Fujitsu Microelectronics Europe
Application Note

an-mb88f332-spe-performance-rev0-10

GRAPHICS CONTROLLERS

MB88F332 'INDIGO'

SPRITE ENGINE PERFORMANCE

APPLICATION NOTE

GRAPHICS COMPETENCE CENTER

FUJITSU

Revision History

Date	Issue
2009-03-10	Rev 0.01 GCC/HA First draft
2009-07-31	Rev 0.10 GCC/HA New layout, minor corrections, improved description

This document contains X pages.

Warranty and Disclaimer

To the maximum extent permitted by applicable law, Fujitsu Microelectronics Europe GmbH restricts its warranties and its liability for **all products delivered free of charge** (eg. software include or header files, application examples, target boards, evaluation boards, engineering samples of IC's etc.), its performance and any consequential damages, on the use of the Product in accordance with (i) the terms of the License Agreement and the Sale and Purchase Agreement under which agreements the Product has been delivered, (ii) the technical descriptions and (iii) all accompanying written materials. In addition, to the maximum extent permitted by applicable law, Fujitsu Microelectronics Europe GmbH disclaims all warranties and liabilities for the performance of the Product and any consequential damages in cases of unauthorised decompiling and/or reverse engineering and/or disassembling. **Note, all these products are intended and must only be used in an evaluation laboratory environment**.

1. Fujitsu Microelectronics Europe GmbH warrants that the Product will perform substantially in accordance with the accompanying written materials for a period of 90 days form the date of receipt by the customer. Concerning the hardware components of the Product, Fujitsu Microelectronics Europe GmbH warrants that the Product will be free from defects in material and workmanship under use and service as specified in the accompanying written materials for a duration of 1 year from the date of receipt by the customer.

2. Should a Product turn out to be defect, Fujitsu Microelectronics Europe GmbH´s entire liability and the customer´s exclusive remedy shall be, at Fujitsu Microelectronics Europe GmbH´s sole discretion, either return of the purchase price and the license fee, or replacement of the Product or parts thereof, if the Product is returned to Fujitsu Microelectronics Europe GmbH in original packing and without further defects resulting from the customer´s use or the transport. However, this warranty is excluded if the defect has resulted from an accident not attributable to Fujitsu Microelectronics Europe GmbH, or abuse or misapplication attributable to the customer or any other third party not relating to Fujitsu Microelectronics Europe GmbH.

3. To the maximum extent permitted by applicable law Fujitsu Microelectronics Europe GmbH disclaims all other warranties, whether expressed or implied, in particular, but not limited to, warranties of merchantability and fitness for a particular purpose for which the Product is not designated.

4. To the maximum extent permitted by applicable law, Fujitsu Microelectronics Europe GmbH´s and its suppliers´ liability is restricted to intention and gross negligence.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES

To the maximum extent permitted by applicable law, in no event shall Fujitsu Microelectronics Europe GmbH and its suppliers be liable for any damages whatsoever (including but without limitation, consequential and/or indirect damages for personal injury, assets of substantial value, loss of profits, interruption of business operation, loss of information, or any other monetary or pecuniary loss) arising from the use of the Product.

Should one of the above stipulations be or become invalid and/or unenforceable, the remaining stipulations shall stay in full effect

Contents

1 Introduction

This application note describes the performance of MB88F332's integrated sprite engine in more detail than the hardware manual. It should enable the reader to better understand the dependencies within the sprite engine.

It also documents the key factors that have an impact on overall performance. Additionally some calculation examples are given. External flash for storing sprite data is not covered by this document because a performance estimation depends too much on the specific type of flash being used.

2 Sprite Engine Processing Performance

The Sprite processing Engine (SPE) uses a Linebuffer method to process sprites. Therefore the processing ability of the SPE is restricted by the time needed to display a line period. SPE processing performance is defined as the maximum number of pixels or sprites that can be displayed on a line. The SPE processing performance is assessed in units of "**pixels/line**" or "**sprites/line**".

2.1 Performance Effect Factors

Ideally, the SPE supports up to 512 sprites on a line, but there are several factors which influence the actual processing performance (sprite numbers). The following items have an impact on the SPE's performance:

- **System Clock Frequency: (F_{clk} MHz)**

  The system clock frequency can be configured in **CRG** (Clock and Reset Generator). The maximum value is 83.33 **MHz**

- **Line Frequency: F_{line} KHz**

  This depends on the **VDC** (Video Display Controller) settings according to the panel.

