

a dedicated process. This will improve system reliability in case your driver encounters a problem since other drivers will not be affected.

Ensure the device information reported to `IddCxAdapterInitAsync` is accurate. This information determines how the device is reported to the OS and what static features (like support for gamma tables) the device will have available. If some information cannot be known immediately in the `EvtDeviceD0Entry` callback, `IddCx` allows the driver to call `IddCxAdapterInitAsync` at any point after D0 entry, before D0 exit.

Careful attention should be paid to the frame processing loop. This will directly impact the performance of the user's system, so making use of the [Multimedia Class Scheduler Service](#) and DXGI's support for [GPU prioritization](#) should be considered. Any significant work should be performed outside the main processing loop, such as by queuing work in a thread pool. See `SwapChainProcessor::RunCore` for more information.

License

License MIT and CC0 or Public Domain (for changes I made, check with Microsoft for their license), whichever is least restrictive -- Use it

AS IS - NO IMPLICIT OR EXPLICIT warranty This may break your computer, it didn't break mine. It runs in User Mode which means it's less likely to cause system instability like the Blue Screen of Death.

Acknowledgements

See the original repo below:
<https://github.com/roshkins/IddSampleDriver>

Thanks to <https://github.com/akatrevorjay/edid-generator> for the hi-res EDID.