

## Graphics on Replicant 11

This page documents the current progress and future plans for the graphics acceleration on Replicant 11. The original plan can be found at [TasksFunding](#). Since then, through more in depth research and hands-on experience, several things have diverged.

The full effort of Porting Replicant to Android 11 can be tracked at: [PortingToAndroid11](#).

Background information, as well as details on the software components and acronyms used on this document, can be found at [GraphicsResearch](#).

### Graphics stack tasks

status	origin	short description	notes	estimated man-hours	actual man-hours
done	original plan	Set up the development environment.	Required: i9305 phones, LXC Trisquel container (systemd nspawn fails due to old systemd on Trisquel 8), larger SSDs, 1.8V serial-USB adapters (BS101P FT232RL) plus makeshift resistors' banks.	24	24
ongoing	new	Update development environment for Replicant 11.	Android 11 has higher resource consumption during builds. <a href="#">Hacks and workarounds</a> had to be found to be able to build in our machines. Build servers will be set up to get faster builds.	0	8
ongoing	original plan	Read graphics related AOSP documentation.	Never-ending task that, besides actual documentation, involves scouring through source-code, bug trackers, mailing lists and IRC logs.	16	48
ongoing	new	Ask for help.	Bothering free-software developers <sup>1</sup> that have experience with or contribute to graphics sub-systems has been the most fruitful way to clear most roadblocks.	0	0
done	original plan	Use Mesa's llvmpipe backend instead of softpipe.	Merge requests on Mesa: <a href="#">l1402</a> and <a href="#">l1403</a> . There was	40	28

			no need to update LLVM version.		
todo	new	Find out why we are getting away without using the <a href="#">libEGL patch</a> .	Android 11 no longer needs EGL? Take a look at frameworks/native/opengl/libs/EGL/Loader.cpp	0	0
ongoing	original plan	Implement the missing pixel formats in drm/exynos.	Joonas created <a href="#">a patch</a> that allows selecting the BGR format for Exynos FIMD through a <a href="#">boot time module parameter</a> . It uses the VIDCON0 register, which can be set to either VIDCON0_PNRMODE_RGB or VIDCON0_PNRMODE_BGR. Both formats cannot co-exist. TODO: use a string in the module parameter and send upstream. GNUtoo proposed another approach with runtime checks like <pre>.atomic_check(...){ if (using RGB &amp;&amp; asked for BGR) =&gt; return NOT_POSSIBLE; } </pre> If application A starts using RGB, and application B asks for BGR, the kernel refuses. As if the BGR format is removed from <a href="#">the list</a> at runtime. Once no more applications use the display engine, then it's like if it was re-added to the list. There seems to be no practical use for this extra flexibility, as once booted both Android and GNU/Linux will stick to their preferred pixel format. However, at #dri-devel, emersion told us that <a href="#">listing all formats and then failing at the atomic check is preferable upstream</a> .	72	1

todo	new	Get entire stack to use RGB555 pixel format.	Had a huge performance boost on Replicant 6.	0	0
abandoned	original plan	Proper way to use DRM-Master and DRM-Auth with gbm_gralloc and drm_hwcomposer.	<p>DRM-Auth is no longer needed for gbm_gralloc because, on December 2019, <a href="#">DRM_AUTH was dropped from PRIME_TO/FROM_HANDLE ioctls</a>.</p> <p>drm_hwcomposer and gbm_gralloc can now share the display/kms node with no need for DRM Auth.</p> <p>drm_hwcomposer, which uses KMS ioctls, must attach to the node first, in order to become DRM Master.</p> <p>gbm_gralloc should attach after it.</p> <p>Before DRM_AUTH had been dropped, we had tried:</p> <ol style="list-style-type: none"> <li>1. <a href="#">Auth hack</a> (both on /dev/dri/card0)</li> <li>2. vGEM (gbm gralloc on /dev/dri/card1) - gbm gralloc cannot take advantage of exynos hardware planes; memory may not be properly allocated.</li> <li>3. <a href="#">Allow dumb buffers on render node</a> (gbm gralloc on /dev/dri/renderD128) - Dumb buffers are used for scanout. Should not be created on a render node.</li> </ol>	40	8
done	new	Start gbm_gralloc service after drm_hwcomposer.	<a href="#">Android init</a> is quite primitive, but Joonas accomplished this by <a href="#">disabling the gralloc service</a> . It seems that gralloc is later started automatically when needed.	0	2
ongoing	new	Use hardware planes for better composer performance.	Enabling HW planes with drm_hwcomposer was straightforward but led to severe	0	17

