtx (A)
	GetAbuffDA3 (A)
	IntroRutaMB
	getMBSpeed
	getMBGunStatus (A)
	getMBGunRange
	MoveMBtoRuta
	AtackCodeMB
	getMBRadarRange
	GetMBDistanceEli (A)
	MoveMBtoEli
	TurnCBOffCopyBG (A)
	AssignToGroup
	AssignStateSB (A)
	PutAbuffDA3 (A)
	TurnCBBuilderOff
	ShowOnHB2 (A)
	ShowResumen
	ShowCancelMsg
MenuHBLDR	GetAbuffDA3
	CGMenu
	SBMenu
	Conflnv
	GroupsBuilder
	FxMenu
	CoordsMenu
	GData
	GetSBInfo
	ZxCONFIG
	BMrev
	Ubz2
	Shtx
	ExitBLDR
	PutAbuffDA3
MenuHENG	GetAbuffDA3
	Inventory
	SetBomb
	Pinch
	Scanner
	M6
	RcENG
	Shtx
	TxHelpENG
	TxObjENG
	SetT&M
	BEngine
	TongleShowCiclos
	TongleAnimateEli
	TongleEliCoords
	GData
	ExitENG
	PutAbuffDA3
MBi (A)	

MBIntModeSelect	ShowOffMsg (A)
	MBOff (A)
	ShowOnMsg (A)
	MBOn (A)
	MBStop
	MBStart (A)
MBMov (A)	
MBOff	ListSplitterE (A)
MBOn (A)	
MBStart (A)	
MBStop	ListSplitterE (A)
Modificar	EdZxPack
	GLPack [5]
MoveMBtoEli	MBMov (A)
MoveMBtoRuta	MBMov (A)
msg7	Shtx (A)
msg16	Shtx (A)
msg18	Shtx (A)
MsgBBUILDER	Shtx (A)
	NumListCreator (A)
	TxInp
MsgHBUILDER	Shtx (A)
	NumListCreator (A)
	TxInp
MsgMenu	GetAbuffDA3 (A)
	MsgBBUILDER
	MsgHBUILDER
	MsgOBUILDER
	PutAbuffDA3 (A)
MsgOBUILDER	Shtx (A)
	NumListCreator (A)
	TxInp
NBombsInc (A)	
NPinchInc (A)	
NumListCreator (A)	

OBBuilder	Shtx (A)
	GetAbuffDA3 (A)
	WorldKeysInc (A)
	OBClr (A)
	NBombsInc (A)
	NPinchInc (A)
	EliPowerInc (A)
	WorldMapInc (A)
	MBIntModeSelect
	ShowOffMsg (A)
	PBOff (A)
	ShowOnMsg (A)
	PBOn (A)
	OBOff (A)
	OBOOn (A)
	ABOff (A)
	ABOn (A)
	CESstop
	CESstart (A)
	FxOff (A)
	FxOn (A)
	FxGetState (A)
	ListaIndices
	NumListCreator (A)
	TxInp
	GoalcInc (A)
	GetGroupTTOB
	TongleGroups
	msg18
	CodeInp
	ListaVelocidades (A)
	ListSplitterU (A)
	TurnCBOffCopyBG (A)
	AssignToGroup
	AssignStateSB (A)
	PutAbuffDA3 (A)
	TurnCBBUILDEROff
	ShowOnHB2 (A)
	ShowResumen
	ShowCancelMsg

OBClr (A)

OBOff (A)

OBOOn (A)

PBBuilder	Shtx (A)
	GetAbuffDA3 (A)
	PBMov (A)
	PBOff (A)
	PBOn (A)
	mg18
	CodeInp
	ShowOffMsg (A)
	ShowOnMsg (A)
	MBOff
	MBOn (A)
	OBOff (A)
	OBOOn (A)
	ABOff (A)
	ABOn (A)
	CESstop
	CESstart (A)
	FxOff (A)
	FxOn (A)
	FxGetState (A)
	ListaIndices
	GetGroupT*TPB
	TongleGroups
	NumListCreator (A)
	ListaCSM (A)
	IfPBclashMBTurnMBOff
	TurnCBOffCopyBG (A)
	AssignToGroup
	AssignStateSB (A)
	PutAbuffDA3 (A)
	TurnCBBuilderOff
	ShowOnHB2 (A)
	ShowResumen
	ShowCancelMsg

PBMov (A)

PBOff (A)

PBOn (A)

Pinch	M3 (A)
	SearchMBGetIndex (A)
	MBOff
	Shtx (A)

Play	PlayZxPack
	PlayAxPack

PlayAxPack	AxPackIni	AxLoader
------------	-----------	----------

PlayZxPack	ZxPackIni	ZxLoader
------------	-----------	----------

PutAbuffDA3 (A)

Rc (A)

RcENGN	Shtx (A)
	RcStop
	GetAbuffDA3 (A)
	PutAbuffDA3 (A)
	RcStart
	RxErase (A)

Rcl	RclZxPack (A)
	RclAxPack (A)

RclAxPack (A)
---------------

RclZxPack (A)
---------------

RcStart	BMBBackup
---------	-----------

RcStop	TryCompressFile (A)
--------	---------------------

Recuperar	Rcl
	Anm [3]
	Grb [3]
	GLPack [3]

RestoreKBStd (A)
------------------

RndRApndRToStr (A)
--------------------

RxErase (A)
-------------

Save	BMBBackup
	TryCompressFile (A)

SBIini (A)
------------

SBMenu	GetAbuffDA3 (A)
	CBMode
	MBBuilder
	msg16
	PBBuilder
	OBBuilder
	ABBuilder
	CEBuilder
	YWBuilder
	ConfSB
	PutAbuffDA3 (A)

SBnFixer	MBi (A)
----------	---------

Scanner	M3 (A)
	SearchOBGetIndex (A)
	MBOff
	Shtx (A)

SCCE (A)
----------

SCMB (A)
----------

SCUE (A)
----------

SearchABGetIndex (A)		
SearchMBGetIndex (A)		
SearchOBGetIndex (A)		
SearchPBGetIndex (A)		
SetBomb	Shtx (A)	
SetT&M (A)		
ShowCancelMsg	Shtx (A)	
ShowEliCoords (A)		
ShowEliPower (A)		
ShowOffMsg (A)		
ShowOnHB2 (A)		
ShowOnMsg (A)		
ShowResumen	Shtx (A)	
Shtx (A)		
SLoNO (A)		
SpeedDwENGNG (A)		
SpeedUpENGNG (A)		
StdGClear	GLPackCleaner (A)	
StdGLoader	InvBG (A)	
	YouWinMsjStd (A)	
	GLPackLoader	ShowOnHB2 (A)
		Ubz2 (A)
		ToDMemList (A)
		aSLStd1 (A)
		aSLStd2 (A)
aSLStd3 (A)		
ListSplitter (A)		
SxErase (A)		
T&M (A)		
TeleportLoader (A)		
TestEliInZone (A)		
TestMP	TxInp	
	Shtx (A)	



ToDMemList (A)	
TongleAnimateEli	ShowOffMsg (A)
	ShowOnMsg (A)
TongleEliCoords	ClearMenuLabel (A)
	ShowEliCoords (A)
TongleGroups	MBOff (A)
	PBOff (A)
	OBOff (A)
	ABOff (A)
	CEStop
	FxOff (A)
	MBOOn (A)
	PBOn (A)
	OBOn (A)
	ABOn (A)
	CEStart (A)
	FxOn (A)
TongleShowCiclos	ClearMenuLabel (A)
TryCompressFile (A)	
TxBGList (A)	
TurnCBBuilderOff	ClearMenuLabel (A)
TurnCBBuilderOn (A)	
TurnCBOffCopyBG (A)	
TxBSpc (A)	
TxHelpENGn	Shtx (A)
TxInp	TxBGList (A)
	Ubz2 (A)
	ShowOnHB2 (A)
	TxUpd (A)
	TxBSpc (A)
TxObjENGn	Shtx (A)
TxUpd (A)	
Ubz2 (A)	
WorldKeysInc (A)	
WorldMapInc (A)	
WormLoader (A)	
YouWinMsjStd	Shtx (A)

YW01	M3 (A)
	Shtx (A)
YW02B	NumListCreator (A)
	Shtx (A)
	M3 (A)
YW02D	NumListCreator (A)
	Shtx (A)
	M3 (A)
YW03	NumListCreator (A)
YW04	NumListCreator (A)
YWBuilder	GetAbuffDA3 (A)
	YW01
	YW02B
	YW02D
	YW03
	YW04
	AssignToGroup
	AssignStateSB (A)
	ShowOnHB2 (A)
	ShowResumen
	PutAbuffDA3 (A)
	TurnCBBUILDEROff
	Shtx (A)
ZxCFA (A)	
ZxCFP	GetAbuffDA3 (A)
	SIoNO (A)
	NumListCreator (A)
	PutAbuffDA3 (A)
ZxCONFIG	GetAbuffDA3 (A)
	EdZxName
	EdZxMaster
	GetMP
	ConfInv
	ConfM1
	ConfSB
	ConfKb
	FxMenu
	BoxGrobSelection
	PutAbuffDA3 (A)
ZxIni1	ShowOnHB2 (A)
	BMver (A)
	InvBG (A)
	YouWinMsjStd
	RestoreKBStd (A)
	MBi (A)
ZxIni2	ShowOnHB2 (A)

ZxLoader	CkDescomp (A)
	GetBMver (A)
	BMver (A)
	ShowOnHB2 (A)
	StdGLoader
	ToDMemList (A)
	ExpandB
	bgIni
	M0

ZxLoadV1	MBi (A)
----------	---------

ZxLoadV2 (A)
--------------

ZxPackBuilder	BVCreator
	ZxIni1
	SBnFixer
	MapIni (A)
	GLPackSelection
	ZxIni2
	ExpandB
	EdZxName
	EdZxMaster
	bgIni
	M0

ZxPackIni	ZxLoader	CkDescomp (A)
		GetBMver (A)
		BMver (A)
		ShowOnHB2 (A)
		StdGLoader
		ToDMemList (A)
		ExpandB
		bgIni
		M0

ZxToZxPack	BMver (A)
	BMBackup

## Anexo 2. Clasificación de los Módulos

Módulos del Builder (total: 287):

ABBuilder	CEg0Restore	GetAIGDistanceEli
ABOff	CEIni	GetBMver
ABOn	CEStart	GetCBLISTToInv
ACC	CEStop	GetCEZoneCorners
aEli	cgLoader	GetGroupTTCE
AIGMov	CGMenu	GetGroupTTOB
Ajustes	ChangeEliAnimation	GetGroupTTPB
Anm	CkDescomp	GetMBDistanceEli
AnmBGroundUpdater	CkGoalc	getMBGunRange
AnmBuilder	CkIsGLPack	getMBGunStatus
AnmBuilderDatSECSUST	ClearMenuLabel	getMBRadarRange
AnmCEc1Fixer	CloseENG	getMBSpeed
AnmGrobARunner	CodeInp	GetMP
AnmRunner	CodeInpCompKeys	GetSBInfo
AnmT1Executer	ConfInv	GetZxPack
AnmT2Executer	ConfKb	GLPack
AnmT3Executer	ConfM1	GLPackCleaner
aSLStd1	ConfSB	GLPackExtractor
aSLStd2	CoordsMenu	GLPackIni
aSLStd3	CoordsSaver	GLPackInpName
AssignStateSB	CoordsViewer	GLPackLoader
AssignToGroup	CopyBox0	GLPackSelection
AttackCodeAIG	Crear	GLPackStdOff
AttackCodeMB	Del	GLPackStdOn
AxLoader	DelAxPack	GLPStd1
AxPackBuilder	DelayFMDwENG	GLPStd2
AxPackIni	DelayFMUpENG	GLPStd3
BEngine	DelZxPack	GMapCBBBuilder
bgIni	EdZxMaster	GoalcInc
BLDR	EdZxName	Grb
BMBBackup	EdZxPack	GrbBuilder
BMrev	EliIni	GroupsBuilder
BMver	Eliminar	IfPBCLashMBTurnMBOff
Bomb	EliPowerDec	IndTest
BorderGenerator	EliPowerDecFast	InFE
BoxGrobSelection	EliPowerInc	InitT&M
BVCreator	ENG	Inst
BZAxPack	eNRunner	Instalar
CBBBuilder	Esc	InstCompresor
CBMode	ExitBLDR	IntroRutaMB
CE01	ExitENG	InvBG
CE02A	ExpandB	InvCEL2CBLIST
CE02B	fgIni	Inventory
CE03	FxGetState	L4bz
CE04A	FxMenu	L5bz
CE04B	FxOff	L7bz
CE05	FxOn	LabelBuilder
CE06	GBuffIni	ListaAlcancesArma
CE07	GData	ListaCSM
CEBuilder	GetAbuffDA3	ListaDelayFM

