

UnityOS on the LG Ally

A 2010 Verizon slider phone, rebuilt as a pocket console for a fleet of coding agents. Android is gone; the stock 2.6.32 kernel boots a Buildroot userland from the SD card, and one static Rust program draws every pixel on a 600 MHz ARMv6 core. These notes cover the hardware, the boot chain, the reverse engineering, the OS, the UI internals, the remote-control plane, and what it costs to run, with the measurements behind each claim.



3.3 s kernel start to the first UI frame	9.5 MB RAM in use, whole system, of 173 MB	4.5–5.3 MB UI resident while idle on the fleet	0.84 s held awake per background check (was 20 s)	0.58 s end of speech to final text, home network	131 commits, 13 Oct 2026 to 14 Oct 2026
--	--	--	---	--	---

Contents

1	Hardware	3	Round 2: present path and CPU floor	8	10	Limits, verification and sources	14
	Storage after the conversion	3	Round 3: composition	8		How it was built	15
	What the 2010 kernel allows	3	Memory, layout and decoding	9		Sources	15
2	Boot chain	4	6	Screens	10		
3	Reverse engineering	5	7	Connectivity and the control plane			
	Audio playback order	5			12		
	Modem low-power mode	6		Remote control: ally	12		
4	The operating system	7	8	Dictation	13		
5	allyui internals	8	9	Performance and power	14		
	Present: rotation moved to the MDP	8		Sleep	14		

Scope. Measured on the phone unless marked "host" (a laptop build of the same code) or "model" (a projection from captured payloads and fake-time tests). Phone figures were taken with USB attached; battery-only suspend has not been qualified, so no figure here is a battery-life claim. The 44-page companion report covers the interface screen by screen; these notes keep to numbers and mechanisms.

LAYER	WHAT IT IS	SIZE OR COST
Hardware	LG Ally VS740 (2010): MSM7627, one ARM1136JF-S core, 256 MiB, 480 × 800 panel, slide QWERTY	173,848 kB visible to Linux
Boot	Stock kernel + a 30-line stage-1 initramfs in <code>boot</code> ; root on SD p1 (ext3)	first frame 3.3 s after kernel start
OS	Buildroot, musl, everything static; busybox init; 8 long-lived programs	about 25 MB root filesystem; Android's <code>/system</code> was 128.5 MB
UI	allyui: one Rust binary, RGB565 canvas, MDP-rotated present, no toolkit, no GPU	24.8k lines; 4.5–5.3 MB idle RSS
Hardware support	Audio, camera, flash, sensors, Bluetooth, modem LPM, recovered from LG's binaries	camera 2.5k, OS and tools 4.1k lines
Off the phone	The agent hub (REST + SSE), a speech service on a laptop GPU, a public relay	speech service 2.1k lines
Control plane	ally CLI: lock, inject input, capture, record, deploy with rollback, over pinned SSH	29.4 KiB/s uploads through the relay
Tests	181 Rust unit tests, a speech-service suite, a remote-control suite, phone smoke runs	179 + 65 + 28 passed at release

Table 1 The system at a glance. Line counts are counted from the repository at build time.

Three rounds in two days

ROUND	WHAT SHIPPED	HEADLINE NUMBERS
1	Backups and root; stage-1 boot and Buildroot root on SD; allyui with pairing, fleet, conversations, launch, approvals; speech service; camera, flash, audio, sensors, Bluetooth; charge screen, power menu, LAN/relay failover.	first frame 4.28 → 3.78 s; 15.3 MB RAM in use; UI idle 1.2 % CPU
2	Rotation on the MDP; ARM block copies; the CPU floor fix and bounded input boost; frame pacing; gestures and a bounded picture viewer; modem low-power mode.	touch → pan 17.26 → 4.53 ms; landscape conversation frame 26.0 → 15.4 ms
3	Completion-owned wakelocks, Auto polling, typed decode; retained scrims and local damage; lazy fonts, incremental layout, bounded thumbnails; adaptive dictation; RAM logs; the wireless control plane with deploy rollback; independent QA.	wake hold 20 → 0.84 s; UI RSS 10.7 → 4.5–5.3 MB; LAN dictation 0.92 → 0.58 s

Table 2 Each round ended in review and a phone smoke test; round 2 and 3 work was planned from five research reports measured on the previous build.

1

Hardware

A mid-range Verizon slider from April 2010. Every row was read on the running phone or in the stock firmware; public specifications fill in what the device cannot report.

