

GraphicsD

COLLABORATORS

	<i>TITLE :</i> GraphicsD		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 22, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	GraphicsD	1
1.1	GraphicsD	1
1.2	TMP:Modula-2/GraphicsD.def	3

Chapter 1

GraphicsD

1.1 GraphicsD

Konstanten

a2024MonitorID	a2024fifteenhertzKey	a2024tenhertzKey
alertLayersNoMem	anfracsize	animhalf
b2Bobber	b2Norm	b2Swap
bigBlits	blitMsgFault	borderHit
bottomHit	broadcastBeamcon	broadcastHbstop
broadcastHbstrt	broadcastHsstop	broadcastHsstrt
broadcastVbstop	broadcastVbstrt	broadcastVsstop
broadcastVsstrt	colorMask	colorOn
dblpf	defaultMonitorID	defaultMonitorName
dftchMask	diAvailNochip	diAvailNomonitor
diAvailNotwithgenlock	displayNameLen	dtagDims
dtagDisp	dtagMntr	dtagName
extrahalfbriteKey	extrahalfbritelaceKey	fineScroll
fineScrollMask	fineScrollShift	fromMonitor
hamKey	hamlaceKey	hiresKey
hiresdpf2Key	hiresdpfKey	hireslaceKey
hireslacedpf2Key	hireslacedpfKey	holdnmodify
horizPos	interlace	invalidID
isGrtrX	isGrtrY	isLessX
isLessY	jam1	jam2
leftHit	lmnRegion	lof
loresKey	loresdpf2Key	loresdpfKey
loreslaceKey	loreslacedpf2Key	loreslacedpfKey
m640	mapColor	maxFontMatchWeight
mcompatMixed	mcompatNobody	mcompatSelf
minNtscRow	minPalRow	minVga70Row
minVgaRow	monitorIDmask	monitorSpecType
move	needsNoConcealedRasters	←
needsNoLayerblitDamage		
newLayerInfoCalled	next	normalFont
ntscMonitorID	ntscMonitorName	palMonitorID
palMonitorName	pf2pri	plnCntMsk
plnCntShft	protoMonitorID	ratioFixedpart
ratioUnity	requestA2024	requestNtsc
requestPal	requestSpecial	rightHit
ringtrigger	sht	specialBeamcon

specialMonitorType	spriteAttached	ssGraphics
standardColorclocks	standardDeniseMax	standardDeniseMin
standardHbstop	standardHbstrt	standardHsstop
standardHsstrt	standardMonitorMask	←
standardNtscBeamcon		
standardNtscRows	standardPalBeamcon	standardPalRows
standardVbstop	standardVbstrt	standardViewX
standardViewY	standardVsstop	standardVsstrt
standardXoffset	standardYoffset	superKey
superdpf2Key	superdpfKey	superlaceKey
superlacedpf2Key	superlacedpfKey	sys
taDeviceDPI	toMonitor	topHit
vga70Beamcon	vga70Colorclocks	vga70DeniseMin
vga70Hbstop	vga70Hbstrt	vga70Hsstop
vga70Hsstrt	vga70MonitorName	vga70TotalRows
vga70Vbstop	vga70Vbstrt	vga70Vsstop
vga70Vsstrt	vgaColorclocks	vgaDeniseMin
vgaHbstop	vgaHbstrt	vgaHsstop
vgaHsstrt	vgaMonitorID	vgaMonitorName
vgaTotalRows	vgaVbstop	vgaVbstrt
vgaVsstop	vgaVsstrt	←
vgaextrahalfbriteKey		
vgaextrahalfbritelaceKey	vgaextraloresKey	←
vgaextraloresdpf2Key		
vgaextraloresdpfKey	vgaextraloreslaceKey	←
vgaextraloreslacedpf2Key		
vgaextraloreslacedpfKey	vgahamKey	vgahamlaceKey
vgaloresKey	vgaloresdpf2Key	vgaloresdpfKey
vgaloreslaceKey	vgaloreslacedpf2Key	vgaloreslacedpfKey
vgaproductKey	vgaproductdpf2Key	vgaproductdpfKey
vgaproductlaceKey	vgaproductlacedpf2Key	←
vgaproductlacedpfKey		
viewExtraType	viewportExtraType	vposrlof
vrtclPos	vrtclPosShift	vtagEndCm
wait		

Typ-Deklarationen

AnalogSignalInterval	AnalogSignalIntervalPtr	AnimComp
AnimCompPtr	AnimOb	AnimObPtr
AreaInfo	AreaInfoPtr	←
AvailabilityFlagSet		
AvailabilityFlags	BitMap	BitMapPtr
BitScaleArgs	BitScaleArgsPtr	Bob
BobFlagSet	BobFlags	BobPtr
ChipRevSet	ChipRevs	ClipRect
ClipRectPtr	CollTable	CollTablePtr
ColorFontColors	ColorFontColorsPtr	ColorMap
ColorMapFlagSet	ColorMapFlags	ColorMapPtr
ColorMapType	ColorTextFont	←
ColorTextFontFlagSet		
ColorTextFontFlags	ColorTextFontPtr	CopIns
CopInsPtr	CopList	CopListPtr
Copinit	CopinitPtr	Cprlist
CprlistPtr	DBufPacket	DBufPacketPtr
DimensionInfo	DimensionInfoPtr	DisplayFlagSet
DisplayFlags	DisplayInfo	DisplayInfoHandle

