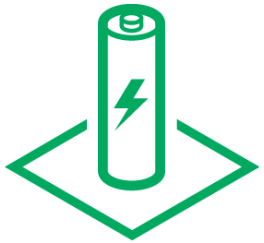
ARM Frame Buffer Compression
AFBC

October 2016

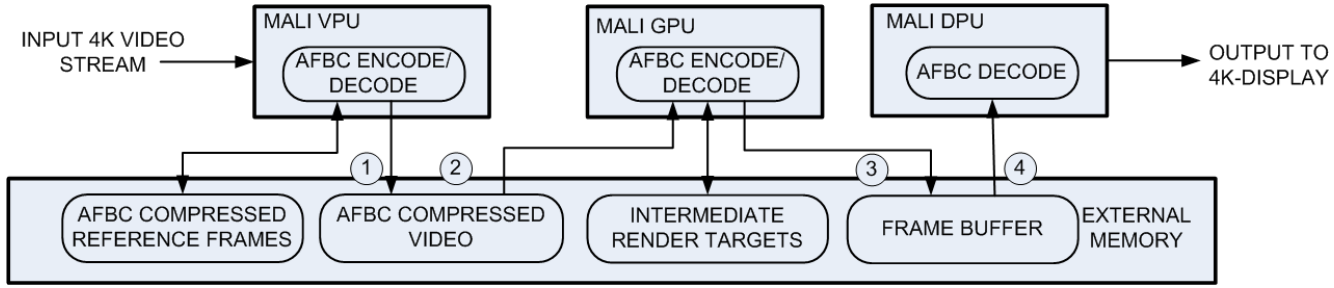
AFBC Standalone Value Proposition

Bandwidth Savings
>50% Image Compression

- Significant SoC level bandwidth reduction
  - System-wide efficiency across AFBC-enabled multimedia IP
- SoC power savings of >500mW, as a result of bandwidth reduction
  - When AFBC is used across GPU, VPU and DPU, for 4K stream



Power Savings
>500mW



- AFBC standalone is scalable and easy of integrate
  - Provides a cost effective way of introducing AFBC benefits to 3rd party IP
  - Continued investment to support new pixel formats and align features with latest Mali IP