ListaGroupsBuilder	PBOff	ToDMemList
ListaGrupos	PBOn	TongleAnimateEli
ListaIndices	Pinch	TongleEliCoords
ListaPotenciasRadar	Play	TongleGroups
ListaTiposdeGrupo	PlayAxPack	TongleShowCiclos
ListaVelocidades	PlayZxPack	TryCompressFile
ListSplitter	PutAbuffDA3	TurnCBBuilderOff
ListSplitterE	Rc	TurnCBBuilderOn
ListSplitterU	RcENGn	TurnCBOffCopyBG
M0	Rcl	TxBGList
M1Rx	RclAxPack	TxBSpC
M1ZxSx	RclZxPack	TxHelpENGn
M2	RcStart	TxInp
M3	RcStop	TxObjENGn
M5	Recuperar	TxUpd
M6	RestoreKBStd	Ubz2
Map1	RndRApndRToStr	WorldKeysInc
Map2	RxErase	WorldMapInc
Map3	Save	WormLoader
MapIni	SBIIni	YouWinMsjStd
MapMenu	SBMenu	YW01
MBBuilder	SBnFixer	YW02B
MBi	Scanner	YW02D
MBIntModeSelect	SCCE	YW03
MBMov	SCMB	YW04
MBOff	SCUE	YWBuilder
MBOn	SearchABGetIndex	ZxCFA
MBStart	SearchMBGetIndex	ZxCFP
MBStop	SearchOBGetIndex	ZxCONFG
MenuHBLDR	SearchPBGetIndex	ZxIni1
MenuHENGn	SetBomb	ZxIni2
Modificar	SetT&M	ZxLoader
MoveMBtoEli	ShowCancelMsg	ZxLoadV1
MoveMBtoRuta	ShowEliCoords	ZxLoadV2
msg16	ShowEliPower	ZxPackBuilder
msg18	ShowOffMsg	ZxPackIni
msg7	ShowOnHB2	ZxToZxPack
MsgBBBuilder	ShowOnMsg	
MsgHBuilder	ShowResumen	
MsgMenu	Shtx	
MsgOBuilder	SIoNO	
NBombsInc	SpeedDwENGn	
NPinchInc	SpeedUpENGn	
NumListCreator	StdGClear	
OBBuilder	StdGLoader	
OBClr	SxErase	
OBOff	T&M	
OBOOn	TeleportLoader	
PBBuilder	TestEliInZone	
PBMov	TestMP	

## Módulos del Engine (Total 177):

ABOff	GBuffIni	OBOOn	TongleShowCiclos
ABOn	GData	PBMov	TryCompressFile
ACC	GLPStd1	PBOff	TxBGList
AIGMov	GLPStd2	PBOn	TxBSpC
Ajustes	GLPStd3	Pinch	TxHelpENG
AnmCEc1Fixer	GLPackCleaner	Play	TxInp
AnmGrobARunner	GLPackLoader	PlayAxPack	TxObjENG
AnmRunner	GMapCBBUILDER	PlayZxPack	TxUpd
AnmT1Executer	GetAIGDistanceEli	PutAbuffDA3	Ubz2
AnmT2Executer	GetAbuffDA3	Rc	WorldKeysInc
AnmT3Executer	GetBMver	RcENG	WorldMapInc
AtackCodeAIG	GetMBDistanceEli	RcStart	WormLoader
AtackCodeMB	GetMP	RcStop	YouWinMsjStd
AxLoader	GetZxPack	Rcl	ZxLoadV1
AxPackIni	GoalcInc	RclAxPack	ZxLoadV2
BLDR	IfPBCLashMBTurnMBOff	RclZxPack	ZxLoader
BMBakup	IndTest	RndRApndRToStr	ZxPackIni
BMrev	InfE	RxErase	ZxToZxPack
BMver	InitT&M	SBIIni	aEli
BZAxPack	Inst	SCCE	aSLStd1
Bomb	InstCompresor	SCMB	aSLStd2
CEIni	InvBG	SCUE	aSLStd3
CESart	InvCEL2CBList	SIoNO	bgIni
CESop	Inventory	Save	cgLoader
CEg0Restore	ListSplitter	Scanner	eNRunner
ChangeEliAnimation	ListSplitterE	SearchABGetIndex	fgIni
CkDescomp	ListSplitterU	SearchMBGetIndex	msg18
CkGoalc	M0	SearchOBGetIndex	
CkIsGLPack	M1Rx	SearchPBGetIndex	
ClearMenuLabel	M1ZxSx	SetBomb	
CloseENG	M2	SetT&M	
CodeInp	M3	ShowEliCoords	
CodeInpCompKeys	M5	ShowEliPower	
CopyBox0	M6	ShowOffMsg	
Del	MBMov	ShowOnHB2	
DelAxPack	MBOff	ShowOnMsg	
DelZxPack	MBOn	Shtx	
DelayFMDwENG	MBStart	SpeedDwENG	
DelayFMUpENG	MBStop	SpeedUpENG	
EliIni	MBi	StdGClear	
EliPowerDec	Map1	StdGLoader	
EliPowerDecFast	Map2	SxErase	
EliPowerInc	Map3	T&M	
ENG	MenuHENG	TeleportLoader	
Esc	MoveMBtoEli	TestEliInZone	
ExitENG	MoveMBtoRuta	TestMP	
ExpandB	NBombsInc	ToDMemList	
FxGetState	NPinchInc	TongleAnimateEli	
FxOff	OBClr	TongleEliCoords	
FxOn	OBOff	TongleGroups	

El módulo BLDR se conserva en el Engine para no alterar las referencias a los módulos de la librería en los archivos de juego, pero está oculto.

### Módulos contenedores de gráficos:

aSLStd1	GLPStd1	TxBGList
aSLStd2	GLPStd2	InvBG
aSLStd3	GLPStd3	

### Módulos contenedores de listas no comprimidas:

L7bz (usado por ListaDelayFM)	ListaAlcancesArma	ListaCSM
L5bz (usado por ListaGrupos)	ListaPotenciasRadar	ListaGroupsBuilder
L4bz (usado por ListaIndices)	ListaTiposdeGrupo	
	ListaVelocidades	

### Módulos comprimidos:

aSLStd1	GLPStd1	L4bz	TxBGList
aSLStd2	GLPStd2	L5bz	InvBG
aSLStd3	GLPStd3	L7bz	BMrev

Nota: el módulo BMrev es creado y comprimido automáticamente por Link )

### Módulos contenedores de mensajes de texto:

msg7
msg16
msg18

### Módulos que no usan tabla de definiciones de variables (\$TDV):

Ajustes	GetAbuffDA3	ListaPotenciasRadar	PlayZxPack
aSLStd1	GetBMver	ListaTiposdeGrupo	PutAbuffDA3
aSLStd2	GetZxPack	ListaVelocidades	Rcl
aSLStd3	GLPStd1	ListSplitter	RclAxPack
AxLoader	GLPStd2	ListSplitterE	RclZxPack
AxPackIni	GLPStd3	ListSplitterU	Recuperar
BLDR	InfE	MBi	RxErase
BMrev	Instalar	Modificar	ShowCancelMsg
BMver	InvBG	msg16	ShowOffMsg
BZAxPack	L4bz	msg18	ShowOnHB2
Crear	L5bz	msg7	ShowOnMsg
Del	L7bz	NumListCreator	SloNO
DelAxPack	ListaAlcancesArma	Play	TxBGList
DelZxPack	ListaDelayFM	PlayAxPack	Ubz2
Eliminar	ListaGrupos		YouWinMsjStd
ENG	ListaIndices		ZxPackIni

## Anexo 3. Variables Lambda

NÚMERO		NOMBRE	TIPO	DESCRIPCIÓN
DEC	HEX			
1	1	BMver	\$	Versión de BM. Tiene <b>restricción posicional</b> por el módulo GetBMver.
2	2	dH	#	Distancia vertical a desplazar ventana en GBUFF $= dH_{pix} * Y_{Jump}$ .
3	3	dW	#	Distancia horizontal a desplazar ventana en GBUFF $= dW_{pix} * X_{Jump}$ .
4	4	eH+dH	#	Variable especial = $eH + dH$ . Creada para evitar la repetición de este cálculo en cada ciclo de M1ZxSx.
5	5	eW+dW	#	Variable especial = $eW + dW$ . Creada para evitar la repetición de este cálculo en cada ciclo de M1ZxSx.
6	6	MBActual	#	Número de la primera variable lambda del MB con que se está trabajando en un momento determinado.
7	7	PBActual	#	Número de la primera variable lambda del PB con que se está trabajando en un momento determinado.
8	8	OBActual	#	Número de la primera variable lambda del OB con que se está trabajando en un momento determinado.
9	9	ABActual	#	Número de la primera variable lambda del AB con que se está trabajando en un momento determinado.
10	A	CEActual	#	Número de la primera variable lambda del CE con que se está trabajando en un momento determinado.
11	B	MBi	#	Número de la primera variable lambda de los MB.
12	C	PBi	#	Número de la primera variable lambda de los PB.
13	D	OBi	#	Número de la primera variable lambda de los OB.
14	E	ABi	#	Número de la primera variable lambda de los AB.
15	F	CEi	#	Número de la primera variable lambda de los CE.
16	10	Box0	grob	Gráfico que está debajo de Eli. Este sustituye a Eli cuando se retira de una posición.
17	11	Xeli	#	Coordenada X de Eli en GBUFF = $X_{pix} * X_{Jump}$ .
18	12	Yeli	#	Coordenada Y de Eli en GBUFF = $Y_{pix} * Y_{Jump}$ .
19	13	Xpix	#	Coordenada X de Eli en ABUFF.
20	14	Ypix	#	Coordenada Y de Eli en ABUFF.
21	15	XJump	#	Cantidad de pixeles de salto horizontal = $BoxGrobW + InterBox$ .
22	16	YJump	#	Cantidad de pixeles de salto vertical = $BoxGrobH + InterBox$ .
23	17	eLEN	#	Cantidad de diapositivas de la secuencia de animación de Eli (Seceli).
24	18	CeLEN	#	Contador de las diapositivas de Eli. Indica cuál diapositiva de la secuencia es la actual.
25	19	Seceli	lista	Secuencia actual de la animación de Eli.
26	1A	AnimateEli?	flag	Indica si la animación de Eli está activada. Usada por aEli. Modificada por TongleAnimateEli
27	1B	EliGrob	grob	Gráfico básico de Eli. Usado para calcular dimensiones de Eli y por M1ZxSx y M1Rx cuando AnimateEli? esta desactivado.
28	1C	eNList	lista	Contiene todas las secuencias de Eli.
29	1D	eNSecini	#	Índice en eNList de la secuencia con que se iniciará el juego. El valor estandar es #1
30	1E	EliBoost	#	Contiene la cantidad de espacio de impulso de Eli. Actualizada por M2. Utilizada por algunos PB para ser movidos.
31	1F	EliPowDown?	flag	Indica si la reducción cíclica de energía de Eli está activada. Usada por ACC
32	20	FMDelayOn?	flag	Indica si el retardo del primer movimiento de Eli está activado. Usado por M1ZxSx, y modificado por DelayFMDwENGn y DelayFMUpENGn.
33	21	FMDelay?	flag	Indica si esta activado el retardo del primer movimiento de Eli.
34	22	DelayFM	%	Tiempo de retardo del primer movimiento de Eli. Usado por M1ZxSx y BEngine. Modificado por DelayFMUpENGn y DelayFMDwENGn.
35	23	DelayAniEli	%	Tiempo de retardo de la animación de Eli. Usada por aEli. Modificado por SpeedUpENGn y SpeedDwENGn
36	24	Map?	flag	Indica al BLDR si ya hay un mapa (Editando ZxPack) o si se va a crear un mapa nuevo. Usado por M0. Indica al ENGn si el jugador posee el mapa, para mostrarlo al presionar las teclas V, W o X.
37	25	Map	grob	Mapa para ABUFF.
38	26	MapH	#	Altura del Mapa en ABUFF.
39	27	MapW	#	Anchura del Mapa en ABUFF.
40	28	2MapH	#	Variable especial = $2 * MapH$ . Usada para acceder. Acceder al area de bloques especiales den ABUFF.
41	29	Hgbuffinpix	#	Altura del mapa en GBUFF = $MapH * Y_{Jump}$ .