DisplayInfoPtr	DrawModeSet	DrawModes
ExtendedNode	ExtendedNodePtr	FontFlagSet
FontFlags	FontStyleSet	FontStyles
GelsInfo	GelsInfoPtr	GfxBase
GfxBasePtr	Isrvstr	IsrvstrPtr
Layer	LayerFlagSet	LayerFlags
LayerInfo	LayerInfoPtr	LayerPtr
LongProc	MonitorInfo	MonitorInfoPtr
MonitorSpec	MonitorSpecPtr	NameInfo
NameInfoPtr	Point	PointPtr
PropertyFlagSet	PropertyFlags	QueryHeader
QueryHeaderPtr	RasInfo	RasInfoPtr
RastPort	RastPortFlagSet	RastPortFlags
RastPortPtr	Rect32	Rect32Ptr
Rectangle	RectanglePtr	Region
RegionPtr	RegionRectangle	RegionRectanglePtr
SimpleSprite	SimpleSpritePtr	SpecialMonitor
SpecialMonitorPtr	TTextAttr	TTextAttrPtr
TextAttr	TextAttrPtr	TextExtent
TextExtentPtr	TextFont	↔
TextFontExt0FlagSet		
TextFontExt0Flags	TextFontExtension	↔
TextFontExtensionPtr		
TextFontPtr	TmpRas	TmpRasPtr
UCopList	UCopListPtr	VSprite
VSpriteFlagSet	VSpriteFlags	VSpritePtr
VTags	View	ViewExtra
ViewExtraPtr	ViewModeSet	ViewModes
ViewPort	ViewPortExtra	ViewPortExtraPtr
ViewPortPtr	ViewPtr	

1.2 TMP:Modula-2/GraphicsD.def

```
DEFINITION MODULE GraphicsD; (*$ Implementation:=FALSE *)
(* 10-Mar-1992/cn *)
```

```
FROM SYSTEM IMPORT ADDRESS,BITSET,BYTE,LONGSET,WORD,SHIFT,SHORTSET;
```

```
FROM ExecD IMPORT
  Interrupt , Library , List , Message , MinList , Node , NodePtr , NodeType ,
  SignalSemaphore , SignalSemaphorePtr , TaskPtr , MsgPortPtr ;
```

```
FROM Hardware IMPORT
  BltnodePtr , BeamConFlags , BeamConFlagSet ;
```

```
FROM UtilityD IMPORT
  HookPtr ,tagUser, TagItemPtr ;
```

```
TYPE
```

```
(*
  Die Datentypen von Graphics referenzieren sich oft gegenseitig.
  Aus diesem Grund werden alle Zeiger zuerst deklariert.
*)
```

```
(*36*) AnalogSignalIntervalPtr =POINTER TO AnalogSignalInterval ;
  AnimCompPtr =POINTER TO AnimComp ;
```

```

AnimObPtr =POINTER TO  AnimOb ;
AreaInfoPtr =POINTER TO  AreaInfo ;
BitMapPtr =POINTER TO  BitMap ;
(*36*) BitScaleArgsPtr =POINTER TO  BitScaleArgs ;
BobPtr =POINTER TO  Bob ;
ClipRectPtr =POINTER TO  ClipRect ;
CollTablePtr =POINTER TO  CollTable ;
(*36*) ColorFontColorsPtr =POINTER TO  ColorFontColors ;
ColorMapPtr =POINTER TO  ColorMap ;
(*36*) ColorTextFontPtr =POINTER TO  ColorTextFont ;
CopinitPtr =POINTER TO  Copinit ;
CopInsPtr =POINTER TO  CopIns ;
CopListPtr =POINTER TO  CopList ;
CprlistPtr =POINTER TO  Cprlist ;
DBufPacketPtr =POINTER TO  DBufPacket ;
(*36*) DimensionInfoPtr =POINTER TO  DimensionInfo ;
(*36*) DisplayInfoPtr =POINTER TO  DisplayInfo ;
(*36*) ExtendedNodePtr =POINTER TO  ExtendedNode ;
GelsInfoPtr =POINTER TO  GelsInfo ;
GfxBasePtr =POINTER TO  GfxBase ;
IsrvstrPtr =POINTER TO  Isrvstr ;
LayerPtr =POINTER TO  Layer ;
LayerInfoPtr =POINTER TO  LayerInfo ;
(*36*) MonitorInfoPtr =POINTER TO  MonitorInfo ;
(*36*) MonitorSpecPtr =POINTER TO  MonitorSpec ;
(*36*) NameInfoPtr =POINTER TO  NameInfo ;
(*36*) PointPtr =POINTER TO  Point ;
(*36*) QueryHeaderPtr =POINTER TO  QueryHeader ;
RasInfoPtr =POINTER TO  RasInfo ;
RastPortPtr =POINTER TO  RastPort ;
(*36*) Rect32Ptr =POINTER TO  Rect32 ;
RectanglePtr =POINTER TO  Rectangle ;
RegionPtr =POINTER TO  Region ;
RegionRectanglePtr =POINTER TO  RegionRectangle ;
SimpleSpritePtr =POINTER TO  SimpleSprite ;
(*36*) SpecialMonitorPtr =POINTER TO  SpecialMonitor ;
TextAttrPtr =POINTER TO  TextAttr ;
(*36*) TextExtentPtr =POINTER TO  TextExtent ;
(*36*) TextFontExtensionPtr =POINTER TO  TextFontExtension ;
TextFontPtr =POINTER TO  TextFont ;
TmpRasPtr =POINTER TO  TmpRas ;
(*36*) TTextAttrPtr =POINTER TO  TTextAttr ;
UCopListPtr =POINTER TO  UCopList ;
(*36*) ViewExtraPtr =POINTER TO  ViewExtra ;
ViewPtr =POINTER TO  View ;
(*36*) ViewPortExtraPtr =POINTER TO  ViewPortExtra ;
ViewPortPtr =POINTER TO  ViewPort ;
VSpritePtr =POINTER TO  VSprite ;