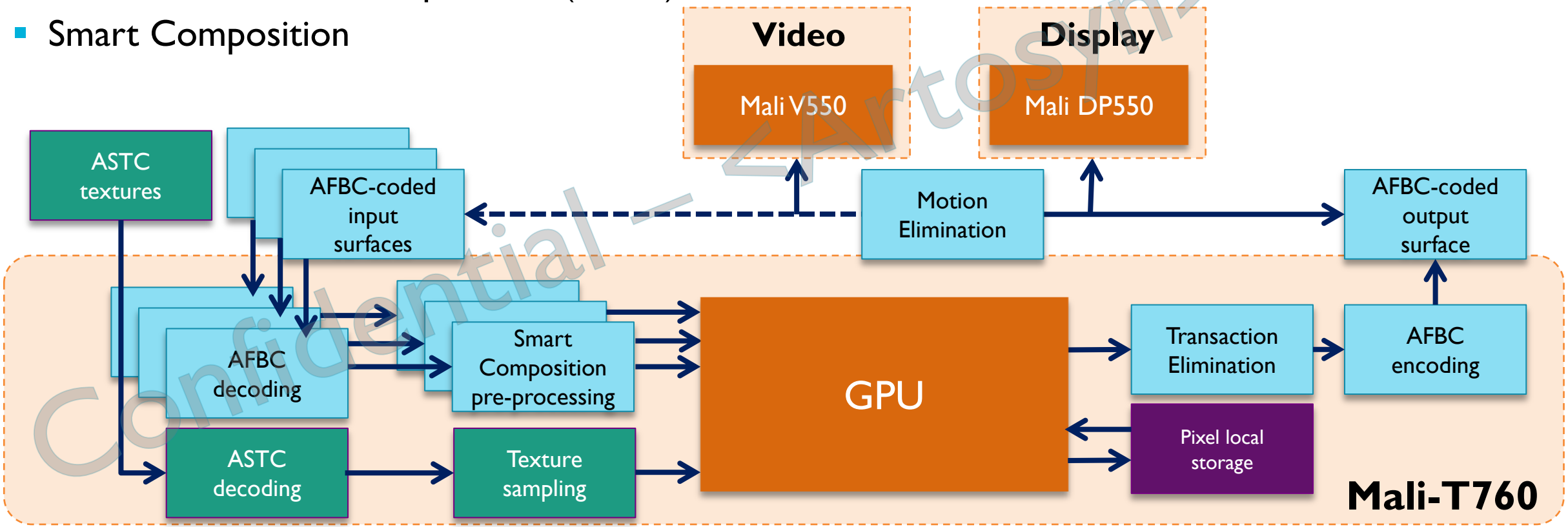
Scalability and
Ease of Integration

AFBC 1.2 EAC– March 2017

ARM

AFBC: Key part of Mali SoC-level power saving story

- Adaptive Scalable Texture Compression (ASTC)
