

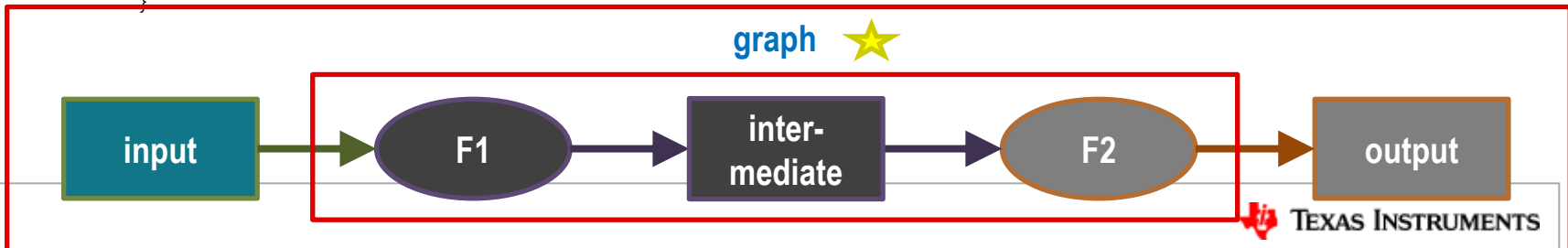
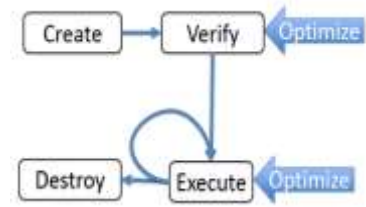
Introduction to OpenVX: OpenVX example

OpenVX Graph

```
vx_context context = vxCreateContext();
vx_image input = vxCreateImage( context, 640, 480, VX_DF_IMAGE_U8 );
vx_image output = vxCreateImage( context, 640, 480, VX_DF_IMAGE_U8 );

vx_graph graph = vxCreateGraph( context );
vx_image intermediate = vxCreateVirtualImage( graph, 640, 480, VX_DF_IMAGE_U8 );
vx_node F1 = vxF1Node( graph, input, intermediate );
vx_node F2 = vxF2Node( graph, intermediate, output );

if ( vxVerifyGraph( graph ) == VX_SUCCESS ) {
    while(...) {
        // ... write to input image ...
        vxProcessGraph( graph ); // or vxScheduleGraph( graph );
        // ... read from output image ...
    }
}
```



For more information

- Khronos OpenVX v1.1 video tutorials

<https://youtu.be/JZZCNcflqqs?list=PLYO7XTAX41FP01wTyWfwiNW3xq9IDRAnO>

- Jacinto 7 Processor SDK Automotive download:

<http://www.ti.com/tool/PROCESSOR-SDK-JACINTO-DRA8X-TDA4X>

- Processor SDK Linux Automotive (PSDKLA) user guide:

[\\${PSDKLA_INSTALL_PATH}/docs/linux/index.html](#)

- Processor SDK RTOS Automotive (PSDKRA) user guide:

[\\${PSDKRA_INSTALL_PATH}/index.html](#)

- For additional questions, refer to the E2E community forums:

<https://e2e.ti.com/support/processors/f/791>



©2020 Texas Instruments Incorporated. All rights reserved.

The material is provided strictly "as-is" for informational purposes only and without any warranty.
Use of this material is subject to TI's **Terms of Use**, viewable at [TI.com](https://www.ti.com)