PART	DETAIL	SOURCE
Model	LG Ally VS740, board "aloha" (also sold as Apex and Axis). Released April 2010, discontinued July 2011. Android 2.2 (FRG83D) on kernel 2.6.32.9 PREEMPT, built 18 Dec 2010.	<code>/proc/cpuinfo</code> , <code>uname</code> ; public specs
SoC	Qualcomm MSM7627: ARM11 application core (Linux), ARM9 modem core running AMSS, QDSP5 audio/video DSP, MDP display processor with a PPP blit engine.	kconfig ARCH_MSM7X27
CPU	ARM1136JF-S r1p5, ARMv6 + VFPv2, no NEON , 16 KB data cache. 122.88–600 MHz, ondemand. BogoMIPS 479.7.	<code>/proc/cpuinfo</code> , <code>cpufreq</code>
Memory	256 MiB in package; bootloader passes <code>mem=223M</code> ; kernel reports 173,848 kB after carve-outs.	<code>/proc/meminfo</code>
Display	3.2" TFT, 480 × 800, about 292 ppi, on the MDP's LCDC. fbdev <code>msmfb30_30001</code> : RGB565, 960-byte stride, virtual 480 × 1600 (two pages).	fbdev ioctl
Input	<code>touch_mcs6000</code> capacitive, multitouch protocol A; <code>kbd_pp2106</code> four-row QWERTY + d-pad; <code>gpio-slide-detect</code> (EV_SW SW_LID); <code>7k_handset</code> keys; volume, power and camera keys.	<code>/proc/bus/input</code>
Radios	BCM4329 over SDIO: 802.11 b/g/n (stock <code>wireless.ko</code> , LG NVRAM) and Bluetooth 2.1 + EDR on <code>/dev/ttyHS0</code> , no LE. CDMA 800/1900 + EV-DO Rev. A on the ARM9; that network no longer exists.	kconfig; docs
Camera	MT9T111, 2048 × 1536 YUV (sold as 3.2 MP AF); AAT1270 flash/torch driver for a white LED.	kconfig
Sensors, lights	KR3DH accelerometer, AKM8973 compass, GP2AP002 proximity, dock hall sensor; no ambient-light sensor. ROHM BD6084GU backlights; RGB notification LED.	i2c survey
Storage	512 MiB NAND as 10 MTD partitions (492.6 MiB mapped); 32 GB microSD (29.7 GiB).	<code>/proc/mtd</code>
Power	1,500 mAh removable. Charging is run by the modem core over RPC; Linux only reads state. LG rated 500 h standby (about 3 mA average).	public specs; docs

Table 3 Hardware, as read on the phone under the new OS.

Storage after the conversion

PARTITION	SIZE	NOW
<code>mtd0 boot</code>	5 MiB	Stock kernel + new stage-1 cpio. The only internal partition written.
<code>mtd2 recovery</code>	5 MiB	Stock recovery with root <code>adb</code> and <code>flash_image</code> : the way back.
<code>system</code> , <code>userdata</code> , <code>cache</code>	178 / 152 / 128 MiB	Android, untouched, unused. Every partition was dumped and checksummed first.
AMSS, APPSBL	19.5 MiB / 896 KiB	Modem firmware, bootloader (reads back 0xFF).
SD p1 ALLYR00T	2 GiB	ext3 root, mounted read-only first.
SD p2 ALLYDATA	27.7 GiB	ext3 <code>/data</code> , <code>data=ordered</code> : tokens, Wi-Fi, keys, settings.

What the 2010 kernel allows

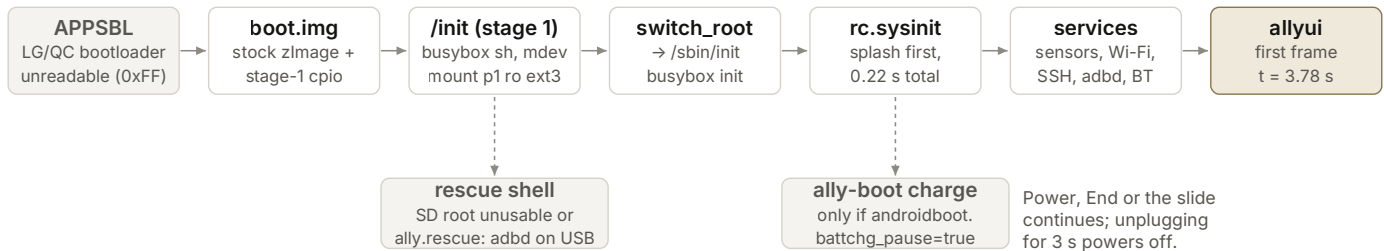
- **ext2/ext3 only**, no ext4; **no devtmpfs**: busybox `mdev`, and programs create their own nodes from `/sys/class/.../dev`.
- **Android power management**: early suspend and wakelocks; `mem` to `/sys/power/state` suspends once no wakelock is held.
- **Plain ARMv6 TLS**: no TPIDRURO thread register, which rules out ARMv6K userlands (§3).
- **No BlueZ 5 daemon**: it needs the 3.4 management API, so pairing is implemented directly (§3).
- **A VT with fbcon**: the UI switches to `KD_GRAPHICS` so text never draws over it.

2

Boot chain

The stock kernel stays: it is the only one with drivers for this board. A stage-1 initramfs packed beside it hands over to the SD card. Nothing else on internal flash changed.