```

```

Rectangle =RECORD
minX,minY,maxX,maxY:INTEGER;
END;

```

```

Rect32 =RECORD
minX,minY,maxX,maxY:LONGINT
END;

```

```
Point =RECORD
x,y:INTEGER;
END;
```

```
BitMap =RECORD
bytesPerRow:CARDINAL;
rows:CARDINAL;
flags:SHORTCARD;
depth:SHORTCARD;
pad:CARDINAL;
planes:ARRAY [0..7] OF ADDRESS;
END;
```

```
Layer =RECORD
front: LayerPtr ;
back: LayerPtr ;
clipRect: ClipRectPtr ;
rp: RastPortPtr ;
bounds: Rectangle ;
reserved:ARRAY [0..3] OF BYTE;
priority:CARDINAL;
flags:CARDINAL;
superBitMap: BitMapPtr ;
superClipRect: ClipRectPtr ;
window:ADDRESS;
scrollX:INTEGER;
scrollY:INTEGER;
cr: ClipRectPtr ;
cr2: ClipRectPtr ;
crnew: ClipRectPtr ;
superSaveClipRects: ClipRectPtr ;
cliprects: ClipRectPtr ;
layerInfo: LayerInfoPtr ;
lock: SignalSemaphore ;
backFill: HookPtr ;
reserved1:LONGCARD;
clipRegion: RegionPtr ;
saveClipRects: RegionPtr ;
width,height:INTEGER;
reserved2:ARRAY [0..17] OF BYTE;
damageList: RegionPtr ;
END;
```

```
ClipRect =RECORD
next: ClipRectPtr ;
prev: ClipRectPtr ;
lobs: LayerPtr ;
bitMap: BitMapPtr ;
bounds: Rectangle ;
p1: ClipRectPtr ;
p2: ClipRectPtr ;
reserved:LONGINT;
flags:LONGINT;
END;
```

```
CONST
needsNoConcealedRasters=01H;
```

```
needsNoLayerblitDamage=02H;
```

```
isLessX=1;  
isLessY=2;  
isGrtrX=4;  
isGrtrY=8;
```

```
CONST
```

```
borderHit=0;  
topHit=1;  
bottomHit=2;  
leftHit=4;  
rightHit=8;
```

```
CONST
```

```
move=0;  
wait=1;  
next=2;  
sys=13;  
sht=14;  
lof=15;
```

```
TYPE
```

```
  CopIns =RECORD  
  CASE opCode:CARDINAL OF  
    | move:  
      destAddr:INTEGER;  
      destData:INTEGER;  
    | wait:  
      vWaitPos:INTEGER;  
      hWaitPos:INTEGER;  
    | next:  
      nxtlist: CopListPtr ;  
  END;  
END;
```

```
  Cprlist =RECORD  
  next: CprlistPtr ;  
  start:ADDRESS;  
  maxCount:INTEGER;  
END;
```

```
  CopList =RECORD  
  next: CopListPtr ;  
  copList: CopListPtr ;  
  viewPort: ViewPortPtr ;  
  copIns: CopInsPtr ;  
  copPtr: CopInsPtr ;  
  copLStart:ADDRESS;  
  copSStart:ADDRESS;  
  count:INTEGER;  
  maxCount:INTEGER;  
  dyOffset:INTEGER;  
  (*34*) cop2Start:ADDRESS;  
  (*34*) cop3Start:ADDRESS;  
  (*34*) cop4Start:ADDRESS;  
  (*34*) cop5Start:ADDRESS;
```

END;

```
UCopList =RECORD
next: UCopListPtr ;
firstCopList: CopListPtr ;
copList: CopListPtr ;
END;
```

```
Copinit =RECORD
vsynchBlank:ARRAY [0..1] OF CARDINAL;
diwstart:ARRAY [0..3] OF CARDINAL;
diagstrt:ARRAY [0..3] OF CARDINAL;
sprstrtup:ARRAY [0..(2*8*2)-1] OF CARDINAL;
wait14:ARRAY [0..1] OF CARDINAL;
normHBlank:ARRAY [0..1] OF CARDINAL;
genloc:ARRAY [0..3] OF CARDINAL;
jump:ARRAY [0..(2*2)-1] OF CARDINAL;
waitForever:ARRAY [0..1] OF CARDINAL;
sprstop:ARRAY [0..3] OF CARDINAL;
END;
```

CONST

```
m640=08000H;
plnCnTMsk=07H;
plnCnTShft=12;
pf2pri=40H;
colorOn=200H;
dblpf=400H;
holdnmodify=800H;
interlace=04H;
```

```
fineScroll=0FH;
fineScrollShift=04H;
fineScrollMask=0FH;
horizPos=07FH;
vrtclPos=01FFH;
vrtclPosShift=07H;
dftchMask=0FFH;
vposrlof=08000H;
```

TYPE

```
ExtendedNode =RECORD
succ: NodePtr ;
pred: NodePtr ;
type: NodeType ;
pri:SHORTINT;
name:ADDRESS;
subsystem:SHORTCARD;
subtype:SHORTCARD;
library:LONGINT;
init:PROCEDURE():LONGINT;
END;
```