  For example, 33.3 **KHz** (resolution=800x480/1200x480) or 16.9 **KHz** (resolution=640x160).

- **Display Width: $W_{display}$ Pixels**

  This is derived from the setting of the register ***DPWD*** in the Display Area Register ***SPEDPAR***

- **Sprite Color Format: Y_{spr} bit/pixel**

  It can be set to 1,2,4,8,16 or 32.

- **Sprite Width: W_{spr} Pixels**

- **Pattern Data Source Memory (SRAM or Internal Flash)**

  Please note that SRAM is much faster than the internal flash memory. The access speed of the internal flash can be configured via the register ***MTIMING1*** **(INTFLASH).**

- **Alpha Table on/off**

  The alpha table only exists with pattern data in memory (SRAM/IntFlash) when both ***SPESnCR2.ATS*** and ***SPESnCR2.ATE*** of the sprite are set to "1". When data from the alpha table is read into the SPE, it affects the SPE performance. When the sprite color format is one of 1bpp/2bpp/4bpp/8pp/RGB565, it may decrease the performance. When the color format is 32bpp/ARGB1555 or the alpha value in the LUT is used (index color format only), the alpha table on/off does not decrease the performance of the SPE.

Furthermore additional load on the AHB will decrease SPE performance when data transfers to the SPE are in progress parallel to other AHB transfers.

3 Performance Estimation

Please refer to figure 3-1 as a scheme to identify the performance relevant part in your application. The calculation rules given in chapter 4 will allow to determine the Min(U,V). Read thoroughly the formulae and calculate the values for your application. Chapter 5 will also provide some example data.

