Using GStreamer for hardware accelerated video decoding on Intel® Atom™ Processor E6xx series

September 2010
## Contents

Executive Summary .................................................................................................................. 3  
Background............................................................................................................................. 4  
Video Decode Engine............................................................................................................. 5  
Intel® Embedded Media and Graphics Drivers....................................................................... 7  
GStreamer Overview.............................................................................................................. 7  
Configuring GStreamer for Hardware Acceleration............................................................... 8  
Summary............................................................................................................................... 10  
References............................................................................................................................. 11
Executive Summary

This paper introduces the video decode hardware acceleration capability of the Intel® Atom™ processor E6xx series [1] System-on-Chip. A new breed of graphics drivers named Intel® Embedded Media and Graphics Driver supports the platforms based on the Intel® Atom™ processor E6xx series SoC running Linux based Operating Systems such as MeeGo* [2] and Fedora*.

The GStreamer framework allows various multimedia applications to playback high definition videos on the Intel® Atom™ processor E6xx series SoC with low power requirements and CPU utilization.

This paper focuses on the software stack needed to enable the hardware accelerated video decode capability of the SoC. GStreamer [3] and other multimedia components compatible with MeeGo/Fedora including Intel® EMGD [6], that are part of the video decode software stack are presented in detail. The GStreamer framework allows various multimedia applications to playback high definition videos on the Intel® Atom™ processor E6xx series SoC with low power requirements and CPU utilization.


Background

Intel recently unveiled the Intel® Atom™ processor E6xx series SoC for various market segments like IP Phones, Printers and In-Vehicle Infotainment systems for cars. The Intel® Atom™ processor E6xx series solution which focuses on delivering internet and mobile experience to consumers is a single chip solution combining an Intel® Atom™ processor, the memory controller hub, graphics engine and video engine. With benefits such as Intel architecture, low carbon footprint and flexible solution which helps in reducing bill of materials and board real estate for embedded applications, the Intel® Atom™ processor E6xx series also enables consumers to playback High Definition video(s) on their embedded devices.

The demand for high definition videos on the embedded device has exploded in recent times with connectivity to Internet and availability of content with great graphic details. To meet this demand and the intense computing requirements of video processing algorithms, the Intel® Atom™ processor E6xx series video engine has dedicated hardware accelerator support to decode high definition video in real-time while doing so with a very low power requirement.

GStreamer is a multimedia framework that provides a unified approach to handling various types of videos from different sources. It also hides the OS, hardware and other platform components from the multimedia applications ranging from video playback, video chat to video streaming etc. Intel provides Media Infrastructure Accelerator (MI-X), a proprietary solution, to be used in the GStreamer multimedia framework as a plug-in. MI-X libraries hardware accelerates the GStreamer video decode and rendering process by processing and offloading the data to the Intel® Atom™ processor E6xx series video engine through the Intel EMGD driver. The communication between MI-X and Intel EMGD is based on industry standard VAAPI [5] specification.
Other multimedia framework like FFmpeg [4] that the Intel® Atom™ processor E6xx series SoC supports will not be discussed in this paper. A difference in performance exists when choosing one framework over the other. GStreamer framework is also supported for hardware accelerated Video Encoding on the Intel® Atom™ processor E6xx series SoC discussion of which is beyond the scope of this paper. This paper aims at introducing basic GStreamer concepts and how the MI-X plug-in is used within GStreamer. This paper will serve as a basic guide to enable hardware accelerated video playback along with information on software stack logistics.

### Video Decode Engine


Table 1 summarizes the video codec profile/level/maximum resolution/maximum frame rate supported by the Intel® Atom™ processor E6xx series SoC for each codec standard accelerated in hardware.

