Application Report

Feb 2012

TMS320DM81XX Display Output Configuration

Video Surveillance Applications

ABSTRACT

TMS320DM81XX supports multiple displays; the display sub system needs to be configured to tune the display output as per usecase requirements. Displays can be tuned for different resolutions and different contents. They can be enabled or disabled as per need. Many other parameters about displays can be configured from application. This app note gives an overview of such parameters and explains how these parameters can be configured from application using MCFW APIs

1 Overview

TI816X supports four displays – HDMI, HDCOMP (VGA), DVO2 and SD TI814X supports all these except HDCOMP. There are different VESA resolutions supported on these displays. There are several static time parameters and runtime variable that can be used to configure display output. Following sections describe briefly about how this can be done.

2 Statically changing display resolution

MCFW supports change in display resolution for VGA display at static time. Following VESA resolutions are tested on display VGA, ON-chip HDMI, OFF-chip HDMI.

- 1080P60 (1920 * 1080)
- 720P (1280 * 720)
- XGA (1024 * 768)
- SXGA (1280 * 1024)

By default, DVO2 and HDCOMP VENCS are tied together for ti816x (For ti814x HDMI and DVO2 are tied together). It means when these VENCs are configured same resolution must be passed to DVO2 and HDCOMP. Also, same content will be displayed on both outputs.

It possible to change to VGA resolution from MCFW demo. By default it supports 1080P60 on all the three displays (HDMI, HDCOMP and DVO2). Once VGA resolution is changed demo needs to be rebuilt. Please refer code below that sets VGA display to show XGA resolution

```
Vdis_params_init(&vdisParams);
/* Override the context here as needed */
```

```
vdisParams.deviceParams[VDIS_DEV_HDMI].resolution= VDIS_FORMAT_1080P60;
      /* If HDCOMP and DVO2 are tied together they must have same resolution */
vdisParams.deviceParams[VDIS_DEV_HDCOMP].resolution= VDIS_FORMAT_XGA;
vdisParams.deviceParams[VDIS_DEV_DVO2].resolution =
      vdisParams.deviceParams[VDIS_DEV_HDCOMP].resolution;
vdisParams.deviceParams[VDIS_DEV_SD].resolution = VDIS_FORMAT_NTSC;
Vdis_init(&vdisParams);
……..
```

3 Dynamically changing display resolution

MCFW allows dynamically changing the display resolution through display settings run time menu. Please refer demo DM81XX_DVR_RDK_DemoGuide for details of options that are supported and how to use the menu. Following resolutions are tested for run time display resolution change

VENCS Resolutions	On-chip HDMI	Off-chip HDMI	HDCOMP(only ti816x)	SDTV
1920 * 1080 @ 60	Yes	Yes	Yes	No
1280 * 720 @ 60	Yes	Yes	Yes	No
1024 * 768 @ 60 (XGA)	Yes	Yes	Yes	No
1280 * 1024 @ 60 (SXGA)	Yes	Yes	Yes	No
720 * 480 (NTSC)	Not tested	Not tested	Not tested	Yes
720 * 576 (PAL)	Not tested	Not tested	Not tested	Yes

Tied VENCS always show same resolution since they are driven by same clock e.g. in ti816x HDCOMP and DVO2 are tied together, if you change resolution for DVO2 from 1080P to 720P same effect should be replicated to HDCOMP display.

Untied VENCS can be configured for different supported resolutions at the same time. Change in display resolution is achieved by setting VENC timing through sysfs entries.

In current TI814x application grpx plane is used to demonstrate downscaling and same graphics plane is used between HDMI and SDTV. When the display resolution is changed dynamically the graphics logo is downscaled to resolution newly set for display. Graphics log downscaling is achieved using FBDEV ioctls on A8 side. This is only supported in TI814x and not in TI816X.

4 Tied VENCs and Mesh Configuration

Following are the possible combinations of tied VENCs

TI816X

- HDMI and HDCOMP (VGA)
- HDMI and DVO2
- HDCOMP and DVO2

TI814X

- HDMI and DVO2

Selection of these options can be done using API `Vdis_tiedVencInit()` from demo application file as shown below

vdisParams.deviceParams[VDIS_DEV_HDMI].resolution = VSYS_STD_1080P_60;
vdisParams.deviceParams[VDIS_DEV_HDCOMP].resolution = VSYS_STD_1080P_60;
Vdis_tiedVencInit(VDIS_DEV_HDCOMP, VDIS_DEV_DVO2, &vdisParams);

This call will set `tiedDevicesMask` internally in VDIS to tie HDCOMP and DVO2 together. This call will also set resolution for the tied vencs. Please note resolution for the tied vencs should be same.