CONST

```
ssGraphics=02H;

viewExtraType=1;
```

```
viewportExtraType=2;
specialMonitorType=3;
monitorSpecType=4;

TYPE
  LongProc =PROCEDURE():LONGINT;

  MonitorSpec =RECORD
    node: ExtendedNode ;
    flags:BITSET;
    ratioh:LONGINT;
    ratiov:LONGINT;
    totalRows:CARDINAL;
    totalColorclocks:CARDINAL;
    deniseMaxDisplayColumn:CARDINAL;
    beamCon0:CARDINAL;
    minRow:CARDINAL;
    special: SpecialMonitorPtr ;
    openCount:CARDINAL;
    transform: LongProc ;
    translate: LongProc ;
    scale: LongProc ;
    xoffset:CARDINAL;
    yoffset:CARDINAL;
    legalView: Rectangle ;
    maxoscan: LongProc ;
    videoscan: LongProc ;
    deniseMinDisplayColumn:CARDINAL;
    displayCompatible:LONGCARD;
    displayInfoDataBase: List ;
    displayInfoDataBaseSemaphore: SignalSemaphore ;
    reserved00:LONGCARD;
    reserved01:LONGCARD;
  END;

CONST
  toMonitor=0;
  fromMonitor=1;

  standardXoffset=9;
  standardYoffset=0;

  requestNtsc=1;
  requestPal=2;
  requestSpecial=4;
  requestA2024=8;

  defaultMonitorName="default.monitor";
  ntscMonitorName="ntsc.monitor";
  palMonitorName="pal.monitor";

  standardMonitorMask=requestNtsc+requestPal;

  standardNtscRows=262;
  standardPalRows=312;
  standardColorclocks=226;
  standardDeniseMax=455;
```

```
standardDeniseMin=93;
standardNtscBeamcon=0000H;
standardPalBeamcon=displayPal;

specialBeamcon= BeamConFlagSet {
    varVBlank, lolDis, varVSync, varBeam, cSBlank};

minNtscRow=21;
minPalRow=29;

standardViewX=81H;
standardViewY=2CH;
standardHbstrt=06H;
standardHsstrt=0BH;
standardHsstop=1CH;
standardHbstop=2CH;
standardVbstrt=0122H;
standardVsstrt=02A6H;
standardVsstop=03AAH;
standardVbstop=1066H;

vgaColorclocks=standardColorclocks DIV 2;
vgaTotalRows=standardNtscRows*2;
vgaDeniseMin=59;

minVgaRow=29;

vgaHbstrt=08H;
vgaHsstrt=0EH;
vgaHsstop=1CH;
vgaHbstop=1EH;
vgaVbstrt=0000H;
vgaVsstrt=0153H;
vgaVsstop=0235H;
vgaVbstop=0CCDH;

vgaMonitorName="vga.monitor";

vga70Colorclocks=standardColorclocks DIV 2;
vga70TotalRows=449;
vga70DeniseMin=59;
minVga70Row=35;
vga70Hbstrt=08H;
vga70Hsstrt=0EH;
vga70Hsstop=1CH;
vga70Hbstop=1EH;
vga70Vbstrt=0000H;
vga70Vsstrt=02A6H;
vga70Vsstop=0388H;
vga70Vbstop=0F73H;

vga70Beamcon=specialBeamcon/ BeamConFlagSet {vSyncTrue};

vga70MonitorName="vga70.monitor";

broadcastHbstrt=01H;
broadcastHsstrt=06H;
```

```

broadcastHsstop=17H;
broadcastHbstop=27H;
broadcastVbstrt=0000H;
broadcastVsstrt=02A6H;
broadcastVsstop=054CH;
broadcastVbstop=1C40H;
broadcastBeamcon= BeamConFlagSet {lolDis,cSBlank};

```

```

ratioFixedpart=4;
ratioUnity=SHIFT (LONGCARD (1),ratioFixedpart);

```

TYPE

```

  AnalogSignalInterval =RECORD
    strt:CARDINAL;
    stop:CARDINAL;
  END;

```

```

  SpecialMonitor =RECORD
    node: ExtendedNode ;
    flags:BITSET;
    doMonitor: LongProc ;
    reserved1: LongProc ;
    reserved2: LongProc ;
    reserved3: LongProc ;
    hblank: AnalogSignalInterval ;
    vblank: AnalogSignalInterval ;
    hsync: AnalogSignalInterval ;
    vsync: AnalogSignalInterval ;
  END;

```

TYPE

```

  DisplayInfoHandle =ADDRESS;

```

CONST

```

dtagDisp=800000000H;
dtagDims=80001000H;
dtagMntr=80002000H;
dtagName=80003000H;

```

TYPE

```

  QueryHeader =RECORD
    structID:LONGCARD;
    displayID:LONGCARD;
    skipID:LONGCARD;
    length:LONGCARD;
  END;

```

TYPE

```

  PropertyFlags =(
    isLace,isDualpf,isPf2pri,isHam,isEcs,isPal,isSprites,isGenlock,isWb,
    isDraggable,isPanelled,isBeamsync,isExtrahalfbrite,is13,is14,is15,is16
  );
  PropertyFlagSet =SET OF PropertyFlags ;

```

```

  AvailabilityFlags =(
    availNochips,availNomonitor,availNotwithgenlock,avf3,
    avf4,avf5,avf6,avf7,avf8
  );

