The Xbox One System on a Chip and Kinect Sensor

The Xbox One entertainment console's system on a chip (SoC) is one of the largest consumer designs to date, with five billion transistors. The Xbox One Kinect image and voice sensor uses time-of-flight technology to provide high-resolution, low-latency, lighting-independent 3D image sensing. Together, Kinect and the SoC provide unique voice and gesture interaction with high-performance games and other entertainment applications.

John Sell
Patrick O'Connor
Microsoft

The Xbox One system on a chip (SoC) incorporates five billion transistors to provide high-performance computation, graphics, audio processing, and audio-video I/O for multiple simultaneous applications and system services. The Xbox One Kinect adds low-latency 3D image and voice sensing. Together, the SoC and Kinect provide unique voice and gesture control.

The Xbox One system (see Figure 1) includes the Kinect image and audio sensors, console, and wireless controllers. It recognizes individual users, who can use voice and movement within many applications, switch instantly between functions, and combine games, TV, and music while interacting with friends via services such as Skype audio and video.

Figure 2 shows a block diagram of the system. The main SoC contains all of the principal computation components. The southbridge chip expands the SoC I/O to access optical disc, hard disc, flash storage, HDMI input, Kinect, and wireless devices.

Main SoC

A single SoC departs from the initial implementations of previous high-performance consoles.[1] One chip enables the most efficient allocation of memory and other resources. It avoids the latency, bandwidth limitations, and power consumption of communicating between computation chips.

Microsoft collaborated with Advanced Micro Devices (AMD) to develop the SoC. Static RAM (SRAM) and GPU circuits with redundancy comprise more than 50 percent of the 370 mm^2 chip, resulting in yield comparable to much smaller designs.

Figure 3 shows the SoC organization. The SoC provides simultaneous system and user services, video I/O, voice recognition, and 3D image recognition.

Significant features include unified, but not uniform, main memory; universal host-guest virtual memory management; high-bandwidth CPU cache coherency; and power islands matching features and performance to active tasks.

Main memory

Main memory consists of 8 Gbytes of low-cost DDR3 external DRAM and 32 Mbytes of internal SRAM. This provides necessary bandwidth while saving power and considerable cost over wider or faster external DRAM-only alternatives.

Published by the IEEE Computer Society

Peak DRAM bandwidth is 68 Gbytes per second. Peak SRAM bandwidth ranges between 109 and 204 Gbytes per second, depending on the mix of transactions. Sustainable total peak bandwidth is about 200 Gbytes per second.

Memory management unit (MMU) hardware maps guest virtual addresses to guest physical addresses to physical addresses for virtualization and security.[2] The implementation increases the size of translation look-aside buffers (TLBs) that it uses to cache fully translated page addresses and uses large pages where possible to avoid significant performance impact from the 2D address translation.

System software manages physical memory allocation. System software and hardware

Figure 1. Xbox One Kinect, console, and wireless controller. The system recognizes individual users, who can use voice and movement to switch between functions; users can also interact with friends through various services, including Skype.

