======================================
================================

The Flaming Bird Disassembler - (c) Phœnix corp. 1992-94

Files of extension

======================================
================================

Intro

Most of the users of a désassembleur are a day confronted has a particular case which was not originally planned by the programmer, as for example a gigantic specific zone of data to the program to be dismantled organized in a known way but which éxigerait to cross(spend) days whole tops. To counter in this probleme were create the scripts, goes out of interpreted mini-language allowing to manage most of the simple, but fast limited cases when it is a question of treating(handling) relatively complex zones, even whole file. He can for example be practical to be able to dismantle systematically certain type(chap) of file (bookshops, dated) in a effective and fast way. It is in this optics whom(that) were create the files of extension.

If you write files of extension which you judge a certain interest (notament for the dismantling of known files of type(chap)), send I, they will be spread(broadcasted) with the next versions of TFBD and will take advantage has all the users (I shall send back to(dismiss) you the diskette, with an update of TFBD if necessary).

======================================
================================

Doc

The files of extension are type(chap) TLK's modules $(BC, accommodated in the directory 1:Expand. Their auxtype has a particular meaning: he(it) détèrmine in which measure the extension can apply to the file in the course of dismantling.

Bit   15: ' Load File ' only
Bits   8-14: Reserved (0)
Bits   0-7: type(chap) of file treated(handled) (0 = everything typifies)

For example, an extension of auxtype 80B3 will apply only to the applications P16 and GS / BONES, $8000 in any segment of file OMF, $0006 in files binary and $0000 in all the files.

The extensions can etre multi-segments and contain resources, but the field Entry of the main segment has to clock(point,stick) at a particular structure, definie as follows:

```
00 Length    Signature    ASC " TFBD "
04 Word      Version      $0100)
06 Word      Reserved     ( Must be put has 0)
08 Length    TFBD_Call
0C...   At the beginning of the code of the extension
```

The field ' TFBD_Call ' will be set up by TFBD at the time of the load of the extension and will contain a long jump (JMPL) towards a point of internal entrance(entry) to TFBD.

'Version' contains the minimum version number of TFBD necessary for the maid éxécution of the extention. At the moment, the only possible value is $0100 (v1.0).

The signature is necessarily there ascii superieur.

The extensions are called by TFBD via a JSL, and have to by consequent end by a RTL.

Attention: an extention is not a command(order) shell, and does not on no account have to end by a Quit GS / bone.

Pile(Battery) and direct page:

The extention is called with, already set up, its own pile(battery) and its own direct page. The total size(cutting) of the buffer is of $800, is $100 for the direct page and $the 700 for the pile(battery), what should be sufficient(self-important) in most of the cases. If the extension has a more important need in pile(battery) for example, to her to take care with assigning the other one either to have its own segment DP / Stack.

Values of the direct page:

The past various data have the extension are stored in his(her,its) direct page, as follows:

```
$80    Length        ObjPtr      Timekeeper on the file / segment
$84    Length        ObjLength       Length of the segment
$88    Word      SegNum      Number of the segment
8A     Word      FileType    Type(Chap) of the file
8C     Length        AuxType         Auxiliary type(chap)
$90    Length        SegHead         Timekeeper on the segment to
header
```

$94	Length		OMF_Hnd	Handle on information OMF
$98	Word	MemID		ID of the extension
$9A-AF		Rffe		Reserved

In the case of a file not relogeable, the number of segment is has 1, the segment to header contains not significant information and there is no information OMF. memID is specific has the extension, and it makes what she wants with. But as any GS / BONE application, it is asked to leave the memory in the state where it found her(it) by entering.

Attention: the éxécution of an extention which can be suspended by the user, this one has to assign some memory (if need be) only with ID which TFBD communicated him(her) (and possibly its auxiliary IDs). She does not have to build up it to itself of the other one via _GetNewID for example.

Calls(Appeals) to the Flaming Bird Disassembler:

The calls(appeals) have TFBD are made in the style of toolsets: parameters crossed(spent) by the pile(battery), the number of function(office) in X then JSL. Normally, a file of macro and of equates is delivered with the soft: if you do not have him(it), bawl, parceque I not decrirai not here the organization of the register X. I have the macro, I use(get) it (also, I don't care royally numbers of tools when I schedule(program)!).