			<p>graphics corruption. <a href="#">Disabling devfreq</a> fixed the corruption. Tentative explanation: display controller frequency gets too low for timely DMA transfer of overlays. <a href="#">Reported upstream. Devfreq on Exynos is known to be a little bit broken.</a></p> <p>TODO:</p> <ul style="list-style-type: none"> <li>* Lock display controller frequencies through sysfs and re-enable devfreq, or try to <a href="#">remove low freq OPP steps.</a></li> <li>* Make sure that drm_hwcomposer is using all 4 available HW planes on exynos (1 primary, 3 overlay). Joonas used <a href="#">dumpsys</a> and added prints to <a href="#">validateDisplay</a>, finding at most 3 planes in use: avail_planes is how many HW planes we have and layers_.size() is how many are in the composition.</li> <li>* Add support for rotation.</li> <li>* Debug <a href="#">drm-hwcomposer-in-termittent-alpha.mp4</a> (Dim Layer sent by SurfaceFlinger).</li> <li>* Enable the cursor plane.</li> </ul>		
ongoing	new	Use Skia instead of HWUI to render the Canvas.	<p>Unlike Replicant 6, none of the usual system props (e.g. ro.kernel.qemu=1, ro.config.avoid_gfx_accel=1) would yield the expected performance. Got there by forcing <a href="#">hardwareAccelerated=false</a> on all apps.</p> <p>TODO:</p> <ul style="list-style-type: none"> <li>* Turn this dirty hack into a system property that can be toggled on the device tree.</li> <li>* Test</li> </ul>	0	22

			ro.kernel.qemu.gles=0		
ongoing	original plan	Create test scenarios and check if the graphics stack works as expected.	Stock apps work. Check <a href="#">Tested apps</a> bellow for the current status with apps that require advanced graphics features. TODO: compliance tests.	40	14
ongoing	original plan	Make the graphics stack work with vGEM driver besides drm/exynos.	vGEM seem to be the proper dri node for Mesa's <a href="#">kms_swrast driver</a> . We are currently using a <a href="#">simple hack</a> that kms_swrast to use drm/exynos instead. Should rather use <a href="#">vGEM</a> .	40	4
tentative	new	Combine <a href="#">kmsro</a> with kms_swrast on vGEM render node?	Is kms_swrast working on top of the vGEM render node able to share PRIME buffers with the display node (Exynos)? If not, would adding kmsro to the mix help? Architectural ideia: kmsro + kms_swrast on vGEM render node -> PRIME -> drm_hwcomposer and gbm_gralloc on display node (Exynos) Advantages: - no need to copy buffers between kms_swrast and Exynos (PRIME takes care of that); - can take advantage of HW planes.	0	0
ongoing	original plan	Document the design decisions.	Done at this wiki page plus <a href="#">GraphicsResearch</a> and the presentation at <a href="#">ContributorsMeetin gJuly2019</a> .	16	64
ongoing	new	Try out the <a href="#">Android Go low RAM switches</a> .	Check their impact on graphics rendering performance and overall system usability.	0	1
todo	new	Test gbm_gralloc with camera.	So far we've only been testing gbm_gralloc with	0	0

			Lima and Exynos. However, the Gralloc HAL will be used to allocate buffers that will be shared with other devices as well, such as the camera.		
todo	new	Fix screen recorder.	Seems to fail with some EGL issue.	0	1
total sum:				288	240

<sup>1</sup> A big thanks to Joonas Kylmäla, Paul Kocialkowski, Denis Carikli, Andrés Domínguez, Mauro Rossi, Erico Nunes, Tomeu Vizoso, Daniel Stone, Emil Velikov, Andrzej Hajda, Marek Szyrowski and LiquidAcid.

## SwiftShader tasks

status	origin	short description	notes	estimated man-hours	actual man-hours
done	original plan	Find a way to use SwiftShader instead of Mesa.	Joonas got there <a href="#">with ranchu composer (from Android Emulator and the default gralloc</a> , plus a <a href="#">patch to support UDIV/SDIV emulation in the kernel</a> .	40	0
done	new	Use LLVM as backend instead of SubZero.	Found a <a href="#">SwiftShader revision</a> that uses LLVM and is still compatible with Android 9 frameworks/native. No noticeable performance difference.	0	6
done	new	Do UDIV/SDIV emulation on JIT compiled shader code instead of kernel patch. Avoids performance penalty of interrupt handling.	It seems that <a href="#">SwiftShader does not send the processor model (microarchitecture) to LLVM</a> , leaving it without a way to decide <a href="#">whether the processor has hardware division</a> . Fixed upon update to Replicant 10. SwiftShader now uses LLVM 7 instead of 3, which fixed this.	0	30
todo	new	Use <code>drm_hwcomposer</code> instead of <code>ranchu</code> . Advantages: uses hardware planes and DRM nodes	<a href="#">Joonas was close</a> .	0	0

		instead of direct framebuffer.			
todo	new	Use mainline SwiftShader. Brings in a Vulkan software renderer for Replicant.	Difficult to due incompatibilities with <a href="#">frameworks/native</a> . Check if is fixed with Replicant 11.	0	0
total sum:				40	36