- Transaction Elimination (TE)
- ARM Frame Buffer Compression (AFBC)
- Smart Composition
- Motion Elimination
- Pixel Local Storage extensions



ARM

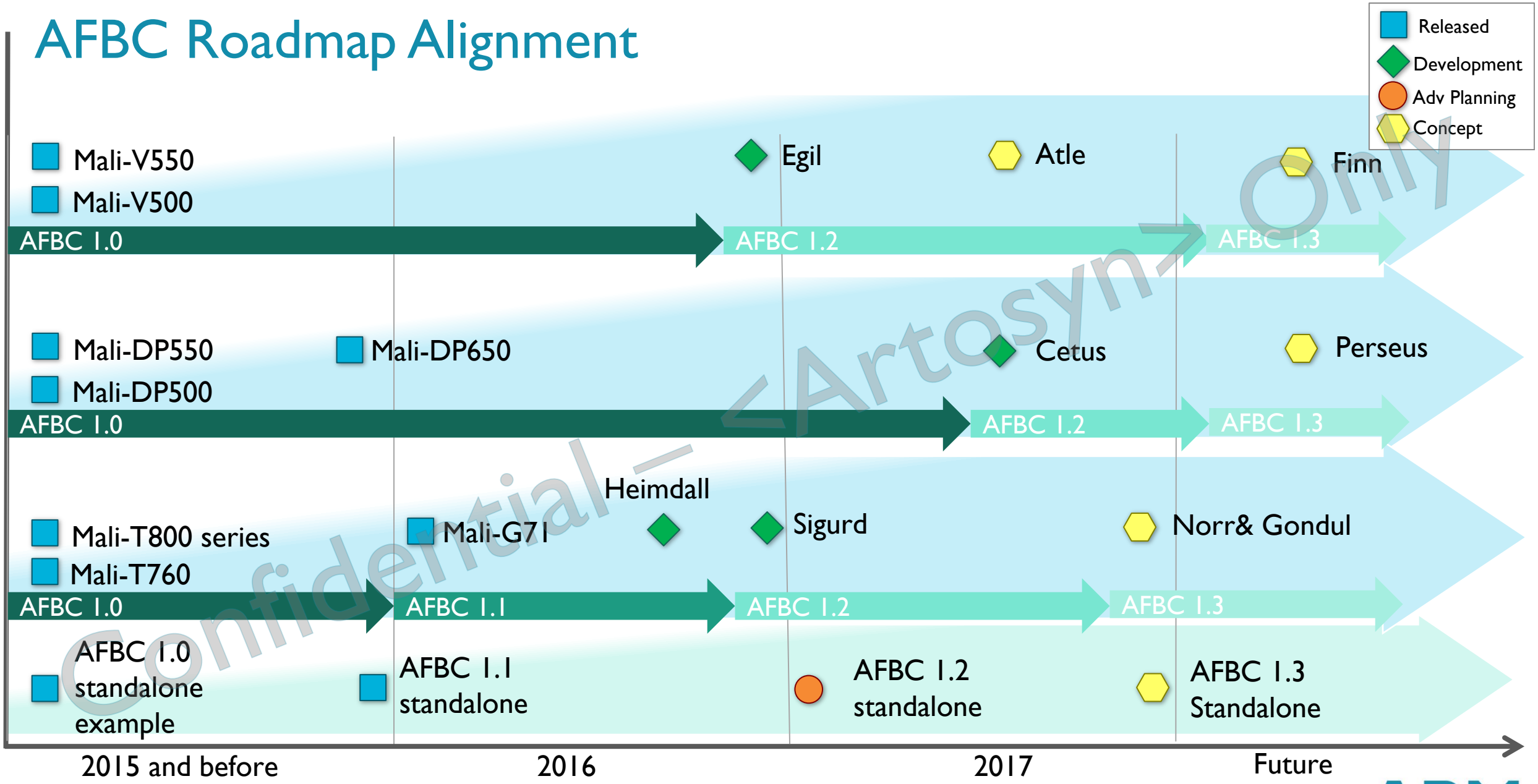
What is AFBC?

