

## BADGE – Data Sheet

### General Description

**BADGE** – BitSim’s Accelerated Display Graphics Engine IP block for ASIC & FPGA, is an advanced graphic controller.

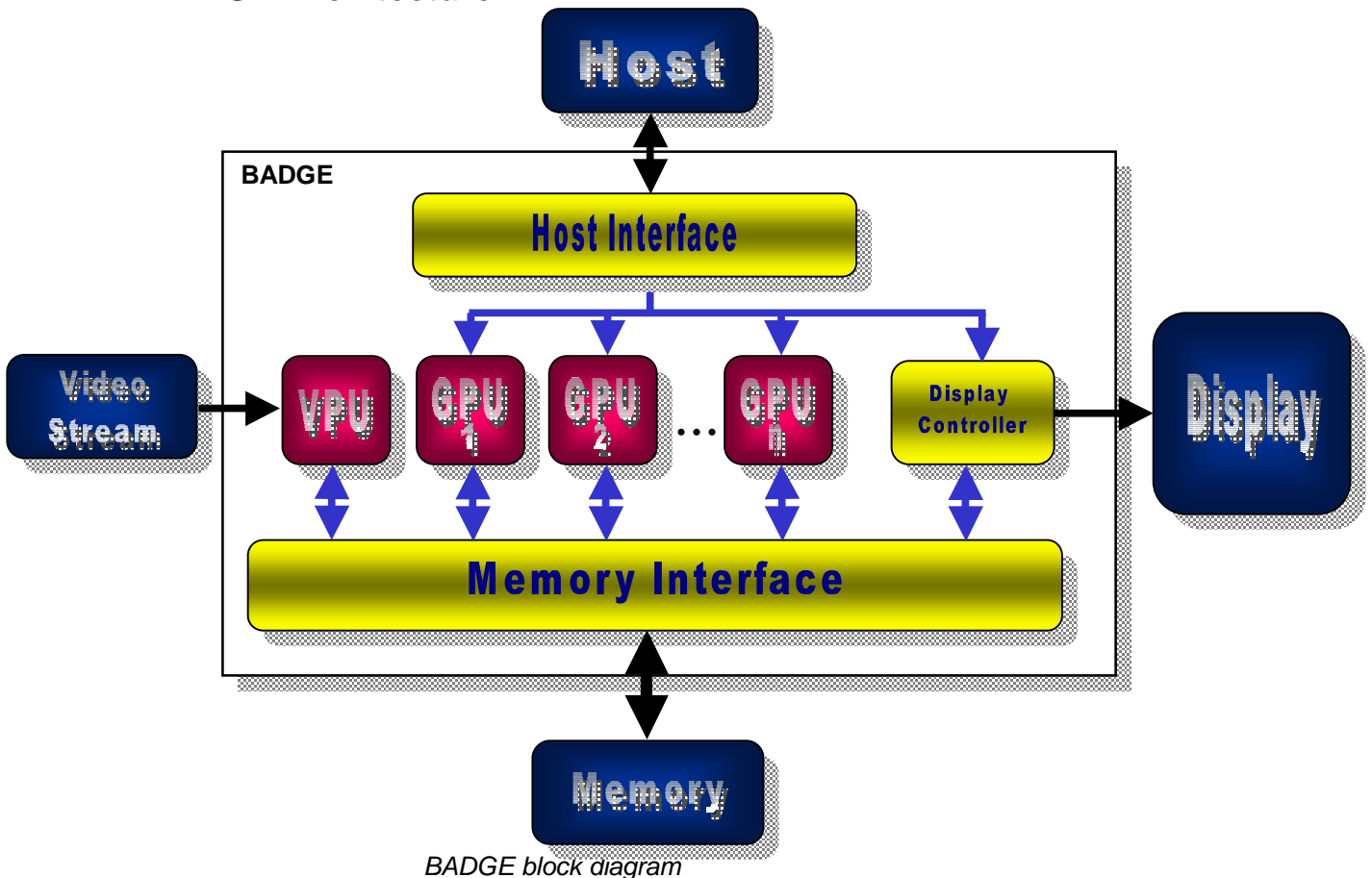
**BADGE** is an adaptable IP-block for ASIC and FPGA.

**BADGE** is easy to use and to implement. The only external components needed are a memory and a display. The processor may be a separate component or may be embedded with BADGE in the FPGA or ASIC. For analog video, an ADC is needed.

