



Don't Bake Your Graphics Cards!

Rajneesh Bhardwaj
Alex Deucher

Mysteries and Myths

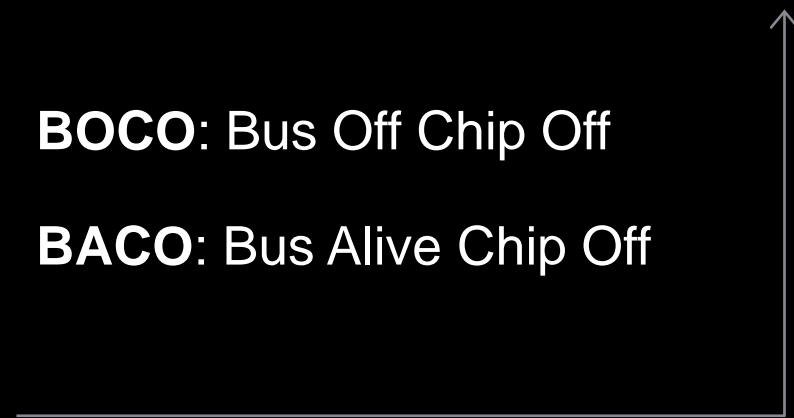
- Baking the graphics cards for repair?
 - https://www.reddit.com/r/pcmasterrace/comments/6zfrf1/should_you_bake_your_graphics_card/
- Expert mode && “I am feeling lucky” operation
- This talk is not about heat!
- It’s about the technology that keeps the graphics card cool!

Bus and Chip Jargon

- A = alive
- B = bus
- C = chip
- O = off

BOCO: Bus Off Chip Off

BACO: Bus Alive Chip Off



Cont'd.

- Device States : D3 / D3Hot / D3Cold
- BACO : It is an idle state of the dGPU which is employed in idle scenarios for long idle power requirements. BACO is entered when dGPU has been idle for several seconds and display has gone blank. Driver support is required to save the video memory and other required information as part of BACO entry sequence
- BOCO : Powers down the GPU completely and offers more power savings than BACO

BOCO

- Introduced with AMD PowerXpress laptops
- Mostly handled by ACPI ATPX and _PR3 methods
- BUS interface remains off
- Device disappears from bus
- More wakeup latency but more power savings

BACO

- Low power mode at long idle
- Applied for dGPU, hybrid GPU
- Cross collaboration of hardware, firmware, and software
- Important PLLs such as SPLL, DPLL etc. are offloaded
- BUS interface remains on
- Fan control
- Less wakeup latency with power savings

Linux Power Management Schemes

- System wide
 - Suspend to RAM
 - Suspend to idle (S2Idle)
- Component wide
 - Runtime power management
 - ```
struct dev_pm_ops {
 runtime_suspend
 suspend
 freeze
 runtime_idle
 ...
}
```
    - `pm_runtime_autosuspend()`
    - `pm_runtime_mark_last_busy()`
    - `pm_runtime_get_sync()`
    - `pm_runtime_put_autosuspend()`

# OTHER DEPENDENCIES

- DISPLAY AUDIO

- Audio codecs need to participate in runtime power management
- <https://lists.freedesktop.org/archives/amd-gfx/2019-November/043097.html>

- KFD

- Idle detection for user mode queues based on per process device data (pdd)
- <https://lists.freedesktop.org/archives/amd-gfx/2020-February/045504.html>

# GPU RESET

- `amdgpu_reset_method`
  - (-1 = auto (default), 0 = legacy, 1 = mode0, 2 = mode1, 3 = mode2, 4 = baco)
- GPU reset using BACO and RAS issues
  - <https://github.com/RadeonOpenCompute/ROCK-Kernel-Driver/blob/7b4f1de71ea335e965fba590f4d030de52644137/drivers/gpu/drm/amd/amdgpu/soc15.c#L508>
- `/sys/kernel/debug/dri/N/amdgpu_gpu_recover` can be used to manually trigger a GPU reset at the next fence wait and internally it may use BACO if applicable



# UPSTREAM STATUS

Yes, Its available upstream!

<https://lists.freedesktop.org/archives/amd-gfx/2020-February/045504.html>

<https://lists.freedesktop.org/archives/amd-gfx/2019-February/031552.html>

```
bool amdgpu_device_supports_boco(struct drm_device dev);
bool amdgpu_device_supports_baco(struct drm_device dev);
bool amdgpu_device_is_peer_accessible(struct amdgpu_device adev,
struct amdgpu_device peer_adev);
int amdgpu_device_baco_enter(struct drm_device dev);
int amdgpu_device_baco_exit(struct drm_device dev);
int smu_baco_get_state(struct smu_context smu, enum smu_baco_state state);
int smu_baco_enter(struct smu_context smu);
int smu_baco_exit(struct smu_context smu);
bool smu_baco_is_support(struct smu_context smu);
bool amdgpu_dpm_is_baco_supported(struct amdgpu_device adev)
```



**Questions?**



# DISCLAIMER AND ATTRIBUTIONS

## DISCLAIMER

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. Any computer system has risks of security vulnerabilities that cannot be completely prevented or mitigated. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.

THIS INFORMATION IS PROVIDED 'AS IS.' AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS, OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION. AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY RELIANCE, DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

©2020 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, Radeon and combinations thereof are trademarks of Advanced Micro Devices, Inc. Linux is a trademark of Linus Torvalds and OpenCL is a trademark of Apple Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

**AMD** 