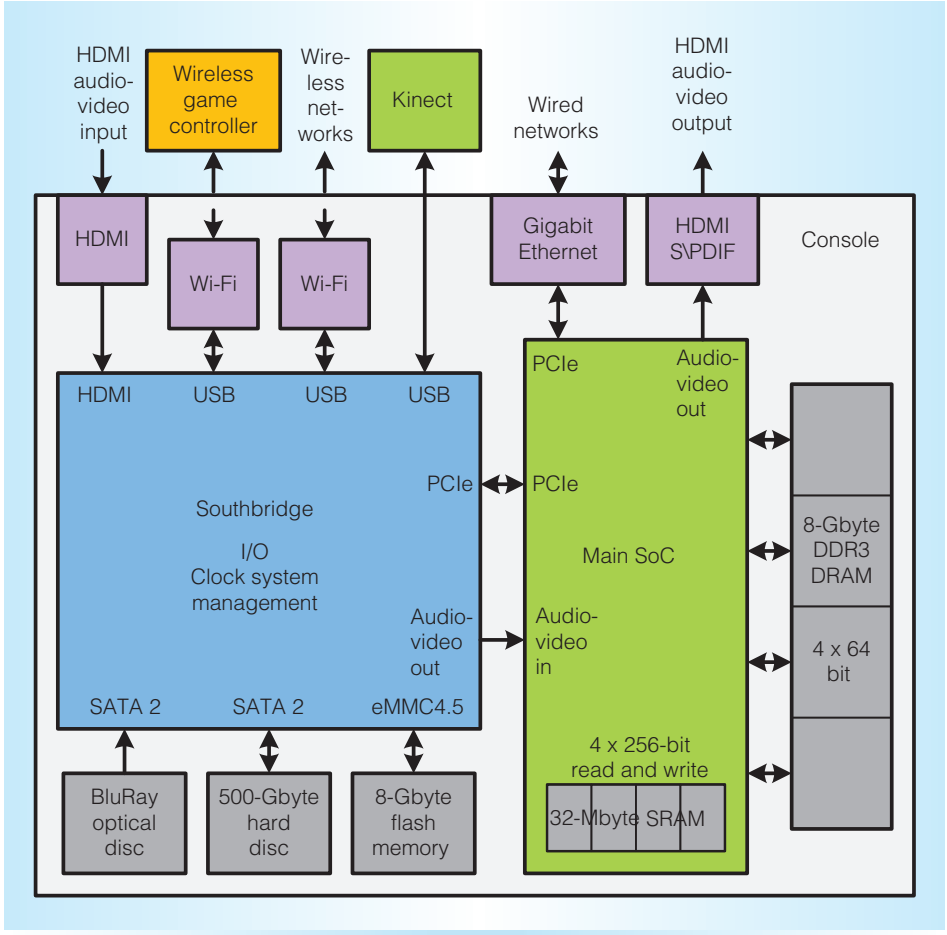

Figure 2. Block diagram of the Xbox One system. The main system on a chip (SoC) contains all the principal computation components.

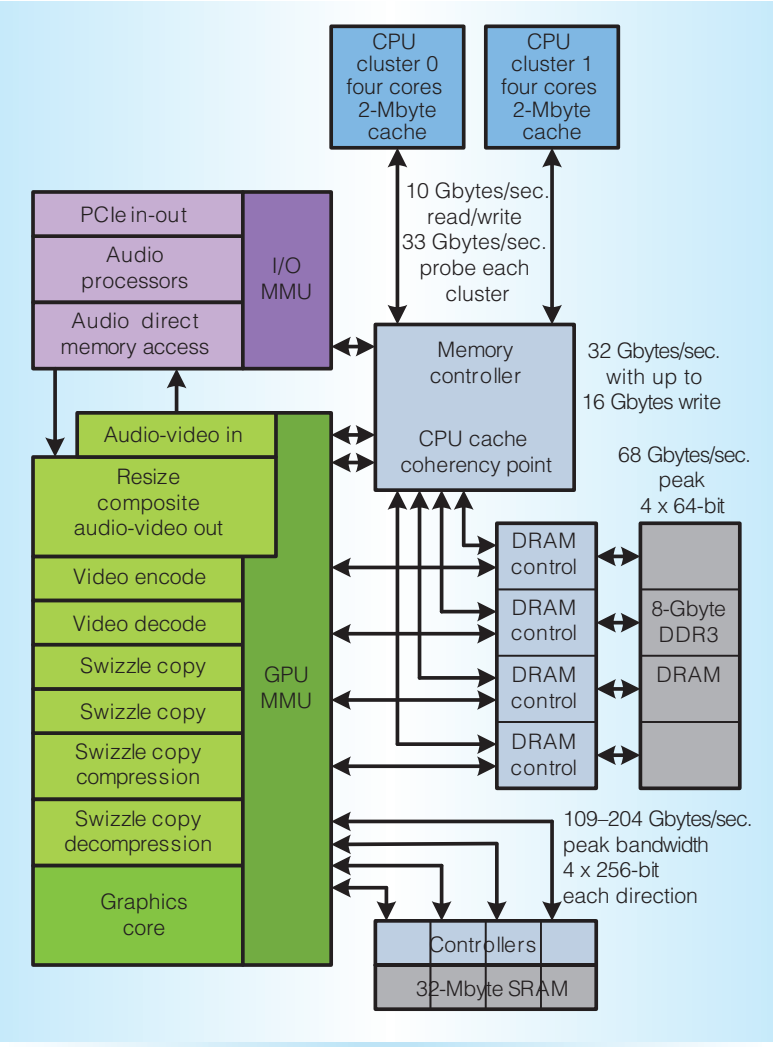

Figure 3. The SoC organization. The SoC provides simultaneous system and user services, video I/O, voice recognition, and 3D image recognition.

keep page tables synchronized so that the CPU, GPU, and other processors can share memory and pass pointers rather than copying data, and so that a linear data structure in a GPU or CPU virtual space can have physical pages scattered in DRAM and SRAM. The unified memory system frees applications from the mechanics of where data is located, but GPU-intensive applications can specify which data should be in SRAM for best performance.

The GPU graphics core and several specialized processors share the GPU MMU, which supports 16 virtual spaces. The design implements the audio processors and audio direct memory access (DMA) as internal PCI devices. They and PCI Express (PCIe) I/O share the I/O MMU, which supports virtual spaces for each PCI bus, device, and function. Each CPU core has its own MMU (CPU access to SRAM maps through a CPU MMU and the GPU MMU).