**BADGE** supports several memory types and this support can be extended to other memories.

**BADGE** supports several CPU bus types and this support can be extended to other CPU bus types.

### BADGE Architecture



The VPU (Video Processing Unit) transfers the information in a digital video stream into BADGE. The GPUs (Graphics Processing Units) are tailored to accelerate a specific graphic task and they work simultaneously or can be chosen independently.

Example of such tasks:

- Draw pixels, lines and rectangles
- Write text of various fonts, sizes and colors
- Copy, resize and recolor rectangles
- Draw/Move graphical objects – e.g. "sprites"
- Analog and Digital Video

As **BADGE** is modular it is convenient to include only the required GPUs or VPUs performing just the graphics operations needed in a system. This makes the design trimmed, excluding unused logic on a minimal area of silicon using a minimum of power.

If BADGE is implemented in an FPGA, it is even possible to dynamically reconfigure BADGE during operation.

## Features

- Fully synchronous, synthesizable and technology independent RTL code
- Capable of drawing shapes such as pixels, lines and rectangles
- Capable of drawing text of various fonts, sizes and colors
- Capable of copying, resizing and recolor rectangles – e.g. "BitBlit" and ROP
- Capable of drawing/moving graphical objects – e.g. "sprites"
- Supports *multi-buffered frame memory* – which eliminates flicker when graphical objects move
- Supports Analog Video – with external A/D circuit
- Support of alpha blending (2 variants) **(New)**
- Supports Digital Video – uncompressed (SDI) and compressed (MPEG)
- Anti-aliasing **(New)**
- Text/Graphics overlay
- Hardware Window – Picture-In-Picture support
- Hardware-cursor support, by making use of sprites
- Programmable frame rate
- Up to 4096 x 4096 pixels display resolution
- Generic color depth up to 24 bits per pixel
- Supports several memory types such as ZBT-SRAM, SDRAM, DDR etc.
- Supports address mapped linear frame buffer or single address command based interface
- Generic data bus width between BADGE and video memory
- Supports several bus types such as Avalon (Nios), CoreConnect/OPB (PowerPC, MicroBlaze) and Intel Xscale CPU-bus
- CPU-data bus of 32, 16 and 8-bit supported (8-bit **New**)
- BADGE has been used with displays such as:
  - Sharp display LQ065T9DR51, 400x240 (WQVGA)
  - Sharp display LQ057Q3DC12, 320x240 (QVGA) **(New)**
  - Samsung displays LTM150XH-T01, LTM150XH-L04, 1024x768 (XGA)
  - AU Optronics A070VW01 7" 800x480 (WVGA)
- Supports serial LVDS and parallel LVTTTL TFT-interface
- Supports DVI **(New)**
- Supports Display Power Sequencing
- Supports DE Only Mode, for displays which do not use hsync and vsync inputs
- Support of Portrait mode **(New)**
- A Test Pattern Generator is included, for debug purpose
- WinCE driver (CETK approved **New**)
- Linux driver (accelerated Frame Buffer for Linux **New**)
- API, for non OS users **(New)**

## LITE option

BADGE can be delivered in a LITE version, in which BADGE only acts as a display-controlling device with no support for graphic acceleration (drawing lines, rectangles, text support etc). However, support for video memory access on a pixel-by-pixel basis and Hardware cursors is included. Additional features from the list above can be added upon request.

## Future Enhancements

Since BADGE is implemented in 'hard SW', the following implementation specifics can be changed easily:

- Common memory for BADGE and Host
- The display resolution can easily be enlarged
- The color depth can be easily enlarged
- New Host Interfaces: new CPUs, PCI etc

## Applications

- Test and Measurement Instrumentation
- Medical Instrumentation
- Industrial Equipment
- Gaming and Amusement Machines



*BADGE: Implemented on BitSim's High Speed FPGA Platform board, HIPPO.*

## Interfaces

### General

| Signal Name | I/O | Width | Description             |
|-------------|-----|-------|-------------------------|
| Reset_n     | I   | 1     | Global reset            |
| Clk_mem     | I   | 1     | Memory controller clock |
| Clk_core    | I   | 1     | Core clock              |

*General Interface*

### Host Bus Interface

The host bus interfaces below are some typical examples. Other interfaces are also supported.