42	2A	Wgbuffinpix	#	Anchura del mapa en GBUFF = MapW * XJump.
43	2B	InterBox	#	Número de pixeles entre bloques en GBUFF
44	2C	MBf	#	Número de la última variable lambda de los MB. Usada por BMBBackup.
45	2D	MBActualA	#	Usada por algunos módulos para guardar una copia de MBActual.
46	2E	MBNum	#	Cantidad de MB del juego.
47	2F	MBGrobList	lista	Lista de gráficos para los MB.
48	30	ListMBtoRun	lista	Usada por SCMB (Sistema de control de los MB). Contiene la lista de MBs que serán evaluados en cada ciclo de M1ZxSx o M1Rx.
49	31	SizeListMBtoRun	#	Contiene la cantidad de Mbss de la lista ListMBtoRun
50	32	RunListMBtoRun?	flag	Usado por SCMB. Sirve para desactivar dicho sistema indicándole si debe o no evaluar los MB de la lista ListMBtoRun.
51	33	ListListMBtoRun	lista	Grupos de MBs. Lista que contiene 16 sublistas para los 16 grupos de MB.
52	34	MBStep	#	Indica al módulo MBMove en que dirección ha de mover el MB.
53	35	PBf	#	Número de la última variable lambda de los PB. Usada por BMBBackup.
54	36	PBActualA	#	Usada por algunos módulos para guardar una copia de PBActual.
55	37	PBNum	#	Cantidad de PB del juego.
56	38	PBGrobList	lista	Lista de gráficos para los PB.
57	39	ListListPBtoRun	lista	Grupos de PBs. Lista que contiene 16 sublistas para los 16 grupos de PB
58	3A	OBf	#	Número de la última variable lambda de los OB. Usada por BMBBackup.
59	3B	OBAActualA	#	Usada por algunos módulos para guardar una copia de OBAActual.
60	3C	OBNum	#	Cantidad de OB del juego.
61	3D	OBGrobList	lista	Lista de gráficos para los OB.
62	3E	ListListOBtoRun	lista	Grupos de OBs. Lista que contiene 16 sublistas para los 16 grupos de OB
63	3F	ABf	#	Número de la última variable lambda de los AB. Usada por BMBBackup.
64	40	ABNum	#	Cantidad de AB del juego.
65	41	ABActualA	#	Usada por algunos módulos para guardar una copia de ABActual.
66	42	ABGrobList	lista	Lista de gráficos para los AB.
67	43	ListListABtoRun	lista	Grupos de ABs. Lista que contiene 16 sublistas para los 16 grupos de AB
68	44	CEf	#	Número de la última variable lambda de los CE. Usada por BMBBackup.
69	45	CENum	#	Cantidad de CE del juego.
70	46	ListCEtoRun	lista	Usada por SCCE (Sistema de control de los CE). Contiene la lista de CEs que serán evaluados en cada ciclo de M1ZxSx o M1Rx.
71	47	SizeListCEtoRun	#	Contiene la cantidad de CEs de la lista ListCEtoRun.
72	48	ListListCEtoRun	lista	Grupos de CEs. Lista que contiene 16 sublistas para los 16 grupos de CE.
73	49	RunListCEtoRun?	flag	Usado por SCCE. Sirve para desactivar dicho sistema indicándole si debe o no evaluar los CE de la lista ListCEtoRun.
74	4A	ListUEtoRun	lista	Usada por SCUE (Sistema de control de los UE). Contiene la lista de UEs que serán evaluados en cada ciclo de M1ZxSx o M1Rx.
75	4B	RunListUEtoRun?	flag	Usado por SCUE (Sistema de control de los UE). Sirve para desactivar dicho sistema indicándole si debe o no evaluar los UE de la lista ListUEtoRun.
76	4C	SizeListUEtoRun	#	Contiene la cantidad de UEs de la lista ListUEtoRun
77	4D	CBGrobList	lista	Lista de gráficos para los CB.
78	4E	BoxGrobindex	#	Contiene el índice del gráfico de CBGrobList utilizado para los CB y guardado en Boxgrob. Utilizado por GLPackLoader. El valor por defecto es #1
79	4F	BoxGrob	grob	Gráfico usado para los CB.
80	50	BoxGrobH	#	Altura del gráfico BoxGrob.
81	51	BoxGrobW	#	Anchura del gráfico BoxGrob.
82	52	Nciclos	#	Usado por reloj de BM y por AAC. Contador de ciclos de M1ZxSx o M1Rx.
83	53	cNciclos	#	Usado por AAC. Indica el número del ciclo en que se volverá a restar energía a Eli.
84	54	Nciclosinc	#	Usado por AAC. Indica cada cuantos ciclos se restará energía a Eli.
85	55	bgGrobList	lista	Lista con los gráficos de fondo de la Zona. Usada por bgIni.
86	56	bgGrobList?	flag	Indica si la Zona contiene gráficos de fondo. Usado por bgIni al cargar el juego.
87	57	fgGrobList	lista	Lista de gráficos de sobrefondo. Usado por fgIni.
88	58	fgGrobList?	flag	Indica si la Zona tiene gráficos de sobrefondo. Usado por fgIni al ser llamado por M0.
89	59	InvBGrobStd?	flag	Indica si el gráfico fondo del inventario es estandar (inf no utilizado).
90	5A	InvBGrob	grob	Gráfico de fondo de ventana de inventario.
91	5B	InvBGh	#	Altura de ventana de inventario
92	5C	InvBGw	#	Anchura de ventana de inventario.
93	5D	InvdX	#	Distancia desde la línea derecha de la ventana actual hasta la del inventario.
94	5E	InvdY	#	Distancia desde la línea superior de la ventana actual hasta la del inventario.
95	5F	InvX	#	Coordenada X de ventana de inventario.
96	60	InvY	#	Coordenada Y de ventana de inventario.
97	61	InvGrob0	grob	Usada para guardar gráfico de fondo de ventana de inventario.
98	62	TeleT?	flag	Usada por los telepuertos. Autoriza/desautoriza teletransportaciones. Esto es necesario para evitar que Eli sea devuelto luego de aparecer en el telepuerto de destino.

99	63	TeleportFree?	flag	Indicando a BUILDER si hay un telepuerto sin enlazar
100	64	TelepDestino		Usada por los telepuertos para decir a los otros la actual del telepuerto de destino.
101	65	Goaln	#	Indica la cantidad de metas que se debe lograr para ganar el juego. Usada por CkGoalc.
102	66	Goalc	#	Cuando este contador llega a cierto valor se gana el juego.
103	67	Goal?	flag	Indica si se va a verificar el contador de metas. Esta verificación consiste en comprobar si Goalc y Goaln tienen el mismo valor. Usada por CkGoalc.
104	68	YouWin	comp	Programa que se ejecuta cuando se gana el juego.
105	69	YouWinStd?	flag	Indica si el mensaje de salida YouWin que se ejecuta al ganar el juego es estándar. Usado por CkGoalc, y desactivada por YW01, YW02B y YW02D.
106	6A	aniH	#	Altura de las diapositivas de la animación. Usada por M3.
107	6B	aniW	#	Anchura de las diapositivas de la animación. Usada por M3.
108	6C	aniX	#	Coordenada X en GBUFF de animación actual. Usada por M3.
109	6D	aniY	#	Coordenada Y en GBUFF de animación actual. Usada por M3.
110	6E	anig0	grob	Usado por M3 para guardar el gráfico de fondo de la animación.
111	6F	aniDelay	%	Tiempo entre diapositivas de animaciones. Usada por M3.
112	70	aniLEN	#	Indica a M3 la cantidad de diapositivas de la animación actual.
113	71	aniFlag	flag	Indica a M3 si luego de ejecutar la animación actual debe devolver GBUFF a su estado inicial o si debe dejar el último grob de la animación.
114	72	aniMode	#	Indica a M3 el modo de animación actual: #2 para Gxor, #1 para Gor y Otro bint cualquiera para GROB!
115	73	Xbomb	#	Usado por Bomb. Contiene la coordenada X de la bomba actual en GBUFF.
116	74	Ybomb	#	Usado por Bomb. Contiene la coordenada Y de la bomba actual en GBUFF.
117	75	Bomb?	flag	Indica si hay una bomba colocada. Ayuda a que no se coloque más de una bomba a la vez. Usada por SetBomb.
118	76	Tbomb	#	Usado por Bomb. Contador de bomba. Cuando Tbomb llega a #A se produce la detonación de la bomba.
119	77	CTbomb	#	Usado por Bomb. Contador de ciclos de M1 para la bomba. Se incrementa en #1 en cada ciclo de M1. Cuando CTbomb llega a #2 se reinicia y Tbomb es incrementado en #1
120	78	BombType	#	Usada por Bomb. Indica el tipo de bomba.
121	79	Xbombpix	#	Usado por Bomb. Contiene la coordenada X de la bomba actual en ABUFF.
122	7A	Ybombpix	#	Usado por Bomb. Contiene la coordenada Y de la bomba actual en ABUFF.
123	7B	BombGrob	grob	Gráfico para el contador de bombas. Actualizado por GLPackLoader. Utilizado por Bomb. Décimos gráfico de CBGrobList.
124	7C	AfterBombGrob	grob	Usada por Bomb. Contiene el gráfico que se colocara en GBUFF luego de la explosión. Gráfico #B de CBGrobList.
125	7D	txList	lista	Usada por Shtx para guardar la lista con el texto a mostrar.
126	7E	txN	#	Usada por Shtx para guardar la cantidad de líneas del texto a mostrar.
127	7F	txW	#	Usada por Shtx para guardar la anchura en píxeles de la ventana de texto.
128	80	txH	#	Usada por Shtx para guardar la altura en píxeles de la ventana de texto.
129	81	txG0	grob	Usado por Shtx para guardar gráfico de fondo de ventana de texto.
130	82	txX1	#	Usada por Shtx para guardar coordenada X de ventana de texto.
131	83	txY1	#	Usada por Shtx para guardar coordenada Y de ventana de texto.
132	84	ShtxDelay	%	Usada por Shtx. Contiene el tiempo de espera luego de presentar mensaje. Es una fracción de tiempo durante la cual el teclado no responde. Sirve para evitar que se quite el mensaje sin leerlo.
133	85	ShtxDelay0	%	Usada por Shtx. Contiene el valor por defecto de ShtxDelay.
134	86	ShtxKEY	#	Número de la tecla presionada luego del mensaje mostrado por módulo Shtx. Actualizado por Shtx.
135	87	EliPower	#	Cantidad de Energía de Eli.
136	88	EliPowerIni	#	Cantidad de energía inicial de Eli.
137	89	NBombs	#	Cantidad inicial de bombas o arma 1
138	8A	NPinch	#	Cantidad de municiones de Pinch o arma 2.
139	8B	PinchOn?	flag	Indica si se posee el detonador Pinch.
140	8C	WorldKeys	#	Cantidad de llaves en el inventario.
141	8D	WorldMap?	flag	Indica si el jugador posee el mapa de la zona. En caso afirmativo se le muestra al presionar las teclas correspondientes.
142	8E	Shield?	flag	Arma que permite pasar por zonas de reducción de energía sin ser afectado.
143	8F	ScannerOn?	flag	Indica si Eli posee el Detector de OBs.
144	90	S+?	flag	El resultado de escaneo es positivo. Se ha detectado un OB.
145	91	F1	flag	Estado del interruptor F1.
146	92	F2	flag	Estado del interruptor F2.
147	93	F3	flag	Estado del interruptor F3.
148	94	F4	flag	Estado del interruptor F4.

149	95	F5	flag	Estado del interruptor F5.
150	96	F6	flag	Estado del interruptor F6.
151	97	F7	flag	Estado del interruptor F7.
152	98	F8	flag	Estado del interruptor F8.
153	99	F9	flag	Estado del interruptor F9.
154	9A	FA	flag	Estado del interruptor FA.
155	9B	FB	flag	Estado del interruptor FB.
156	9C	FC	flag	Estado del interruptor FC.
157	9D	FD	flag	Estado del interruptor FD.
158	9E	FE	flag	Estado del interruptor FE.
159	9F	FF	flag	Estado del interruptor FF.
160	A0	ListListFxtorun	lista	Grupos de Fx. Lista que contiene 16 sublistas para los 16 grupos de Fx
161	A1	TxMode	#	Usada para indicarle a TxInp el modo en que debe operar.
162	A2	TxBG	grob	Usada por TxInp para guardar barra de fondo del texto.
163	A3	TxNChars	#	Usada por TxInp para guardar el número de caracteres por línea.
164	A4	InpX	#	Usada por TxInp para guardar coordenada X relativa del texto.
165	A5	InpY	#	Usada por TxInp para guardar coordenada Y relativa del texto.
166	A6	InpK	#	Usada por TxInp para guardar el número de la tecla actual presionada durante la entrada de texto.
167	A7	InpText	\$	Usado por TxInp para guardar texto introducido.
168	A8	Inpc	#	Usado por TxInp como contador de caracteres.
169	A9	Inpg0	grob	Usado por TxInp para guardar gráfico de fondo.
170	AA	Inppl	#	Usado por TxInp para guardar el plano de la tecla presionada.
171	AB	InpFinish?	flag	Usada por TxInp para indicarle a su bucle interno terminar entrada de caracteres.
172	AC	InpXpix	#	Usada por TxInp para guardar coordenada X absoluta del texto.
173	AD	InpYpix	#	Usada por TxInp para guardar coordenada Y absoluta del texto.
174	AE	TxBGH	#	Usado por TxInp para guardar altura de barra de texto.
175	AF	TxBGW	#	Usado por TxInp para guardar anchura de barra de texto.
176	B0	TxFlag	flag	Usada por TxInp para indicarle al próximo programa si se presiono ENTER o CANCEL.
177	B1	TxBBGx	#	Usada por TxInp para guardar coord relativa x de fondo negro ABUFF.
178	B2	TxBBGy	#	Usada por TxInp para guardar coord relativa y de fondo negro ABUFF.
179	B3	TxBBGh	#	Usada por TxInp para guardar altura de fondo negro.
180	B4	TxBBGw	#	Usada por TxInp para guardar anchura de fondo negro.
181	B5	TxBBGxpix	#	Usada por TxInp para guardar coord relativa x de fondo negro GBUFF.
182	B6	TxBBGypix	#	Usada por TxInp para guardar coord relativa y de fondo negro GBUFF.
183	B7	TxInfl	lista	Usada por TxInp para guardar el texto de ayuda.
184	B8	TxList?	flag	Indica si existe un texto de bienvenida a la Zona. Usado por M1Rx y M1ZxSx.
185	B9	TxHelp?	flag	Indica si existe un texto de ayuda. Usado por TxHelpENG.
186	BA	TxObj?	flag	Indica si existe un texto sobre los objetivos de la Zona. Usado por TxObjENG.
187	BB	TxList	\$	Texto de bienvenida a la Zona. Usado por M1Rx y M1ZxSx.
188	BC	TxHelp	\$	Texto de ayuda. Usado por TxHelpENG.
189	BD	TxObj	\$	Texto sobre los objetivos de la Zona. Usado por TxObjENG.
190	BE	MenuOn?	flag	Indica al GBuffIni si debe activar el menú (GBuffIni es llamado por M0 al iniciar el juego).
191	BF	ShowCiclos?	flag	Indica si se debe mostrar el reloj de BM. Usada por M1ZxSx y M1Rx. Modificada por TongleShowCiclos.
192	C0	ShowEliCoords?	flag	Indica a BEngine, M1ZxSx y M1Rx si se debe mostrar las coordenadas de Eli. Modificada por TongleEliCoords.
193	C1	T&M?	flag	Indica si el sistema T&M está activado. Esto permite que se pueda activar y desactivar con la misma tecla.
194	C2	T&Mx	#	Coordenada X de T&M. Actualizada por InitT&M. Utilizada por SetT&M
195	C3	T&My	#	Coordenada Y de T&M. Actualizada por InitT&M. Utilizada por SetT&M
196	C4	ticks0	#	Conteo de ticks inicial
197	C5	dticks	#	Variación en conteo de ticks
198	C6	T&Mg0	grob	Gráfico de fondo de T&M en GBUFF. Actualizada por InitT&M. Utilizada y actualizada por SetT&M
199	C7	MEMi		Cantidad de memoria libre cuando se arranca BM.
200	C8	DMemList	lista	Contiene datos sobre consumo de memoria.
201	C9	Nvars	#	Cantidad de variables.
202	CA	VKEY	#	Tecla actual durante reproducción de replay. Actualizada por M1Rx. Utilizada por M2
203	CB	VKEYc	#	Usada por sistema Rx como contador de teclas presionadas. Tiene <b>restricción posicional</b> por el módulo RcStop.