## Ilvmpipe optimization tasks

status	origin	short description	notes	estimated man-hours	actual man-hours
todo	original plan	Setup a testing and benchmarking environment.	Profiling: turn on profiling switch on Mesa + simpleperf? Benchmarks: <a href="#">android-fps-count</a> , <a href="#">OxBenchmark</a> , <a href="#">GearsES2</a> Conformance: <a href="#">dEQP</a> , <a href="#">Android CTS</a> , <a href="#">piglit</a> , <a href="#">freedreno/tests-*</a> , <a href="#">glmark2</a>	40	1
todo	original plan	Disable expensive OpenGL operations.		24	0
todo	original plan	Recap matrix operations and study ARM NEON.		48	0
todo	original plan	Profile apps to find the most used GLES operations.		32	0
todo	original plan	Use <a href="#">Ne10 library</a> or <a href="#">Neon Intrinsics</a> for the most used GLES operations.	Optimizations have to be done on LLVM and not on Ilvmpipe. Ilvmpipe only outputs LLVM IR. LLVM already has autovectorization for ARM NEON, try it.	80	0
todo	original plan	Fix bugs, re-write the code where needed, get it stable.		80	0
total sum:				304	1

## Lima driver tasks

status	origin	short description	notes	estimated man-hours	actual man-hours
done	original plan	Rebase <a href="#">Lima's Linux kernel DRM driver</a> on top of	Done by others. Lima DRM driver was accepted into	80	0

		<a href="#">forkbomb's Midas on Mainline kernel.</a>	mainline Linux, which also has forkbomb's patches and is now used on Replicant 11.		
done	original plan	Replace mainline Mesa for <a href="#">Lima's Mesa</a> (with their driver).	Done by others. Lima is now on mainline Mesa. <a href="#">Lima wiki</a>	16	0
done	new	Lima DRM driver bringup on Exynos.	Lima development is done on AllWinner devices. We expected <a href="#">some issues to get it working on Exynos.</a> Although there were encouraging reports by <a href="#">ChronoMonochrome</a> , <a href="#">hexdump0815</a> and Viciouss ( <a href="#">manifest</a> , <a href="#">xda</a> ). Joonas added Lima to Replicant 10 and faced no major bringup issues.	0	1
done	new	Fully test proper architecture.	drm_hwcomposer and gbm_gralloc on card0 (Exynos) -> PRIME -> Mesa on renderD129 (Lima) Advantages: - no need to copy buffers between Lima and Exynos (PRIME takes care of that); - can take advantage of HW planes.	0	1
done	new	Fix graphics corruption with hardware planes.	Corruption happened when compositing GL planes with non-GL planes. E.g. on <a href="#">Shader Editor</a> , run a shader and open a menu. Disabling devfreq didn't solve it (as it did with llvmpipe). Was due to having <a href="#">gbm_gralloc working on Lima's render node</a> , which cannot do contiguous memory allocation.	0	1
todo	new	Fix video play.	Joonas reported that the Big Buck Bunny video fails at os_get_total_physical_memory call from Mesa, which is	0	0

			called from lima_screen.c		
todo	new	Advertise GLES 2. <a href="#">Shader Editor</a> can only detect GLES 1.		0	0
todo	original plan	Build and test thoroughly with <a href="#">synthetic</a> and real applications.	Use conformance tests to figure out the <a href="#">current GLES implementation status</a> .	40	0
abandoned	original plan	Create a fallback mechanism that uses the software renderer for GLES functions not yet implemented in Lima.	There is no sane way to switch between different GLES drivers at the function level. Abandoned in favour of the tasks bellow.	100	1
done	new	Lima as SurfaceFlinger backend.	This is the default (SurfaceFlinger using the default GLES implementation). No problems found.	0	0
done	new	Lima as HWUI (SkiaGL) backend.	This is the default (SkiaGL using the default GLES implementation). No problems found.	0	0
todo	new	Lima on a per-app basis.	Lima will at most support GLES2. Therefore it may not work with certain apps depending on their GLES usage. We can re-work the <a href="#">per process libagl/llvmpipe patch</a> into a patch that switches between Lima and a software renderer (llvmpipe or SwiftShader).	0	0
total sum:				236	4