The design provides 32 Gbytes/second peak DRAM access with hardware-maintained CPU cache coherency for data shared by the CPU, GPU, and other processors. Hardware-maintained coherency improves performance and software reliability.

The implementation restricts shared CPU-cache-coherent data (and PCIe and audio data, most of which is CPU cache coherent) to DRAM for simplification and cost savings. GPU SRAM access and non-CPU-cache-coherent DRAM access bypass CPU cache coherency checking.

CPU

The CPU contains eight AMD Jaguar single-thread 64-bit x86 cores in two clusters of four.[3] The cores contain individual first-level code caches and data caches. Each cluster contains a shared 2-Mbyte second-level cache.

The CPU cores operate at 1,750 MHz in full performance mode. Each cluster can operate at different frequencies. The system selectively powers individual cores and clusters to match workload requirements.

Jaguar provides good performance and excellent power-performance efficiency. The CPU contains minor modifications from earlier Jaguar implementations to support two clusters and increased CPU cache-coherent bandwidth.

GPU

Figure 4 shows the graphics core and the independent processors and functions sharing the GPU MMU. The GPU contains AMD graphics technology supporting a customized version of Microsoft DirectX graphics features. Hardware and software customizations provide more direct access to hardware resources than standard DirectX. They also reduce CPU overhead to manage graphics activity and combined CPU-GPU processing. Kinect makes extensive use of combined CPU-GPU computation.