<table>
<thead>
<tr>
<th>Format</th>
<th>Profile</th>
<th>Levels</th>
<th>Max Resolution</th>
<th>Max Frame Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>H.264</td>
<td>Baseline Profile</td>
<td>L1,L1.2,L1.3,L2,L2.2,L3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H.264</td>
<td>Main Profile</td>
<td>L1,L1.2,L1.3,L2,L2.2,L3,3.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H.264</td>
<td>High Profile</td>
<td>L1,L1.2,L1.3,L2,L2.2,L3,3.1</td>
<td>1080p 720p</td>
<td>30 fps 60 fps</td>
</tr>
<tr>
<td>MPEG-4</td>
<td>Simple Profile</td>
<td>L0,L1,L2,L3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPEG-4</td>
<td>DivX HD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPEG-4</td>
<td>Advanced</td>
<td>L0,L1,L2,L3,5</td>
<td>720p</td>
<td>30 fps</td>
</tr>
</tbody>
</table>
Using GStreamer for Hardware accelerated video decoding on Intel® Atom™ Processor E6xx

<table>
<thead>
<tr>
<th>Format</th>
<th>Profile</th>
<th>Levels</th>
<th>Max Resolution</th>
<th>Max Frame Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>VC-1</td>
<td>Simple Profile</td>
<td>LL,ML</td>
<td>1080p 720p</td>
<td>30 fps 60 fps</td>
</tr>
<tr>
<td>VC-1</td>
<td>Main Profile</td>
<td>LL,ML,HL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VC-1</td>
<td>Advanced Profile</td>
<td>L0,L1,L2,L3</td>
<td>1080p 720p</td>
<td>30 fps 60 fps</td>
</tr>
<tr>
<td>WMV9</td>
<td>Simple Profile</td>
<td>LL,ML</td>
<td>1080p 720p</td>
<td>30 fps 60 fps</td>
</tr>
<tr>
<td>WMV9</td>
<td>Main Profile</td>
<td>LL,ML,HL</td>
<td>1080p 720p</td>
<td>30 fps 60 fps</td>
</tr>
<tr>
<td>MPEG-2</td>
<td>Simple Profile</td>
<td>Main Level</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPEG-2</td>
<td>Main Profile</td>
<td>Low, Main, High1440, High Level</td>
<td>1080p 720p</td>
<td>30 fps 60 fps</td>
</tr>
</tbody>
</table>

NOTES:
1. L3.1 may be supported where the toolset use is those common to both Baseline and Main Profile.
2. At L5 resolution, only Simple Profile Toolset is supported.
3. DivX is based on MPEG4 Advanced simple profile but ignores the levels defined by MPEG4. There are two variants of DivX. The “certified” version does not require GMC or quarter pixel motion compensation prediction. The “non-certified” does support these features.

Note that the ability to play video at the stated resolution and frame rates is subject to the capabilities of the rest of the software stack and choice of multimedia framework. We recommend using VLD entry point for MPEG-2, H.264, VC-1, WMV9 and MPEG-4 codec standards which is supported by Intel® Atom™ processor E6xx series SoC.
Intel® Embedded Media and Graphics Drivers

Intel EMGD is enhanced over previous Intel Embedded Graphics Drivers, providing higher performance and improved usability. Among the numerous features like advanced display configuration settings, and 3D graphics acceleration, Intel EMGD also supports Hardware Video Decode Acceleration. Primarily, the Intel EMGD acts as a liaison between the Intel® Atom™ processor E6xx series Video Decode/Display Engine and VAAPI. VAAPI defines the scheme to pass various types of data buffers from the application and codec(s) to a video acceleration driver, such as, Intel EMGD.

GStreamer Overview

The GStreamer framework is used to build multimedia applications like video phone, video conferencing, media player and streaming video etc. GStreamer is based on a plug-in model where 100s of plugins in its project database fit into the framework and provide audio/video codecs and other functionalities. The plugins which can be classified as protocols, sources, formats, codecs, filters and sinks can be linked together to form a pipeline. The pipeline defines the flow of the data as it passes through various categories of plugins. An example of protocol could be a video file stored locally on the system or http: streaming. A sink is an object that accepts the data e.g. decoded video bit-stream from GStreamer pipeline going to the display acting as a sink, or compressed video as output from video encoding pipeline getting transmitted to the network as a sink. 3rd party plugins such as MI-X provided by Intel can also act as decoder filters for various video types (MPEG-4, VC-1 and H.264) or can act as a sink to display video decoded from the hardware.
Without going into the technical details of GStreamer, a video pipeline example is shown in Figure 1 which lets the media player application connect a file source protocol, with a dvd demux format, video only output of which is sunk into an mpeg2 decoder. After the video data gets decoded it gets accepted by a sink element such as xvimagesink which is responsible for displaying the video onto the screen connected with the platform running a Linux system with X window system on it.