```

```

);
AvailabilityFlagSet =SET OF  AvailabilityFlags ;

DisplayInfo =RECORD
header: QueryHeader ;
notAvailable: AvailabilityFlagSet ;
propertyFlags: PropertyFlagSet ;
resolution: Point ;
pixelSpeed:CARDINAL;
numStdSprites:CARDINAL;
paletteRange:CARDINAL;
spriteResolution: Point ;
pad:ARRAY [0..3] OF SHORTCARD;
reserved2:ARRAY [0..1] OF LONGCARD;
END;

CONST
diAvailNochip= AvailabilityFlagSet {availNochips};
diAvailNoMonitor= AvailabilityFlagSet {availNoMonitor};
diAvailNotwithgenlock= AvailabilityFlagSet {availNotwithgenlock};

TYPE
DimensionInfo =RECORD
header: QueryHeader ;
maxDepth:CARDINAL;
minRasterWidth:CARDINAL;
minRasterHeight:CARDINAL;
maxRasterWidth:CARDINAL;
maxRasterHeight:CARDINAL;
nominal: Rectangle ;
maxOScan: Rectangle ;
videoOScan: Rectangle ;
txtOScan: Rectangle ;
stdOScan: Rectangle ;
pad:ARRAY [0..13] OF SHORTCARD;
reserved:ARRAY [0..1] OF LONGCARD;
END;

MonitorInfo =RECORD
header: QueryHeader ;
mspc: MonitorSpecPtr ;
viewPosition: Point ;
viewResolution: Point ;
viewPositionRange: Rectangle ;
totalRows:CARDINAL;
totalColorClocks:CARDINAL;
minRow:CARDINAL;
compatibility:INTEGER;
pad:ARRAY [0..35] OF SHORTCARD;
DefaultViewPosition: Point ;
PreferredModeID:LONGCARD;
reserved:ARRAY [0..1] OF LONGCARD;
END;

CONST
mcompatMixed=0;
mcompatSelf=1;

```

```
mcompatNobody=-1;

displayNameLen=32;

TYPE
    NameInfo =RECORD
        header: QueryHeader ;
        name:ARRAY [0..displayNameLen-1] OF CHAR;
        reserved:ARRAY [0..1] OF LONGCARD;
    END;

CONST
    invalidID=-1;

    monitorIDmask=0FFFF1000H;

    defaultMonitorID=00000000H;
    ntscMonitorID=00011000H;
    palMonitorID=00021000H;

    loresKey=00000000H;
    hiresKey=00008000H;
    superKey=00008020H;
    hamKey=00000800H;
    loreslaceKey=00000004H;
    hireslaceKey=00008004H;
    superlaceKey=00008024H;
    hamlaceKey=00000804H;
    loresdpfKey=00000400H;
    hiresdpfKey=00008400H;
    superdpfKey=00008420H;
    loreslacedpfKey=00000404H;
    hireslacedpfKey=00008404H;
    superlacedpfKey=00008424H;
    loresdpf2Key=00000440H;
    hiresdpf2Key=00008440H;
    superdpf2Key=00008460H;
    loreslacedpf2Key=00000444H;
    hireslacedpf2Key=00008444H;
    superlacedpf2Key=00008464H;
    extrahalfbriteKey=00000080H;
    extrahalfbritelaceKey=00000084H;

    vgaMonitorID=00031000H;

    vgaextraloresKey=00031004H;
    vgaloresKey=00039004H;
    vgaproductKey=00039024H;
    vgahamKey=00031804H;
    vgaextraloreslaceKey=00031005H;
    vgaloreslaceKey=00039005H;
    vgaproductlaceKey=00039025H;
    vgahamlaceKey=00031805H;
    vgaextraloresdpfKey=00031404H;
    vgaloresdpfKey=00039404H;
    vgaproductdpfKey=00039424H;
    vgaextraloreslacedpfKey=00031405H;
```

```
vgaloreslacedpfKey=00039405H;
vgaproductlacedpfKey=00039425H;
vgaextraloresdpf2Key=00031444H;
vgaloresdpf2Key=00039444H;
vgaproductdpf2Key=00039464H;
vgaextraloreslacedpf2Key=00031445H;
vgaloreslacedpf2Key=00039445H;
vgaproductlacedpf2Key=00039465H;
vgaextrahalfbriteKey=00031084H;
vgaextrahalfbritelaceKey=00031085H;
```

```
a2024MonitorID=00041000H;
```

```
a2024tenhertzKey=00041000H;
a2024fifteenhertzKey=00049000H;
```

```
protoMonitorID=00051000H;
```

CONST

```
ringtrigger=01H;
anfracsize=06H;
animhalf=020H;
```

```
b2Norm=0;
b2Swap=1;
b2Bobber=2;
```