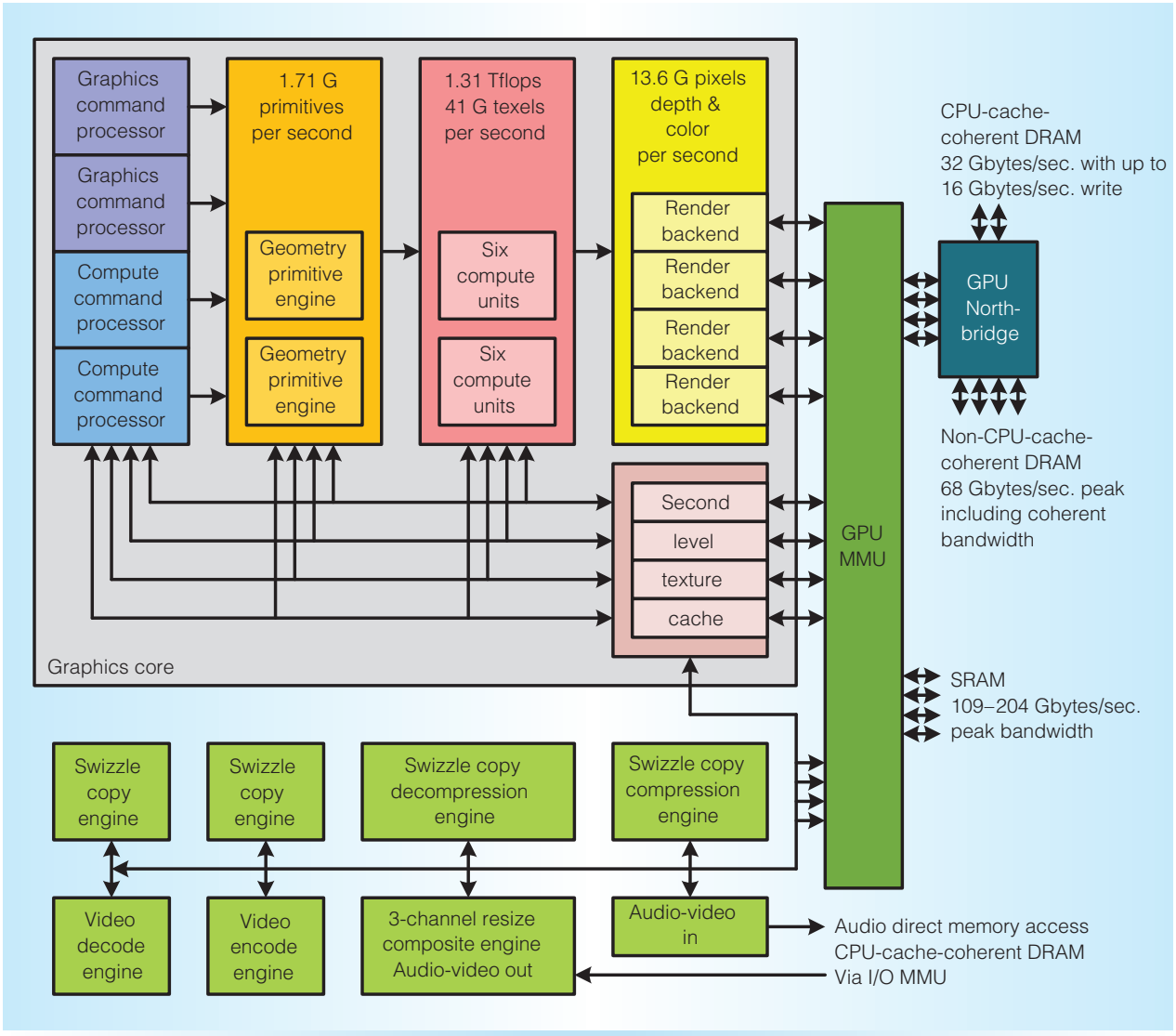

Figure 4. The GPU and independent processors sharing the GPU memory management unit (MMU). The GPU contains Advanced Micro Devices graphics technology supporting a customized version of Microsoft DirectX graphics features.

The graphics core contains two graphics-command and two computation-command processors. Each command processor supports 16 work streams. The two geometry-primitive engines, 12 computing units, and four render-backend depth and color engines in the graphics core support two independent graphics contexts.

The graphics core operates at 853 MHz in full performance mode. System software selects lower frequencies and powers the graphics core and computing unit resources to match tasks.

Independent GPU processors and functions

Eight independent processors and functions share the GPU MMU. These engines support applications and system services. They augment GPU and CPU processing, and are more power-performance efficient at their tasks than the graphics core and the CPU.

Four of the engines provide copy, format conversion, compression, and decompression services. The video decode and encode engines support multiple streams and a range of formats. The audio-video input and

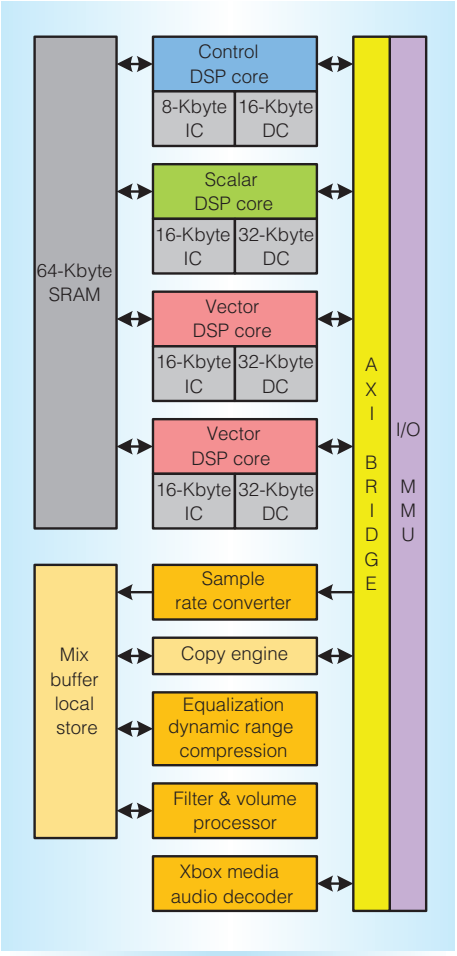

Figure 5. Audio processors. These processors support applications and system services with multiple work queues. Collectively, they are the equivalent of two CPU cores dedicated to audio processing.

output engines support multiple streams, synchronization, and digital rights management. Audio-video output includes resizing and compositing three images and saving results in main memory in addition to the display output.

Audio processors

The SoC contains eight audio processors and supporting hardware (see Figure 5). The processors support applications and system services with multiple work queues. Collectively, they would require two CPU cores to match their audio-processing capability.

The four digital signal processor cores are Tensilica-based designs incorporating standard and specialized instructions. Two include single-precision vector floating-point support, totaling 15.4 billion operations per second. The other four audio processors implement

- sample rate conversion,
- equalization and dynamic range compression,
- filter and volume processing, and
- 512 stream Xbox Media Audio format decompression.

The audio processors use the I/O MMU. This path to main memory provides lower latency than the GPU MMU path. Low latency is important for games, which frequently make instantaneous audio decisions, and for Kinect audio processing.

Xbox One Kinect

The Xbox One Kinect is the second-generation Microsoft 3D image and audio sensor. It is integral to the Xbox One system. The 3D image and audio sensors and the SoC computation capabilities operating in parallel with games and other applications provide an unprecedented level of voice, gesture, and physical interaction with the system.

Image sensor goals and requirements

User experience drove the image sensor goals, which include the following:

- Resolution sufficient for software to reliably detect and track the range of human sizes from young children to large adults (a limiting dimension is a small child's wrist diameter, which is approximately 2.5 cm).
- Camera field of view wide enough for users to interact close to the camera in small spaces and relatively far away in larger rooms.
- Camera dynamic range sufficient for users throughout the space with widely varying clothing colors.
- Lighting independence.
- Stability and repeatability.
- Sufficiently low latency for natural-feeling gesture and physical interaction.