204	CC	VKEYList	lista	Usada por sistema Rx para guardar la lista de teclas presionadas. (VKEYList = Virtual Key List) Tiene <b>restricción posicional</b> por el módulo RcStop.
205	CD	VKEYrec?	flag	Usada por sistema Rx para autorizar iniciar proceso de grabación.
206	CE	VKEYplay?	flag	Usada para indicar que se debe tomar las pulsaciones de teclas de la lista VKEY y no del teclado. Esto es, que se esta reproduciendo un replay. Usada por M0 para elegir entre M1ZxSx y M1Rx. Usada también por M1Rx Tiene <b>restricción posicional</b> por el módulo RcStop.
207	CF	MBClash?	flag	Indica si al intentar mover un MB ha ocurrido algún choque que lo impidiera. Actualizada por MBMov.
208	D0	MBClashX	#	Coord X en ABUFF del objeto con que se ha producido un choque al intentar mover un MB. Actualizada por MBMov.
209	D1	MBClashY	#	Coord Y en ABUFF del objeto con que se ha producido un choque al intentar mover un MB. Actualizada por MBMov.
210	D2	MBClash1?	flag	Indica si al intentar mover un MB ha ocurrido algún choque con un CB que lo impidiera. Actualizada por MBMov.
211	D3	MBClash2?	flag	Indica si al intentar mover un MB ha ocurrido algún choque con un MB, PB o AB que lo impidiera. Actualizada por MBMov.
212	D4	MBClash3?	flag	Indica si al intentar mover un MB ha ocurrido algún choque con un OB que lo impidiera. Actualizada por MBMov.
213	D5	PBClash?	flag	Indica si al intentar mover un PB ha ocurrido algún choque que lo impidiera. Actualizada por PB Mov.
214	D6	PBClashX	#	Coordenada X en ABUFF del objeto con que se ha producido un choque al intentar mover un PB. Actualizada por PB Mov.
215	D7	PBClashY	#	Coordenada Y en ABUFF del objeto con que se ha producido un choque al intentar mover un PB. Actualizada por PB Mov.
216	D8	PBClash1?	flag	Indica si al intentar mover un PB ha ocurrido algún choque con un CB que lo impidiera. Actualizada por PB Mov.
217	D9	PBClash2?	flag	Indica si al intentar mover un PB ha ocurrido algún choque con un MB, PB o AB que lo impidiera. Actualizada por PB Mov.
218	DA	PBClash3?	flag	Indica si al intentar mover un PB ha ocurrido algún choque con un OB que lo impidiera. Actualizada por PB Mov.
219	DB	HitEli?	flag	Indica si la IA Gusano ha chocado Eli al intentar moverse. Actualizada por AIGMov
220	DC	AIClash?	flag	Indica si la IA Gusano ha chocado con algún bloque o con Eli al intentar moverse. Actualizada por AIGMov
221	DD	AIClashX	#	Coordenada X en ABUFF del objeto con que se ha producido un choque al intentar mover una IA Gusano. Actualizada por AIGMov
222	DE	AIClashY	#	Coordenada Y en ABUFF del objeto con que se ha producido un choque al intentar mover una IA Gusano. Actualizada por AIGMov.
223	DF	AIClash1?	flag	Indica si la IA Gusano ha chocado con algún CB al intentar moverse. Actualizada por AIGMov
224	E0	AIClash2?	flag	Indica si la IA Gusano ha chocado con algún MB, PB o AB al intentar moverse. Actualizada por AIGMov
225	E1	AIClash3?	flag	Indica si la IA Gusano ha chocado con algún OB al intentar moverse. Actualizada por AIGMov
226	E2	XSearch	#	Utilizado por los módulos de localización de bloques para indicar la coordenada X del punto en que se realizará la búsqueda.
227	E3	YSearch	#	Utilizado por los módulos de localización de bloques para indicar la coordenada Y del punto en que se realizará la búsqueda.
228	E4	MB?	flag	Usado por SearchMBGetIndex para indicar si se ha localizado algun MB.
229	E5	MBindex	#	Usado por SearchMBGetIndex para indicar el índice de un MB localizado.
230	E6	PB?	flag	Usado por SearchPBGetIndex para indicar si se ha localizado algun PB.
231	E7	PBindex	#	Usado por SearchPBGetIndex para indicar el índice de un PB localizado.
232	E8	OB?	flag	Usado por SearchOBGetIndex para indicar si se ha localizado algun OB.
233	E9	OBindex	#	Usado por SearchOBGetIndex para indicar el índice de un OB localizado.
234	EA	AB?	flag	Usado por SearchABGetIndex para indicar si se ha localizado algun AB.
235	EB	ABindex	#	Usado por SearchABGetIndex para indicar el índice de un AB localizado.
236	EC	CE?	flag	Reservado para módulo SearchCEGetIndex. Dicho módulo aun no ha sido creado.
237	ED	CEindex	#	Reservado para módulo SearchCEGetIndex. Dicho módulo aun no ha sido creado.
238	EE	GLPackIncluded?	flag	Indica a StdGClear si debe guardar una copia del GLPack utilizado en el archivo de juego, específicamente en la variable GLPackBZ. Esto ocurre cuando el GLPack no es estándar. Variable actualizada por GLPackSelection.
239	EF	GLPackBZ	bz	Sirve para almacenar GLPack no estándar en formato comprimido. Actualizada por GLPackSelection. Utilizada por StdGClear y StdGLoader.
240	F0	GLPStd?	flag	Indica si se esta utilizando un GLPack estándar.
241	F1	GLPStdX	lista	Cuando se evalúa, llama al GLPStd utilizado.



242	F2	aSLStd?	flag	Indica si se está utilizando un aSLStd estándar.
243	F3	aSLStdX	lista	Cuando se evalúa, llama al aSLStd utilizado.
244	F4	aniSecList	lista	Lista de varias secuencias gráficas como los telepuertos, desintegración de Eli y onda de ataque. Es una especie de almacén de secuencias.
245	F5	IndGrob	grob	Contiene un gráfico, del tamaño de un bloque común, creado para los indicadores. Variable actualizada por GLPackLoader utilizando el gráfico 9 de CBGrobList.
246	F6	BuilderMode?	flag	Indica si se está ejecutando BM en modo Builder o Engine. Usada por BMBBackup, M1ZxSx y M2 Tiene <b>restricción posicional</b> por el módulo BMBBackup.
247	F7	BMDataType	#	Indica el tipo de archivo primario BM: #A para Zx, #B para Sx o #C para Rx
248	F8	IniLeftCol	#	Coordenada X de ventana al iniciar el juego. Usada por GBuffIni.
249	F9	IniTopRow	#	Coordenada Y de ventana al iniciar el juego. Usada por GBuffIni.
250	FA	M1Off?	flag	Para autorizar la salida de M1ZxSx o M1Rx se coloca TRUE en esta variable. Esto se hace cuando se quiere terminar un juego, una reproducción de replay o salir del Builder a la pila sin pasar por el Engine.
251	FB	Accion?	flag	Indica el modo de operación de M1ZxSx. TRUE para modo fluido y FALSE para modo no fluido. En este último solo se ejecuta un ciclo si se presiona alguna tecla.
252	FC	GC?	flag	Usado para autorizar a M1ZxSx para que realice una recogida de basura (Gabbage collection).
253	FD	KEY	#	Número de la tecla actual presionada.
254	FE	RepK?	flag	Usada para indicar que se ha presionado determinada tecla de forma repetida o que se ha dejado presionada. Utilizada por M1ZxSx, M2, aEli y BEngine.
255	FF	KeyboardStd?	flag	Indica si la configuración para el teclado es estándar. Usada por ConfKb.
256	100	KUp	#	Asignación de tecla. Contiene el número de la tecla de movimiento hacia arriba.
257	101	KDown	#	Asignación de tecla. Contiene el número de la tecla de movimiento hacia abajo.
258	102	KLeft	#	Asignación de tecla. Contiene el número de la tecla de movimiento hacia la derecha.
259	103	KRight	#	Asignación de tecla. Contiene el número de la tecla de movimiento hacia izquierda.
260	104	KHelp	#	Asignación de tecla. Contiene el número de la tecla de mensaje de ayuda.
261	105	KTakeOb	#	Asignación de tecla. Contiene el número de la tecla para recoger objetos.
262	106	KMap1	#	Asignación de tecla. Contiene el número de la tecla para ver mapa 1 (CBs).
263	107	KMap2	#	Asignación de tecla. Contiene el número de la tecla para ver mapa 2 (MBs, PBs, ABs).
264	108	KMap3	#	Asignación de tecla. Contiene el número de la tecla para ver mapa 3 (OBs).
265	109	KGun1	#	Asignación de tecla. Contiene el número de la tecla para usar arma 1 (Bomba).
266	10A	KGun2	#	Asignación de tecla. Contiene el número de la tecla para usar arma 2 (Pinch).
267	10B	KGun3	#	Asignación de tecla. Contiene el número de la tecla para usar arma 3 (Escáner).
268	10C	KSpeedUp	#	Asignación de tecla. Contiene el número de la tecla para aumentar velocidad de juego.
269	10D	KSpeedDw	#	Asignación de tecla. Contiene el número de la tecla para reducir velocidad de juego.
270	10E	KExit	#	Asignación de tecla. Contiene el número de la tecla para salir del juego.
271	10F	KInventory	#	Asignación de tecla. Contiene el número de la tecla para ver el inventario.
272	110	ZxName	\$	Nombre de la Zona o ZxPack..
273	111	ZxMaster	\$	Nombre o nick del creador de la Zona o ZxPack..
274	112	eH	#	Altura de las diapositivas de la animación de Eli. Altura de los bloques.
275	113	eW	#	Anchura de las diapositivas de la animación de Eli. Anchura de los bloques.
276	114	dHpix	#	Cantidad de bloques de distancia vertical para desplazar ventana GBUFF.
277	115	dWpix	#	Cantidad de bloques de distancia horizontal para desplazar ventana GBUFF.
278	116	XeliB	#	Variable especial = Wgbuffinpix – Xjump. Creada para evitar la repetición de este cálculo en cada ciclo de M1ZxSx.
279	117	YeliB	#	Variable especial = Hgbuffinpix Yjump. Creada para evitar la repetición de este cálculo en cada ciclo de M1ZxSx.
280	118	MasterPassword	\$	Pasword del Master (Encriptado).
281	119	MasterPassword?	flag	Indica si esta activado el Pasword del Master.
				(MBi = #11A)

La **restricción posicional** se refiere a que la variable no puede ser cambiada de posición en la tabla ya que algún módulo la localiza en base a su posición y no a su nombre. Esto es importante cuando se quiere hacer cambios en la tabla.

## Anexo 4. Variables Locales

MODULO QUE LAS CREA	NOMBRES DE VARIABLES
<b>AnmBuilder</b>	AnmMenuOpt
<b>Anm</b>	ANMOPT ANMx1 ANMy1 ANMGLPack ANMw ANMh ANMn NombreAnm
<b>AssigbToGroup</b>	GroupTypeList
<b>AxPackBuilder</b>	AxName AxNroMundos AxZxNamesList AxFlagsList AxZxList AxSxList AxRxList AxOK?
<b>AxPackIni</b>	BMAxID BMAxZxToPlay
<b>BEngine</b>	BEngineOff? <b>CBBuilder?</b> <b>CB+?</b>
<b>BVCreator</b>	Nvars mem
<b>CE05</b>	Actual2 Actual1
<b>CEBuilder</b>	CEMenuOpt
<b>ConfInv</b>	SubMenuOpt
<b>ConfKb</b>	SubMenuOpt
<b>ConfM1</b>	SubMenuOpt
<b>ConfSB</b>	CONFGtipo CONFGindice CONFGaccion CONFGBIGLIST
<b>EdZxPack</b>	BMFileLoaded
<b>ExitBLDR</b>	BEngineOff? ExitBLDROpt
<b>FxMenu</b>	SubMenuOpt
<b>GLPackIni</b>	Tipo NumerodeFilas NumerodeColumnas EspacioEtiquetasX EspacioEtiquetasY
<b>GLPackInpName</b>	Nombred Paquete
<b>GLPackLoader</b>	GLPackX

<b>GLPack</b>	GRBOPT Dimensiones BMGLPack
<b>GrbBuilder</b>	FBstatus
<b>Grb</b>	GRBOPT GRBx1 GRBy1 GRBGLPack GRBw GRBh NombreGrb
<b>GroupsBuilder</b>	GroupsBuilderMenuOpt GList SBindice Gindice GrupodeTrabajo GrupodeTrabajoLEN SBindicePOS Gindice2 GrupodeTrabajo2 BMBuilderc1 BMBuilderf1
<b>MapMenu</b>	MapG MapP
<b>MBBuilder</b>	MBMenuOpt
<b>OBBuilder</b>	OBMenuOpt OBSubMenuOpt
<b>PBBuilder</b>	PBMenuOpt PSubMenuOpt
<b>PlayAxPack</b>	MEMi
<b>PlayZxPack</b>	MEMi
<b>RcStart</b>	recX
<b>ShowResumen</b>	ShowResumenList
<b>TongleGroups</b>	TongleGroupsMode
<b>YW02D</b>	CEA0
<b>YWBuilder</b>	YWMenuOpt
<b>ZxLoader</b>	BMFileLoaded
<b>ZxPackBuilder</b>	CoordsList

## Anexo 5. Estructura de los Componentes

### Estructura de Bloques y Eventos

Los MB, PB y OB constan, cada uno, de 16 partes.