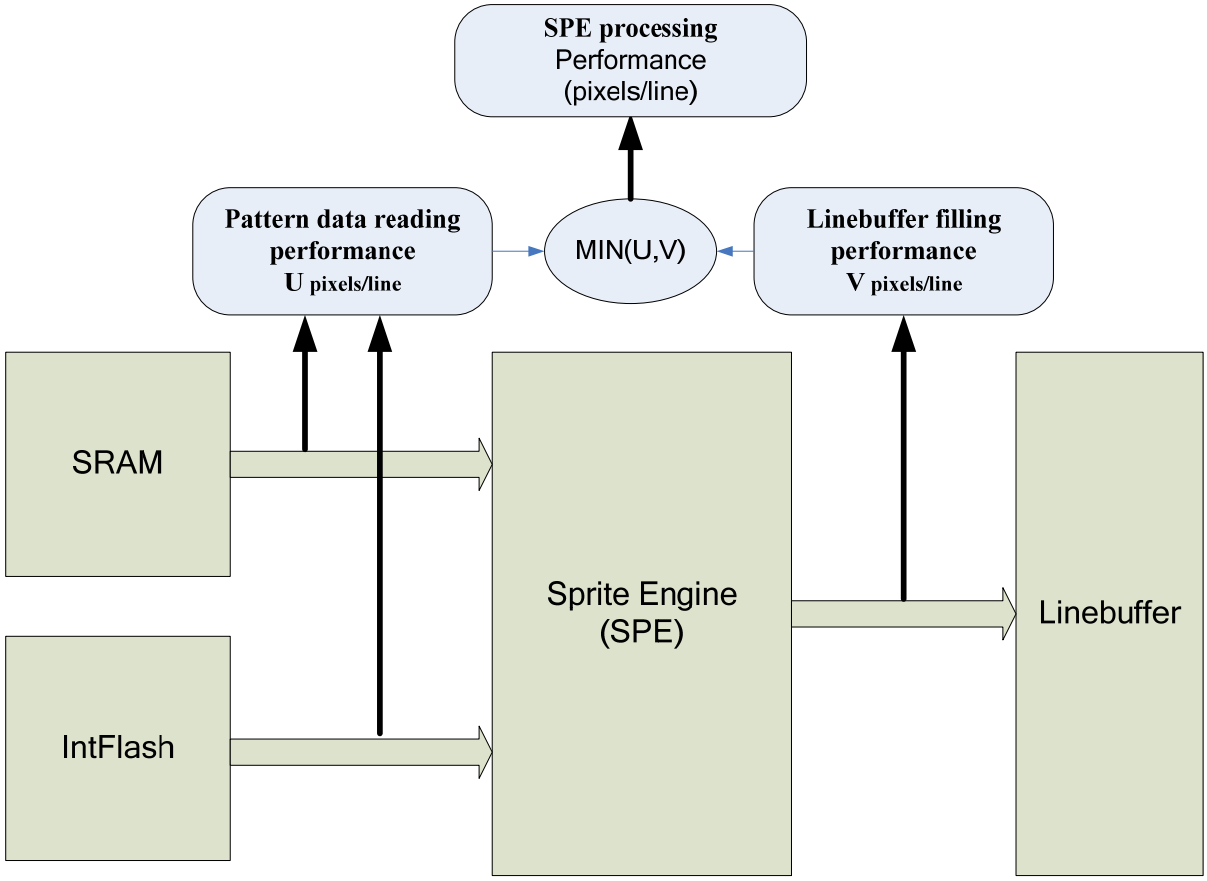

Figure 3-1 SPE processing performance estimation

4 Calculation Rules

Please use the formulae below to estimate the SPE's performance:

[1] Line Period : M cycles (system clock)

$$M = (1 / F_{line}) \times F_{clk} \times 1000 \text{ cycles}$$

[2] Maximum SPE processing time on a line: N cycles (system clock)

$$N = \left(M - \frac{Wdtsplay}{2}\right) \text{ cycles}$$

[3] SPE ideal Linebuffer filling performance on a line: U pixels

$$U = 2 \times N \text{ pixels}$$

[4] Total words of a sprite: T_{spr} words (1word = 32bit)

$$T_{spr} = (W_{spr} \times Y_{spr}) / 32 \text{ words (Alpha off)}$$
$$T_{spr} = (W_{spr} \times (Y_{spr}+4)) / 32 \text{ words (Alpha:4bpp)}$$
$$T_{spr} = (W_{spr} \times (Y_{spr}+ 8)) / 32 \text{ words (Alpha:8bpp)}$$

[5] SRAM reading performance: RC cycles/ word (1word = 32bit)

(1) $T_{spr} \%16 = 0$

$RC = ((T_{spr} /16) \times 20) / T_{spr}$ cycles/word

(2) $T_{spr} \%16 \neq 0$

$RC = ((T_{spr} /16) \times 20 + T_{spr} \%16 + 4) / T_{spr}$ cycles/word

[6] Internal flash reading performance: RC cycles/ word (1word = 32bit)

$RC = (MTIMING1.RACC + MTIMING1.RDSC + 2)$ cycles/word

By default, for RACC=4 and RDSC=1, then

RC = 4+1+2= 7 cycles/word

[7] Pattern data reading performance on a line: V pixels

V = ((N / RC) x 32) / Y_{spr} pixels (Alpha off)

V = ((N / RC) x 32) / (Y_{spr}+4) pixels (Alpha:4bpp)

V = ((N / RC) x 32) / (Y_{spr}+8) pixels (Alpha:8bpp)

Compare the pattern reading performance with the linebuffer filling performance and get estimation of the SPE's processing performance.

MIN (U, V) pixels/line

MIN (U, V) / W_{spr} sprites/line

When using Single color sprite, since the pattern data is not needed to be read from memory, performance of SPE should be U pixels / line or (U / W_{spr}) sprites/line.

Notice : Above formulas are just for getting an approximate processing performance result. It cannot guarantee the actual performance of all applications.

5 Calculation Example

Attributes	Value	Comments
System Clock	83.33MHz	Default
Resolution	800x480	SPEDPWD = 0x00000310
Line Frequency	33.3KHz	HTP=940, HSP=819, HSW=95, HDP=799 VTR=555, VSP=489, VSW=2, VDP=479 DCLK= 31.3MHz
Sprite Format	1bpp	-
Sprite Width	512pixels	-
Pattern Data	SRAM	-
Alpha Table	Off	-

Table 1 Attributes of an example

According to the above table:

F_{clk} = 83.33 MHz
F_{line} = 33.3 KHz
$W_{display}$ = 800 pixels
W_{spr} = 512pixels
Y_{spr} = 1 bit/pixel

Substitute above value into the previous formulas:

M = (1 / 33.3) * 83.33 * 1000 = 2502 cycles **see [1]**
N = 2502 – (800 / 2) = 2102 cycles **see [2]**
U = 2 * 2102 = 4204 pixels **see [3]**
T_{spr} = (512 * 1) / 32 = 16 words **see [4]**
16 %16 = 0 **see [5]**
RC = ((16/16) *20 / 16 = 1.25 word/cycle **see [5]**
V = ((2102 / 1.25) * 32) / 1 = 53811 pixels **see [7]**
MIN (U, V) = MIN (4204, 53811) = 4204 pixels

Finally, the SPE processing performance is:
4204 pixels/line or
4204/512 = 8.2 sprites/line