TYPE

```
VSpriteFlags =(
    vsprite, saveBack, overlay, mustDraw, vf4, vf5, vf6, vf7,
    backSaved, bobUpdate, gelGone, vsOverflow, vf12, vf13, vf14, vf15
);
VSpriteFlagSet =SET OF VSpriteFlags ;
VSprite =RECORD
    nextVSprite: VSpritePtr ;
    prevVSprite: VSpritePtr ;
    drawPath: VSpritePtr ;
    clearPath: VSpritePtr ;
    oldY:INTEGER;
    oldX:INTEGER;
    flags: VSpriteFlagSet ;
    y:INTEGER;
    x:INTEGER;
    height:INTEGER;
    width:INTEGER;
    depth:INTEGER;
    meMask:BITSET;
    hitMask:BITSET;
    imageData:ADDRESS;
    borderLine:ADDRESS;
    collMask:ADDRESS;
    sprColors:ADDRESS;
    vsBob: BobPtr ;
    planePick:SHORTCARD;
    planeOnOff:SHORTCARD;
END;
```



```
BobFlags =(
  saveBob,bobIsComp,bf2,bf3,bf4,bf5,bf6,bf7,
  bWaiting,bDrawn,bobsAway,bobNix,savePreserve,outStep,bf14,bf15
);
BobFlagSet =SET OF  BobFlags ;
```

```
Bob =RECORD
  flags: BobFlagSet ;
  saveBuffer:ADDRESS;
  imageShadow:ADDRESS;
  before: BobPtr ;
  after: BobPtr ;
  bobVSprite: VSpritePtr ;
  bobComp: AnimCompPtr ;
  dBuffer: DBufPacketPtr ;
END;
```

```
AnimComp =RECORD
  flags:INTEGER;
  timer:INTEGER;
  timeSet:INTEGER;
  nextComp: AnimCompPtr ;
  prevComp: AnimCompPtr ;
  nextSeq: AnimCompPtr ;
  prevSeq: AnimCompPtr ;
  animCRoutine:ADDRESS;
  yTrans:INTEGER;
  xTrans:INTEGER;
  headOb: AnimObPtr ;
  animBob: BobPtr ;
END;
```

```
AnimOb =RECORD
  nextOb: AnimObPtr ;
  prevOb: AnimObPtr ;
  clock:LONGINT;
  anOldY:INTEGER;
  anOldX:INTEGER;
  anY:INTEGER;
  anX:INTEGER;
  yVel:INTEGER;
  xVel:INTEGER;
  yAccel:INTEGER;
  xAccel:INTEGER;
  ringYTrans:INTEGER;
  ringXTrans:INTEGER;
  animORoutine:ADDRESS;
  headComp: AnimCompPtr ;
END;
```

```
DBufPacket =RECORD
  bufY:INTEGER;
  bufX:INTEGER;
  bufPath: VSpritePtr ;
  bufBuffer:ADDRESS;
END;
```

```

CollTable =RECORD
collPtrs:ARRAY [0..15] OF ADDRESS
END;

DisplayFlags =(
    ntsc,genloc,pal,todaSafe,df4,df5,df6,df7,
    df8,df9,df10,df11,df12,df13,df14,df15
);
DisplayFlagSet =SET OF DisplayFlags ;

(*36*) ChipRevs =(hrAgnus,hrDenise,cr2,cr3,cr4,cr5,cr6,cr7);
(*36*) ChipRevSet =SET OF ChipRevs ;

CONST
    bigBlits=hrAgnus;

TYPE
    GfxBase =RECORD
        libNode: Library ;
        actiView: ViewPtr ;
        copinit: CopinitPtr ;
        cia:ADDRESS;
        blitter:ADDRESS;
        loFlist:ADDRESS;
        shFlist:ADDRESS;
        blthd: BltnodePtr ;
        blttl: BltnodePtr ;
        bsblthd: BltnodePtr ;
        bsblttl: BltnodePtr ;
        vbsrv: Interrupt ;
        timsrv: Interrupt ;
        bltsrv: Interrupt ;
        textFonts: List ;
        defaultFont: TextFontPtr ;
        modes:BITSET;
        vBlank:SHORTCARD;
        debug:BYTE;
        beamSync:INTEGER;
        bplcon0:BITSET;
        spriteReserved:SHORTCARD;
        bytereserved:SHORTCARD;
        flags:BITSET;
        blitLock:INTEGER;
        blitNest:INTEGER;
        blitWaitQ: List ;
        blitOwner: TaskPtr ;
        waitQ: List ;
        displayFlags: DisplayFlagSet ;
        simpleSprites:ADDRESS;
        maxDisplayRow:CARDINAL;
        maxDisplayColumn:CARDINAL;
        normalDisplayRows:CARDINAL;
        normalDisplayColumns:CARDINAL;
        normalDPMX:CARDINAL;
        normalDPMY:CARDINAL;
        lastChanceMemory: SignalSemaphorePtr ;
        lcMptr:ADDRESS;

```

```

microsPerLine: CARDINAL;
minDisplayColumn: CARDINAL;
(*36*) chipRevBits0: ChipRevSet ;
(*36*) reservedPad: SHORTCARD;
(*36*) reserved: ARRAY [0..3] OF SHORTCARD;
(*36*) monitorId: CARDINAL;
(*36*) hedley: ARRAY [0..7] OF ADDRESS;
(*36*) hedleySprites: ARRAY [0..7] OF ADDRESS;
(*36*) hedleySprites1: ARRAY [0..7] OF ADDRESS;
(*36*) hedleyCount: INTEGER;
(*36*) hedleyFlags: BITSET;
(*36*) hedleyTmp: INTEGER;
(*36*) hashTable: ADDRESS;
(*36*) currentTotRows: CARDINAL;
(*36*) currentTotCclks: CARDINAL;
(*36*) hedleyHint: SHORTCARD;
(*36*) hedleyHint2: SHORTCARD;
(*36*) nreserved: ARRAY [0..3] OF LONGCARD;
(*36*) a2024SyncRaster: ADDRESS;
(*36*) controlDeltaPal: INTEGER;
(*36*) controlDeltaNtsc: INTEGER;
(*36*) currentMonitor: MonitorSpecPtr ;
(*36*) monitorList: List ;
(*36*) defaultMonitor: MonitorSpecPtr ;
(*36*) monitorListSemaphore: SignalSemaphorePtr ;
(*36*) displayInfoDataBase: ADDRESS;
(*36*) lapad: INTEGER;
(*36*) actiViewCprSemaphore: SignalSemaphorePtr ;
(*36*) utilityBase: ADDRESS;
(*36*) execBase: ADDRESS;
END;

CONST
  blitMsgFault=4;

TYPE
  Isrvstr = RECORD
    node: Node ;
    iptr: IsrvstrPtr ;
    code: ADDRESS;
    ccode: ADDRESS;
    carg: LONGINT;
  END;

  LayerFlags = (
    layerSimple, layerSmart, layerSuper, lf3, layerUpdating, lf5, layerBackdrop,
    layerRefresh, layerClipRectsLost
  );
  LayerFlagSet = SET OF LayerFlags ;

  LayerInfo = RECORD
    layer: LayerPtr ;
    lp: LayerPtr ;
    obs: ClipRectPtr ;
    freeClipRects: MinList ;
    lock: SignalSemaphore ;
    head: List ;