#### Avalon Bus Interface

| Signal Name | I/O | Width | Description                                   |
|-------------|-----|-------|---|
| Chipselect  | I   | 1     | Chip select signal to the slave               |
| Read        | I   | 1     | Read request signal to the slave              |
| Readdata    | O   | 32    | Data lines to the slave for read transfers    |
| Write       | I   | 1     | Write request signal to the slave             |
| Writedata   | I   | 32    | Data lines from the slave for write transfers |
| Waitrequest | O   | 1     | Stalls the slave if no immediate respons      |
| Irq         | O   | 1     | Slave interrupt request                       |

*Avalon Bus Interface*

#### CoreConnect/OPB Bus Interface

| Signal Name | I/O | Width | Description                       |
|-------------|-----|-------|-----------------------------------|
| OPB_DBus    | I   | 32    | Data to badge                     |
| OPB_RNW     | I   | 1     | Read/Not write                    |
| OPB_Rst     | I   | 1     | OPB Reset, not used               |
| CS          | I   | 1     | Chip select from address decoding |
| SIn_DBus    | O   | 32    | Data from Badge                   |
| SIn_toutSup | O   | 1     | Timeout suppress                  |
| SIn_xferAck | O   | 1     | Slave Transfer acknowledge        |
| Irq         | O   | 1     | Interrupt                         |

*CoreConnect/OPB Bus Interface*

## Memory Interfaces

The memory interface support can be easily changed to support other memory types besides those described below. G.W. below denotes generic width, that is a generic in VHDL.

### ZBT Memory Interface

Supports both pipelined ZBT and flow-through ZBT.

| Signal Name | I/O | Width | Description          |
|-------------|-----|-------|----------------------|
| maddr       | O   | G.W.  | Address              |
| mdata       | I/O | G.W.  | Data                 |
| mbw         | O   | G.W.  | Byte write enables   |
| mrw         | O   | 1     | R/W select           |
| moe         | O   | 1     | Output enable        |
| mce1        | O   | 1     | Chip enable          |
| mce2n       | O   | 1     | Chip enable          |
| mce2        | O   | 1     | Chip enable          |
| mzz         | O   | 1     | Snooze enable        |
| mlbon       | O   | 1     | Burst/linear select  |
| mcken       | O   | 1     | Clock enable         |
| mldn        | O   | 1     | Address advance/load |

*ZBT Memory Interface*

### SDRAM Memory Interface

| Signal Name | I/O | Width | Description   |
|-------------|-----|-------|---|
| DQ          | I/O | G.W.  | Data input/output   |
| /CAS        | O   | 1     | Column Address Strobe   |
| /RAS        | O   | 1     | Row Address Strobe  |
| /We         | O   | 1     | Write Enable  |
| /S0, /S1    | O   | 2     | Chip Select   |
| CKE0, CKE1  | O   | 2     | Clock Enable, controls internal clock signal  |
| Address     | O   | G.W.  | Address   |
| BA          | O   | G.W.  | Selects banks to be activated during /RAS activity<br>Selects banks to be read/written during /CAS activity |
| DQM         | O   | G.W.  | Data Mask   |

*SDRAM Memory Interface*

## Display Interface

LVTTL

| Signal Name | I/O | Width | Description  |
|-------------|-----|-------|--|
| Enable      | O   | 1     | Indicates active video pixels                      |
| Hsync       | O   | 1     | Horizontal synchronous signal                      |
| DispClk     | O   | 1     | Display clock signal for sampling each data signal |
| Vsync       | O   | 1     | Vertical synchronous signal                        |
| Blue        | O   | G.W.  | Blue data signal                                   |
| Green       | O   | G.W.  | Green data signal                                  |
| Red         | O   | G.W.  | Red data signal                                    |