- AFBC is a lossless image compression that can be deployed by all media processing IPs in a SoC, enabling efficient data sharing among them that result in significant energy savings
  - Preserves original image exactly (bit accurate)
  - 16x16 & 32x8 blocks of pixels are independently stored
  - Random Access down to 4x4 block level
  - 4x4 block independently compressed
  - Wide selection of RGB and YUV pixel formats supported

- AFBC was first introduced in Mali T760 and in now included in all new Mali media processors (GPUs, VPUs, DPUs)
  - Standalone AFBC implementation is also available for easy integration with non-Mali media processing IPs (DPUs, VPUs, ISPs, etc)

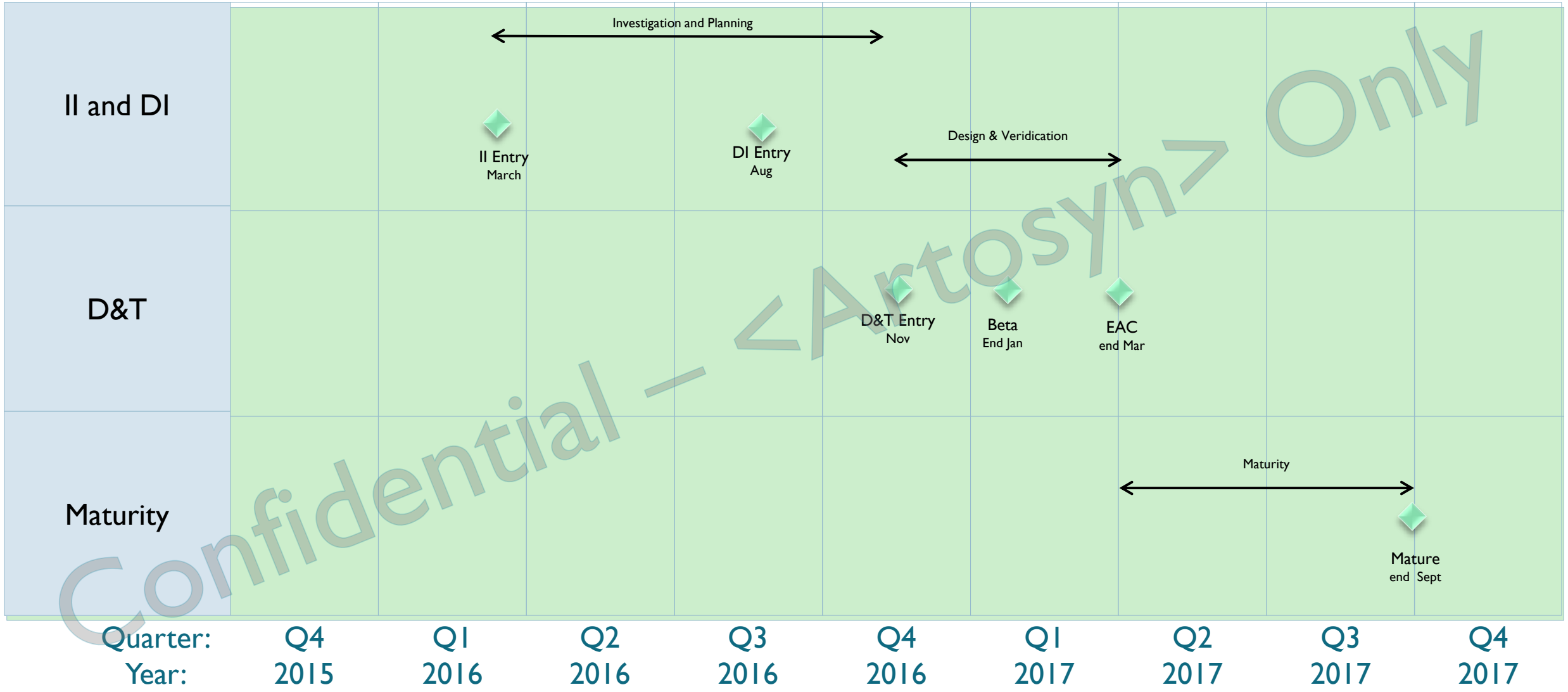
AFBC Roadmap & Features

AFBC Roadmap Alignment



ARM

AFBC 1.2 Standalone Schedule



ARM

Cross-IP Alignment for AFBC

- Mali VPU and DPU IPs may not be fully aligned with standalone AFBC in terms of pixel formats and block sizes supported
  - Some pixel formats are not be relevant for VPU and DPU
  - Mali VPU and DPU implementations will support the required subset of AFBC specification to allow for area efficient implementation, **without compromising cross-IP interoperability**
- AFBC conformance test suite developed to verify SoC level interoperability for every possible combination of AFBC-enabled IPs
  - Out-of-box AFBC standalone implementation is guaranteed to be conformant with all ARM AFBC-enabled IPs
  - Artosyn developed or modified AFBC implementations need to be verified with the AFBC conformance test suite to guarantee interoperability with ARM AFBC-enabled IPs

AFBC 1.0 Features Summary

- Lossless compression format for image buffers
  - Preserves original image exactly (bit accurate)
  - 16x16 pixel blocks independently stored
  - 4x4 pixel blocks independently compressed
- Block size formats
  - 16x16 native block size, 16x8 split block size
  - 8 line buffer in the display IP or any other IPs with raster scan order
- Multiple pixel formats supported*
  - RGB formats for both AFBC encode and decode
  - YUV formats for AFBC decode only in GPU and standalone implementations

*Please refer to AFBC Pixel Formats Matrix for more details

AFBC 1.1 Features Summary

- Additional block sizes supported
  - 32x8 native block size, 32x4 split block size
  - Allows reduced buffer size of 4 lines in the display IP or any other IPs with raster scan order
- YUV support added for both AFBC encode and decode*
  - All input and output pixel formats of the GPU (including 10bit ones) can be compressed with AFBC
  - Improved bandwidth efficiency for all use cases
- Pixel order flexibility
  - Support for linear pixel order by adding tiler/ detiler functionality
  - Allowing efficient implementation of AFBC in all media processing IPs
- Backwards compatibility
  - Same compression algorithm as AFBC 1.0, based on 4x4 pixel block granularity
  - All AFBC 1.0 block sizes and pixel formats are still supported in AFBC 1.1

