Vivante GCNanoUltraV Series
Small Footprint Vector Graphics IP

Hardware Features
V2.0.x

Document Revision 0.89
24 February 2023

This document is compatible with Vivante
GCNanoUltraV hardware versions 2.0.x

Legal Notices

COPYRIGHT INFORMATION

This document contains proprietary information of Vivante Corporation and VeriSilicon Holdings Co., Ltd. They reserve the right to make changes to any products herein at any time without notice and do not assume any responsibility or liability arising out of the application or use of any product described herein, except as expressly agreed to in writing by Vivante and/or VeriSilicon; nor does the purchase or use of a product from Vivante or VeriSilicon convey a license under any patent rights, copyrights, trademark rights, or any other of the intellectual property rights of Vivante, VeriSilicon or third parties.

DISCLOSURE/RE-DISTRIBUTION LIMITATIONS

The information contained herein is not to be used by or disclosed to third parties without the express written permission of an officer of Vivante Corporation or VeriSilicon Holdings Co., Ltd.
(VeriSilicon Distribution **LEVEL D: CONFIDENTIAL – NOT FOR REDISTRIBUTION**).

SoC vendors with an active license may re-package portions of this document for general distribution. Contents which are NOT eligible for redistribution are in orange font enclosed in an orange box. Upon request VeriSilicon provides this document in Word format for adaptive inclusion in documentation intended for your customers. VeriSilicon requests a pre-release copy of the adaptation for review and approval. In the adapted document please identify the technology and features described in this document using the Vivante trademark and include a reference to the copyright owner, for example: "This section describes the Vivante® GPU and contains copyright material disclosed with permission of VeriSilicon."

TRADEMARK ACKNOWLEDGMENT

VeriSilicon® and the VeriSilicon logo design are the trademarks or the registered trademarks of VeriSilicon Holdings Co., Ltd. Vivante® is a registered trademark of Vivante Corporation. All other brand and product names may be trademarks of their respective companies.

For our current distributors, sales offices, design resource centers, and product information, visit our web page located at http://www.verisilicon.com.

For technical support, please email vivante-support@verisilicon.com.

Preface

This document is the primary feature description for Vivante GCNanoUltraV Series vector graphics processing unit IP core.

Audience

This document was prepared for both hardware and software integrators who are familiar with system-on-a-chip (SoC) digital design and related support devices. Those who would benefit from this technical document are:

- Engineers and managers who are evaluating this core for use in a system
- Engineers who are designing this core into a system

Conventions Used in This Document

AHB – Advance High Performance Bus
APB – Advanced Peripheral Bus
AXI – Advanced eXtensible Interface
DDR – Double Data Rate
DMA – Dynamic Memory Access
FE – Graphics Pipeline Front End/Fetch Engine
GPU – Graphics Processing Unit
GUI – Graphical User Interfaces
HI – Host Interface
IM – Imaging Engine
IOT Internet of Things
MC – Memory Controller
MCU Micro Controller Unit
MPU Micro Processor Unit
PE – Pixel Engine
PM – Power Management
RA – Rasterizer
RTL – Resistor Transistor Logic
SoC – System on Chip
TS – Tessellation Engine
VG – Vector Graphics

The word *assert* means to drive a signal true or active. Signals that are active LOW end in an "n."
Hexadecimal numbers are indicated by the prefix "0x" —for example, 0x32CF.
Binary numbers are indicated by the prefix "0b" —for example, 0b0011.0010.1100.1111
Code snippets are given in `Consolas` typeset.

Table of Contents

List of Figures

List of Tables

1 Introduction

This document provides a summary of the functional feature set for the Vivante GCNanoUltraV Series small footprint vector graphics processing unit IP core (GCNanoUltraV GPU IP). The Vivante GCNanoUltraV Series IP defines a next generation high-performance/area UI graphics core designed for hardware acceleration of Vivante's proprietary VGLite™ Graphics API for vector graphics display on a variety of consumer devices.