Application can choose to over write the default venc parameter settings done in `Vdis_params_init()`. For each venc parameters can be set from the demo application. Following example shows how to set parameters for DVO2 similar steps apply for other displays

```
Vdis_params_init(&vdisParams);

vdisParams.deviceParams[VDIS_DEV_DVO2].enable = TRUE;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.vencNodeNum = VDIS_VENC_DVO2;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.aFmt = VDIS_A_OUTPUT_COMPOSITE;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.dvoFidPolarity =
VDIS_POLARITY_ACT_HIGH;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.dvoVsPolarity =
VDIS_POLARITY_ACT_HIGH;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.dvoHsPolarity =
VDIS_POLARITY_ACT_HIGH;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.dvoActVidPolarity =
VDIS_POLARITY_ACT_HIGH;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.dvoFmt = VDIS_DVOFMT_DOUBLECHAN;
vdisParams.deviceParams[VDIS_DEV_DVO2].outputInfo.dataFormat = SYSTEM_DF_YUV422SP_UV;

Vdis_init(&vdisParams);
```

Regarding clocks, VENC ‘D’ clock drives HDMI (on chip HDMI) and VENC ‘A’ clock drives DVO2 and HDCOMP. There is no change regarding SD VENC and SD VENC clock.

If Display Controller configuration needs to be changed, it can be done by modifying default configuration set for gSystem_dctrlTriDisplayConfig in system_dctrl.c. Similarly, a new mesh configuration can be added to same file which would resemble to gSystem_dctrlTriDisplayConfigDvo2 and gSystem_dctrlTriDisplayConfigHdmi as defined for TI816X. To understand the mesh of components in display controller please refer HDVPSS_UserGuide.pdf in hdvpss package, UserGuideHdvpssDisplayDriver section, Macro mapping - TI816X diagram.

If application over writes the VENC configuration as shown above. Same configuration will be used in system_dctrl.c to configure VENC.