*Please refer to AFBC Pixel Formats Matrix for more details

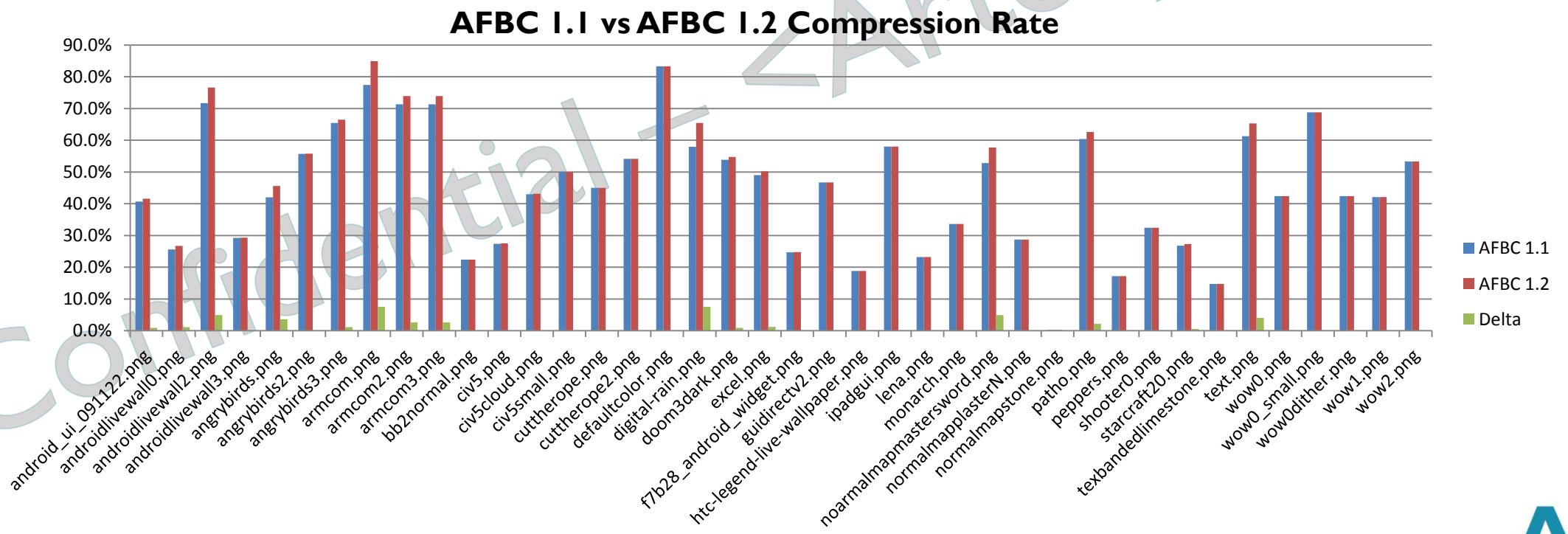
AFBC 1.2 Features Summary

- AFBC 1.2 specification improvements focus on:
  - DDR-optimised data order for headers and body buffers (tiled headers, tiles of superblocks)
    - Improving GPU performance in bandwidth limited scenarios
    - Improving DPU performance in rotation use case
  - Optimising compression for constant colour blocks
  - Full backwards compatibility with AFBC 1.1 specification

- AFBC 1.2 standalone implementation is focused on:
  - Implementing a fully featured version of AFBC 1.2 specification
  - Improving pixel throughput to 2pix/clk
  - Reassuring AFBC standalone is never a bottleneck of system performance
  - Area optimised configurations for each target resolution (8k, 4k, 1080p, 720p)
  - Enable support for both TZMP1 and TZMP2 schemes

Constant Colour Blocks Optimisation in AFBC 1.2

- AFBC 1.2 further improves AFBC compression ratio on images with constant colour blocks
  - UI and 2D-like graphics applications benefit from that
  - No additional benefits for noisy images and most high end 3D graphics applications



ARM

AFBC Pixel Formats Matrix

- AFBC 1.2 doesn't introduce new pixel formats per se, but changes the way existing formats are stored

			T760-r0px		T760-r1px T800-r0px		T800-r1px T800-r2px		G71 Heimdall S.A. 1.1		Sigurd S.A. 1.2		V500		V550		Egil		DP500		DP550 DP650		Cetus	
Pixel format	Block size	AFBC ver.	RD	WR	RD	WR	RD	WR	RD	WR	RD	WR	RD	WR	RD	WR	RD	WR	RD	WR	RD	WR	RD	WR
8/10bit RGB(A)	16x16	1.0	X	X	X	X	X	X	X	X	X	X							X		X		X	X
RGB565	16x16	1.0	X	X	X	X	X	X	X	X	X	X							X		X		X	X
8bit YUV_420	16x16	1.0	X		X		X		X	X	X	X		X		X	X	X	X		X		X	X
10bit YUV_420	16x16	1.0					X		X	X	X	X				X	X	X	X		X		X	X
8bit YUV_422	16x16	1.0	X		X		X		X	X	X	X				X	X	X	X		X		X	X
10bit YUV_422	16x16	1.0								X		X												
8/10bit RGB(A)	16x8 (split blk)	1.0			X	X	X	X	X	X	X	X									X		X	
RGB565	32x8	1.1							X	X	X	X											X	
8bit YUV_420	32x8	1.1							X	X	X	X					X							
10bit YUV_420	32x8	1.1							X		X						X							
8bit YUV_422	32x8	1.1							X		X						X							
8/10bit RGB(A)	32x4 (split blk)	1.1							X	X	X	X											X	
Tiled headers	16x16	1.2									X	X					X	X					X	X
Tiled headers	32x8	1.2									X	X					X						X	
Tiled headers	16x8, 32x4 (split blk)	1.2									X	X											X	