Architectural enhancements bring the look and feel of consumer product GUI rendering and menu displays to embedded and Internet-of-Things (IOT) devices that need ultra-low power, minimal silicon area and zero DDR memory. The Vivante GCNanoUltraV Series cores has a simple, lightweight application programming interface (API) and software architecture that makes customization easy and minimizes memory footprint.

The Vivante GCNanoUltraV Series processor cores are designed specifically for MCU and MPU applications that need hardware accelerated UI displays and effects. Built upon Vivante's successful mass market proven ScalarMorphic™ architecture powering some of the leading smartphones, TVs, tablets, automobiles, and embedded devices, the embedded UI architecture of the GCNanoUltraV Series brings high-end graphics capabilities to MCU/MPU designs.

GCNanoUltraV Series is designed from the ground up to work within the limitations of highly memory and compute resource constrained MCU/MPU solutions including embedded flash memory configurations. These features are absolutely essential for building efficient embedded systems targeting the next generation of displays to meet rising consumer expectations for graphics intensive man-machine interaction.

Our robust embedded UI graphics solution includes easy-to-use software, a VGLite Graphics API to enable the GPU acceleration through customized applications on Linux.

Vivante GCNanoUltraV Series IP is designed for easy integration onto the SoC, providing powerful graphics at low power consumption and the smallest of silicon footprints. The core is delivered as synthesizable RTL. It is technology independent and can be synthesized using a variety of libraries. Dynamic power consumption is minimized by extensive use of localized clock gating.

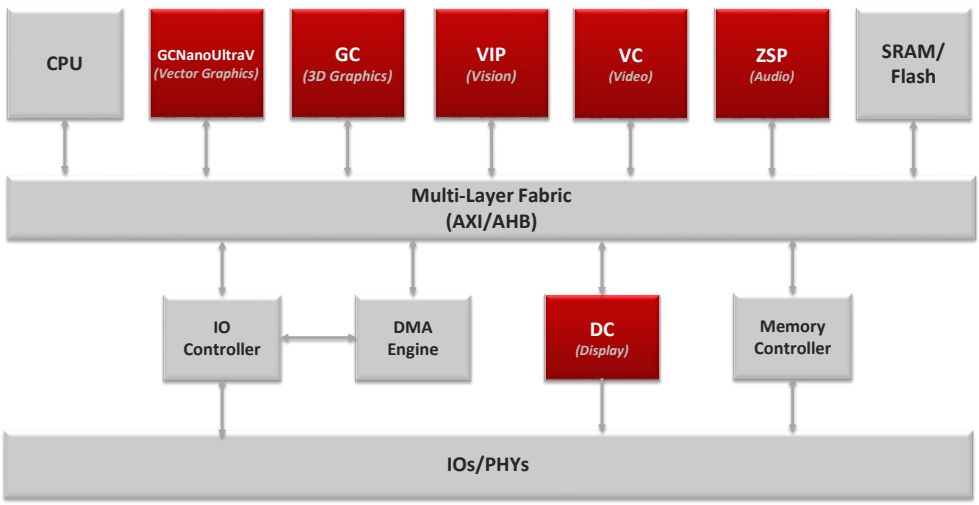

Figure 1. Typical SoC with Vivante GCNanoUltraV IP

1.1 GCCORE Design Description

The main functional blocks of the GCCORE are described here, and a block diagram is shown below.

Host Interface	Allows the GCCORE to communicate with external memory and the CPU through AXI and/or AHB/APB buses. In this block data crosses clock domain boundaries.
Memory Controller	Internal memory management unit that controls the block-to-host memory request interface.
Graphics Pipeline Front End	Inserts high level primitives and commands into the graphics pipeline.
Tessellation Engine	Transforms vertices and control points. Tessellates lines, quadratic and cubic Bezier curves.
Vector Graphics Engine	Rasterizer that converts primitives to pixels.
Imaging Engine	Paint and image generator that colors each pixel.
Pixel Engine	Renderer that combines different sources into the final pixel value.