*LVTTL Display Interface*

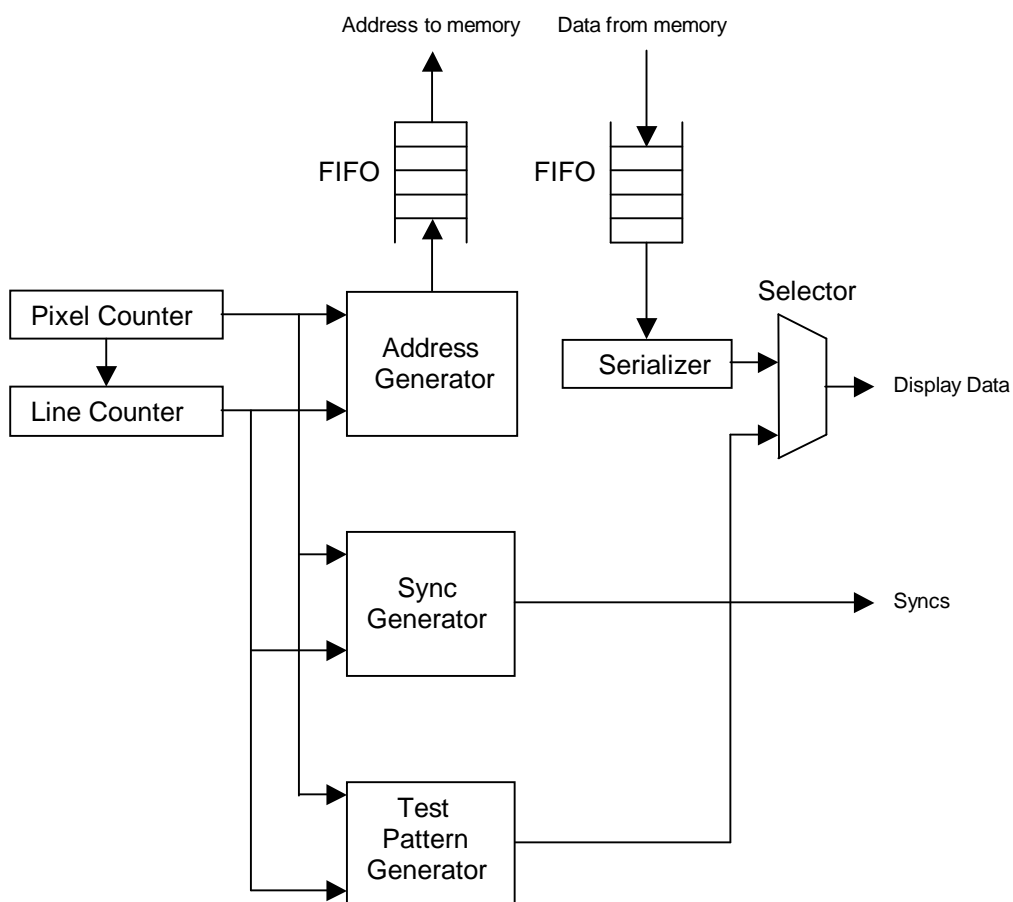
LVDS

| Signal Name    | I/O | Width  | Description        |
|----------------|-----|--------|--------------------|
| TXCLKp, TXCLKn | O   | 1 pair | LVDS Display clock |
| TX3p, TX3n     | O   | 1 pair | LVDS Data          |
| TX2p, TX2n     | O   | 1 pair | LVDS Data          |
| TX1p, TX1n     | O   | 1 pair | LVDS Data          |
| TX0p, TX0n     | O   | 1 pair | LVDS Data          |

*LVDS Display Interface*

## Block Diagram

### Display Controller



*Display Controller Block Diagram*

## BADGE Description

### Interfaces:

**Host Interface:**

Communication with a host processor.

**Memory Controller Interface:**

For reading and writing to the external video memory.

**Display Controller Interface:**

The display controller reads image data from the video memory and outputs it to the display together with display clock, sync and enable signals.

**Video Stream Interface:**

Today, the digital interface ITU-R BT.656 is supported. An external A/D circuit is needed.

## GPUs and VPUs

**MDAGPU:**

The Memory Direct Access GPU lets the host write and read directly in the video memory. This is required for storing image bitmaps, fonts, etc in the video memory.

**SPDGPU:**

The Simple Pixel Drawing GPU is used for drawing points, lines and rectangles, with specified color.

**CHRGPU:**

The CHRGPU is used to accelerate text drawing with various fonts, sizes and colors.

**RCCGPU:**

The ReCtangle Copy GPU performs Rectangle Copying including Raster OPerations (ROP). ROP is normally used in Graphical User Interfaces (GUI), for example inverting and shadowing of icons. Performance: around 110Mpixel/s (for Spartan-3/Cyclone families).

**VPU:**

The Video Processing Unit is used for showing composite, S-video or other analog formats on a display.

**MPEGGPU:**

The MPEG GPU is used to accelerate the decoding of MPEG 1/2 encoded video streams.

