



Proposal for V4L2 workshop by Samsung S.LSI



SW solution team
2011. 03. 14.

System LSI
Semiconductor Division
Samsung Electronics

Agenda

□ Proposals

- **Adding new compressed format**
- **Enhancement of V4L2 control framework**
- **How to control FB device as V4L2 sub-device?**
- **Enhanced TV-Out Driver**

□ Q & A

- **Sensor/Flash/Snapshot functionality**
- **Frame-size switching : preview/snapshot**
- **Entity information ioctl**

Adding new compressed format

FOURCC ?

Related emails

- <http://www.spinics.net/lists/linux-media/msg29822.html>

What is FOURCC?

- FOURCC is short for "four character code" - an identifier for a video codec, compression format, color or pixel format used in media files. < in the fourcc.org>

Current Header File In Videodev2.h

- ❑ MPEG has been only defined as a compressed format

```
/* compressed formats */  
#define V4L2_PIX_FMT_MJPEG      v4l2_fourcc('M', 'J', 'P', 'G') /* Motion-JPEG      */  
#define V4L2_PIX_FMT_JPEG      v4l2_fourcc('J', 'P', 'E', 'G') /* JFIF JPEG      */  
#define V4L2_PIX_FMT_DV        v4l2_fourcc('d', 'v', 's', 'd') /* 1394            */  
#define V4L2_PIX_FMT_MPEG      v4l2_fourcc('M', 'P', 'E', 'G') /* MPEG-1/2/4     */
```

Proposal

- Add new FOURCC type of each codec
 - We need to define new fourcc type of each codec based on codec characteristics

```
/* compressed formats */
#define V4L2_PIX_FMT_MJPEG    v4l2_fourcc('M', 'J', 'P', 'G') /* Motion-JPEG */
#define V4L2_PIX_FMT_JPEG    v4l2_fourcc('J', 'P', 'E', 'G') /* JFIF JPEG */
#define V4L2_PIX_FMT_DV      v4l2_fourcc('d', 'v', 's', 'd') /* 1394 */
#define V4L2_PIX_FMT_MPEG    v4l2_fourcc('M', 'P', 'E', 'G') /* MPEG-1/2/4 */

#define V4L2_PIX_FMT_H264    v4l2_fourcc('H', '2', '6', '4') /* H264 */
#define V4L2_PIX_FMT_H263    v4l2_fourcc('H', '2', '6', '3') /* H263 */
#define V4L2_PIX_FMT_MPEG12  v4l2_fourcc('M', 'P', '1', '2') /* MPEG-1/2 */
#define V4L2_PIX_FMT_MPEG4    v4l2_fourcc('M', 'P', 'G', '4') /* MPEG-4 */
#define V4L2_PIX_FMT_DIVX    v4l2_fourcc('D', 'I', 'V', 'X') /* DivX */
#define V4L2_PIX_FMT_DIVX3    v4l2_fourcc('D', 'I', 'V', '3') /* DivX 3.11 */
#define V4L2_PIX_FMT_DIVX4    v4l2_fourcc('D', 'I', 'V', '4') /* DivX 4.12 */
#define V4L2_PIX_FMT_DIVX500  v4l2_fourcc('D', 'X', '5', '2') /* DivX 5.00 - 5.02 */
#define V4L2_PIX_FMT_DIVX503  v4l2_fourcc('D', 'X', '5', '3') /* DivX 5.03 - x */
#define V4L2_PIX_FMT_XVID    v4l2_fourcc('X', 'V', 'I', 'D') /* Xvid */
#define V4L2_PIX_FMT_VC1      v4l2_fourcc('V', 'C', '1', 'A') /* VC-1 */
#define V4L2_PIX_FMT_VC1_RCV  v4l2_fourcc('V', 'C', '1', 'R') /* VC-1 RCV */
```

Why do we need to add new codec type ?