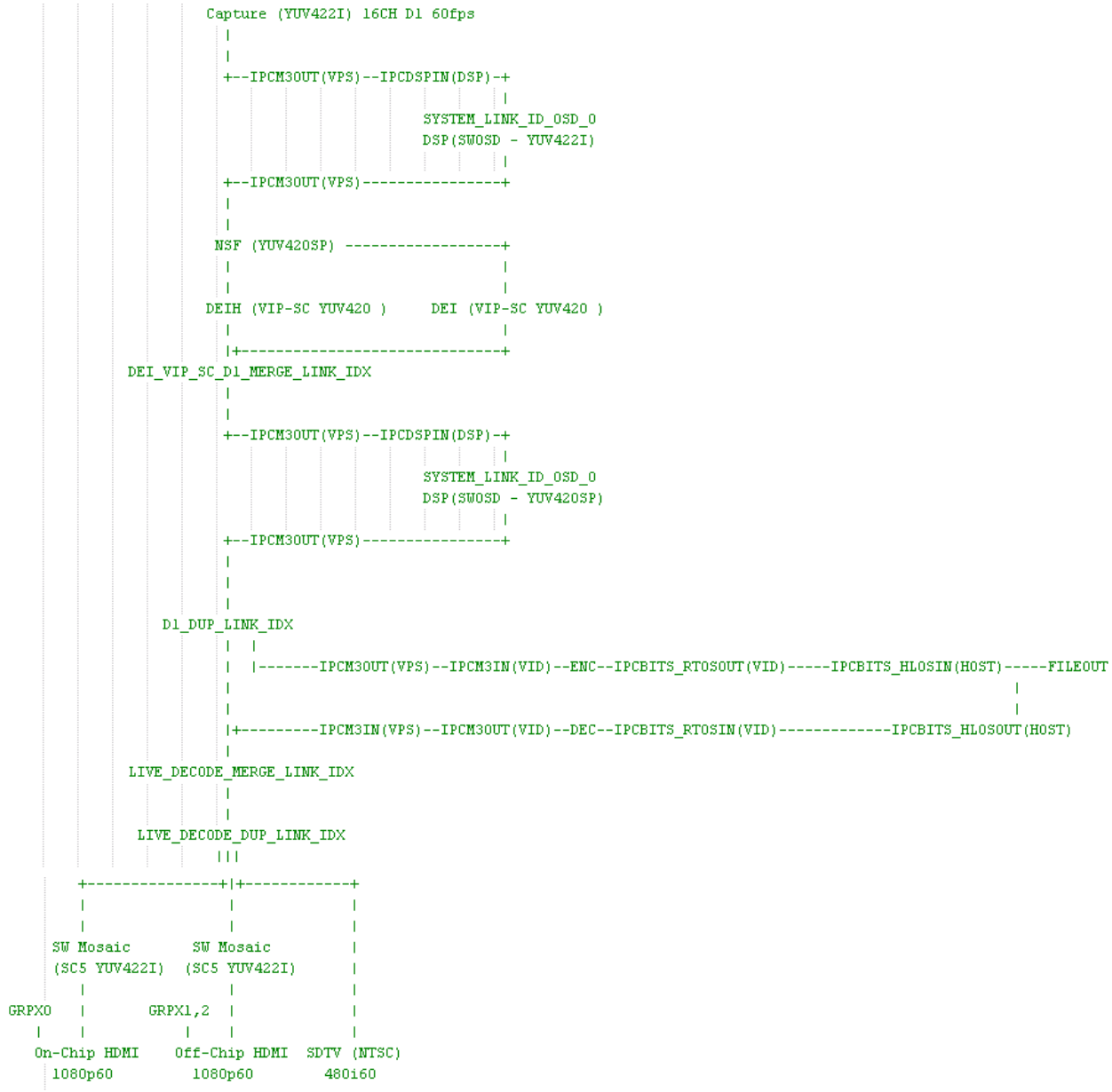
Untied VENCs

Current release does not support untied vencs for TI816X and TI814X.

There are changes needed in system_dctrl.c, system_m3vpss.c and system_dctrl_common.c to support untied vencs

5 Disabling displays

User needs to tweak a demo application in order to disable one particular display. Currently number of displays in the demo is usecase specific e.g. progressive mcfw demo supports on chip HDMI, off chip, VGA and SD Following scenario is supported in progressive demo



Disabling SD

Set enableSdtv variable to FALSE to disable SD display.

Disabling HDMIs

If HDMI display is to be disabled user should ensure dupPrm[LIVE_DECODE_DUP_LINK_IDX].numOutQue is configured correctly.

If there is only one display in the system dup2 Link in the figure above should not be created. Application can be modified to connect merge2 out to display through swms or without swms (as per requirement) by changing nexLink of merge2 and prevLink of chosen display instance.

If there are two HDMI displays parameter dupPrm[LIVE_DECODE_DUP_LINK_IDX].numOutQue must be set to 2 .

dupPrm[LIVE_DECODE_DUP_LINK_IDX].numOutQue = 2;
Also, nexLink ids of dup2 link and prevLink ids of swms Link need to be configured accordingly assuming swms is required.

All these modifications needs to be done in usecase specific create and delete that is present in usecase specific file which will be in dvr_rdk/mcfw/src_linux/mcfw_api/usecases/
Important Note: Since HDCOMP and DVO2 are tied together if you disable the dup output that drives off chip HDMI display instance both off chip HDMI and VGA will be disabled.

6 Live bypass view for SD Display

Display Link supports a feature called “Live bypass view” for SD display. This feature enables captured fields to be directly fed to the display link bypassing any operations such as deinterlacing.
This provides a SPOT TV view for a selected channel.
To enable this feature display link supports multiple input queues.
One queue will feed the live bypass content
Another queue will feed the Mosaic-ed content that is output by the Software Mosaic link.
The application will configure the display link for required number of input queues and select the active queue at create time.
Display link can switch from one queue to another at runtime based on application control
In the SPOT TV view the application can additionally select the channel to be displayed.
In a typical application all capture channels will be fed to the display link and the application can select the channel to be displayed.
Please refer following figure.