Los AB, constan, cada uno, de 5 partes.

Los CE, constan cada uno de 32 partes.

MB	PB	OB	AB	CE
MBtype	PBtype	OBtype	ABtype	CEtype
MBstatus	PBstatus	OBstatus	ABstatus	CEstatus
MBxpix	PBxpix	OBxpix	ABxpix	CEtest
MBypix	PBypix	OBypix	ABypix	CEprogT
MBtest	PBtest	OBtest	ABg0	CEprogF
MBprogT	PBprogT	OBprogT		CEx5pix
MBprogF	PBprogF	OBprogF		CEy5pix
MBL1	PBL1	OBL1		CEx4pix
MBL0	PBL0	OBL0		CEy4pix
MBc1	PBc1	OBc1		CEx3pix
MBc0	PBc0	OBc0		CEy3pix
MBf1	PBf1	OBf1		CEx2pix
MBf0	PBf0	OBf0		CEy2pix
MBg1	PBg1	OBg1		CEx1pix
MBg0	PBg0	OBg0		CEy1pix
MBk	PBk	OBk		CEx0pix
				CEy0pix
				CEL2
				CEL1
				CEL0
				CEc2
				CEc1
				CEc0
				CEf2
				CEf1
				CEf0
				CEg2
				CEg1
				CEg0
				CEk2
				CEk1
				CEk0

Estas variables tienen diferentes usos dependiendo del tipo de componente.



## I. Estructura de los MB

Valores por defecto:

```
(-----)
BINT100  >MBk
Box0     >MBg0
#0       >MBg1
#0       >MBf0
#0       >MBf1
#0       >MBc0
#0       >MBc1
#0       >MBL0
#0       >MBL1
{ }      >MBprogF
{ }      >MBprogT
{TRUE}   >MBtest
Ypix     >MBypix
Xpix     >MBxpix
TRUE     >MBstatus
#0       >MBtype
(-----)
```

Leyenda:

Negro: Variable que se usa  
Verde: Variable no utilizada. Se marcan con (\*)  
Negrita: Variable especial de este tipo de MB

### Bloque Móvil Tipos #1 y #2: PATRULLERO

```
(-----)
MBk          (Energía Inicial)          )
MBg0         (Gráfico de fondo)         )
MBg1         (Alcance de su arma)       )
MBf0         (Velocidad. Cada cuantos ciclos?) )
MBf1         (¿Está armado?)            )
MBc0         (Contador de ciclos)       )
MBc1        (Contador de Ruta)         )
MBL0        (*)                          )
MBL1        (Ruta)                     )
MBprogF      (Código Falso)             )
MBprogT      (Código Verdadero)         )
MBtest       (Código de prueba)         )
MBypix       (Coordenada actual Y)      )
MBxpix       (Coordenada actual X)      )
MBstatus     (Estado: activado/desactivado) )
MBtype       (Tipo. Indica su gráfico)  )
(-----)
```

### Bloque Móvil Tipos #3 y #4

#### PERSEGUIDOR

-----)		
MBk	(Energía Inicial	)
MBg0	(Gráfico de fondo	)
MBg1	(Alcance de su arma	)
MBf0	(Velocidad	)
MBf1	(¿Está armado?	)
MBc0	(Contador de ciclos	)
<b>MBc1</b>	<b>(*)</b>	
<b>MBL0</b>	<b>(Potencia de Radar. Para perseguir</b>	<b>)</b>
<b>MBL1</b>	<b>(*)</b>	
MBprogF	(Código Falso	)
MBprogT	(Código Verdadero	)
MBtest	(Código de prueba	)
MBypix	(Coordenada actual Y	)
MBxpix	(Coordenada actual X	)
MBstatus	(Estado: activado/desactivado	)
MBtype	(Tipo. Indica su gráfico	)
-----)		

### Bloque Móvil Tipos #5 y #6

#### DOBLE MODO

#### (Patrullero y Perseguidor)

-----)		
MBk	(Energía Inicial	)
MBg0	(Gráfico de fondo	)
MBg1	(Alcance de su arma	)
MBf0	(Velocidad	)
MBf1	(¿Está armado?	)
MBc0	(Contador de ciclos	)
<b>MBc1</b>	<b>(Contador de Ruta</b>	<b>)</b>
<b>MBL0</b>	<b>(Potencia de Radar. Para perseguir</b>	<b>)</b>
<b>MBL1</b>	<b>(Ruta</b>	<b>)</b>
MBprogF	(Código Falso	)
MBprogT	(Código Verdadero	)
MBtest	(Código de prueba	)
MBypix	(Coordenada actual Y	)
MBxpix	(Coordenada actual X	)
MBstatus	(Estado: activado/desactivado	)
MBtype	(Tipo. Indica su gráfico	)
-----)		

### Bloque Móvil Tipos #7

#### ESTATICO

-----)		
MBk	(Energía Inicial	)
MBg0	(Gráfico de fondo	)
MBg1	(Alcance de su arma	)
MBf0	(Velocidad	)
MBf1	(¿Está armado?	)
MBc0	(Contador de ciclos	)
<b>MBc1</b>	<b>(*)</b>	
<b>MBL0</b>	<b>(*)</b>	

<b>MBL1</b>	<b>(*)</b>	
MBprogF	(Código Falso	)
MBprogT	(Código Verdadero	)
MBtest	(Código de prueba	)
MBypix	(Coordenada actual Y	)
MBxpix	(Coordenada actual X	)
MBstatus	(Estado: activado/desactivado	)
MBtype	(Tipo. Indica su gráfico	)
(-----)		

## II. Estructura de un archivo ZxPack

{	
\$BMZxPack	(Tipo de paquete)
\$BMver	(Versión de BM)
\$NombredePack	(Nombre de la Zona)
#TamañoZx	(Tamaño al descomprimir en bytes)
{ Zx.BZ }	(archivo de juego comprimidos)
{ Sx.BZ }	(Salvados de juego comprimidos)
{ Rx.BZ }	(Replay de juego comprimidos)
}	

## III. Estructura de un archivo AxPack

{	
\$BMAxPack	(Tipo de paquete)
\$BMver	(Versión de BM)
\$NombredePack	(Nombre de la Aventura)
#NumerodeMundos	
{ NombreZx1 NombreZx2 ... NombreZxN }	(Nombres de mundos)
{ flag1 flag2 ... flagN }	(Permisos de acceso a mundos.)
{ ZxMundo1.BZ ZxMundo2.BZ ... ZxMundoN.BZ }	(Lista de archivos de juego comprimidos)
{ SxMundo1.BZ SxMundo2.BZ ... SxMundoN.BZ }	(Lista de salvados comprimidos)
{ RxMundo1.BZ RxMundo2.BZ ... RxMundoN.BZ }	(Lista de replays comprimidos)
}	

## IV. Estructura de un archivo GLPack:

{	
"BMGLPack" "NombredePaquete" #Dimensiones #NumerodeColumnas #NumerodeFilas	
{CBGrobList}	
{MBGrobList}	
{PBGrobList}	
{OBGrobList}	
{ABGrobList}	
{eNList sin dividir}	(Es dividida por el módulo ListSplitter en GLPackLoader)
}	

Cada {SBGrobList} esta compuesta por 15 gráficos. Uno para cada tipo de SB.

En {CBGrobList} el grob 10 se usa para BombGrob y el grob 11 se usa para AfterBombGrob.

## V. Estructura de un archivo aniSecList (aSLStdX):

```
{
    {Secuencia de integración/desintegración de Eli en los telepuertos}
    {Secuencia de ataque}
    {Secuencia de desaparición de Eli al ganar juego}
}
```

Las nuevas secuencia deben ser agregadas al final de la lista.

## VI. Estructura de un archivo de gráfico complementario:

{gn xn yn fn}

donde:

gn: es el gráfico  
xn: coordenada x en abuff  
yn: coordenada y en abuff  
fn: flag para activar y desactivar carga del gráfico

## VII. Estructura de las listas: bgGrobList y fgGrobList

Poseen la misma estructura:

{ {g1 x1 y1 f1} {g2 x2 y2 f2} {g3 x3 y3 f3} ... {gn xn yn fn} }

## VIII. Estructura de una animación M3:

El módulo M3 requiere en la pila:

{GROB1 GROB2 GROB3 ... GROBN}	(lista de diapositivas o secuencia gráfica)
#m	(modo de anim: #1 = Gor, #2= Gxor, #3= GROB!)
Flag	(dice si se limpia GBUFF al terminar secuencia)
#xpix	(Coordenada x en ABUFF)
#ypix	(Coordenada y en ABUFF)
%t	(tiempo entre diapositivas)

## Anexo 6. Definiciones de Variables

```
(GETLAM- PARTE A -----)

(Version de Blocks Master -----)
DEFINE BMver          1GETLAM

(Aceleracion -----)
DEFINE dh              2GETLAM
DEFINE dw              3GETLAM
DEFINE eh+dh           4GETLAM
DEFINE ew+dw           5GETLAM

(Actuales -----)
DEFINE MBActual        6GETLAM
DEFINE PBActual        7GETLAM
DEFINE OBAActual       8GETLAM
DEFINE ABActual        9GETLAM
DEFINE CEActual       10GETLAM

(SBi -----)
DEFINE MBi             11GETLAM
DEFINE PBi             12GETLAM
DEFINE OBi             13GETLAM
DEFINE ABi             14GETLAM
DEFINE CEi             15GETLAM

(Eli parte 1 -----)
DEFINE Box0            16GETLAM
DEFINE Xeli            17GETLAM
DEFINE Yeli            18GETLAM
DEFINE Xpix            19GETLAM
DEFINE Ypix            20GETLAM
DEFINE XJump           21GETLAM
DEFINE YJump           22GETLAM

DEFINE eLEN            #17 GETLAM
DEFINE CeLEN           #18 GETLAM
DEFINE Seceli          #19 GETLAM
DEFINE AnimateEli?    #1A GETLAM
DEFINE EliGrob         #1B GETLAM
DEFINE eNList          #1C GETLAM
DEFINE eNSecini        #1D GETLAM
DEFINE EliBoost        #1E GETLAM
DEFINE EliPowDown?    #1F GETLAM
DEFINE FMDelayOn?     #20 GETLAM
DEFINE FMDelay?       #21 GETLAM
DEFINE DelayFM         #22 GETLAM
DEFINE DelayAniEli     #23 GETLAM

(Mapa -----)
DEFINE Map?            #24 GETLAM
DEFINE Map             #25 GETLAM
DEFINE MapH            #26 GETLAM
DEFINE MapW            #27 GETLAM
DEFINE 2MapH           #28 GETLAM
DEFINE Hgbuffinpix     #29 GETLAM
DEFINE wgbuffinpix     #2A GETLAM
DEFINE InterBox        #2B GETLAM

(MB -----)
DEFINE MBf             #2C GETLAM
DEFINE MBActualA       #2D GETLAM
DEFINE MBNum           #2E GETLAM
DEFINE MBGrobList      #2F GETLAM
DEFINE ListMBtoRun     #30 GETLAM
DEFINE SizeListMBtoRun #31 GETLAM
DEFINE RunListMBtoRun? #32 GETLAM
DEFINE ListListMBtoRun #33 GETLAM
DEFINE MBStep          #34 GETLAM
```

```
(PB -----)
DEFINE PBf #35 GETLAM
DEFINE PBActua1A #36 GETLAM
DEFINE PBNum #37 GETLAM
DEFINE PBGroblList #38 GETLAM
DEFINE ListListPBtoRun #39 GETLAM
```

```
(OB -----)
DEFINE OBf #3A GETLAM
DEFINE OBActua1A #3B GETLAM
DEFINE OBNum #3C GETLAM
DEFINE OBGroblList #3D GETLAM
DEFINE ListListOBtoRun #3E GETLAM
```

```
(AB -----)
DEFINE ABf #3F GETLAM
DEFINE ABNum #40 GETLAM
DEFINE ABActua1A #41 GETLAM
DEFINE ABGroblList #42 GETLAM
DEFINE ListListABtoRun #43 GETLAM
```

```
(CE -----)
DEFINE Cef #44 GETLAM
DEFINE CEnum #45 GETLAM
DEFINE ListCEtoRun #46 GETLAM
DEFINE SizeListCEtoRun #47 GETLAM
DEFINE ListListCEtoRun #48 GETLAM
DEFINE RunListCEtoRun? #49 GETLAM
```

```
(UE -----)
DEFINE ListUEtoRun #4A GETLAM
DEFINE RunListUEtoRun? #4B GETLAM
DEFINE SizeListUEtoRun #4C GETLAM
```

```
(CB -----)
DEFINE CBGroblList #4D GETLAM
DEFINE BoxGrobindex #4E GETLAM
DEFINE BoxGrobl #4F GETLAM
DEFINE BoxGroblH #50 GETLAM
DEFINE BoxGroblW #51 GETLAM
```

```
(Reloj -----)
DEFINE Nciclos #52 GETLAM
DEFINE cNciclos #53 GETLAM
DEFINE Nciclosinc #54 GETLAM
```

```
(bg -----)
DEFINE bgGroblList #55 GETLAM
DEFINE bgGroblList? #56 GETLAM
```

```
(fg -----)
DEFINE fgGroblList #57 GETLAM
DEFINE fgGroblList? #58 GETLAM
```

```
(ventana Inventario -----)
DEFINE InvBGroblStd? #59 GETLAM
DEFINE InvBGrobl #5A GETLAM
DEFINE InvBGh #5B GETLAM
DEFINE InvBGw #5C GETLAM
DEFINE Invdx #5D GETLAM
DEFINE Invdy #5E GETLAM
DEFINE Invx #5F GETLAM
DEFINE Invy #60 GETLAM
DEFINE InvGrobl0 #61 GETLAM
```

```
(Telepuerto -----)
DEFINE TeleT? #62 GETLAM
DEFINE TeleportFree? #63 GETLAM
DEFINE TelepDestino #64 GETLAM
```

```
(Metas -----)
DEFINE Goaln #65 GETLAM
```

```

DEFINE Goalc                #66 GETLAM
DEFINE Goal?                #67 GETLAM
DEFINE YouWin               #68 GETLAM
DEFINE YouWinStd?           #69 GETLAM

(Animaciones -----)
DEFINE aniH                 #6A GETLAM
DEFINE aniW                 #6B GETLAM
DEFINE aniX                 #6C GETLAM
DEFINE aniY                 #6D GETLAM
DEFINE anig0                #6E GETLAM
DEFINE aniDelay             #6F GETLAM
DEFINE aniLEN               #70 GETLAM
DEFINE aniFlag              #71 GETLAM
DEFINE aniMode              #72 GETLAM

(Bombas -----)
DEFINE Xbomb                #73 GETLAM
DEFINE Ybomb                #74 GETLAM
DEFINE Bomb?                #75 GETLAM
DEFINE Tbomb                #76 GETLAM
DEFINE CTbomb               #77 GETLAM
DEFINE BombType             #78 GETLAM
DEFINE Xbombpix             #79 GETLAM
DEFINE Ybombpix             #7A GETLAM
DEFINE BombGrob             #7B GETLAM
DEFINE AfterBombGrob        #7C GETLAM

(Shtx -----)
DEFINE txList               #7D GETLAM
DEFINE txN                  #7E GETLAM
DEFINE txW                  #7F GETLAM
DEFINE txH                  #80 GETLAM
DEFINE txG0                 #81 GETLAM
DEFINE txX1                 #82 GETLAM
DEFINE txY1                 #83 GETLAM
DEFINE ShtxDelay            #84 GETLAM
DEFINE ShtxDelay0           #85 GETLAM
DEFINE ShtxKEY              #86 GETLAM

(Armas -----)
DEFINE EliPower             #87 GETLAM
DEFINE EliPowerIni          #88 GETLAM
DEFINE NBombs               #89 GETLAM
DEFINE NPinch               #8A GETLAM
DEFINE PinchOn?             #8B GETLAM
DEFINE worldKeys            #8C GETLAM
DEFINE worldMap?            #8D GETLAM
DEFINE Shield?              #8E GETLAM
DEFINE ScannerOn?           #8F GETLAM
DEFINE S+?                  #90 GETLAM

(Fx -----)
DEFINE F1                   #91 GETLAM
DEFINE F2                   #92 GETLAM
DEFINE F3                   #93 GETLAM
DEFINE F4                   #94 GETLAM
DEFINE F5                   #95 GETLAM
DEFINE F6                   #96 GETLAM
DEFINE F7                   #97 GETLAM
DEFINE F8                   #98 GETLAM
DEFINE F9                   #99 GETLAM
DEFINE FA                   #9A GETLAM
DEFINE FB                   #9B GETLAM
DEFINE FC                   #9C GETLAM
DEFINE FD                   #9D GETLAM
DEFINE FE                   #9E GETLAM
DEFINE FF                   #9F GETLAM
DEFINE ListListFxtoRun      #A0 GETLAM

(TxInp -----)
DEFINE TxMode               #A1 GETLAM

