

File created: 26-Oct-2021 10:52:24 {DSK}<home>larry>medley>sources>MAIKOBITBLT.;2

changes to: (FNS \\MAIKO.BITBLTSUB \\MAIKO.BLTCHAR \\MAIKO.PUNTBTLCHAR \\MAIKO.BITBLT.BITMAP
\\MAIKO.BLTSHADE.BITMAP)

previous date: 24-Oct-2021 10:31:31 {DSK}<home>larry>medley>sources>MAIKOBITBLT.;1

Read Table: XCL

Package: INTERLISP

Format: XCCS

; Copyright (c) 1988-1990, 1994 by Venue & Xerox Corporation.

(RPAQQ **MAIKOBITBLTCOMS**

(
;; this file has some optimizations for BITBLT on MAIKO; while PILOTBITBLT opcode still works, these functions directly implement some
;; higher level operations

(FNS \\MAIKO.BITBLTSUB \\MAIKO.BLTCHAR \\MAIKO.PUNTBTLCHAR \\MAIKO.BITBLT.BITMAP \\MAIKO.BLTSHADE.BITMAP
)

;; Save the old \BITBLT.BITMAP, because it handles the OPERATION - MERGE case, where the C code doesn't.

(P (MOVD ' \\BITBLT.BITMAP ' \\MAIKO.OLDBITBLT.BITMAP))
(ADDVARS (\\MAIKO.MOVDS (\\MAIKO.BLTCHAR \\MEDW.BLTCHAR)
(\\MAIKO.BITBLTSUB \\BITBLTSUB)
(\\MAIKO.BITBLT.BITMAP \\BITBLT.BITMAP)
(\\MAIKO.BLTSHADE.BITMAP \\BLTSHADE.BITMAP))))

;; this file has some optimizations for BITBLT on MAIKO; while PILOTBITBLT opcode still works, these functions directly implement some higher level
;; operations

(DEFINEQ

(\\MAIKO.BITBLTSUB

(LAMBDA (PILOTBBT |SourceBitMap| SLX STY |DestinationBitMap| DLX DTY HEIGHT |SourceType| |Operation| |Texture|
|WindowXOffset| |WindowYOffset|)
; Edited 26-Oct-2021 10:06 by larry
; Edited 29-Jun-88 16:24 by

;; replaces \BITBLTSUB on Maiko

(SUBRCALL BITBLTSUB PILOTBBT |SourceBitMap| SLX STY |DestinationBitMap| DLX DTY HEIGHT |SourceType|
|Operation| |Texture| |WindowXOffset| |WindowYOffset|))

(\\MAIKO.BLTCHAR

(LAMBDA (CHARCODE DISPLAYSTREAM DISPLAYDATA)
; Edited 26-Oct-2021 10:22 by larry
; Edited 6-Jul-90 10:14 by matsuda
(SUBRCALL NEW-BLTCHAR CHARCODE DISPLAYSTREAM DISPLAYDATA)))

(\\MAIKO.PUNTBTLCHAR

(LAMBDA (CHARCODE DISPLAYSTREAM DISPLAYDATA)
; Edited 26-Oct-2021 10:21 by larry
; Edited 1-Nov-89 15:26 by takeshi

;; puts a character on a display stream. This function will be called when \maiko.bluchar failed. Punt from subr call

(**DECLARE** (LOCALVARS . T))
(PROG (LOCAL1 RIGHT LEFT CURX CHAR8CODE)
(SETQ CHAR8CODE (\\CHAR8CODE CHARCODE))
CRLP
(COND
((NOT (EQ (|ffetch| (\\DISPLAYDATA DDCHARSET) |of| DISPLAYDATA)
(\\CHARSET CHARCODE)))
(\\CHANGECHARSET.DISPLAY DISPLAYDATA (\\CHARSET CHARCODE))))
(COND
((|ffetch| (\\DISPLAYDATA |DDSlowPrintingCase|) |of| DISPLAYDATA)
(RETURN (COND
((|type?| STREAM DISPLAYSTREAM)
(\\SLOWBLTCHAR CHARCODE DISPLAYSTREAM))
((|type?| WINDOW DISPLAYSTREAM)
(\\SLOWBLTCHAR CHARCODE (FETCH DSP OF DISPLAYSTREAM)))
(T (ERROR "Not Stream or Window" DISPLAYSTREAM))))))
(SETQ CURX (|ffetch| (\\DISPLAYDATA DDXPOSITION) |of| DISPLAYDATA))
(SETQ RIGHT (IPLUS CURX (\\DSPGETCHARIMAGEWIDTH CHAR8CODE DISPLAYDATA)))
(COND
((IGREATERP RIGHT (|ffetch| (\\DISPLAYDATA |DDRRightMargin|) |of| DISPLAYDATA))
; would go past right margin, force a cr
(COND
((IGREATERP CURX (|ffetch| (\\DISPLAYDATA |DDLeftMargin|) |of| DISPLAYDATA))
; don't bother CR if position is at left margin anyway. This also
; serves to break the loop.
(\\DSPPRINTCR/LF (CHARCODE EOL)
DISPLAYSTREAM)
; reuse the code in the test of this conditional rather than repeat it
; here.
(GO CRLP))))))
(|freplace| (\\DISPLAYDATA DDXPOSITION) |of| DISPLAYDATA |with| (IPLUS CURX (\\DSPGETCHARWIDTH CHAR8CODE
DISPLAYDATA)))
; transforms an x coordinate into the destination coordinate.

```

    (SETQ LOCAL1 (|ffetch| (\\DISPLAYDATA DDXOFFSET) |of| DISPLAYDATA))
    (SETQ CURX (IPLUS CURX LOCAL1))
    (SETQ RIGHT (IPLUS RIGHT LOCAL1))
    (COND
      ((IGREATERP RIGHT (SETQ LOCAL1 (|ffetch| (\\DISPLAYDATA |DDClippingRight|) |of| DISPLAYDATA)))
        ; character overlaps right edge of clipping region.
        (SETQ RIGHT LOCAL1)))
    (SETQ LEFT (COND
      ((IGREATERP CURX (SETQ LOCAL1 (|ffetch| (\\DISPLAYDATA |DDClippingLeft|) |of| DISPLAYDATA)))
        CURX)
      (T LOCAL1)))
    (RETURN (COND
      ((AND (ILESSP LEFT RIGHT)
        (NOT (EQ (|ffetch| (PILOTBBT PBTHEIGHT) |of| (SETQ LOCAL1 (|ffetch| (\\DISPLAYDATA
          DDPILOTBBT)
          |of| DISPLAYDATA))))
          0)))
      (.WHILE.TOP.DS. DISPLAYSTREAM (SUBCALL BLTCHAR LOCAL1 DISPLAYDATA CHAR8CODE CURX LEFT
        RIGHT))
      T))))))