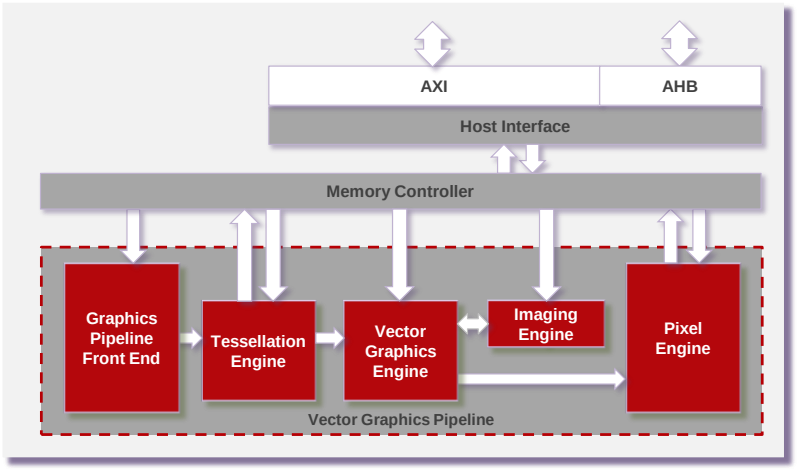

Figure 2. GCNanoUltraV Block Diagram with Vector Graphics Pipeline and AXI+AHB

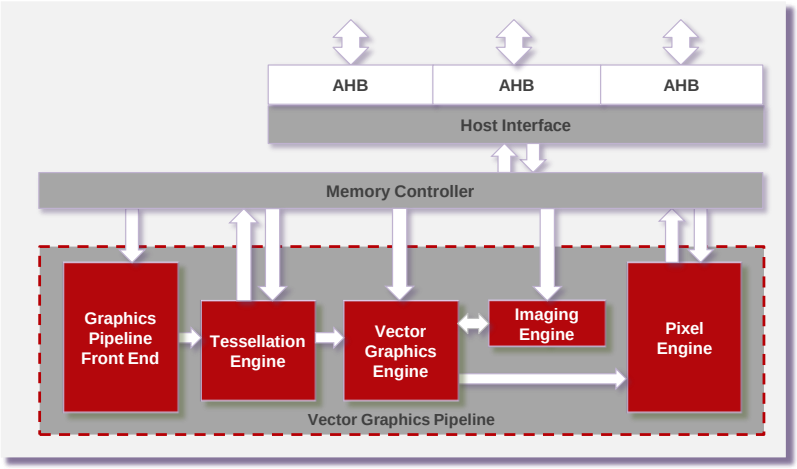

Figure 3. GCNanoUltraV Block Diagram with Vector Graphics Pipeline with 3xAHB

1.2 GCCORE API Support

The following table describes the API support available for the Vivante GCNanoUltraV Series small footprint vector graphics IP.

Table 1. GCNanoUltraV API Support

Feature	GPU Support
Primary API	Vivante's proprietary VGLite® Graphics API
Drivers	Vivante GCNanoUltraV Driver
Operating systems	VGLite Graphics API is platform independent, typical use is Embedded Linux

1.3 Core Variants

The GCNanoUltraV V2.0x Series Vector Graphics is designed in the following primary variant:

GCNanoUltraV This is the default design. GCNanoUltraV Series has a small area. Product revisions may vary, especially in their RGB and YUV format support levels.

Because of its target small footprint design, few options are configurable for this IP. The following table summarizes some key features and options. Typical product features which can be customized by Vivante prior to RTL delivery are listed in the table below.

Table 2. Typical RTL Key Features and Customization Summary

Feature	GCNanoUltraV Default	Alternate Options
Interfaces	1 64-bit AXI + 1 32-bit AHB	3 32-bit AHB
32-bit bus interface	AHB	APB option
Clock source	clk1x	
Image Source Read	YES	
Vivante VGLite API Support	YES	
Transformation Support	3x3 perspective and affine	
Linear Paint	YES	
INDEX1/2/4/8 Input Support	YES	
YUY2 Input Support	YES	
YUV Tiled Input Support	NO	option
ETC2 Compression	NO	option
DECNano Compression/Decompression	NO	option Compress on Output, Decompress Input and Output

For pixel format support, refer to Section 2.5.