```

```

DEFINE TxBG          #A2 GETLAM
DEFINE TxNChars      #A3 GETLAM
DEFINE InpX          #A4 GETLAM
DEFINE InpY          #A5 GETLAM
DEFINE InpK          #A6 GETLAM
DEFINE InpText       #A7 GETLAM
DEFINE Inpc          #A8 GETLAM
DEFINE Inpg0         #A9 GETLAM
DEFINE Inpp1         #AA GETLAM
DEFINE InpFinish?    #AB GETLAM
DEFINE InpXpix       #AC GETLAM
DEFINE InpYpix       #AD GETLAM
DEFINE TxBGH         #AE GETLAM
DEFINE TxBGW         #AF GETLAM
DEFINE TxFlag        #B0 GETLAM
DEFINE TxBBGx        #B1 GETLAM
DEFINE TxBBGy        #B2 GETLAM
DEFINE TxBBGh        #B3 GETLAM
DEFINE TxBBGw        #B4 GETLAM
DEFINE TxBBGxpix     #B5 GETLAM
DEFINE TxBBGypix     #B6 GETLAM
DEFINE TxInFL        #B7 GETLAM

```

(Mensajes de texto -----)

```

DEFINE TxList?       #B8 GETLAM
DEFINE TxHelp?       #B9 GETLAM
DEFINE TxObj?        #BA GETLAM
DEFINE TxList        #BB GETLAM
DEFINE TxHelp        #BC GETLAM
DEFINE TxObj         #BD GETLAM

```

(Pantalla -----)

```

DEFINE MenuOn?       #BE GETLAM
DEFINE ShowCiclos?   #BF GETLAM
DEFINE ShowEliCoords? #C0 GETLAM

```

(T&M -----)

```

DEFINE T&M?          #C1 GETLAM
DEFINE T&Mx          #C2 GETLAM
DEFINE T&My          #C3 GETLAM
DEFINE ticks0        #C4 GETLAM
DEFINE dticks        #C5 GETLAM
DEFINE T&Mg0         #C6 GETLAM

```

(Memoria -----)

```

DEFINE MEMi          #C7 GETLAM
DEFINE DMemList      #C8 GETLAM
DEFINE Nvars         #C9 GETLAM

```

(Replays -----)

```

DEFINE VKEY          #CA GETLAM
DEFINE VKEYc         #CB GETLAM
DEFINE VKEYList      #CC GETLAM
DEFINE VKEYrec?      #CD GETLAM
DEFINE VKEYplay?     #CE GETLAM

```

(Choques -----)

```

DEFINE MBClash?      #CF GETLAM
DEFINE MBClashX      #D0 GETLAM
DEFINE MBClashY      #D1 GETLAM
DEFINE MBClash1?     #D2 GETLAM
DEFINE MBClash2?     #D3 GETLAM
DEFINE MBClash3?     #D4 GETLAM
DEFINE PBClash?      #D5 GETLAM
DEFINE PBClashX      #D6 GETLAM
DEFINE PBClashY      #D7 GETLAM
DEFINE PBClash1?     #D8 GETLAM
DEFINE PBClash2?     #D9 GETLAM
DEFINE PBClash3?     #DA GETLAM
DEFINE HitEli?       #DB GETLAM
DEFINE AIClash?      #DC GETLAM
DEFINE AIClashX      #DD GETLAM

```



```

DEFINE AIClashY          #DE GETLAM
DEFINE AIClash1?         #DF GETLAM
DEFINE AIClash2?         #E0 GETLAM
DEFINE AIClash3?         #E1 GETLAM

(Sistema Search -----)
DEFINE XSearch           #E2 GETLAM
DEFINE YSearch           #E3 GETLAM
DEFINE MB?               #E4 GETLAM
DEFINE MBindex           #E5 GETLAM
DEFINE PB?               #E6 GETLAM
DEFINE PBindex           #E7 GETLAM
DEFINE OB?               #E8 GETLAM
DEFINE OBindex           #E9 GETLAM
DEFINE AB?               #EA GETLAM
DEFINE ABindex           #EB GETLAM
DEFINE CE?               #EC GETLAM
DEFINE CEindex           #ED GETLAM

(GLPack -----)
DEFINE GLPackIncluded?   #EE GETLAM
DEFINE GLPackBZ           #EF GETLAM
DEFINE GLPStd?           #F0 GETLAM
DEFINE GLPStdX           #F1 GETLAM
DEFINE aSLStd?           #F2 GETLAM
DEFINE aSLStdX           #F3 GETLAM
DEFINE aniSecList        #F4 GETLAM
DEFINE IndGrob           #F5 GETLAM

(Builder -----)
DEFINE BuilderMode?      #F6 GETLAM
DEFINE BMDDataType       #F7 GETLAM

(Coordenadas inic Ventana -----)
DEFINE IniLeftCol        #F8 GETLAM
DEFINE IniTopRow         #F9 GETLAM

(Modulo M1 -----)
DEFINE M1off?            #FA GETLAM
DEFINE Accion?           #FB GETLAM
DEFINE GC?               #FC GETLAM

(Teclado -----)
DEFINE KEY                #FD GETLAM
DEFINE Repk?             #FE GETLAM
DEFINE KeyboardStd?      #FF GETLAM
DEFINE KUp                #100 GETLAM
DEFINE KDown              #101 GETLAM
DEFINE KLeft              #102 GETLAM
DEFINE KRight             #103 GETLAM
DEFINE KHelp              #104 GETLAM
DEFINE KTakeob            #105 GETLAM
DEFINE KMap1              #106 GETLAM
DEFINE KMap2              #107 GETLAM
DEFINE KMap3              #108 GETLAM
DEFINE KGun1              #109 GETLAM
DEFINE KGun2              #10A GETLAM
DEFINE KGun3              #10B GETLAM
DEFINE KSpeedUp           #10C GETLAM
DEFINE KSpeedDw           #10D GETLAM
DEFINE KExit              #10E GETLAM
DEFINE KInventory         #10F GETLAM

(ZxName y ZxMaster -----)
DEFINE ZxName             #110 GETLAM
DEFINE ZxMaster           #111 GETLAM

(Eli parte 2 -----)
DEFINE eH                 #112 GETLAM
DEFINE eW                 #113 GETLAM

(Ventana -----)

```

```

DEFINE dHpix          #114 GETLAM
DEFINE dWpix          #115 GETLAM
DEFINE xeliB          #116 GETLAM
DEFINE yeliB          #117 GETLAM

```

(Password -----)

```

DEFINE MasterPassword #118 GETLAM
DEFINE MasterPassword? #119 GETLAM

```

(GETLAM- PARTE B -----)

( MBDATA -----)

```

DEFINE MBtype          :: 6GETLAM      GETLAM ;
DEFINE MBstatus        :: 6GETLAM #1+  GETLAM ;
DEFINE MBxpix          :: 6GETLAM #2+  GETLAM ;
DEFINE MBypix          :: 6GETLAM #3+  GETLAM ;
DEFINE MBtest          :: 6GETLAM #4+  GETLAM ;
DEFINE MBprogT         :: 6GETLAM #5+  GETLAM ;
DEFINE MBprogF         :: 6GETLAM #6+  GETLAM ;
DEFINE MBL1            :: 6GETLAM #7 #+ GETLAM ;
DEFINE MBL0            :: 6GETLAM #8 #+ GETLAM ;
DEFINE MBc1            :: 6GETLAM #9 #+ GETLAM ;
DEFINE MBc0            :: 6GETLAM #A #+ GETLAM ;
DEFINE MBf1            :: 6GETLAM #B #+ GETLAM ;
DEFINE MBf0            :: 6GETLAM #C #+ GETLAM ;
DEFINE MBg1            :: 6GETLAM #D #+ GETLAM ;
DEFINE MBg0            :: 6GETLAM #E #+ GETLAM ;
DEFINE MBk             :: 6GETLAM #F #+ GETLAM ;

```

( PBDATA -----)

```

DEFINE PBtype          :: 7GETLAM      GETLAM ;
DEFINE PBstatus        :: 7GETLAM #1+  GETLAM ;
DEFINE PBxpix          :: 7GETLAM #2+  GETLAM ;
DEFINE PBypix          :: 7GETLAM #3+  GETLAM ;
DEFINE PBtest          :: 7GETLAM #4+  GETLAM ;
DEFINE PBprogT         :: 7GETLAM #5+  GETLAM ;
DEFINE PBprogF         :: 7GETLAM #6+  GETLAM ;
DEFINE PBL1            :: 7GETLAM #7 #+ GETLAM ;
DEFINE PBL0            :: 7GETLAM #8 #+ GETLAM ;
DEFINE PBc1            :: 7GETLAM #9 #+ GETLAM ;
DEFINE PBc0            :: 7GETLAM #A #+ GETLAM ;
DEFINE PBf1            :: 7GETLAM #B #+ GETLAM ;
DEFINE PBf0            :: 7GETLAM #C #+ GETLAM ;
DEFINE PBg1            :: 7GETLAM #D #+ GETLAM ;
DEFINE PBg0            :: 7GETLAM #E #+ GETLAM ;
DEFINE PBk             :: 7GETLAM #F #+ GETLAM ;

```

( OBDATA -----)

```

DEFINE OBtype          :: 8GETLAM      GETLAM ;
DEFINE OBstatus        :: 8GETLAM #1+  GETLAM ;
DEFINE OBxpix          :: 8GETLAM #2+  GETLAM ;
DEFINE OBypix          :: 8GETLAM #3+  GETLAM ;
DEFINE OBtest          :: 8GETLAM #4+  GETLAM ;
DEFINE OBprogT         :: 8GETLAM #5+  GETLAM ;
DEFINE OBprogF         :: 8GETLAM #6+  GETLAM ;
DEFINE OBL1            :: 8GETLAM #7 #+ GETLAM ;
DEFINE OBL0            :: 8GETLAM #8 #+ GETLAM ;
DEFINE OBC1            :: 8GETLAM #9 #+ GETLAM ;
DEFINE OBC0            :: 8GETLAM #A #+ GETLAM ;
DEFINE OBF1            :: 8GETLAM #B #+ GETLAM ;
DEFINE OBF0            :: 8GETLAM #C #+ GETLAM ;
DEFINE OBG1            :: 8GETLAM #D #+ GETLAM ;
DEFINE OBG0            :: 8GETLAM #E #+ GETLAM ;
DEFINE OBk             :: 8GETLAM #F #+ GETLAM ;

