Mousepointer

COLLABORATORS

	TITLE : Mousepointer		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		February 14, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Mousepointer

1.1 AFC Module: mousepointer / Main

** Mousepointer V1.10 - Original By Andrea Galiberti **

Part of Amiga Foundation Classes

Introduction

Author(s) Info

History
AFC doc

Requires: Hardsprite , Displayer , nodemaster

Base: $8002

```
COMMANDS                          BRIEF DESCRIPTION
-----------------------------------------------------------------------

            mousepointer()
            Initialises the mousepointer object

            image()
            Stores the mouse image

            changeImage()
            Changes the mouse image

            hotspot()
            Sets the hotspot

            move()
            Moves the mouse
```

```
x()
Returns x coordinate

y()
Returns y coordinate

update()
Updates mouse position

auto()
Auto updates mouse position

vp()
Returns mouse viewport

version()
Gets mousepointer version


ERROR TABLE
```

1.2 Author(s) Info

Original By: Andrea Galimberti

E Version By: Andrea Galimberti

1.3 Mousepointer / Introduction

Mousepointer.

This MODULE exploits the Hardsprite MODULE TO move a pointer around a viewport. You can move the pointer sprite by hand, or allow the module do it for you by reading the mouse hardware registers (update() method). The mouse sprite can be automatically clipped to a chosen viewport. If you use this module in conjunction with the Displayer module, you can have your mouse sprite automatically pass from one viewport to another taking care of the changing resolution of the different viewports (auto() method). Of course the mouse sprite image can be changed at your will.

1.4 Mousepointer / Error Table

Val (Hex)	Description
none	

1.5 Mousepointer / History

```
  V1.10 - added resourceTracker support

  V1.00 - First release
```

1.6 AFC module: mousepointer / mousepointer()

```
         NAME: mousepointer(number=-1, resTracker=NIL)

  DESCRIPTION: Initialises a mousepointer object.

       INPUTS: number of the (hardware) sprite (from 0 to 7) to be used to
               represent the pointer on the screen.  The default value of
               -1 means that it will be allocated the first sprite
               available.

               resTracker: pointer to an optional resourceTracker object

      RESULTS: FALSE if the requested sprite is already in use and so it
               couldn't be allocated.
               TRUE otherwise.

     SEE ALSO:  hardsprite / hardsprite()
```

1.7 AFC module: mousepointer / image()

```
         NAME: image(height, list:PTR TO INT)

  DESCRIPTION: this method stores the image to be used with the mouse
               sprite.  Whenever this method is called, it will discard
               the previous image.

       INPUTS: height (in lines) of the image,
               a typed list (it MUST be typed to INT) in the following
               form:

                           [ $0000, $0000,
                             $0000, $0000,
                             ...          ]:INT

                               ^      ^
                               |      |
                               |      second bitplane (16 bits wide)
                               |
                               first bitplane (16 bits wide)

               This way you can select one of 4 colours for each pixel of
               the sprite's image.

      RESULTS: returns FALSE if: sprite non allocated (with mousepointer()),
                                 height<=0,
```

pointer to list =NIL.
The hardsprite will raise "No Memory" if it cannot allocate the (CHIP) memory for the image.

SEE ALSO: hardsprite / image()

1.8 AFC module: mousepointer / changeImage()

NAME: changeImage(viewport)

DESCRIPTION: changes the mouse sprite image to the stored one.

INPUTS: pointer to a viewport structure. If this pointer is <>NIL then the change will affect only the chosen viewport; if pointer=NIL then the change is relative to the current view.

RESULTS: FALSE if viewport<0 or mouse not allocated.
Otherwise TRUE.

SEE ALSO:
image()
hardsprite / changeImage()

1.9 AFC module: mousepointer / hotspot()

NAME: hotspot(x,y)

DESCRIPTION: sets the mouse hotspot

INPUTS: x and y coordinates of the mouse hotspot: these coordinates are relative to the top-left hand corner of the mouse image. (Negative values are allowed.)

RESULTS: NONE

SEE ALSO:

1.10 AFC module: mousepointer / update()

NAME: update(viewport, hclip=TRUE, vclip=TRUE)

DESCRIPTION: when called, this method moves the pointer hotspot to the new mouse coordinates relative to the top-left hand corner of the chosen viewport.

INPUTS: viewport: pointer to a viewport structure; if this pointer is NIL then the pointer movements are relative to the current view.

```
                hclip: TRUE= pointer hotspot cannot exceed the horizontal
                dimensions of the viewport (this flag is switched
                automatically to FALSE if viewport=NIL, because in this
                case there aren't any bounds to check for).

                vclip: same as hclip, but for the vertical direction.

       RESULTS: NONE

      SEE ALSO:
                 auto()
```

1.11 AFC module: mousepointer / move()

```
          NAME: move(viewport, x,y)

   DESCRIPTION: moves the mouse image to the desired location on a
                viewport (either Intuition or hand made), taking care of
                the hotspot position.

        INPUTS: viewport: pointer to a viewport structure,
                x and y coordinates of the site where to move the mouse
                hotspot.

       RESULTS: NONE

      SEE ALSO:
```

1.12 AFC module: mousepointer / x()

```
                          NAME: x()

   DESCRIPTION: returns the x coordinate of the mouse hotspot

        INPUTS: NONE

       RESULTS: x coordinate of mouse hotspot

      SEE ALSO:
                 y()
```

1.13 AFC module: mousepointer / y()

```
                          NAME: y()

   DESCRIPTION: returns the y coordinate of the mouse hotspot.

        INPUTS: NONE
```

```
     RESULTS: y coordinate of mouse hotspot

    SEE ALSO:
               x()
```

1.14 AFC module: mousepointer / auto()

```
                         NAME: auto(vo:PTR TO displayer, hclip=TRUE)

 DESCRIPTION: when called, moves the pointer hotspot to the new
              coordinates, taking care of changing viewport if necessary
              and automatically switching to the new resolution.  One
              thing you must be aware of:  when you position the mouse
              sprite in the first viewport you have also to set the
              correct mouse hotspot for that resolution; the auto()
              method then takes care of rescaling the hotspot coordinates
              when the resolution changes.

      INPUTS: vo: pointer to a displayer (see the BuildView module
              documentation); this pointer contains all the information
              concerning dimensions and resolutions of all the viewports
              displayed in the current view.

              hclip: TRUE= pointer hotspot cannot exceed the horizontal
              dimensions of the current viewport.

     RESULTS: NONE

    SEE ALSO:
               update()
```

1.15 AFC module: mousepointer / vp()

```
                         NAME: vp()

 DESCRIPTION: returns the number (starting from 0) of the viewport the
              mouse sprite is in.  This value is meaningful only if you
              are using the auto() method to move the mouse.

      INPUTS: NONE

     RESULTS: number of viewport the mouse is in

    SEE ALSO:
               auto()
```

1.16 AFC module: mousepointer / version()

```
         NAME: version()

  DESCRIPTION: returns the version number of the Mouse module.

       INPUTS: NONE

      RESULTS: version, revision of mousepointer

     SEE ALSO:
```