## Example of Device Utilization & Performance

| <b>Implementation example:<br/>BADGE Lite</b> | <b>Altera FPGA</b><br><i>Cyclone II C5<br/>(BADGE uses ~60%)</i>  | <b>Xilinx FPGA</b><br><i>Virtex 300-5</i>  | <b>Xilinx FPGA</b><br><i>Spartan-3 200<br/>(BADGE uses ~65%)</i>  |
|---|---|--|---|
| <i>Video memory</i>                           | SDRAM, 1M x 32 bits   | 2 ZBT SRAM, 64k x 36 bits each, 10 ns access   | SDRAM, 1M x 32 bits   |
| <i>Display</i>                                | 1024 x 768 pixels, 16 bits per pixel, 60 frames per second  | 400 x 240 pixels, 18 bits per pixel, 50 frames per second  | 1024 x 768 pixels, 16 bits per pixel, 60 frames per second  |
| <i>System clock frequency</i>                 | 100 MHz   | 40 MHz (to all BADGE blocks except the memory controller: 80 MHz)  | 100 MHz   |
| <i>Display clock frequency</i>                | 50 MHz  | 10 MHz   | 50 MHz  |
| <i>Number of global clock-nets</i>            | 2   | 2  | 2   |
| <i>Other info</i>                             | The data bus between the FPGA and the video memory is 32 bits wide, which enables 2 simultaneously pixel operations/accesses. Burst access is used to guarantee performance. Multipixel-operations (concurrent access of multiple pixels in the memory) double the performance. | The data bus between the FPGA and the video memory is 72 bits wide, which enables 4 simultaneously pixel operations/accesses. The performance in terms of pixel updating is 40 Mega pixel per second for each GPU - 80 Mega pixel per second for the system. Multipixel-operations (concurrent access of multiple pixels in the memory) quadruple the performance. | The data bus between the FPGA and the video memory is 32 bits wide, which enables 2 simultaneously pixel operations/accesses. Burst access is used to guarantee performance. Multipixel-operations (concurrent access of multiple pixels in the memory) double the performance. |

*Example of different implementation examples*

| Functional Unit     | Size in LE (Altera) | Size in LC (Xilinx) | Description   |
|---------------------|---------------------|---------------------|---|
| Host Ctrl (Avalon)  | 199                 | N.A.                | Avalon host controller block.                             |
| Host Ctrl (OPB)     | N.A.                | 160                 | OPB host controller block.                                |
| Memory Ctrl (ZBT)   | N.A.                | 305                 | ZBT memory controller block.                              |
| Memory Ctrl (SDRAM) | ~280                | ~280                | SDRAM memory controller block.                            |
| Display Ctrl        | *                   | *                   | Display controller block.                                 |
| MDAGPU              | *                   | *                   | Memory Direct Access GPU. Host access to graphics memory. |
| SPDGPU              | *                   | *                   | Simple Pixel Drawing GPU. Pixel drawing.                  |
| CHRGPU              | *                   | *                   | Character GPU. Text writing.                              |
| RCCGPU              | *                   | *                   | Rectangle Copy GPU (BitBlit & ROP)                        |
| VPU                 | *                   | *                   | Video Processing  |
| MPEGGPU             |                     | *                   | Accelerate decoding of MPEG video streams                 |

*Preliminary LE/LC utilization per functional block*

*\* Contact BitSim for exact sizes*

## BADGE Configurations

|             |  |
|-------------|--|
| BADGE Lite  | <ul style="list-style-type: none"> <li>• Acts as a display-controlling device</li> <li>• Pixel-by-pixel access</li> <li>• HW Cursor</li> </ul> |
| BADGE 2D    | <ul style="list-style-type: none"> <li>• 2D graphics acceleration</li> </ul>   |
| BADGE Video | <ul style="list-style-type: none"> <li>• Analog Video or</li> <li>• Digital Video</li> </ul>   |
| BADGE Full  | <ul style="list-style-type: none"> <li>• Analog or Digital Video and</li> <li>• 2D acceleration</li> </ul>                                     |

*Table of different configurations of BADGE*

BADGE will fit into some of the smallest versions of the Altera Cyclone-II and Xilinx Spartan-3 families, depending on which BADGE configuration.

## Deliverables

|   |  |
|---|--|
| Documentation                             | <ul style="list-style-type: none"><li>• Data Sheet</li><li>• User Guide</li></ul>            |
| Design File Formats                       | <ul style="list-style-type: none"><li>• VHDL or Netlist</li><li>• Constraints File</li></ul> |
| Verification                              | <ul style="list-style-type: none"><li>• Test Bench</li><li>• Command File</li></ul>          |
| Reference Design available (not included) | <ul style="list-style-type: none"><li>• Demo HW Platform, including test SW</li></ul>        |
| Simulation Tool                           | <ul style="list-style-type: none"><li>• ModelSim script</li></ul>                            |
| Support                                   | <ul style="list-style-type: none"><li>• Provided by BitSim AB</li></ul>                      |

*Table of Deliverables*

## Contact Information

### BitSim AB

#### Stockholm

S:t Eriksgatan 33  
SE-112 39 Stockholm  
Sweden  
Tel +46 8 54 55 56 00  
Fax +46 8 54 55 56 11

#### Växjö

Sandvägen 2  
SE-352 45 Växjö  
Sweden  
Tel +46 470 74 29 70  
Fax +46 470 74 22 67

[www.bitsim.com](http://www.bitsim.com)