□ Because

- By media container parser, exact pixel format info. is decided
- By VIDIOC_S_FMT,
 - pixelformat should be transferred to the driver
- MFC HW codec should know specific format information

Let's use FOURCC as much as possible

□ In the fourcc.org

In the fourcc.org

FOURCC ▲	Name/Info Link ↕	Owner ↕	Description, Products Using the Codec, etc. ↕
avc1	H.264	Apple	Apple's version of the MPEG4 part 10/H.264 standard apparently.
X264	H.264	?	This FOURCC was originally registered by a company called XiWave but their web presence has disappeared. I am now informed that it is used by a popular open source H.264 implementation.
SEDC	Samsung MPEG-4	Samsung	MPEG-4 hardware and software codec used in Samsung digital video products.
WVC1	SMPTE VC1	Microsoft	Microsoft's implementation of the SMPTE VC1 codec.
XVID	XVID MPEG-4	XVID	Codec is available in source form from XVID web site. Can also be downloaded as part of the Gordian Knot Codec Pack. This FOURCC is also supported by LEAD's MCMP Codec.
DIV3	DivX MPEG-4	DivX	Low motion codec (optimised for low motion source material?). Several sources tell me that this is an old and illegal codec that should not be used to encode new material.
DIV4	DivX MPEG-4	DivX	Fast motion codec. Several sources tell me that this is an old and illegal codec that should not be used to encode new material.
DIV5	?	?	Apparently almost as old as DIV3 and DIV4. Changing DIV5 AVI's FOURCC to DIV3 or DIV4 seems to allow them to play just fine.
DIVX	DivX	OpenDivX	This FOURCC code is used for versions 4.0 and later of the DivX codec. DivX, "the MP3 of video," is the popular and market-leading MPEG-4 video codec that is emerging as the standard for full screen, full motion, DVD-quality video over IP-based networks. Apparently version 5 also encodes using FOURCC DX50 . The Microsoft codec pack available here apparently supports Divx 6. This FOURCC is supported by LEAD's MCMP Codec.