ARM

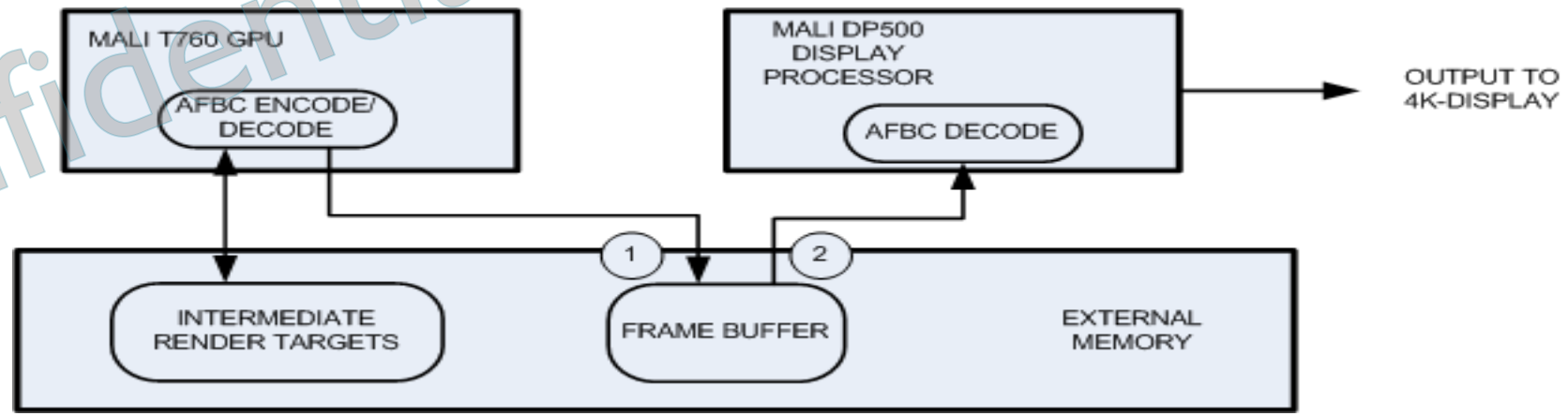
Investigations for features beyond AFBC 1.2

- Compression improvements for GPU buffers
  - Improved compression formats for depth buffers and volume textures
  - Lossy compression for selected render targets
- Improved efficiency for compressed video frames
  - Lossy compression for ISP-to-VPU path
  - New macroblock sizes, optimised for low bitrate YUV formats
- Optimisations for VR use case
  - New formats for optimised timewarp support

AFBC Power Savings

AFBC SoC synergies in 3D apps

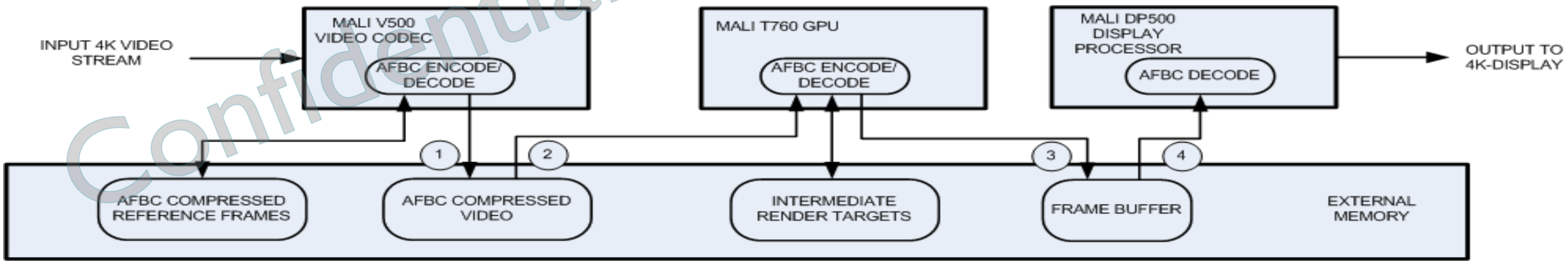
- A SoC that has AFBC support for both GPU and Display Processor running any 3D application will compress:
  - All GPU Intermediate renders (saves GPU write/read bandwidth)
  - 2x Screen Sized Buffers (saves frame buffer write bandwidth from GPU + frame buffer read bandwidth from display)
- Assuming that on average 70% of the AFBC bandwidth saving comes from intermediate renders and 30% from final frame buffer (pre-scalled), 1080p native application resolution, frame buffer scaled to 4K, 60fps, 0.11 mJ/MB (typical for LPDDR3)
  - SoC savings for GLB 2.1.2 Egypt: **1580 MB/sec** or **175 mW**
  - SoC savings for GLB 2.5.1 Egypt: **1395 MB/sec** or **154 mW**
  - SoC savings for GLB 2.7 T-Rex: **3070 MB/sec** or **340 mW**