This Vivante design is customizable, including customizations for RGB color formats and YUV formats. Please note that all variations presented in this document may not be immediately available “off the shelf” as Ready-IP. Check with your Vivante personnel for schedule and availability.

2 GPU Core Module Features

2.1 GPU Host and Memory Interface Features

Table 3. Host and Memory Interface Features

Feature	GPU Support
Interfaces	1 64-bit AXI + 1 32-bit AHB (default) or 3x32-bit AHB. APB alternate option for AHB.
Code and data memory location restrictions	Unrestricted; arbitrary memory reads and writes
Physical address	32 bits
Read write request size support	Memory controller supports 8, 16, 32, 64 byte read and write requests
Resource locks with CPU	Semaphore lock
Clock domains	3 domains: core clock (from clk1x), register access AHB (from HCLK pin) or alternately APB (from PCLK pin), and memory data access from AXI (ACLK pin) or alternately AHB (from ACLK pin)

2.2 GPU Power Management Features

Table 4. Power Management Features

Feature	GPU Support
Low power CMOS technology compatible	Yes
Automatic clock gating of flip flops and rams	Yes
Global clock gating of unused macro blocks	Yes
Active (ON), Idle, Standby and Sleep (OFF) Programmable Power Modes	Yes

2.3 GPU Command Processor Features

Table 5. Command Processor Features

Feature	GPU Support
Command list structure	Call memory buffer
GPU register access	AHB access to selected GPU registers
GPU-CPU synchronization	Synchronization occurs via Interrupt queues.
Command OPCODES for GCNanoUltraV	The Command set is different than that of most other GCCORES: 00 END 01 SEMAPHORE 02 STALL 03 STATE 04 DATA 06 CALL 07 RETURN 08 NOP Additional Command OPCODES are provided for Path Data control.

2.4 GCNanoUltraV Series Graphics Hardware Features

Table 6. Graphics Hardware Features

Feature	Description
Area	Feature set optimized for small footprint
Blending	8 Porter Duff blending modes
Rotation	To any angle
Fill rules	Odd/even and non-zero
Paint paths	Linear and curved paint paths
Compression	ETC2
Coordinate systems and transformations	Image drawing uses a 3x3 perspective transformation matrix
Paints	Solid and linear paint
Image filters and interpolation	Point, Linear and Bi-linear blends
Pixel formats for image interpolation	Pixel formats for image interpolation. *See following Section 2.5 for detail.*

2.5 Format Support with GCNanoUltraV IP

Table 7. Graphics Formats Available

<table>
<tr><th>Feature</th><th>GPU Support</th></tr>
<tr><td>Image formats

The graphics engine supports these formats for source image and destination render targets.