```

```

    longreserved:LONGINT;
    flags: LayerFlagSet ;
    count:SHORTINT;
    lockLayersCount:SHORTINT;
    layerInfoExtraSize:CARDINAL;
    blitbuff:ADDRESS;
    layerInfoExtra:ADDRESS;
END;

CONST
    lmnRegion=-1;
    newLayerInfoCalled=01H;
    alertLayersNoMem=083010000H;

TYPE
    AreaInfo =RECORD
        vctrTbl:ADDRESS;
        vctrPtr:ADDRESS;
        flagTbl:ADDRESS;
        flagPtr:ADDRESS;
        count:INTEGER;
        maxCount:INTEGER;
        firstX:INTEGER;
        firstY:INTEGER;
    END;

    TmpRas =RECORD
        rasPtr:ADDRESS;
        size:LONGINT;
    END;

    GelsInfo =RECORD
        sprRsrvd:SHORTINT;
        flags:SHORTCARD;
        gelHead: VSpritePtr ;
        gelTail: VSpritePtr ;
        nextLine:ADDRESS;
        lastColor:ADDRESS;
        collHandler: CollTablePtr ;
        leftmost:INTEGER;
        rightmost:INTEGER;
        topmost:INTEGER;
        bottommost:INTEGER;
        firstBlissObj:ADDRESS;
        lastBlissObj:ADDRESS;
    END;

    DrawModes =(dm0,complement,inversvid);
    DrawModeSet =SET OF DrawModes ;

    FontStyles =(
        underlined,bold,italic,extended,fs4,fs5,colorFont,tagged
    );
    FontStyleSet =SET OF FontStyles ;

    FontFlags =(
        romFont,diskFont,revPath,tallDot,wideDot,proportional,designed,removed

```

```

);
FontFlagSet =SET OF FontFlags ;

RastPortFlags =(
firstDot,oneDot,dBuffer,areaOutline,rpf4,noCrossFill,rpf6,rpf7,rpf8
);
RastPortFlagSet =SET OF RastPortFlags ;

RastPort =RECORD
layer: LayerPtr ;
bitMap: BitMapPtr ;
areaPtrn:ADDRESS;
tmpRas: TmpRasPtr ;
areaInfo: AreaInfoPtr ;
gelsInfo: GelsInfoPtr ;
mask:SHORTCARD;
fgPen:SHORTCARD;
bgPen:SHORTCARD;
aOlPen:SHORTCARD;
drawMode: DrawModeSet ;
areaPtSz:SHORTCARD;
linPatCnt:SHORTCARD;
dummy:BYTE;
flags: RastPortFlagSet ;
linePtrn:CARDINAL;
x:INTEGER;
y:INTEGER;
minterms:ARRAY [0..7] OF SHORTCARD;
penWidth:INTEGER;
penHeight:INTEGER;
font: TextFontPtr ;
algoStyle: FontStyleSet ;
txFlags: FontFlagSet ;
txHeight:CARDINAL;
txWidth:CARDINAL;
txBaseline:CARDINAL;
txSpacing:INTEGER;
user:ADDRESS;
longreserved:ARRAY [0..1] OF LONGINT;
wordreserved:ARRAY [0..6] OF WORD;
reserved:ARRAY [0..7] OF BYTE;
END;

CONST
jam1= DrawModeSet {};
jam2= DrawModeSet {dm0};

spriteAttached=080H;
normalFont= FontStyleSet {};

TYPE
RegionRectangle =RECORD
next: RegionRectanglePtr ;
prev: RegionRectanglePtr ;
bounds: Rectangle ;
END;

```

```
Region =RECORD
bounds: Rectangle ;
regionRectangle: RegionRectanglePtr ;
END;
```

TYPE

```
BitScaleArgs =RECORD
srcX: CARDINAL;
srcY: CARDINAL;
srcWidth: CARDINAL;
srcHeight: CARDINAL;
srcXFactor: CARDINAL;
srcYFactor: CARDINAL;
destX: CARDINAL;
destY: CARDINAL;
destWidth: CARDINAL;
destHeight: CARDINAL;
xDestFactor: CARDINAL;
yDestFactor: CARDINAL;
srcBitMap: BitMapPtr ;
destBitMap: BitMapPtr ;
flags: LONGCARD;
xDDA: CARDINAL;
yDDA: CARDINAL;
reserved1: LONGINT;
reserved2: LONGINT;
END;
```

```
SimpleSprite =RECORD
posctlldata: ADDRESS;
height: CARDINAL;
x: CARDINAL;
y: CARDINAL;
num: INTEGER;
END;
```

```
TextAttr =RECORD
name: ADDRESS;
ySize: CARDINAL;
style: FontStyleSet ;
flags: FontFlagSet ;
END;
```

```
TTextAttr =RECORD
name: ADDRESS;
ySize: CARDINAL;
style: FontStyleSet ;
flags: FontFlagSet ;
tags: TagItemPtr ;
END;
```

CONST

```
taDeviceDPI=tagUser+1;
maxFontMatchWeight=32767;
```

TYPE

```
TextFont =RECORD
```

```

message: Message ;
ySize: CARDINAL;
style: FontStyleSet ;
flags: FontFlagSet ;
xSize: CARDINAL;
baseline: CARDINAL;
boldSmear: CARDINAL;
accessors: CARDINAL;
loChar: CHAR;
hiChar: CHAR;
charData: ADDRESS;
modulo: CARDINAL;
charLoc: ADDRESS;
charSpace: ADDRESS;
charKern: ADDRESS;
END;