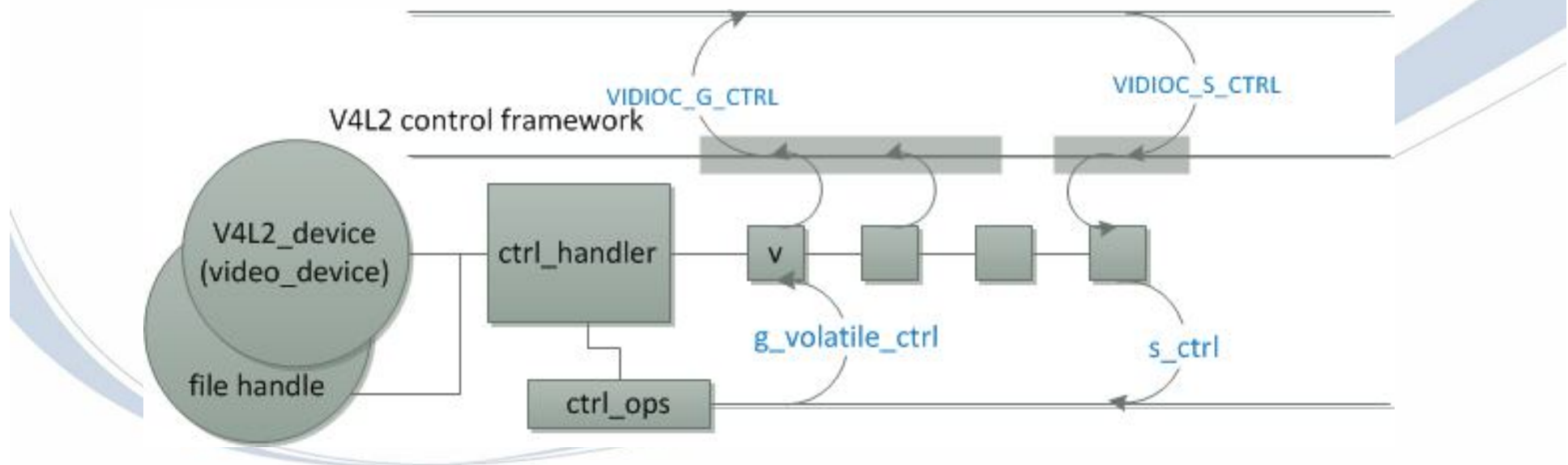
Enhancement of V4L2 Control Framework

Current Operation Flow

❑ Related Emails

- <http://www.spinics.net/lists/linux-media/msg27975.html>

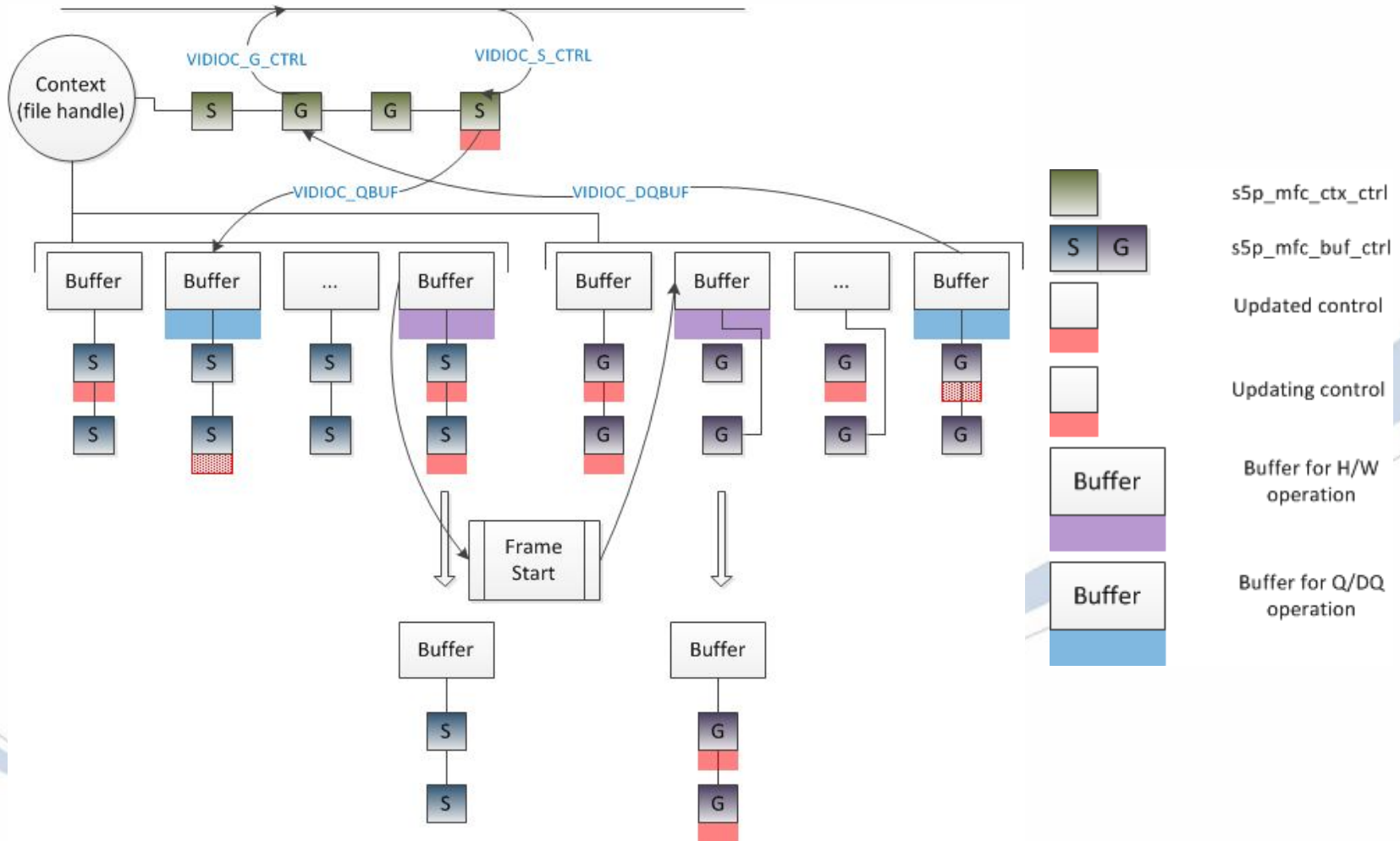
❑ Current operation flow



Limitation of Current Framework

- ❑ Current framework only support **per-device node** or per-filehandle operation(if there is small change)
- ❑ But, some devices need to set or get control value **per-buffer based**
 - ❑ Ex> In the M2M device, some configuration is required based on per-buffer
 - ❑ frametag in decoder (S_CTRL, G_CTRL)
 - ❑ I frame period, bitrate/framerate (S_CTRL)

Newly Required Operation



Proposal

□ Add the code for operation per buffer control

● Support per-filehandle as a pre-requisite

- **Hans agreed to support per-filehandle such as add ctrl_handler to 'struct v4l2_fh'**
- **We just started to apply per-filehandle to our codec driver**