```

( ABDATA -----)

```

DEFINE ABtype          :: 9GETLAM      GETLAM ;
DEFINE ABstatus        :: 9GETLAM #1+  GETLAM ;
DEFINE ABxpix          :: 9GETLAM #2+  GETLAM ;
DEFINE ABypix          :: 9GETLAM #3+  GETLAM ;
DEFINE ABg0            :: 9GETLAM #4+  GETLAM ;

```

```

( CEDATA -----)
DEFINE CEtype          :: 10GETLAM          GETLAM ;
DEFINE CEstatus        :: 10GETLAM #1+      GETLAM ;
DEFINE CEtest          :: 10GETLAM #2+      GETLAM ;
DEFINE CEprogt         :: 10GETLAM #3+      GETLAM ;
DEFINE CEprogF         :: 10GETLAM #4+      GETLAM ;
DEFINE CEx5pix         :: 10GETLAM #5+      GETLAM ;
DEFINE CEy5pix         :: 10GETLAM #6+      GETLAM ;
DEFINE CEx4pix         :: 10GETLAM #7+      GETLAM ;
DEFINE CEy4pix         :: 10GETLAM #8+      GETLAM ;
DEFINE CEx3pix         :: 10GETLAM #9+      GETLAM ;
DEFINE CEy3pix         :: 10GETLAM BINT10 #+ GETLAM ;
DEFINE CEx2pix         :: 10GETLAM BINT11 #+ GETLAM ;
DEFINE CEy2pix         :: 10GETLAM BINT12 #+ GETLAM ;
DEFINE CEx1pix         :: 10GETLAM BINT13 #+ GETLAM ;
DEFINE CEy1pix         :: 10GETLAM BINT14 #+ GETLAM ;
DEFINE CEx0pix         :: 10GETLAM BINT15 #+ GETLAM ;
DEFINE CEy0pix         :: 10GETLAM BINT16 #+ GETLAM ;
DEFINE CEL2           :: 10GETLAM BINT17 #+ GETLAM ;
DEFINE CEL1           :: 10GETLAM BINT18 #+ GETLAM ;
DEFINE CEL0           :: 10GETLAM BINT19 #+ GETLAM ;
DEFINE Cec2           :: 10GETLAM BINT20 #+ GETLAM ;
DEFINE Cec1           :: 10GETLAM BINT21 #+ GETLAM ;
DEFINE Cec0           :: 10GETLAM BINT22 #+ GETLAM ;
DEFINE Cef2           :: 10GETLAM BINT23 #+ GETLAM ;
DEFINE Cef1           :: 10GETLAM BINT24 #+ GETLAM ;
DEFINE Cef0           :: 10GETLAM BINT25 #+ GETLAM ;
DEFINE CEg2           :: 10GETLAM BINT26 #+ GETLAM ;
DEFINE CEg1           :: 10GETLAM BINT27 #+ GETLAM ;
DEFINE CEg0           :: 10GETLAM BINT28 #+ GETLAM ;
DEFINE CEK2           :: 10GETLAM BINT29 #+ GETLAM ;
DEFINE CEK1           :: 10GETLAM BINT30 #+ GETLAM ;
DEFINE CEK0           :: 10GETLAM BINT31 #+ GETLAM ;

```

(PUTLAM- PARTE A -----)

(Version de Blocks Master -----)

```

DEFINE >BMver          1PUTLAM

```

(Aceleracion -----)

```

DEFINE >dH              2PUTLAM
DEFINE >dw              3PUTLAM
DEFINE >eH+dH          4PUTLAM
DEFINE >ew+dw          5PUTLAM

```

(Actuales -----)

```

DEFINE >MBActual        6PUTLAM
DEFINE >PBActual        7PUTLAM
DEFINE >OBActual        8PUTLAM
DEFINE >ABActual        9PUTLAM
DEFINE >CEActual       10PUTLAM

```

(Sbi -----)

```

DEFINE >MBi             11PUTLAM
DEFINE >PBi             12PUTLAM
DEFINE >OBi             13PUTLAM
DEFINE >ABi             14PUTLAM
DEFINE >CEi             15PUTLAM

```

(Eli parte 1 -----)

```

DEFINE >Box0            16PUTLAM
DEFINE >Xeli            17PUTLAM
DEFINE >Yeli            18PUTLAM
DEFINE >Xpix            19PUTLAM
DEFINE >Ypix            20PUTLAM
DEFINE >XJump           21PUTLAM
DEFINE >YJump           22PUTLAM

```

```

DEFINE >eLEN            #17 PUTLAM

```

```

DEFINE >CeLEn          #18 PUTLAM
DEFINE >SeceLi          #19 PUTLAM
DEFINE >AnimateEli?     #1A PUTLAM
DEFINE >EliGrob         #1B PUTLAM
DEFINE >eNList          #1C PUTLAM
DEFINE >eNSecini        #1D PUTLAM
DEFINE >EliBoost        #1E PUTLAM
DEFINE >EliPowDown?     #1F PUTLAM
DEFINE >FMDelayOn?      #20 PUTLAM
DEFINE >FMDelay?        #21 PUTLAM
DEFINE >DelayFM          #22 PUTLAM
DEFINE >DelayAniEli     #23 PUTLAM

```

(Mapa -----)

```

DEFINE >Map?            #24 PUTLAM
DEFINE >Map              #25 PUTLAM
DEFINE >MapH             #26 PUTLAM
DEFINE >MapW             #27 PUTLAM
DEFINE >2MapH            #28 PUTLAM
DEFINE >Hgbuffinpix      #29 PUTLAM
DEFINE >wgbuffinpix      #2A PUTLAM
DEFINE >InterBox         #2B PUTLAM

```

(MB -----)

```

DEFINE >MBf              #2C PUTLAM
DEFINE >MBActualA        #2D PUTLAM
DEFINE >MBNum            #2E PUTLAM
DEFINE >MBGrobList       #2F PUTLAM
DEFINE >ListMBtoRun      #30 PUTLAM
DEFINE >SizeListMBtoRun  #31 PUTLAM
DEFINE >RunListMBtoRun?  #32 PUTLAM
DEFINE >ListListMBtoRun  #33 PUTLAM
DEFINE >MBStep           #34 PUTLAM

```

(PB -----)

```

DEFINE >PBf              #35 PUTLAM
DEFINE >PBActualA        #36 PUTLAM
DEFINE >PBNum            #37 PUTLAM
DEFINE >PBGrobList       #38 PUTLAM
DEFINE >ListListPBtoRun  #39 PUTLAM

```

(OB -----)

```

DEFINE >OBf              #3A PUTLAM
DEFINE >OBActualA        #3B PUTLAM
DEFINE >OBNum            #3C PUTLAM
DEFINE >OBGrobList       #3D PUTLAM
DEFINE >ListListOBtoRun  #3E PUTLAM

```

(AB -----)

```

DEFINE >ABf              #3F PUTLAM
DEFINE >ABNum            #40 PUTLAM
DEFINE >ABActualA        #41 PUTLAM
DEFINE >ABGrobList       #42 PUTLAM
DEFINE >ListListABtoRun  #43 PUTLAM

```

(CE -----)

```

DEFINE >CEf              #44 PUTLAM
DEFINE >CENum            #45 PUTLAM
DEFINE >ListCEtoRun      #46 PUTLAM
DEFINE >SizeListCEtoRun  #47 PUTLAM
DEFINE >ListListCEtoRun  #48 PUTLAM
DEFINE >RunListCEtoRun?  #49 PUTLAM

```

(UE -----)

```

DEFINE >ListUEtoRun      #4A PUTLAM
DEFINE >RunListUEtoRun?  #4B PUTLAM
DEFINE >SizeListUEtoRun  #4C PUTLAM

```

(CB -----)

```

DEFINE >CBGrobList       #4D PUTLAM
DEFINE >BoxGrobindex     #4E PUTLAM
DEFINE >BoxGrob          #4F PUTLAM

```

```

DEFINE >BoxGrobH      #50 PUTLAM
DEFINE >BoxGrobW      #51 PUTLAM

(Reloj -----)
DEFINE >Nciclos      #52 PUTLAM
DEFINE >cNciclos      #53 PUTLAM
DEFINE >Nciclosinc    #54 PUTLAM

(bg -----)
DEFINE >bgGrobList    #55 PUTLAM
DEFINE >bgGrobList?   #56 PUTLAM

(fg -----)
DEFINE >fgGrobList    #57 PUTLAM
DEFINE >fgGrobList?   #58 PUTLAM

(ventana Inventario -----)
DEFINE >InvBGrobStd?   #59 PUTLAM
DEFINE >InvBGrob      #5A PUTLAM
DEFINE >InvBGh        #5B PUTLAM
DEFINE >InvBGw        #5C PUTLAM
DEFINE >InvdX         #5D PUTLAM
DEFINE >InvdY         #5E PUTLAM
DEFINE >InvX          #5F PUTLAM
DEFINE >InvY          #60 PUTLAM
DEFINE >InvGrob0      #61 PUTLAM

(Telepuerto -----)
DEFINE >TeleT?        #62 PUTLAM
DEFINE >TeleportFree? #63 PUTLAM
DEFINE >TelepDestino   #64 PUTLAM

(Metas -----)
DEFINE >Goaln         #65 PUTLAM
DEFINE >Goalc         #66 PUTLAM
DEFINE >Goal?         #67 PUTLAM
DEFINE >Youwin        #68 PUTLAM
DEFINE >YouwinStd?    #69 PUTLAM

(Animaciones -----)
DEFINE >aniH          #6A PUTLAM
DEFINE >aniW          #6B PUTLAM
DEFINE >aniX          #6C PUTLAM
DEFINE >aniY          #6D PUTLAM
DEFINE >anig0         #6E PUTLAM
DEFINE >aniDelay      #6F PUTLAM
DEFINE >aniLEN        #70 PUTLAM
DEFINE >aniFlag       #71 PUTLAM
DEFINE >aniMode       #72 PUTLAM

(Bombas -----)
DEFINE >Xbomb         #73 PUTLAM
DEFINE >Ybomb         #74 PUTLAM
DEFINE >Bomb?        #75 PUTLAM
DEFINE >Tbomb         #76 PUTLAM
DEFINE >CTbomb        #77 PUTLAM
DEFINE >BombType      #78 PUTLAM
DEFINE >Xbombpix      #79 PUTLAM
DEFINE >Ybombpix      #7A PUTLAM
DEFINE >BombGrob      #7B PUTLAM
DEFINE >AfterBombGrob #7C PUTLAM

(Shtx -----)
DEFINE >txList        #7D PUTLAM
DEFINE >txN           #7E PUTLAM
DEFINE >txW           #7F PUTLAM
DEFINE >txH           #80 PUTLAM
DEFINE >txG0          #81 PUTLAM
DEFINE >txX1          #82 PUTLAM
DEFINE >txY1          #83 PUTLAM
DEFINE >ShtxDelay     #84 PUTLAM
DEFINE >ShtxDelay0    #85 PUTLAM

```

```

DEFINE >ShtxKEY          #86 PUTLAM

(Armas -----)
DEFINE >EliPower          #87 PUTLAM
DEFINE >EliPowerIni       #88 PUTLAM
DEFINE >NBombs            #89 PUTLAM
DEFINE >NPinch            #8A PUTLAM
DEFINE >PinchOn?          #8B PUTLAM
DEFINE >worldkeys         #8C PUTLAM
DEFINE >worldMap?         #8D PUTLAM
DEFINE >Shield?           #8E PUTLAM
DEFINE >ScannerOn?        #8F PUTLAM
DEFINE >S+?               #90 PUTLAM

(Fx -----)
DEFINE >F1                #91 PUTLAM
DEFINE >F2                #92 PUTLAM
DEFINE >F3                #93 PUTLAM
DEFINE >F4                #94 PUTLAM
DEFINE >F5                #95 PUTLAM
DEFINE >F6                #96 PUTLAM
DEFINE >F7                #97 PUTLAM
DEFINE >F8                #98 PUTLAM
DEFINE >F9                #99 PUTLAM
DEFINE >FA                #9A PUTLAM
DEFINE >FB                #9B PUTLAM
DEFINE >FC                #9C PUTLAM
DEFINE >FD                #9D PUTLAM
DEFINE >FE                #9E PUTLAM
DEFINE >FF                #9F PUTLAM
DEFINE >ListListFxtorun #A0 PUTLAM

(TxInp -----)
DEFINE >TxMode            #A1 PUTLAM
DEFINE >TxBG              #A2 PUTLAM
DEFINE >TxNChars          #A3 PUTLAM
DEFINE >InpX              #A4 PUTLAM
DEFINE >InpY              #A5 PUTLAM
DEFINE >InpK              #A6 PUTLAM
DEFINE >InpText           #A7 PUTLAM
DEFINE >Inpc              #A8 PUTLAM
DEFINE >Inpg0             #A9 PUTLAM
DEFINE >Inpp1             #AA PUTLAM
DEFINE >InpFinish?        #AB PUTLAM
DEFINE >InpXpix           #AC PUTLAM
DEFINE >InpYpix           #AD PUTLAM
DEFINE >TxBGH             #AE PUTLAM
DEFINE >TxBGW             #AF PUTLAM
DEFINE >TxFlag            #B0 PUTLAM
DEFINE >TxBBGx            #B1 PUTLAM
DEFINE >TxBBGy            #B2 PUTLAM
DEFINE >TxBBGh            #B3 PUTLAM
DEFINE >TxBBGw            #B4 PUTLAM
DEFINE >TxBBGxpix         #B5 PUTLAM
DEFINE >TxBBGypix         #B6 PUTLAM
DEFINE >TxInfl            #B7 PUTLAM

(Mensajes de texto -----)
DEFINE >TxList?           #B8 PUTLAM
DEFINE >TxHelp?           #B9 PUTLAM
DEFINE >TxObj?            #BA PUTLAM
DEFINE >TxList            #BB PUTLAM
DEFINE >TxHelp            #BC PUTLAM
DEFINE >TxObj             #BD PUTLAM

(Pantalla -----)
DEFINE >MenuOn?           #BE PUTLAM
DEFINE >ShowCiclos?       #BF PUTLAM
DEFINE >ShowEliCoords?    #C0 PUTLAM

(T&M -----)
DEFINE >T&M?              #C1 PUTLAM