ARM

AFBC SoC Synergies in 3D UI with video stream

- A SoC that has AFBC support for GPU, Video Codec and Display Processor running a 3D UI with video stream will compress:
  - Video Decoder reference frames (saves Video decoder write/read bandwidth)
  - All GPU Intermediate renders (saves GPU write/read bandwidth)
  - 4x Screen Sized Buffers
- Assuming a 3D UI application with 3 full screen layers at native resolution of 1080p, 4K "Life in Garden" video stream, 4K display, 60fps, 0.11 mJ/MB (typical for LPDDR3)
  - SoC savings: **4800 MB/sec** or **534 mW**



ARM

Standalone AFBC Options

AFBC Standalone

- AFBC 1.1 standalone and newer versions are provided as fully verified products with the same quality guarantees as any other ARM IP

- AFBC 1.1 standalone and newer versions to be available in 2 variants:
  - AFBC standalone: Configurable RTL of productised quality that will be able to cover the most common use cases for integration with display, video and ISP
  - AFBC core codec: Core AFBC codec will be available to partners that wish to add/remove features beyond the available configuration options of AFBC standalone
    - Encoder core will be provided as macroblock level module
    - Decoder core will be provided as 4x4 pixel block level module

- Conformance test suite is also provided to cutomers that plan to use AFBC core codec or modify AFBC standalone

AFBC Standalone (2)

AFBC 1.1 and 1.2 standalone deliverables include:

- AFBC Compression Format Specification
- AFBC C software implementation
  - AFBC encoder C model
  - AFBC decoder C model
- AFBC RTL example documentation
  - AFBC Release Note
  - AFBC standalone Technical Reference Manual
- AFBC RTL example implementation
  - AFBC standalone encoder synthesisable RTL
  - AFBC standalone decoder synthesisable RTL
  - AFBC core block encoder synthesisable RTL
  - AFBC core block decoder synthesisable RTL
- AFBC RAM integration testbench
- AFBC conformance test suite + documentation

AFBC 1.1 Standalone configurations and area

Resolution & number of layers supported	AFBC 1.1 Decoder Area (TSMC CLN28HPM S31, 700MHz, post layout with 85% utilization)
1 layer @ 8K or 2 layers @ 4K or 4 layers @ 1080p or 4 layers @ 720p	0.37 mm2
1 layer @ 4K or 2 layers @ 1080p or 3 layers @ 720p	0.28 mm2
1 layer @ 1080p	0.23 mm2*
1 layer @720p	0.21 mm2*

Resolution & number of layers supported	AFBC 1.1 Encoder Area (TSMC CLN28HPM S31, 700MHz, post layout with 85% utilization)
1 layer @ 8K (or smaller resolution)	0.3 mm2
1 layer @ 4K (or smaller resolution)	0.21 mm2*
1 layer @ 1080p (or smaller resolution)	0.16 mm2*
1 layer @ 720p (or smaller resolution)	0.14 mm2*

*These configurations have not been verified in EAC release, but can be verified on Artosyn demand

ARM

AFBC Standalone business model

- Different options for licensing standalone AFBC:

1. Free of charge AFBC specification licence, to be used in SoCs along with Mali GPU or VPU
   Deliverables:
   - AFBC 1.1 Specification & Conformance test suite
   - No option for support
2. Chargeable AFBC standalone product license, to be used in SoCs along with Mali GPU or VPU
   Deliverables:
   - AFBC 1.1 Specification & Conformance test suite
   - C-model
   - Fully verified standalone & core blocks RTL, with accompanied documentation
   - Support and maintenance available for extra fee

- No royalties charged for standalone AFBC for either option 1 or 2

Thank You!