Notes:
• Your design variant may be customized and not include support for all possible formats.</td><td>With hardware V 2.0.x
<table>
<tr><th>Bit Depth</th><th>Format</th><th>Supported for Source IMAGE</th><th>Supported for Destination</th></tr>
<tr><td>1</td><td>INDEX1</td><td>YES</td><td>NO</td></tr>
<tr><td>2</td><td>INDEX2</td><td>YES</td><td>NO</td></tr>
<tr><td>4</td><td>A4</td><td>Yes</td><td>NO</td></tr>
<tr><td>4</td><td>INDEX4</td><td>YES</td><td>NO</td></tr>
<tr><td>8</td><td>A8</td><td>Yes</td><td>Yes</td></tr>
<tr><td>8</td><td>INDEX8</td><td>YES</td><td>NO</td></tr>
<tr><td>8</td><td>L8</td><td>Yes</td><td>Yes</td></tr>
<tr><td>8</td><td>A2R2G2B2</td><td>Yes</td><td>Yes</td></tr>
<tr><td>12</td><td>NV12_TILED</td><td>NO (option)</td><td>NO</td></tr>
<tr><td>12</td><td>YV12_TILED</td><td>NO (option)</td><td>NO</td></tr>
<tr><td>16</td><td>A1R5G5B5</td><td>Yes</td><td>Yes</td></tr>
<tr><td>16</td><td>A4R4G4B4</td><td>Yes</td><td>Yes</td></tr>
<tr><td>16</td><td>R5G6B5</td><td>Yes</td><td>Yes</td></tr>
<tr><td>16</td><td>YUY2</td><td>YES</td><td>NO</td></tr>
<tr><td>16</td><td>YUY2_TILED</td><td>NO (option)</td><td>NO</td></tr>
<tr><td>16</td><td>NV16_TILED</td><td>NO (option</td><td>NO</td></tr>
<tr><td>16</td><td>YV16_TILED</td><td>NO (option)</td><td>NO</td></tr>
<tr><td>24</td><td>A8R5G6B5</td><td>Yes</td><td>Yes</td></tr>
<tr><td>24</td><td>R8G8B8</td><td>Yes</td><td>Yes</td></tr>
<tr><td>32</td><td>A8R8G8B8</td><td>Yes</td><td>Yes</td></tr>
<tr><td>32</td><td>X8R8G8B8</td><td>Yes</td><td>Yes</td></tr>
</table></td></tr>
<tr><td>Color component swizzle support</td><td>Supported color component swizzles are:
SWIZZLE_ABGR, SWIZZLE_ARGB, SWIZZLE_BGRA, SWIZZLE_RGBA</td></tr>
<tr><td>DECNano format compression support (optional)</td><td>
<table>
<tr><th>Bit Depth</th><th>Format</th><th>Supported for Source</th><th>Supported for Destination</th></tr>
<tr><td>32</td><td>A8R8G8B8</td><td>Optional</td><td>Optional</td></tr>
<tr><td>32</td><td>X8R8G8B8</td><td>Optional</td><td>Optional</td></tr>
</table>
Note: DECNano can be bypassed by software configuration.</td></tr>
<tr><td>ETC2 format compression support (optional)</td><td>
<table>
<tr><th>Bit Depth</th><th>Format</th><th>Supported for Source</th><th>Supported for Destination</th></tr>
<tr><td>32</td><td>A8R8G8B8</td><td>Yes</td><td>NO</td></tr>
</table></td></tr>
</table>


VeriSilicon Distribution Level D:
Confidential - Not for Redistribution

3 Encode and Decode Hardware Features

Key features of the DEC compression unit are shown in the following table.

Table 8. Encode Decode Hardware Features

Feature	DEC Support
Input and Output Pixel Formats	See Data Formats Section
Compression Scheme	Lossy Compression and decompression
Compression Modes	3: No sample; Horizontal only; Horizontal and vertical
Compression De-compression Ratio	1.6 to 3.0 dependent on color format and compression mode
Pixel Data Organization Support (in pixels)	Tile (4x4) and Raster Scan (16x1)
User controls*	Compression is controlled by the GPU
Compression Units	64 Bytes
Burst Size	4 Bytes
By-pass Scheme	Uncompressed data travels through the GPU pipeline directly bypassing the DEC compression/decompression logic then travels across the AXI interface through standard memory-to-the memory display path. DECNano bypass can be configured via software.
Codec Pixel Data Rate	1 pixel per cycle encode/decode for 32bpp
Bandwidth Reduction	2-3X with lossy compression
Maximum Concurrent Channels	DECNano: 1 read path and 1 write path DECNanoE: 1 write path DECNanoD: 1 read path
Tile Status Support	NO
Error Notification	Yes

*Note: The DECNano does not have any user-controllable features. Functions such as stream support, stream identification, flush and outstanding requests should be implemented outside the DECNano if necessary.

3.1 DECNano Lossy Data Compression Overview

The DECNano Series supports the following formats, tile modes and Compression modes.

- Formats are 32 bit-per-pixel ARGB8 and XRGB8, specified in signal **format**. A/X is in the upper bits and B in the lower bits.
- Tile mode is either linear 16x1 pixels or tiled 4x4 pixels per tile. Each tile occupies 64 Bytes.
- Three compression modes are supported:
  - None
  - Horizontal sample
  - Horizontal and vertical sample. Note this mode is not supported for linear scanlines.