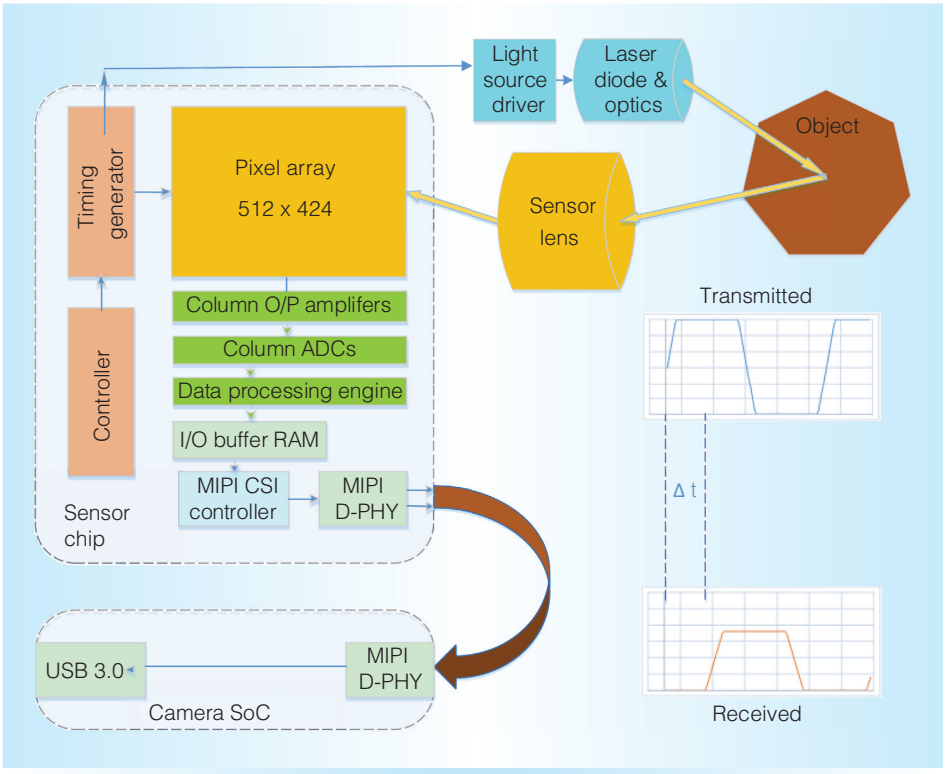

Figure 6. 3D image sensor system. The system comprises the sensor chip, a camera SoC, illumination, and sensor optics.

These goals led to the following key requirements:

- Field of view of 70 degrees horizontal by 60 degrees vertical.
- Aperture F# < 1.1.
- Depth resolution within 1 percent of distance.
- Minimum software resolvable object less than 2.5 cm.
- Operating range from 0.8 to 4.2 meters (m) from the camera.
- Illumination from the camera and operation independent of room lighting.
- Maximum of 14 milliseconds (ms) exposure time.
- Less than 20 ms latency from the beginning of each exposure to data delivered over USB 3.0 to main system software.
- Depth accuracy within 2 percent across all lighting, color, users, and other conditions in the operating range.

In order to meet these challenging and, in some cases, contradictory requirements, Microsoft developed a full custom 3D CMOS image sensor and system based on time-of-flight technology.[4]

Time-of-flight camera architecture

Figure 6 shows the 3D image sensor system. The system consists of the sensor chip, a camera SoC, illumination, and sensor optics. The SoC manages the sensor and communications with the Xbox One console.

The time-of-flight system modulates a camera light source with a square wave. It uses phase detection to measure the time it takes light to travel from the light source to the object and back to the sensor, and calculates distance from the results.

The timing generator creates a modulation square wave. The system uses this signal to modulate both the local light source (transmitter) and the pixel (receiver).

The light travels to the object and back in time Δt. The system calculates Δt by estimating the received light phase at each pixel with knowledge of the modulation frequency. The system calculates depth from the speed of light in air: 1 cm in 33 picoseconds.

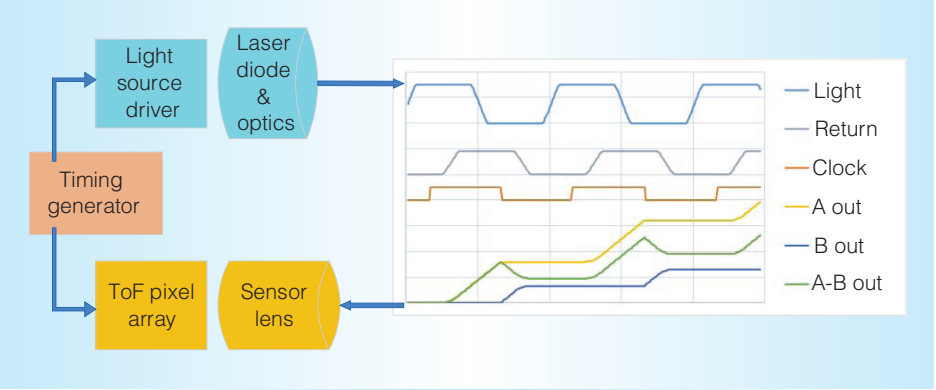

Figure 7. Time-of-flight sensor and signal waveforms. Signals "Light" and "Return" denote the envelope of the transmitted and received modulated light. "Clock" is the local gating clock at the pixel, while "A out" and "B out" are the voltage output waveforms from the pixel.