```

(\\MAIKO.BITBLT.BITMAP

```

(LAMBDA (SOURCEBITMAP SOURCELEFT SOURCEBOTTOM DESTBITMAP DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT
  SOURCETYPE OPERATION TEXTURE CLIPPINGREGION CLIPPEDSOURCELEFT CLIPPEDSOURCEBOTTOM)
  ; Edited 14-Jun-90 16:47 by TS

```

;; SUN version of \\BITBLT.BITMAP. For all but the MERGE case, use C code. For the MERGE case, use the old code.

;; Jun-14 Now, C function, bitblt_bitmap, has PUNT case for MREGE case(Takeshi)

;; \\MAIKO.OLDBITBLT.BITMAP is obsolete, \\PUNT.BITBLT.BITMAP is there.

;;

```

;; (COND ((EQ SOURCETYPE 'MERGE) (\\MAIKO.OLDBITBLT.BITMAP SOURCEBITMAP SOURCELEFT SOURCEBOTTOM DESTBITMAP
;; DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT SOURCETYPE OPERATION TEXTURE CLIPPINGREGION
;; CLIPPEDSOURCELEFT CLIPPEDSOURCEBOTTOM)) (T (SUBCALL BITBLT.BITMAP SOURCEBITMAP SOURCELEFT SOURCEBOTTOM
;; DESTBITMAP DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT SOURCETYPE OPERATION TEXTURE CLIPPINGREGION
;; CLIPPEDSOURCELEFT CLIPPEDSOURCEBOTTOM)))

```

```

(SUBCALL BITBLT.BITMAP SOURCEBITMAP SOURCELEFT SOURCEBOTTOM DESTBITMAP DESTINATIONLEFT DESTINATIONBOTTOM
  WIDTH HEIGHT SOURCETYPE OPERATION TEXTURE CLIPPINGREGION CLIPPEDSOURCELEFT CLIPPEDSOURCEBOTTOM)))

```

(\\MAIKO.BLTSHADE.BITMAP

```

(LAMBDA (TEXTURE DESTINATIONBITMAP DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT OPERATION CLIPPINGREGION)
  ; Edited 14-Jun-90 16:49 by TS

```

```

(DECLARE (LOCALVARS . T))

```

;; C function, bitshade_bitmap, has PUNT case \\PUNT.BLTSHADE.BITMAP(Takeshi)

```

(SUBCALL BLTSHADE.BITMAP TEXTURE DESTINATIONBITMAP DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT OPERATION
  CLIPPINGREGION))

```

)

;; Save the old \\BITBLT.BITMAP, because it handles the OPERATION - MERGE case, where the C code doesn't.

```

(MOVD ' \\BITBLT.BITMAP ' \\MAIKO.OLDBITBLT.BITMAP)

```

```

(ADDTOPVAR \\MAIKO.MOVDS (\\MAIKO.BLTCHAR \\MEDW.BLTCHAR)
  (\\MAIKO.BITBLTSUB \\BITBLTSUB)
  (\\MAIKO.BITBLT.BITMAP \\BITBLT.BITMAP)
  (\\MAIKO.BLTSHADE.BITMAP \\BLTSHADE.BITMAP))

```

```

(PUTPROPS MAIKOBITBLT COPYRIGHT ("Venue & Xerox Corporation" 1988 1989 1990 1994))

```

FUNCTION INDEX

| | | | | | |
|-----------------------------|---|-------------------------------|---|---------------------------|---|
| \\MAIKO.BITBLT.BITMAP | 2 | \\MAIKO.BLTCHAR | 1 | \\MAIKO.PUNTBLTCHAR | 1 |
| \\MAIKO.BITBLTSUB | 1 | \\MAIKO.BLTSHADE.BITMAP | 2 | | |

VARIABLE INDEX

| | |
|---------------------|---|
| \\MAIKO.MOVDS | 2 |
|---------------------|---|