```

TYPE

```

TextFontExt0Flags = (noRemFont) ;
TextFontExt0FlagSet = SET OF TextFontExt0Flags ;

TextFontExtension = RECORD
matchWord: CARDINAL;
flags0: TextFontExt0FlagSet ;
flags1: SHORTSET;
backPtr: TextFontPtr ;
origReplyPort: MsgPortPtr ;
tags: TagItemPtr ;
oFontPatchS: ADDRESS;
oFontPatchK: ADDRESS;
END;

```

TYPE

```

ColorFontColors = RECORD
reserved: CARDINAL;
count: CARDINAL;
colorTable: POINTER TO ARRAY [0..1000] OF CARDINAL;
END;

ColorTextFontFlags = (
colorfont, greyfont, antialias, ctf3, ctf4, ctf5, ctf6, ctf7,
ctf8, ctf9, ctf10, ctf11, ctf12, ctf13, ctf14, ctf15
);
ColorTextFontFlagSet = SET OF ColorTextFontFlags ;

```

CONST

```

mapColor = colorFont;
colorMask = ColorTextFontFlagSet {colorfont..ctf3};

```

TYPE

```

ColorTextFont = RECORD
tf: TextFont ;
flags: CARDINAL;
depth: SHORTCARD;
fgColor: SHORTCARD;
low: SHORTCARD;
high: SHORTCARD;

```

```
planePick:SHORTCARD;  
planeOnOff:SHORTCARD;  
colorFontColors: ColorFontColorsPtr ;  
charData:ARRAY[0..7] OF ADDRESS;  
END;
```

```
TextExtent =RECORD  
width:CARDINAL;  
height:CARDINAL;  
extent: Rectangle ;  
END;
```

```
CONST  
vtagEndCm=00000000H;
```

```
TYPE  
VTags =(  
vtagChromaKeyClr:=tagUser,  
vtagChromaKeySet,  
vtagBitplaneKeyClr,  
vtagBitplaneKeySet,  
vtagBorderblankClr,  
vtagBorderblankSet,  
vtagBordernotransClr,  
vtagBordernotransSet,  
vtagChromaPenClr,  
vtagChromaPenSet,  
vtagChromaPlaneSet,  
vtagAttachCmSet,  
vtagNextbufCm,  
vtagBatchCmClr,  
vtagBatchCmSet,  
vtagNormalDispGet,  
vtagNormalDispSet,  
vtagCoerceDispGet,  
vtagCoerceDispSet,  
vtagViewportextraGet,  
vtagViewportextraSet,  
vtagChromaKeyGet,  
vtagBitplaneKeyGet,  
vtagBorderblankGet,  
vtagBordernotransGet,  
vtagChromaPenGet,  
vtagChromaPlaneGet,  
vtagAttachCmGet,  
vtagBatchCmGet,  
vtagBatchItemsGet,  
vtagBatchItemsSet,  
vtagBatchItemsAdd,  
vtagVpmodeidGet,  
vtagVpmodeidSet,  
vtagVpmodeidClr,  
vtagUserclipGet,  
vtagUserclipSet,  
vtagUserclipClr  
);
```

```
ViewModes =(
vm0,genlocVideo,lace,vm3,vm4,superHires,pfba,extraHalfbrite,
genlocAudio,vm9,dualpf,ham,extendedMode,vpHide,sprites,hires
);
ViewModeSet =SET OF ViewModes ;
```

```
ViewPort =RECORD
next: ViewPortPtr ;
colorMap: ColorMapPtr ;
dspIns: CopListPtr ;
sprIns: CopListPtr ;
clrIns: CopListPtr ;
uCopIns: UCopListPtr ;
dWidth:INTEGER;
dHeight:INTEGER;
dxOffset:INTEGER;
dyOffset:INTEGER;
modes: ViewModeSet ;
spritePriorities:SHORTCARD;
extendedModes:SHORTSET;
rasInfo: RasInfoPtr ;
END;
```

```
View =RECORD
viewPort: ViewPortPtr ;
lofCprList: CprlistPtr ;
shfCprList: CprlistPtr ;
dyOffset:INTEGER;
dxOffset:INTEGER;
modes: ViewModeSet ;
END;
```

```
ViewExtra =RECORD
n: ExtendedNode ;
view: ViewPtr ;
monitor: MonitorSpecPtr ;
END;
```

```
ViewPortExtra =RECORD
n: ExtendedNode ;
viewPort: ViewPortPtr ;
displayClip: Rectangle ;
END;
```

```
RasInfo =RECORD
next: RasInfoPtr ;
bitMap: BitMapPtr ;
rxOffset:INTEGER;
ryOffset:INTEGER;
END;
```

TYPE

```
ColorMapType =(colorMapTypeV33,colorMapTypeV36);
```

```
ColorMapFlags =(
colormapTransparency,colorPlaneTransparency,borderBlanking,
```

```
borderNoTransparency,videoControlBatch,userCopperClip,cm6,cm7
);
ColorMapFlagSet =SET OF   ColorMapFlags ;

ColorMap =RECORD
flags: ColorMapFlagSet ;
type: ColorMapType ;
count:CARDINAL;
colorTable:ADDRESS;
vpe: ViewPortExtraPtr ;
transparencyBits:ADDRESS;
transparencyPlane:SHORTCARD;
reserved1:SHORTCARD;
reserved2:CARDINAL;
vp: ViewPortPtr ;
normalDisplayInfo:ADDRESS;
coerceDisplayInfo:ADDRESS;
batchItems: TagItemPtr ;
vpModeID:LONGCARD;
END;

END GraphicsD.noimp
```