● Add per-buffer control handling routine

- **Should this be included in the V4l2 ctrl framework ?**
Probably or maybe not
- **Alternative1: (Videobuffer2 + Driver) or Driver only**
We tried to implement it in the driver Only, it works,
- **But Laurent mentioned the way using frame metadata**

How to control Frame Buffer device as V4L2 sub-device

Subdev pool proposed by SPRC

- ❑ **Acquiring subdevs from other devices using subdev pool**
 - <http://www.mail-archive.com/linux-media@vger.kernel.org/msg21831.html>
 - subdev pool uses its own list to save the *sd.
 - It simplifies V4L2 subdevice management by introducing a pool of named subdevs.

Conflict of Platform driver data

- ❑ **Current sub device pointer is saved in platform driver data using `v4l2_set_subdevdata` to find the `*sd` in other driver**

- ❑ **Some platform driver uses the platform driver data for its own purpose.**
 - **For example, display controller can be the source of Capture interface**
 - **The FB driver embed `*sd` inside a larger struct which is saved in a larger struct using `platform_set_drvdata`.**
 - **Conflict: `v4l2_set_subdevdata` and `platform_set_drvdata` also uses `pdev->dev->p->driver_data`.**

Proposal

- ❑ **There are some solutions but it doesn't look nice.**
 - **Because the main purpose of driver_data is for the platform driver not for v4l2 host driver.**

Enhanced TV-Out Driver

Why is frame buffer interface required ?

□ Recent Trend

- From Market

- TV became one of LCDs

- From Customers

- They want to use frame buffer interface for TV out

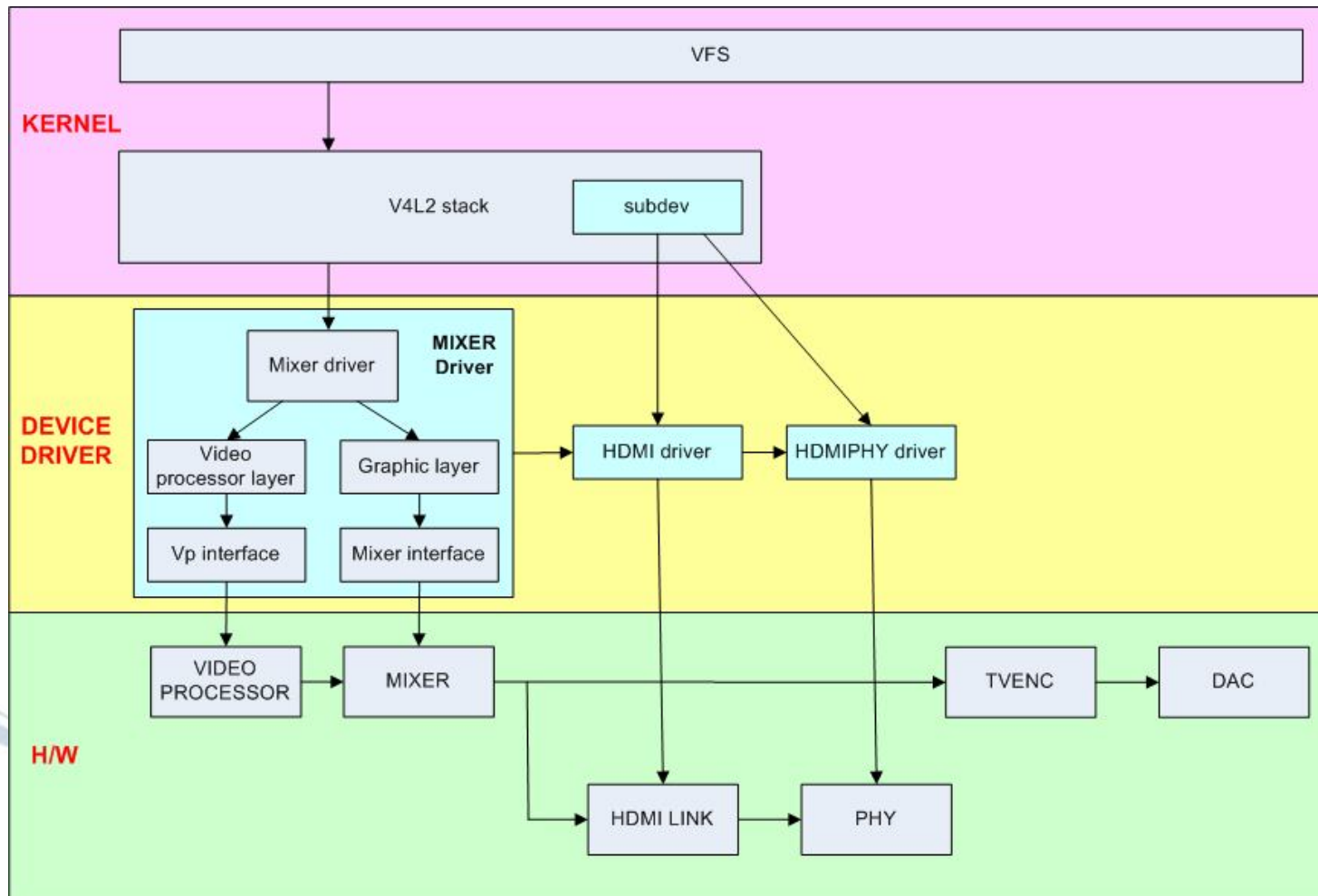
- From Chrome & LiMO(SLP)