Table 9. Compression Ratios Per Compression Mode

Color Subsample / Compression Mode	ARGB	XRGB
None	1.6:1	1.6:1
Horizontal Only (raster or tile)	2.1:1	2.1:1
Horizontal and Vertical (tile only)	2.6:1	3.0:1

4 Software API

The Vivante VGLite® Graphics API is designed specifically for use with Vivante GCNanoUltraV Series vector graphics processing unit IP. It provides mechanisms for hardware accelerated two-dimensional vector and raster graphics, thereby allowing the user to implement customized applications and drivers for their hardware accelerated graphical user interface.

4.1 Vivante VGLite Graphics API

Vivante's platform independent VGLite Graphics API (Application Programming Interface) is designed to support menu driven user interfaces optimized for a system's overall resource requirements. Its goal is to provide maximum performance, while keeping the memory footprint to a minimum. The typical environment is Embedded Linux. The Vivante VGLite Graphics API allows for fine granularity in memory usage and is appropriate for use in cases where only one of the available rendering classes is used.

4.1.1 Reference

The document **Vivante Programming: VGLite Vector Graphics API** is the current reference document for the VGLite API. It provides a set of functions for menu UIs with Vivante vector graphics GCNanoUltraV Series cores.

4.1.2 Organization of the VGLite Graphics API

The VGLite API is partitioned to provide controls for three main areas of functionality:

- **Initialization** for hardware and software initialization
- **Blit** for raster rendering
- **Draw** for draw operations

Hardware supported Blit and/or Draw capabilities available for software control through the API include:

Table 10. Blit and Draw Controls in VGLite API

Capability	GCNanoUltraV GPU with VGLite API Support
Fill	Non-zero or Even-Odd
Path coordinate formats	S8, S16, S32, FP32
HW Anti-alias levels	Low, Medium, High
Blending	SrcOver, DstOver, SrcIn, DstIn, Screen, Multiply, Additive, Subtractive
Data Formats	Linear: INDEX1/2/4/8, ARGB2/4/8, XRGB8, RGB8, A8RGB565, A4/8, L8, YUY2 and BRGA/X variants. Tiled YUV formats if supported by hw. (refer to VGLite API documentation for full list of formats)
Path definitions	bounding box, path quality, coordinate format, buffer pointer, path length, path data

Document Revision History

This section describes top level differences in the versions of this document.

Document Revision	Doc Date	Compatible Hardware	Description
0.89	2023-02-24	GCNanoUltraV	Update document template. Section 1.3: Add DECNano option Section 2.5: Add DECNano format support. Section 3: Inserted for DECNano
0.88	2022-09-02	GCNanoUltraV	Update document template. Miscellaneous format and text refinements. Section 3.1.2 Table 8: add YUY2.
0.87	2022-05-30	GCNanoUltraV	Update document template. Miscellaneous refinements. Section 2.5 Table 7: Remove SWIZZLE_UV, SWIZZLE_VU.
0.86	2022-04-12	GCNanoUltraV	Section 2.4, Table 6: Add row for image filters
0.85	2022-02-02	GCNanoUltraV	Legal Notices, General: Update branding layout to include VeriSilicon. Miscellaneous format refinements.
0.84	2022-01-11	GCNanoUltraV	Section 1.1: Add block diagram for AXI+AHB. Section 1.3: Refine Interface options. Update INDEX as supported by default. Remove ARGB2 (also default, not a separate option). Section 2.1and various: Update available interface options. Section 2.3: remove WAIT. Section 2.4: update format description to reference next table. Section 2.5 Table 7: remove DI references and update support levels.
0.83	2021-12-10	GCNanoUltraV	Section 1.3: Add format options for INDEX and ARGB2. Section 2.3: remove WAIT. Section 2.4: update format description to reference next table. Section 2.5 Table 7: add optional formats. Add note for YUV targets.
0.82	2021-11-12	GCNanoUltraV	Section 1.3 and various: Remove variant F. Section 1.3 Table 1. Add more key features.
0.81	2021-10-01	GCNanoUltraV	Section 2.5 Table 7 and Section 3.1.2 Table 8: Add tiled formats.
0.80	2021-07-13	GCNanoUltraV	Initial Edition