A difference however is to be noted between the calls(appeals) has TFBD and the calls(appeals) to tools: it is useless to reserve of the space in the pile(battery) for the values of return.

* ------Offsets and Addresses

Most of the getting(touching) calls(appeals) have the object require not an address but an offset, that is the byte of the code that we reference, and not the address that he(it) would occupy loaded(charged) there memoire. For example for a file SYS, the address $2000 is the offset $0000 . As very often an extension can fetch an address in the code (in a table of addresses for example), it is previously necessary to transform her(it) into offset. This problem does not generally settle(arise) with the code OMF, a file OMF containing only offsets.

Two calls(appeals) are planned for this kind of thing:

- AddrToOffset:

Parameters of entrance(entry):   Exit(Release):

```
Previous tells              Previous tells
----------------4           ----------------4
                            Segment to number
                            ----------------2
Address                     Offset
----------------< - POSTAL AREA    ----------------< - POSTAL AREA
```

Errors: expBadOffset ($0101)

- OffsetToAddr: inverse Call(Appeal). Converts offsets in address, in function(office) ORG of the code. The parameters are the opposite of those of AddrToOffset.

Errors: expBadAddr ($0102)

For the problem which consists in fetching an address directly in the code, for example an address referencée by a JMP or a DA, a specific call(appeal) is planned: EvaluateEA, who avoids to the programmer of extensions to set himself counts the segment referenced by the address either the value of the DBR for an addressing absolute report, for example.

- EvaluateEA: evaluation of an effective address

```
Entrance(Entry):                        Exit(Release):

Previous tells              Previous tells
----------------8           ----------------< - POSTAL AREA
Segment to number
----------------6
Offset
----------------4


EARecPtr

----------------< - POSTAL AREA
```

EARecPtr is a timekeeper on a structure in which EvaluateEA is going to register its answer; she(it) is organized as follows:

0	Word	Kind
2	Word	Size (bytes)
4	Word	Shift count
6	Length	Reference

Kind indicates of which type(chap) is the reference:

      0:      Lo Word = Offset, Hi Word = Segment
      1:      Offset / segment opérande of an instruction 816
      2:      Offset / segment via a recording OFFSET
      3:      Absolute address (extern(day pupil) in the code)
      4:      Address in direct page
      5:      Indeterminé ("Maybe" an address)

If TFBD finds no address in this place (for example if the offset which is crossed(spent) to him(her) clocks at an instruction or in the middle of a constant), he(it) turns(returns) the error to expNoAddr ($0103).

* ------Alignments

      Two calls(appeals) exist to move in the code without caring about the dismantling:

- PreviousInst: go back up(raise) of an instruction in the code

      Entrance(Entry):   Exit(Release):

      Previous tells   Previous tells
      ----------------2   ----------------2
      Offset   New offset
      ----------------< - POSTAL AREA   ----------------< - POSTAL AREA

- NextInst: advance of an instruction

      Entrance(Entry):   Exit(Release):

      Previous tells   Previous tells
      ----------------2   ----------------2
      Offset   New offset
      ----------------< - POSTAL AREA   ----------------< - POSTAL AREA

The offset is necessarily in the common(current) segment.

If the offset is not aligned over the beginning of the instruction, he(it) is considered as such; " PreviousInst offset " will not align the offset over the beginning of the common(current) instruction but over it he of the previous one (it is the purpose).

Funny property: an enchainement _PreviousInst - _NextInst aligns the offset on the common(current) instruction. But this function(office) has a special command(order):

- Align: align the offset on the instruction in which it points(clocks,welds).

          Entrance(Entry):   Exit(Release):

          Previous tells   Previous tells
          ----------------2   ----------------2
          Offset   New offset
          ----------------< - POSTAL AREA   ----------------< - POSTAL AREA

Has no effect if Offset is already aligned.

* ------diverse Calls(Appeals)

- Message: post(show) a court (max 77 tanks) text on the line of bottom(stocking).
                 The text has to be a PString (byte + ascii).