- Platforms based on X window use frame buffer interface when they use TV as output device



TV-Out Driver From SPRC (1 / 2)

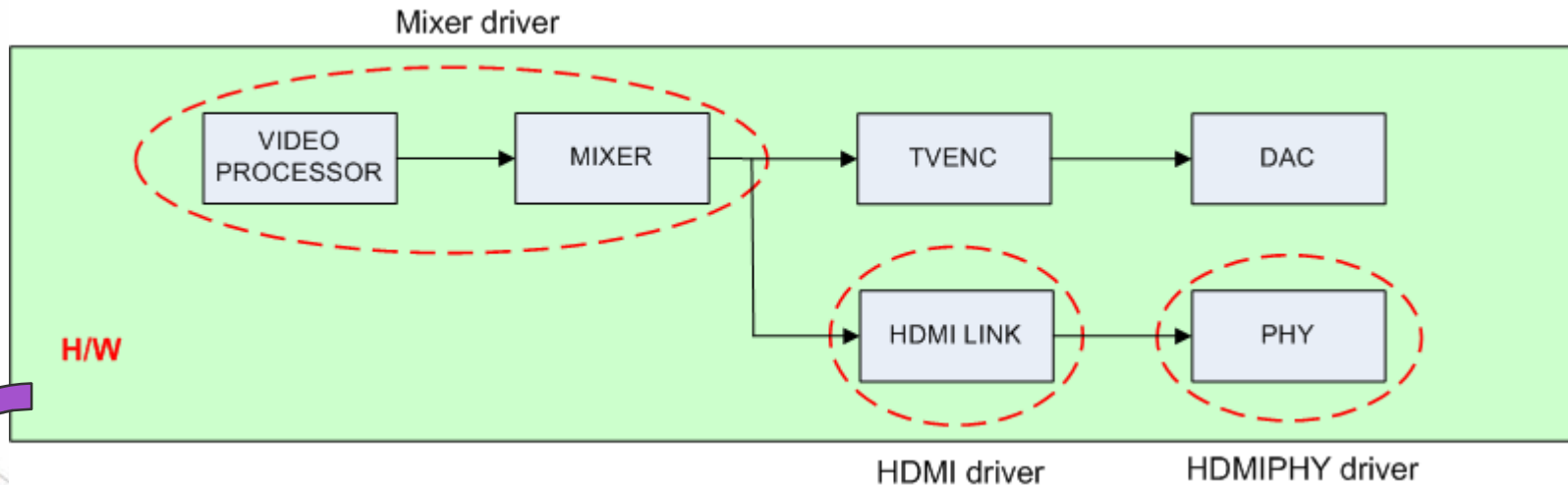
□ Architecture



TV-Out Driver From SPRC (2 / 2)