Differential pixels

Figure 7 shows the time-of-flight sensor and signal waveforms. A laser diode illuminates the subjects. The time-of-flight differential pixel array receives the reflected light.

A differential pixel distinguishes the time-of-flight sensor from a classic camera sensor. The modulation input controls conversion of incoming light to charge in the differential pixel's two outputs. The timing generator creates clock signals to control the pixel array and a synchronous signal to modulate the light source. The waveforms illustrate phase determination.

The light source transmits the light signal. It travels out from the camera, reflects off any object in the field of view, and returns to the sensor lens with some delay (phase shift) and attenuation.

The lens focuses the light on the sensor pixels. A synchronous clock modulates the pixel receiver. When the clock is high, photons falling on the pixel contribute charge to the *A*-out side of the pixel. When the clock is low, photons contribute charge to the *B*-out side of the pixel.

The $(A - B)$ differential signal provides a pixel output whose value depends on both the returning light level and the time it arrives with respect to the pixel clock. This is the essence of time-of-flight phase detection.

Some interesting properties of the pixel output lead to a useful set of output images:

- $(A + B)$ gives a "normal" grayscale image illuminated by normal ambient (room) lighting ("ambient image").
- $(A - B)$ gives phase information after an arctangent calculation ("depth image").
- $\sqrt{(\sum (A - B)^2)}$ gives a grayscale image that is independent of ambient (room) lighting ("active image").

Chip optical and electrical parameters determine the quality of the resulting image. It does not depend significantly on mechanical factors. Multiphase captures cancel linearity errors, and simple temperature compensation ensures that accuracy is within specifications.

Key benefits of the time-of-flight system include the following:

- One depth sample per pixel: $X - Y$ resolution is determined by chip dimensions.
- Depth resolution is a function of the signal-to-noise ratio and modulation frequency: that is, transmit light power, receiver sensitivity, modulation contrast, and lens *f*-number.
- Higher frequency: the phase to distance ratio scales directly with modulation frequency resulting in finer resolution.
- Complexity is in the circuit design. The overall system, particularly the mechanical aspects, is simplified.

An additional benefit is that the sensor outputs three possible images from the same pixel data: depth reading per pixel, an "active" image independent of the room and ambient lighting, and a standard "passive" image based

on the room and ambient lighting. For more information, see work by Bamji et al.[5]

Dynamic range

High dynamic range is important. To provide a robust experience in multiplayer situations, we want to detect someone wearing bright clothes standing close to the camera and simultaneously detect someone wearing dark clothes standing at the back of the play space.

With time of flight, depth resolution is a function of the signal-to-noise ratio at the sensor, where *signal* is the received light power and *noise* is a combination of shot noise in the light and circuit noise in the sensor electronics. We want to exceed a minimum signal-to-noise ratio for all pixels imaging the users in the room, independent of how many users there are, what clothes they are wearing, or where they are in the room.

For an optical system, the incident power density falls off with the square of distance. Reflectivity of typical clothes can vary from more than 95 percent to less than 10 percent. This requires that the sensor show a per-pixel dynamic range in excess of 2,500×.

A photographer can adjust aperture and shutter time in a camera to achieve optimal exposure for a subject. The Kinect time-of-flight system must keep the aperture wide open to minimize the light power required. It takes two images back to back with different but fixed shutter times of approximately 100 and 1,000 μs, and selects the best result pixel by pixel. The design provides nondestructive pixel reading, and light integration involves reading each pixel multiple times to select the best result.

Sensing over long range with fine resolution

The system measures the phase shift of a modulated signal, then calculates depth from the phase using

$$2d = \frac{phase}{2\pi} \cdot \frac{\mathrm{c}}{fmod},$$

where depth is d, c is the speed of light, and $fmod$ is the modulation frequency.

Increasing the modulation frequency increases resolution—that is, the depth resolution for a given phase uncertainty. Power limits what modulation frequencies can be practically used, and higher frequency increases phase aliasing.