```

```

DEFINE >T&Mx          #C2 PUTLAM
DEFINE >T&My          #C3 PUTLAM
DEFINE >ticks0        #C4 PUTLAM
DEFINE >dticks        #C5 PUTLAM
DEFINE >T&Mg0        #C6 PUTLAM

```

(Memoria -----)

```

DEFINE >MEMi          #C7 PUTLAM
DEFINE >DMemList      #C8 PUTLAM
DEFINE >Nvars         #C9 PUTLAM

```

(Replays -----)

```

DEFINE >VKEY          #CA PUTLAM
DEFINE >VKEYc         #CB PUTLAM
DEFINE >VKEYList      #CC PUTLAM
DEFINE >VKEYrec?      #CD PUTLAM
DEFINE >VKEYplay?     #CE PUTLAM

```

(Choques -----)

```

DEFINE >MBClash?      #CF PUTLAM
DEFINE >MBClashX      #D0 PUTLAM
DEFINE >MBClashY      #D1 PUTLAM
DEFINE >MBClash1?     #D2 PUTLAM
DEFINE >MBClash2?     #D3 PUTLAM
DEFINE >MBClash3?     #D4 PUTLAM
DEFINE >PBClash?      #D5 PUTLAM
DEFINE >PBClashX      #D6 PUTLAM
DEFINE >PBClashY      #D7 PUTLAM
DEFINE >PBClash1?     #D8 PUTLAM
DEFINE >PBClash2?     #D9 PUTLAM
DEFINE >PBClash3?     #DA PUTLAM
DEFINE >Hiteli?       #DB PUTLAM
DEFINE >AIClash?      #DC PUTLAM
DEFINE >AIClashX      #DD PUTLAM
DEFINE >AIClashY      #DE PUTLAM
DEFINE >AIClash1?     #DF PUTLAM
DEFINE >AIClash2?     #E0 PUTLAM
DEFINE >AIClash3?     #E1 PUTLAM

```

(Sistema Search -----)

```

DEFINE >XSearch       #E2 PUTLAM
DEFINE >YSearch       #E3 PUTLAM
DEFINE >MB?           #E4 PUTLAM
DEFINE >MBindex       #E5 PUTLAM
DEFINE >PB?           #E6 PUTLAM
DEFINE >PBindex       #E7 PUTLAM
DEFINE >OB?           #E8 PUTLAM
DEFINE >OBindex       #E9 PUTLAM
DEFINE >AB?           #EA PUTLAM
DEFINE >ABindex       #EB PUTLAM
DEFINE >CE?           #EC PUTLAM
DEFINE >CEindex       #ED PUTLAM

```

(GLPack -----)

```

DEFINE >GLPackIncluded? #EE PUTLAM
DEFINE >GLPackBZ       #EF PUTLAM
DEFINE >GLPStd?        #F0 PUTLAM
DEFINE >GLPStdX        #F1 PUTLAM
DEFINE >aSLStd?        #F2 PUTLAM
DEFINE >aSLStdX        #F3 PUTLAM
DEFINE >aniSecList     #F4 PUTLAM
DEFINE >IndGrob        #F5 PUTLAM

```

(Builder -----)

```

DEFINE >BuilderMode?   #F6 PUTLAM
DEFINE >BMDDataType    #F7 PUTLAM

```

(Coordenadas inic ventana -----)

```

DEFINE >IniLeftCol     #F8 PUTLAM
DEFINE >IniTopRow       #F9 PUTLAM

```

(Modulo M1 -----)

```

DEFINE >M1off?      #FA PUTLAM
DEFINE >Accion?      #FB PUTLAM
DEFINE >GC?          #FC PUTLAM

```

(Teclado -----)

```

DEFINE >KEY          #FD PUTLAM
DEFINE >Repk?        #FE PUTLAM
DEFINE >KeyboardStd? #FF PUTLAM
DEFINE >KUp          #100 PUTLAM
DEFINE >KDown        #101 PUTLAM
DEFINE >KLeft        #102 PUTLAM
DEFINE >KRight       #103 PUTLAM
DEFINE >KHelp        #104 PUTLAM
DEFINE >KTakeob      #105 PUTLAM
DEFINE >KMap1        #106 PUTLAM
DEFINE >KMap2        #107 PUTLAM
DEFINE >KMap3        #108 PUTLAM
DEFINE >KGun1        #109 PUTLAM
DEFINE >KGun2        #10A PUTLAM
DEFINE >KGun3        #10B PUTLAM
DEFINE >KSpeedUp     #10C PUTLAM
DEFINE >KSpeedDw     #10D PUTLAM
DEFINE >KExit        #10E PUTLAM
DEFINE >KInventory   #10F PUTLAM

```

(ZxName y ZxMaster -----)

```

DEFINE >ZxName       #110 PUTLAM
DEFINE >ZxMaster     #111 PUTLAM

```

(Eli parte 2 -----)

```

DEFINE >eH           #112 PUTLAM
DEFINE >eW           #113 PUTLAM

```

(Ventana -----)

```

DEFINE >dHpix        #114 PUTLAM
DEFINE >dwpix        #115 PUTLAM
DEFINE >Xelib        #116 PUTLAM
DEFINE >Yelib        #117 PUTLAM

```

(Password -----)

```

DEFINE >MasterPassword #118 PUTLAM
DEFINE >MasterPassword? #119 PUTLAM

```

(PUTLAM- PARTE B -----)

( MBDATA -----)

```

DEFINE >MBtype       :: 6GETLAM      PUTLAM ;
DEFINE >MBstatus      :: 6GETLAM #1+  PUTLAM ;
DEFINE >MBxpix        :: 6GETLAM #2+  PUTLAM ;
DEFINE >MBypix        :: 6GETLAM #3+  PUTLAM ;
DEFINE >MBtest        :: 6GETLAM #4+  PUTLAM ;
DEFINE >MBprogT       :: 6GETLAM #5+  PUTLAM ;
DEFINE >MBprogF       :: 6GETLAM #6+  PUTLAM ;
DEFINE >MBL1          :: 6GETLAM #7 #+ PUTLAM ;
DEFINE >MBL0          :: 6GETLAM #8 #+ PUTLAM ;
DEFINE >MBC1          :: 6GETLAM #9 #+ PUTLAM ;
DEFINE >MBC0          :: 6GETLAM #A #+ PUTLAM ;
DEFINE >MBf1          :: 6GETLAM #B #+ PUTLAM ;
DEFINE >MBf0          :: 6GETLAM #C #+ PUTLAM ;
DEFINE >MBg1          :: 6GETLAM #D #+ PUTLAM ;
DEFINE >MBg0          :: 6GETLAM #E #+ PUTLAM ;
DEFINE >MBk           :: 6GETLAM #F #+ PUTLAM ;

```

( PBDATA -----)

```

DEFINE >PBtype       :: 7GETLAM      PUTLAM ;
DEFINE >PBstatus      :: 7GETLAM #1+  PUTLAM ;
DEFINE >PBxpix        :: 7GETLAM #2+  PUTLAM ;
DEFINE >PBypix        :: 7GETLAM #3+  PUTLAM ;
DEFINE >PBtest        :: 7GETLAM #4+  PUTLAM ;
DEFINE >PBprogT       :: 7GETLAM #5+  PUTLAM ;
DEFINE >PBprogF       :: 7GETLAM #6+  PUTLAM ;

```



```

DEFINE >PBL1          :: 7GETLAM #7 #+ PUTLAM ;
DEFINE >PBL0          :: 7GETLAM #8 #+ PUTLAM ;
DEFINE >Pbc1          :: 7GETLAM #9 #+ PUTLAM ;
DEFINE >Pbc0          :: 7GETLAM #A #+ PUTLAM ;
DEFINE >PBf1          :: 7GETLAM #B #+ PUTLAM ;
DEFINE >PBf0          :: 7GETLAM #C #+ PUTLAM ;
DEFINE >PBg1          :: 7GETLAM #D #+ PUTLAM ;
DEFINE >PBg0          :: 7GETLAM #E #+ PUTLAM ;
DEFINE >PBk           :: 7GETLAM #F #+ PUTLAM ;

( OBDATA -----)
DEFINE >OBtype        :: 8GETLAM          PUTLAM ;
DEFINE >OBstatus      :: 8GETLAM #1+      PUTLAM ;
DEFINE >OBxpix        :: 8GETLAM #2+      PUTLAM ;
DEFINE >OBypix        :: 8GETLAM #3+      PUTLAM ;
DEFINE >OBtest        :: 8GETLAM #4+      PUTLAM ;
DEFINE >OBprogT       :: 8GETLAM #5+      PUTLAM ;
DEFINE >OBprogF       :: 8GETLAM #6+      PUTLAM ;
DEFINE >OBL1          :: 8GETLAM #7 #+    PUTLAM ;
DEFINE >OBL0          :: 8GETLAM #8 #+    PUTLAM ;
DEFINE >Obc1          :: 8GETLAM #9 #+    PUTLAM ;
DEFINE >Obc0          :: 8GETLAM #A #+    PUTLAM ;
DEFINE >OBf1          :: 8GETLAM #B #+    PUTLAM ;
DEFINE >OBf0          :: 8GETLAM #C #+    PUTLAM ;
DEFINE >OBg1          :: 8GETLAM #D #+    PUTLAM ;
DEFINE >OBg0          :: 8GETLAM #E #+    PUTLAM ;
DEFINE >OBk           :: 8GETLAM #F #+    PUTLAM ;

( ABDATA -----)
DEFINE >ABtype        :: 9GETLAM          PUTLAM ;
DEFINE >ABstatus      :: 9GETLAM #1+      PUTLAM ;
DEFINE >ABxpix        :: 9GETLAM #2+      PUTLAM ;
DEFINE >ABypix        :: 9GETLAM #3+      PUTLAM ;
DEFINE >ABg0          :: 9GETLAM #4+      PUTLAM ;

( CEDATA -----)
DEFINE >CEtype        :: 10GETLAM         PUTLAM ;
DEFINE >CEstatus      :: 10GETLAM #1+     PUTLAM ;
DEFINE >CEtest        :: 10GETLAM #2+     PUTLAM ;
DEFINE >CEprogT       :: 10GETLAM #3+     PUTLAM ;
DEFINE >CEprogF       :: 10GETLAM #4+     PUTLAM ;
DEFINE >CEx5pix       :: 10GETLAM #5+     PUTLAM ;
DEFINE >CEy5pix       :: 10GETLAM #6+     PUTLAM ;
DEFINE >CEx4pix       :: 10GETLAM #7+     PUTLAM ;
DEFINE >CEy4pix       :: 10GETLAM #8+     PUTLAM ;
DEFINE >CEx3pix       :: 10GETLAM #9+     PUTLAM ;
DEFINE >CEy3pix       :: 10GETLAM BINT10 #+ PUTLAM ;
DEFINE >CEx2pix       :: 10GETLAM BINT11 #+ PUTLAM ;
DEFINE >CEy2pix       :: 10GETLAM BINT12 #+ PUTLAM ;
DEFINE >CEx1pix       :: 10GETLAM BINT13 #+ PUTLAM ;
DEFINE >CEy1pix       :: 10GETLAM BINT14 #+ PUTLAM ;
DEFINE >CEx0pix       :: 10GETLAM BINT15 #+ PUTLAM ;
DEFINE >CEy0pix       :: 10GETLAM BINT16 #+ PUTLAM ;
DEFINE >CEL2          :: 10GETLAM BINT17 #+ PUTLAM ;
DEFINE >CEL1          :: 10GETLAM BINT18 #+ PUTLAM ;
DEFINE >CEL0          :: 10GETLAM BINT19 #+ PUTLAM ;
DEFINE >CEC2          :: 10GETLAM BINT20 #+ PUTLAM ;
DEFINE >CEC1          :: 10GETLAM BINT21 #+ PUTLAM ;
DEFINE >CEC0          :: 10GETLAM BINT22 #+ PUTLAM ;
DEFINE >CEf2          :: 10GETLAM BINT23 #+ PUTLAM ;
DEFINE >CEf1          :: 10GETLAM BINT24 #+ PUTLAM ;
DEFINE >CEf0          :: 10GETLAM BINT25 #+ PUTLAM ;
DEFINE >CEg2          :: 10GETLAM BINT26 #+ PUTLAM ;
DEFINE >CEg1          :: 10GETLAM BINT27 #+ PUTLAM ;
DEFINE >CEg0          :: 10GETLAM BINT28 #+ PUTLAM ;
DEFINE >CEk2          :: 10GETLAM BINT29 #+ PUTLAM ;
DEFINE >CEk1          :: 10GETLAM BINT30 #+ PUTLAM ;
DEFINE >CEk0          :: 10GETLAM BINT31 #+ PUTLAM ;

(-----)

```