□ Limitation

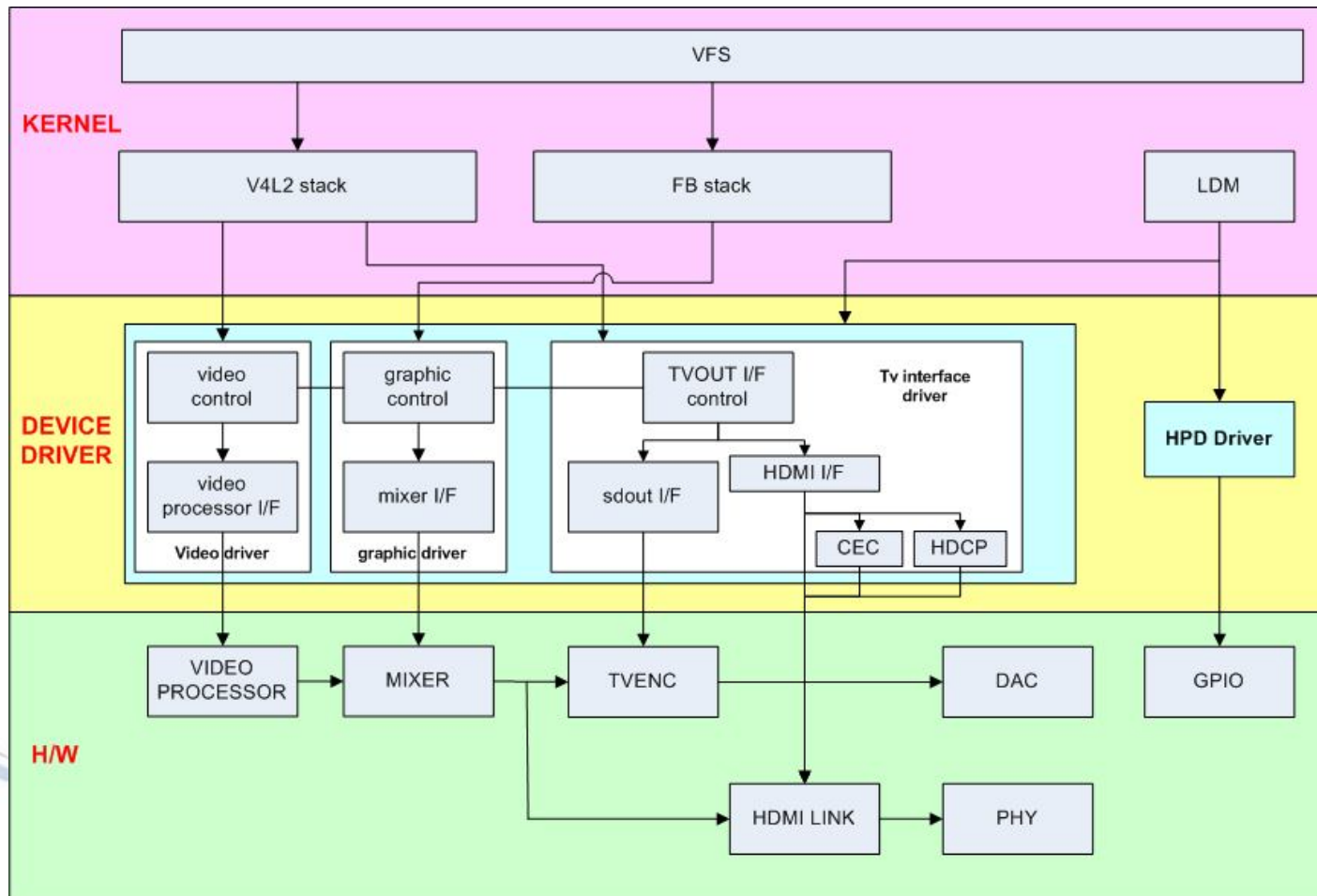
- Mixer driver uses only V4L2 interface
- VP and Mixer are tightly coupled for Mixer driver
- Mixer driver can't support frame buffer interface