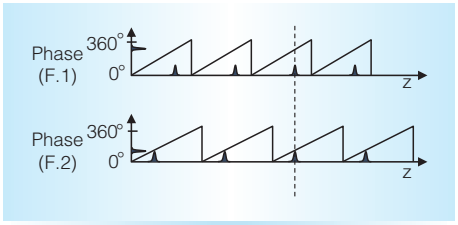

Figure 8. Phase to depth calculation from multiple modulation frequencies. Each individual single-frequency phase result (vertical axis) produces an ambiguous depth result (horizontal axis), but combining multiple frequency results disambiguates the result.

Phase wraps around at 360°. This causes the depth reading to alias. For example, aliasing starts at a depth of 1.87 m with an 80-MHz modulation frequency.

Kinect acquires images at multiple modulation frequencies (see Figure 8). This allows ambiguity elimination as far away as the equivalent of the beat frequency of the different frequencies, which is greater than 10 m for Kinect, with the chosen frequencies of approximately 120 MHz, 80 MHz, and 16 MHz.

Depth image

The GPU in the main SoC calculates depth from the phase information delivered by the camera. This takes a small part of each frame time.

Figure 9 shows a depth image captured at a distance of approximately 2.5 m, direct

Figure 9. Depth image captured at a distance of 2.5 meters. The fine detail of facial features and folds in the clothing are clearly visible.

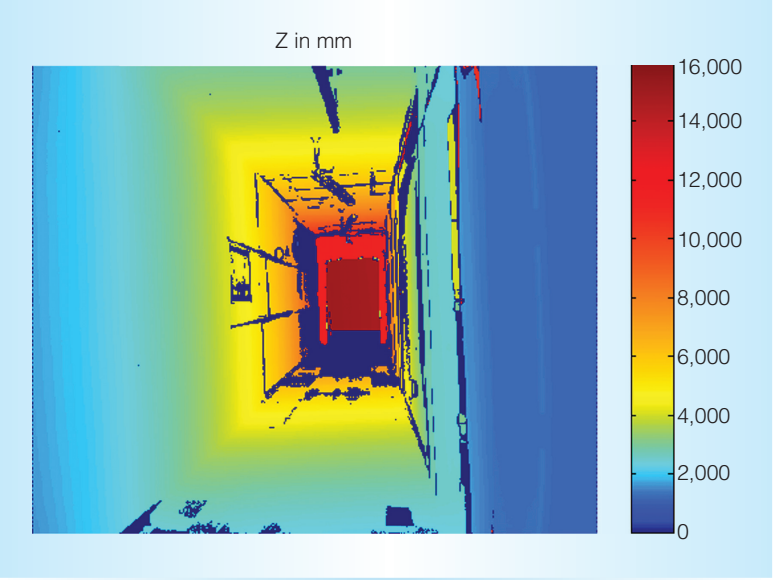

Figure 10. Depth range (denoted by a color map) looking down a long corridor. The depth changes smoothly along the wall, well beyond 10 m, without any trace of aliasing due to phase-wrap.

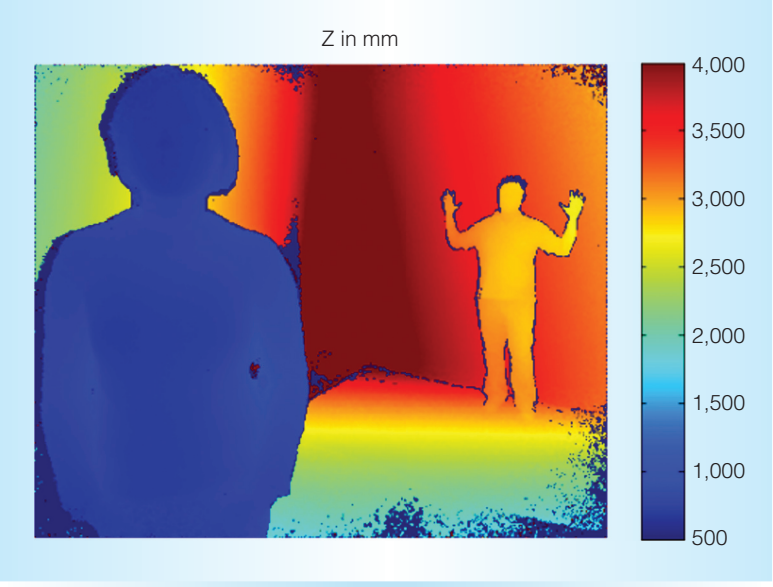

Figure 11. Dynamic range figure recognition. The system captures figures close to the camera and far from the camera clearly.

from the camera, without averaging or further processing. The coloring is a result of test software that assigns a color to each recognized user for engineering use.

Figure 10 illustrates de-aliasing performance. It shows an image of a long corridor. The system obtains smooth depth readings out to 16 m in this example without wrapping.

Figure 11 illustrates the wide dynamic depth range applied to human figure recognition. One figure is close to the camera and the other is far away. The system captures both clearly.

