Lesson 2
Playstation Vita Development

Playstation Vita System Architecture

Sam Serrels and Benjamin Kenwright[1]*

Abstract
An introduction to developing with the system architecture of the Sony Playstation Vita. The features and functions of the hardware will be discussed along with information on the internal processing of the Vita.

Keywords
Sony, PSVita, PlayStation, Setup, Windows, SDK, Development, ELF, SELF, Programming, Visual Studio, Debugging

[1] *Edinburgh Napier University, School of Computer Science, United Kingdom*: b.kenwright@napier.ac.uk

Contents

Figure 1. PSVita Overview -

Introduction

About the Edinburgh Napier University Game Technology Playstation Vita Development Lessons Edinburgh Napier University Game Technology Lab is one of the leading game teaching and research groups in the UK - offering students cutting edge facilities that include Sony's commercial development kits. Furthermore, within the Edinburgh Napier Game Technology group are experienced developers to assist those students aspiring to releasing their own games for Playstation. Student have constant access to he Sony Development Kits (DevKits) and encourage enthusiastic students to design and build their own games and applications during their spare time.

1. The PS Vita

Sony's Playstation Vita (officially abbreviated to PS Vita) is a portable handheld game console released at the start of 2012. It is the successor to the PlayStation Portable(the PSP) as part of the PlayStation brand of gaming devices.

The Old The PSP followed a conventional gaming system design, it was very much a standard video games console, only shrunken down into a small form factor. Using custom designed chips with custom instruction sets.

The New The PS Vita is designed with a more modern approach, a single custom chip, using an ARM Cortex-A9 as the CPU, coupled with a PowerVR SGX543MP4+ GPU. Also within the chip is the Main Memory(512MB - 282MB usable), 128MB of Video Memory and other hardware such as dedicated video codes.

Similarities This new Design is very similar to the modern smartphone. The ARM Cortex-A9 MPCore as been used in hundreds of popular products and has a reputation for

power efficiency and multi-core optimisations. The PowerVR SGX543 GPU is also widely used, almost always in conjunction with an ARM core. Sony has chosen to go with a now standard design, and it's evidently a very good design.

So is it a phone?   Although the primary brains of the PS Vita are similar to a smartphone, it isn't secretly running Android in the background. The Software is all custom and tailor made for the system, with the focus on being the best API for games. What we have is the best of both worlds, a tried and tested processing architecture that is designed for battery powered devices, running with software built for games that knows how to bridge between your software and the strengths of the hardware.

1.1 External Media
PlayStation Vita Card   The Primary media for games is the "PlayStation Vita Card", when you buy a game, this the physical form that it comes on. It's a proprietary flash storage card, with the capacity for 2 GB or 4 GB. The bonus to using flash storage rather than optical media, is that some of the card can actually be *written* to. This can be used for storing save data or game patches.

PlayStation Vita Memory Card   The media for general storage is the "PlayStation Vita Memory Card". It looks very similar to a microSD card, but is actually a proprietary device and is only compatible with the PS Vita. The sizes available range from 4 GB to 64 GB, this space can be used to store save games, media, and whole games, downloaded from the on-line store.
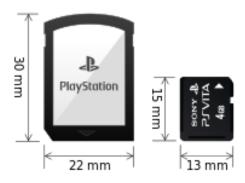

Figure 2. PSVita External Media - Left: The Vita Card, Right: Vita Memory Card

1.2 Other Hardware
Touch Panels   The PS Vita has two capacitive sensing touch panels, one on the front covering the screen, and one on the rear of the device. The maximum number of touch points is 6 for the front touch panel and 4 for the rear touch panel.

Cameras   There is both a front and rear camera on the device, both run at 640x480 (VGA) at 60 frames/s, or at 320x240 at 120 frames/s.

Motion Sensors   For sensing motion, there is an internal three-axis accelerometer, the three-axis gyroscope and a magnetometer.

Wireless Communication   Bluetooth and Wireless 802.11b/g/n wifi are supported. Some models of the PS Vita Support 3G networking using a simcard, the Devkit does not have this feature, but can emulate it.

HDMI Ouput   The screen can be mirrored to output over HDMI at a resolution of 480p or 720p. Audio is also routed over HDMI. When outputting to 720p, the 960x544 display buffer is either displayed as is on the larger screen (with black bars), or it can be scaled up.

Components   The Playstation Vita development lessons are divided into subsequent parts. The article lessons give the students a chance to work within a commercial console environment. In addition, it allows the students to port their games/applications to the PSVita. The different lessons include:

1.3 Full Specification
CPU
      ARM Cortex-A9 - 4 Cores - 0.8 GHz to 2 GHz
      L1 - 32 KB Instruction Cache, 32 KB Data Cache
      L2 - 2 MB shared by all cores
Memory
      LPDDR2 400MHz
      Retail: 512MB RAM, 282MB Useable
      Devkit: 1GB RAM, 538MB Useable
Video Memory
      VRAM: 128 MB VRAM, 112MB Useable
GPU
      PowerVR SGX 543MP4+ (4 cores)
      Tile Based Deferred Rendering
      Shader Model 3.x (With some extras)
      On-Chip MSAA x4/x2 processing
Screen
      OLED/LCD - 5 inches (16:9) - 960 x 544
Touch
      Font and Rear capacitive panels
      Max touch points: Front - 6, Rear - 4

2. Runtime API

When a game is launched on the PS Vita, full control of the system is given to the application. The System software is still running in the background, but only to manage the hardware and abstract the true low level operations. This background process does take up valuable RAM, but you could not use the system without it.

2.1 GXM, the graphics API
The GXM library is the library that handles rendering, shaders and keeping the GPU happy. It is very similar to the GCM library used on the PS3 in terms of render calls when drawing

a frame, but the initial set-up and the work done before and after each frame differ significantly.

Shaders Shaders are programmed in the Nvidia CG language, and compiled on the development PC with a custom SONY Cg compiler. In a traditional render set-up you would load these compiled shader files on the GPU, then tell the GPU/shader the format of any incoming data (Stride/frequency etc...), then send the data, either as an input or as a uniform.

Shader Patching This process changes for the PS Vita. As the GPU uses tile based rendering, to boost performance shaders have to be 'Patched' with the format of the data before they are sent to the tiles to do work. This means that if you have to render two objects, with the same shader, but each object has a slightly different layout of vertices, you would need to patch the shader twice. This isn't a big deal, as it's easy to implement around, but it's one of the many quirks of a tile based GPU.

Rendering Due to the way data is passed around internally in the GPU, and the way it orders it's processing stages, particularly the hidden surface removal pass, data for each rendered object must be kept until every other object is rendered. This seems counter intuitive, but it's the result of some clever processing that in the end results in higher performance.
Even if it is for the best, it still sounds complex to implement, however GXM has got your back here.

Scenes When you fire off a render call (E.G sceGxmDraw or sceGxmSetVertexProgram), this doesn't actually execute, it gets saved into a **Scene** object. Once you have done all you rendering (even though nothing has actually been rendered) you pass this scene off to GXM which replays through it, does some magic and then renders your scene all at once.

3. Conclusion

The vita is an interesting machine, with a lot of power. Sony has shipped many products before this, and as a result the development tools have reached a well refined state. In the past, console developing was a complex task, requiring expert knowledge of every register and memory address. Now deploying an debugging is a seamless experience and the official documentation is fully featured and well written.
The hardware is modern and reactively standard, the ARM instruction set especially. In conclusion, the PS Vita is one of the best consoles to develop for at this point in time.