It's difficult to meet requirements from market & customers

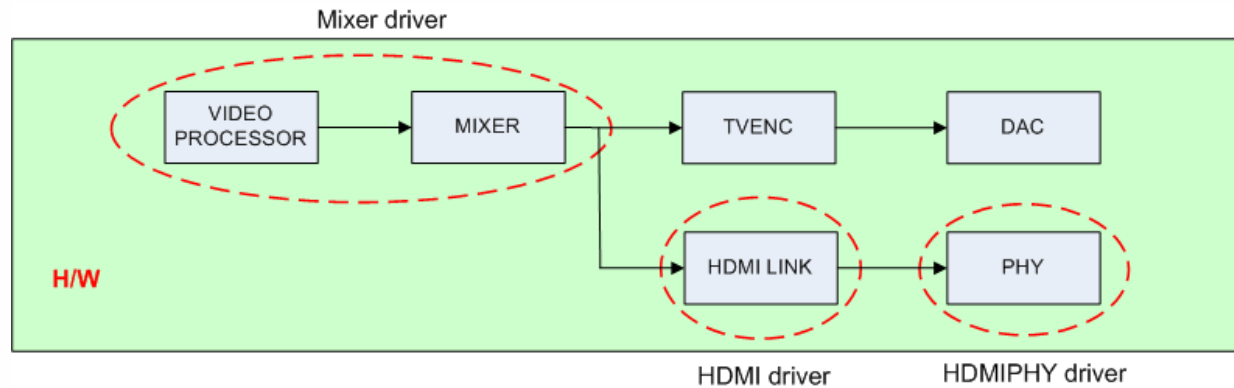
Proposal For TV-Out Driver (1 / 2)

□ Architecture of S.LSI's TV out driver

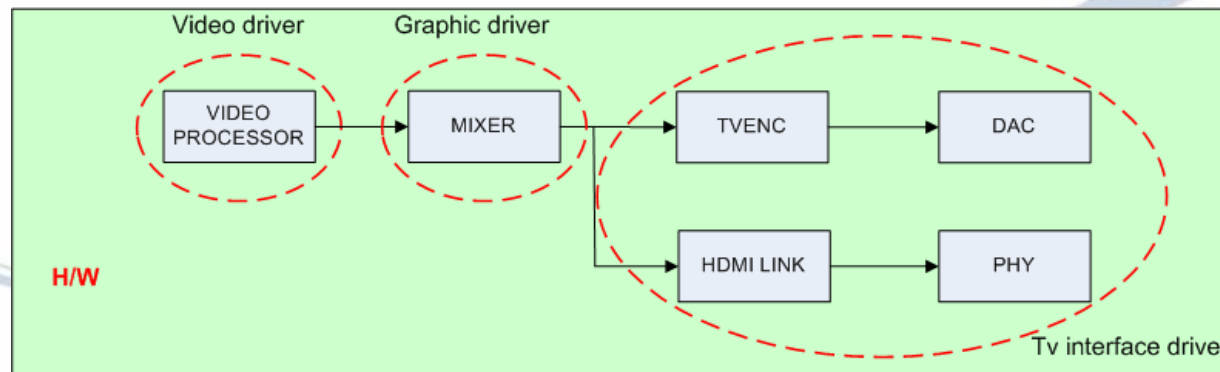


Proposal For TV-Out Driver (2 / 2)

□ Driver structure of SPRC



□ Driver structure of S.LSI



Q & A

Sensor/Flash/Snapshot functionality

□ Related emails

- <http://www.mail-archive.com/linux-media@vger.kernel.org/msg28192.html>
- <http://www.mail-archive.com/linux-media@vger.kernel.org/msg28490.html>

Sensor/Flash/Snapshot functionality

❑ Proposal (by guennadi)

- Make the “snapshot mode” as distinct from “preview mode” for flash control in camera driver.
 - flash control incl. led, lense, exposure are very complexed according to sensor.
 - Using VIDIOC_S/G_PARM to switching mode

❑ Comment

- Positive
 - Because flash control interface in V4L2 is shortage, it is required to list the functionalities.
- Negative
 - User did not want multi interface in camera scenario.
 - Flash control is possible just using i2c sub-device in V4L2.
 - Media control framework(MC) supports that case.

❑ Conclusion

- Discussion in detail at “Warsaw”

Frame-size switching: preview / snapshot

- ❑ The aim is to lessen shutter lag.

- ❑ Switching sequence from preview to snapshot
 - STREAMOFF: camera off
 - REQBUFS(0) : buffer free
 - S_FMT
 - REQBUFS(n): buffer allocation
 - QBUF: needs cache operation
 - STREAMON: camera on
 - DQBUF: get still-shot

- ❑ Needs solution to lessen the switching time.
 - Buffer free, allocation takes some time.
 - Cache operation, flush, spends long time.

Entity information ioctl

Entity information ioctl

- In case of UVC(USB Video Class) driver, suggest new ioctl for supported control list
- New API transferring specific information of device driver (Entity type)

What are these things concretely ?