          Entrance(Entry):   Exit(Release):

          Previous tells   Previous tells
          ----------------4   ----------------< - POSTAL AREA

          G-string to clock(point,stick)

          ----------------< - POSTAL AREA

- Error: Post(Show) an error message (" > ERR: " + Beep + Message), waits
                A key(touch) then returns the hand has the extention.
                The parameters are the same that for Message.

* ------Constants

        The following calls(appeals) serve has to define a constant in a place of the code. The parameters are identical to all the constants:

          Entrance(Entry):   Exit(Release):

          Previous tells   Previous tells
          ----------------6   ----------------< - POSTAL AREA

Segment to number
----------------4
Offset
----------------2
Constant size
----------------< - POSTAL AREA

The call(appeal) is made via the macro _AddConst followed by the type(chap) of constant (dw, str) in small letters:

```
PHW   SegNum
PHW   Offset
PHW   CSize
_AddConst ctype          Or     AddConst ctype; Seg; Off; cSize
```

For the constants of fixed length dw, da, ddb, adrl and adr the parameter cSize must be multiple of the length of the constant; otherwise, he(it) is aligned on the closest multiple superior(higher education). He(it) is taken such which that for: db, ds, asc, hex, rev, str and strl. For G-strings, the length includes the byte or the word of length, then attention at the risk of bad dismantling. A length of zero can provoke small surprises also...

* ------Labels

The definition of a label requires 4 parameters: the place (Seg / Off) where we place him, his(her,its) radius of action, the name and his type(chap) (lbl, ent, equ). His(her,its) radius of action represents the number of bytes from the label which will be sent under the shape " Label+n ". Generally, it is the length of the instruction or the constant in front of which we place him(it). A radius of action of zero will automatically be calculated by TFBD as indicated precedament. Call(appeal):

- SetLabel: definition of a label.

Macro: _SetLabel lType or SetLabel lType; Seg; Off; FieldSize; StrPtr
        With lType = lbl, ent or equ.

Entrance(Entry):   Exit(Release):

Previous tells   Previous tells
----------------10   ----------------< - POSTAL AREA
Segment to number
----------------8
Offset
----------------6

Field size
        ----------------4

        Name to clock(point,stick)

        ----------------< - POSTAL AREA

Any label précédament defined has the given offset will be erased.

* ------Directives

- AddOrigin (Seg, Off, Org / 4): definition of an origin of assembly.

        Entrance(Entry):   Exit(Release):

        Previous tells   Previous tells
        ----------------8   ----------------< - POSTAL AREA
        Segment
        ----------------6
        Offset
        ----------------4

        Origin

        ----------------< - POSTAL AREA

- AddMX (Seg, Off, MX): change the size(cutting) of registers

        MX: Word of the shape $0000, $0001, $0100 or $0101 .
                Weighty byte Fort: M
                    :::   Weak(Weakness): X   (0=16b, 1=8b)

- AddReloc (Seg, Off, Size, Disp, Shift, rSeg, rOff): Relocation.

        Seg / Off: Place where to place the relocation
        Size: Number of bytes of the reference
        Disp: Deplacement [+-80, 7F]
        Shift: Decalage in bits, < 0 towards the right
        RSeg / rOff: reference

- AddExtReloc (Seg, Off, Size, Disp, Shift, Addr / 4): external Relocation.

Seg / Off, Size, Disp, Shift: idem that AddReloc
Addr: address of the reference

- AddDataBank (Seg, Off, 0, DBR): change the value of the DBR.

      Seg / Off: Place where to make the change.
      0:           Reserved - Put has zero
      DBR: New value of the DBR.
                   For $B=K, DBR=8000.

- AddOffset (Seg, Off, 0,0): place an offset.

      Seg / Off: Place where to place the offset
      0:           Reserved - Put has zero
                   Should serve has to treat(handle) differents types(chaps) of offsets,
even
                   Differences between labels (Label2-Label1 by ex).

- AddComment (Seg, Off, @String): place a comment

      Seg / Off: Place where to place the comment.
      @String: Timekeeper on the pString of the comment.

======================================
==============================

He(It) is normally delivered with TFBD by the samples of extensions with their sources Merlin; do not hesitate has to go to make it a small tour(ballot) to see how's that walking(step), it is made for that.

======================================
==============================
                                    Of That all folk...
======================================
==============================