Video Processing Requirements on SoC Infrastructures

Pieter van der Wolf and Tomas Henriksson
NXP Semiconductors Research
Eindhoven, The Netherlands
pieter.van.der.wolf@nxp.com

Abstract
Applications from the embedded consumer domain put challenging requirements on SoC infrastructures, i.e. interconnect and memory. Specifically, video applications demand large storage capacity and high bandwidth while data accesses can be irregular. The SoC architectures used for implementing these applications typically contain a heterogeneous collection of processing elements and use a single interface to off-chip DRAM in order to provide the required storage capacity at a low cost. Proper integration of interconnect and memory architecture is required to achieve the required bandwidths and latencies for accessing memory. The application requirements as well as the characteristics and constraints for accessing memory are key inputs for NoC design. Future memory technologies may cause a paradigm shift by offering high-bandwidth memory access, possibly via multiple memory interfaces.

1. Introduction
Embedded consumer systems that perform advanced media processing typically contain one or more Systems-on-Chip (SoCs) for processing video, audio and graphics. Specifically, video processing poses interesting challenges due to the large data sets that need to be processed, stored, and communicated, while satisfying real-time constraints. The SoCs typically contain a heterogeneous collection of processing elements (PEs), glued together with a SoC infrastructure. This SoC infrastructure contains both interconnect and memory. Networks-on-Chip (NoCs) have emerged as scalable on-chip interconnects that can help to e.g. reduce wire density and offer quality-of-service support for connections [1]. The memory architecture defines the characteristics and constraints for accessing memory by the PEs and has a high impact on the performance, cost and power consumption of a SoC. A key challenge in SoC integration is the definition of a SoC infrastructure that satisfies the communication requirements of the PEs used for executing the application tasks while minimizing cost and power consumption.

The aim of this paper is to present requirements from the embedded consumer domain on SoC infrastructures. Specifically, we investigate video applications and their implementations in order to derive such requirements.

We present a number of key characteristics of video applications in Section 2. Section 3 investigates the architectures used for implementing video applications. In Section 4 we present the requirements on SoC infrastructures. Section 5 briefly looks into future memory technologies and conclusions are drawn in Section 6.

2. Video Applications
There is a variety of video processing functions like decoding, encoding, de-interlacing, picture rate up-conversion, scaling, rotation, noise reduction, sharpness improvement, etc. [2]. The storage requirements and data access patterns of these functions have a huge impact on SoC architectures. Several video functions employ temporal algorithms, which need access to previous frames or fields when processing a frame or field. This is an important characteristic since the previous frames or fields need to be stored in memory. Table 1 lists the required storage for a single frame for various resolutions, video formats, and bit depths.

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Format</th>
<th>Bit depth</th>
<th>Memory size</th>
</tr>
</thead>
<tbody>
<tr>
<td>640 x 480 (VGA)</td>
<td>aRGB</td>
<td>8 bit</td>
<td>1.2 MB</td>
</tr>
<tr>
<td>720 x 576 (SD)</td>
<td>YUV4:2:0</td>
<td>8 bit</td>
<td>0.6 MB</td>
</tr>
<tr>
<td>1920 x 1080 (HD)</td>
<td>YUV4:2:0</td>
<td>8 bit</td>
<td>3.0 MB</td>
</tr>
<tr>
<td>1920 x 1080 (HD)</td>
<td>RGB</td>
<td>12 bit</td>
<td>8.9 MB</td>
</tr>
</tbody>
</table>

Video functions like scaling and rotation have regular access patterns. Video functions like video decoding, de-interlacing, and picture rate up-conversion typically employ motion compensation. Motion compensation is a technique to predict a block of a frame using a block of another frame. A motion vector determines which block is used. Standards like H.264 have few limitations on these motion vectors, possibly leading to highly irregular access patterns.

Bandwidth requirements for video applications are typically high and will further increase as resolutions, bit depths, and frame rates increase. For example, a SD video stream of 25 frames/s consumes 15.6 MB/s, while a HD RGB 12bit stream at a frame rate of 120Hz consumes 1120 MB/s.