![Figure 1: GStreamer pipeline for a basic video player](image)

**Configuring GStreamer for Hardware Acceleration**

With capabilities of hardware video decode acceleration provided by the Intel® Atom™ processor E6xx series Video Decode Engine and Intel EMGD, GStreamer can use the full potential of the platform by following the simple process outlined below. First we will list the components needed to configure
GStreamer for hardware accelerated video decode and then in the next steps the paper will illustrate sample instructions to playback a video. Make sure to install all the OS specific GStreamer packages on your system before proceeding.

**Components Needed**

**Media Player:**

A media player such as a GUI based Totem player or command-line based gst-launch tool available as part of GStreamer package can be used to create a player pipeline automatically or manually by providing options to the player.

**Source:**

A Movie file conformant with GStreamer supported container format and codec supported in Table 1 is required as a source to be played.

**3rd Party GStreamer Plugins (MI-X):**

MI-X plugins provided by Intel along with the Intel EMGD package constitutes a GStreamer decoder element and GStreamer sink element which will be responsible for communicating with Intel EMGD via the industry standard VA-API specification. The MixVideoDecoder element will parse the compressed elementary video bit-stream and send decode specific data to Intel EMGD for handling.

**MI-X licensing:**

MI-X is provided by Intel as a proprietary solution for evaluation purposes. If customers wish to use it in a production solution, they should consult with their legal department on the need of a codec license for permissible use.
Creating a GStreamer Pipeline

As shown in Figure 1, in order to enable hardware acceleration, the example
declare and sink blocks will be replaced by MixVideoDecoder and
MixVideoSink element respectively.

A Media File can be played by building the GStreamer pipeline using
command line utility gst-launch

An example of playing hardware accelerated VC-1 video clip using
gst-launch:

gst-launch filesrc location=./sample.wmv ! asfdemux ! queue !
MixVideoDecoder ! MixVideoSink

An example of playing hardware accelerated H.264/MPEG-4 part 2
video clip using gst-launch:

gst-launch filesrc location=./sample.mp4 ! qtdemux ! queue !
MixVideoDecoder ! MixVideoSink

An example of playing hardware accelerated DivX video file using
gst-launch:

gst-launch filesrc location=./sample.avi ! avidemux! queue !
MixVideoDecoder ! MixVideoSink

Playback controls are not available when using gst-launch.

Summary

This paper introduced the concept of hardware accelerated video decoding on
the Intel® Atom™ processor E6xx series SoC platform using GStreamer
Framework on MeeGo/Fedora OS. The role of Intel EMGD and MI-X libraries is
critical to enable the outlined solution. Details on all the components needed
and steps to configure GStreamer to enable hardware accelerated video
deck were outlined with help of sample instructions. For detailed
instructions users should refer to Intel EMGD User’s Guide.
Using GStreamer for Hardware accelerated video decoding on Intel® Atom™ Processor E6xx

References

Author

Abhishek Girotra is a Graphics Software TME with Embedded and Communications Group (ECG) at Intel Corporation.

Acronyms

API
CPU Central Processing Unit
EMGD Embedded Media and Graphic Drivers
IP Internet Protocol
MI-X Media Infrastructure Accelerator
OS Operating System
SoC System-on-Chip
VAAPI Video Acceleration Application Programming Interface
VLD Variable Length Decoding
Using GStreamer for Hardware accelerated video decoding on Intel® Atom™ Processor E6xx

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL’S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. This paper is for informational purposes only. THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Intel disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.


*Other names and brands may be claimed as the property of others.

Copyright © 2010 Intel Corporation. All rights reserved.