Face recognition

Face recognition is important for a personalized user experience. It is difficult to achieve high-quality results in many situations with normal photography because of the wide variety of room lighting conditions. The photo in Figure 12 shows how room lighting and the resulting shadowing can dramatically change how a person looks to a camera—in this case, from a lamp next to the TV.

Figure 13 shows the same scene captured with the Kinect 3D sensor. The sensor data provides an "active" image that is independent of the wide variation in room lighting.

The resolution from the 3D sensor is lower than the Kinect high-definition RGB (red, green, blue) camera. However, the "active" image more than compensates so that the system can provide robust face recognition to applications.

A five-billion transistor SoC is a large device for a consumer product in 2014, but it provides competitive performance and features without the costs and limitations of a multiple-chip design. It sets the stage to take full advantage of improved cost, power, and performance that future semiconductor processes provide.

The Xbox One's unified main memory supports efficient data sharing between the CPU, GPU, and other processors. The combination of DRAM and SRAM provides the bandwidth necessary for the system's high performance at the lowest cost. Development tool enhancements will continue to improve graphics-intensive application decisions about which data to locate in SRAM.

The combination of the Kinect low-latency 3D image and voice sensing and the computation performance of the SoC enable the system to recognize users as they approach, detect actions and voice commands, and provide a personalized user interface that

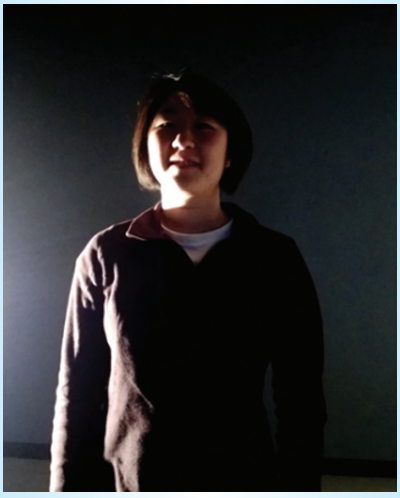

Figure 12. High-contrast ambient lighting situation. Room lighting and shadowing can dramatically change how a person appears to a camera.

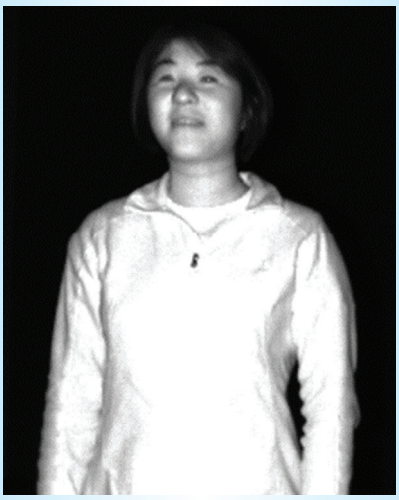

Figure 13. Kinect image in a high-contrast ambient lighting situation. The 3D sensor data provides an image that is independent of variation in the room lighting.

is efficient, simple, and enjoyable. It supports features in applications such as a physical workout that provides real-time, personalized feedback and encouragement while noting improvement from previous sessions. Ongoing development of the technology promises many new user interface and application experiences. MICRO

References

1. J. Andrews and N. Baker, "Xbox 360 System Architecture," *IEEE Micro*, Mar./Apr. 2006, pp. 25-37.
2. *AMD-V Nested Paging*, white paper, Advanced Micro Devices, July 2008.
3. J. Rupley, "AMD's 'Jaguar': A Next Generation Low Power x86 Core," Hot Chips 24, 2012.
4. D. Piatti and F. Rinaudo, "SR-4000 and CamCube3.0 Time of Flight (ToF) Cameras: Tests and Comparison," *Remote Sensing*, vol. 4, no. 4, 2012, pp. 1069-1089.
5. C.S. Bamji et al., "A 512×424 CMOS 3D Time-of-Flight Image Sensor with Multi-Frequency Photo-Demodulation up to 130MHz and 2GS/s ADC," to be published in *Proc. Int'l Solid-State Circuits Conf.*, 2014.

John Sell is a hardware architect at Microsoft, and chief architect of the Xbox One SoC. His research interests include computation architecture, memory systems, quality of service (QoS), and security. Sell has an MS in electrical engineering and computer science from the University of California at Berkeley.

Patrick O'Connor is a senior director of engineering at Microsoft, where he's responsible for hardware and software development of sensors and custom silicon. His research interests include specification and validation of hardware and algorithms for highly functional natural user interfaces. O'Connor has a BS in electrical engineering from Trinity College, Dublin. He is a member of IEEE.

Direct questions and comments about this article to John Sell, Microsoft, 1085 La Avenida, Mountain View, CA 94043; jsell@microsoft.com.