3. Architectures
Video processing applications are implemented by partitioning them into coarse-grain tasks that are mapped onto PEs. The PEs can be function-specific hardware or programmable processors. The programmable processors typically have caches and are used for control-dominated parts or when flexibility is required. The PEs are integrated by connecting them to a SoC infrastructure, typically by means of a load / store interface, with dedicated wires for address and data. Example interfaces are AMBA-AXI and OCP.
SoCs for video processing typically use a unified off-chip DRAM to provide the required storage capacity. For example, HD H.264 decoding requires 15 MB for frame memories (YUV4:2:0, 8bit). Implementing such amounts of on-chip memory is typically not cost effective. On-chip SRAM is about 7mm² per MB in 65nm technology. Embedded DRAM has a higher density but requires additional mask layers for manufacturing, adding to cost.

Cost constraints further require a low pin-count, as this impacts the packaging cost. Therefore SoCs typically contain a single DRAM interface of the lowest data width possible. As an example, the gross bandwidth of a 64-bit DDR2-400 DRAM interface is 3.2GB/s. However, DRAMs need time to setup a data transfer, and this time can not always be overlapped with another data transfer, e.g. due to a bank conflict. Therefore, DRAM accesses need to have large burst sizes, e.g. 128B, in order to use the DRAM interface efficiently. Whether all this data is actually needed depends on the spatial locality of the data accesses. The sizes of the DRAM bursts need to be aligned with the line boundaries, more data has to be accessed than really needed. Motion compensation traffic for H.264 decoding is typically in the range of 100-200MB/s for SD resolution and 500-900MB/s for HD resolution. Due to the irregularity in the accesses, caches are not effective in reducing the bandwidth to off-chip DRAM for worst-case video sequences, although they help for the average case. For SoCs that integrate a set of video functions, the accumulated bandwidth to off-chip DRAM is typically in the 1-6GB/s range.

4. Requirements on SoC Infrastructures

Considering the characteristics of video processing and the constraints imposed by PEs and DRAM, we can derive the following requirements for SoC infrastructures, and NoCs in particular. A large portion of the traffic is load/store traffic for accessing frame memories. This traffic consists of requests and responses. Upon a load operation this is a small request, whereas the response is a data burst. Upon a store operation the request consists of an address and a data burst, whereas an acknowledgement may be returned to the PE.

The memory accesses are directed to a single target, namely the memory interface to off-chip DRAM. Fitting the huge bandwidth requirements in the bottleneck to off-chip DRAM, while satisfying requirements on latency in a predictable way, is a formidable design challenge.

Function-specific PEs that perform regular processing typically have a latency constraint for individual memory accesses, but the latency does not have to be short if a small buffer on the side of the PE is used. For some PEs the latency must be guaranteed under worst-case conditions. Memory accesses issued by programmable processors are typically irregular. They can be cache refills, prefetches or copybacks issued by the cache controller, with a burst size equal to the cache block size. Especially the cache refills require low latency, since the processor may be stalling for the result. However, low latency does not have to be guaranteed for every individual memory access. The deadlines of a task on a programmable processor are typically in the milliseconds range, and hence a total latency must be guaranteed for multiple memory accesses in a time window [3]. These latencies need to be guaranteed by the complete infrastructure, with arbitration occurring both in the NoC and in the DRAM arbiter. Therefore the NoC and the DRAM arbiter need to be well-aligned. For example, upon a load operation, NoC arbitration should allow the data burst from DRAM to travel as fast as possible if the requesting PE is waiting for the (last word of the) data burst [4].

5. Future Memory Technologies

Future memory integration technologies may cause a paradigm shift that helps to remove the bottleneck to off-chip DRAM. For example, technologies such as Z-RAM [5] and STT-RAM promise to offer densities for embedded memory of a factor 2.5-5 higher than eDRAM. Another direction is to keep memory on a separate die, but integrate it with the logic die via 3D stacking, either using microbumps or through-silicon vias. These memory integration technologies may allow a large number of connections, possibly with multiple memory interfaces. However, there are still issues, concerning e.g. technology, manufacturing, standardization and cost, which need to be resolved before such memory integration technology can be applied on a wide scale in the embedded consumer domain.

6. Conclusions

SoC architectures for video processing are dominated by the use of unified off-chip DRAM. Most bandwidth is consumed by PEs performing bursty memory accesses on the single memory target. Caches should be applied selectively. The SoC infrastructure must guarantee that different kinds of latency constraints are met. Storage and bandwidth requirements are expected to further increase in the future. New memory technologies could have a large impact on SoC architectures for video processing.

References