## 2D optimization tasks

status	origin	short description	notes	estimated man-hours	actual man-hours
todo	new	Investigate the possibility of using <a href="#">Pixman</a> or <a href="#">Exynos G2D</a> as RenderEngine for SurfaceFlinger.	There are interesting reports of people using <a href="#">G2D to hardware-accelerat e X11 EXA</a>	0	1
todo	new	Accelerate Skia with G2D.	Rework old patches ( <a href="#">Hillenbrand 2013</a> and <a href="#">raymanfx 2016</a> ) to make them	0	1

			work on current Skia.		
total sum:				0	2

## Tested apps

app	Device	Replicant 6		Replicant 11			notes
		libagl	LLVMpipe	Lima	LLVMpipe	SwiftShader	
Fennec F-Droid <sup>1</sup>	GT-I9300	crashes	slow	fast			Needs GLES 2.0
LibreOffice Viewer <sup>2</sup>	GT-I9300	crashes	slow	cannot test (missing storage)			
Red Reader <sup>3</sup>	GT-I9300	crashes <sup>4</sup>	usable	cannot test (no network)			
Shader Editor <sup>5</sup>	GT-I9300	crashes	7 fps	30 fps (HW planes off) 40 fps (HW planes on) Freezes when changing resolution.			FPS measured on default shader with 1/1 resolution.
Marine Compass <sup>6</sup>	GT-I9300	bad render	bad render	crashes			Only uses GLES 1.0
Gears <sup>7</sup>	GT-I9300	crashes	no render	crashes			
GL TRON <sup>8</sup>	GT-I9300	4 fps	2 fps	23 fps			Has a nice FPS counter.
Tor-browser 10.0.5 <sup>9</sup>	GT-I9250	Untested	Crash <sup>10</sup> [11]	No support for GT-I9250 yet			
Tor-browser 10.0.5 <sup>9</sup>	GT-I9300	Untested	Works <sup>10</sup> [12] a bit slow	No support for network yet			
Replica Island	GT-I9300	Untested	Untested	Fast enough			Retest with specific versions

<sup>1</sup> [https://f-droid.org/en/packages/org.mozilla.fennec\\_fdroid](https://f-droid.org/en/packages/org.mozilla.fennec_fdroid)

<sup>2</sup> <https://f-droid.org/en/packages/org.documentfoundation.libreoffice>

<sup>3</sup> <https://f-droid.org/en/packages/org.quantumbadger.redreader>

<sup>4</sup> <https://github.com/QuantumBadger/RedReader/issues/279>

<sup>5</sup> <https://f-droid.org/en/packages/de.markusfisch.android.shadereditor>

<sup>6</sup> <https://f-droid.org/en/packages/net.pierrox.mcompass>

<sup>7</sup> <https://f-droid.org/en/packages/com.jeffboody.GearsES2clair>

<sup>8</sup> <https://f-droid.org/en/packages/com.glTron>

<sup>9</sup> <https://dist.torproject.org/torbrowser/10.0.5/tor-browser-10.0.5-android-armv7-multi.apk>

<sup>10</sup> Tested on replicant-6.0 0004 RC3:

```
adb root ; adb shell
# setprop ro.kernel.qemu 1
# setprop ro.kernel.qemu.gles 0
# killall surfaceflinger
```

The device was using the default graphics settings (llvmpipe).

<sup>11</sup> It crashed when clicking on the URL bar with the following error:

```
12-04 00:35:36.364 4302 4322 I Gecko : Can't find symbol 'eglGetNativeClientBufferANDROID'.
12-04 00:35:36.364 4302 4322 I Gecko : Can't find symbol 'eglQuerySurfacePointerANGLE'.
12-04 00:35:36.364 4302 4322 I Gecko : Can't find symbol 'eglDupNativeFenceFDANDROID'.
12-04 00:35:36.364 4302 4322 I Gecko : Can't find symbol 'eglQueryDisplayAttribEXT'.
12-04 00:35:36.364 4302 4322 I Gecko : Can't find symbol 'eglQueryDeviceAttribEXT'.
12-04 00:35:36.364 4302 4322 I Gecko : Can't find symbol 'eglStreamConsumerGLTextureExternalAttribsNV'.
12-04 00:35:36.364 4302 4322 I Gecko : Can't find symbol 'eglCreateStreamProducerD3DTextureANGLE'.
12-04 00:35:36.365 4302 4322 I Gecko : Can't find symbol 'eglStreamPostD3DTextureANGLE'.
12-04 00:35:36.365 4302 4322 I Gecko : Can't find symbol 'eglSwapBuffersWithDamageEXT'.
```

<sup>12</sup> To compare with the GT-I9250, # getprop | grep qemu returns nothing.

## Files

---

drm-hwcomposer-intermittent-alpha.mp4	6.76 MB	01/21/2021	dl lud
---------------------------------------	---------	------------	--------