NORMAL BOOT



The rescue path needs no SD card and no network: a broken root still leaves adb over USB.

Figure 1 Bootloader to first frame, with the rescue and charge branches.

Boot image. `os/mkboot.py` writes a new `cpio`, gzips it and packs an Android v0 boot image in pure Python: kernel base `0x200000` (kernel `0x208000`, ramdisk `0x1200000`, tags `0x200100`), 2048-byte pages, command line `mem=223M console=ttyMSM0,115200n8 androidboot.hardware=aloha`. Flashed from stock recovery with `flash_image` and read back with a matching MD5.

Stage 1. A 30-line busybox script: mount `proc`, `sysfs` and a 1 MB tmpfs `/dev`, `mdev -s`, wait up to 10 s for `mmcblk0p1`, mount it read-only, check `/sbin/init`, move `/dev`, `switch_root`. Any failure, or `ally.rescue` on the command line, loops `adbd` on USB instead: a broken root is always repairable.

Charge boot. Ghidra on stock `init` showed the charger boot is recognised by exactly one token, `androidboot.battchg_pause=true`. `ally-boot` charge keys off that token alone, before `/data` is mounted, so a warm reboot with a cable always reaches the UI.

Shutdown. `ally-power off|reboot|recovery` records the request and signals busybox `init` (USR2 or TERM). `rc.shutdown` TERMS everything, waits 2 s, KILLS, syncs, unmounts, then calls `reboot-to` to itself because busybox cannot pass a reason such as `recovery`. Power-off ends in the kernel's `msm_pm_power_off` (read through `/dev/kmem`): `proc_comm PCOM_POWER_DOWN` to the modem.

SECONDS AFTER KERNEL START · FROM KERNEL-LOG MARKERS

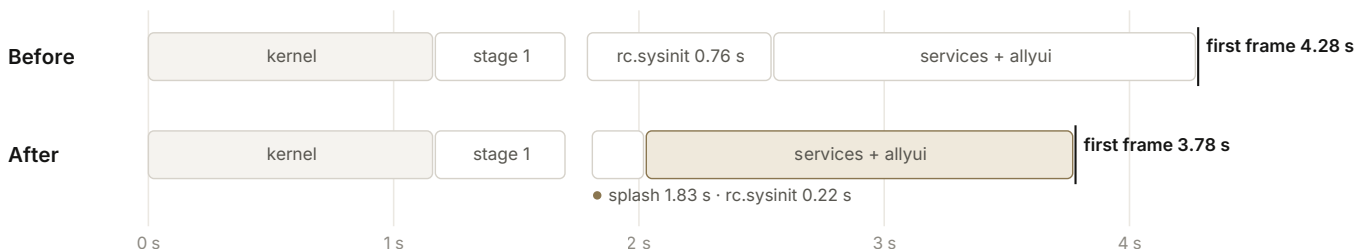


Figure 2 Kernel start to first frame, from kernel-log markers. Round 3's lazy fonts later cut the UI's own start (process to first frame) from 600–718 ms to 191–231 ms; the whole chain now measures 3.3 s. Shutdown and bootloader add about 12 s before `t = 0`.

STEP	BEFORE	AFTER	CHANGE
<code>rc.sysinit</code>	0.76 s	0.22 s	splash drawn first; the second full <code>mdev -s</code> (0.23 s) moved to the background
Splash on screen	—	1.83 s	<code>ally-boot</code> draws it in the UI's typeface, rotated if the slide is open
UI process → first frame	600, 718, 683 ms	231, 196, 191 ms	fonts loaded lazily from static data (three cold starts each)
Kernel → first frame	4.28 s	3.72–3.80 s, then 3.3 s	round 1 boot work, then the round 3 UI start
Reboot → SSH reachable	22–25 s		mostly shutdown, bootloader and Wi-Fi association

Table 4 Boot timings on the phone.

3

Reverse engineering

The kernel had the drivers; the knowledge of how to drive them lived in LG's closed Android libraries. Each subsystem was brought up by decompiling those binaries and replaying their exact call sequences from plain Linux. A coding agent drove Ghidra 12.1.4 headless through REA (analyze, search, decompile, xrefs; about 17 s per call on `liboemcamera.so`), with `objdump`, `/proc/kallsyms`, `/dev/kmem` reads and small static probes on the phone. Every probe ran with a timeout and a remote escape.

SUBSYSTEM	READ FROM	FINDING	RESULT
Userland	Alpine armhf crashes	Alpine targets ARMv6K and reads the thread pointer from CP15 TPIDRURO. This kernel (<code>CONFIG_CPU_V6</code>) never loads it, so every binary segfaulted at start.	Buildroot for exactly <code>arm1136jfs</code> ; <code>musl</code> reads TPIDRURO only if <code>AT_HWCAP</code> says TLS.
Audio	<code>libaudio.so</code> , <code>AudioStreamOutMSM72xx::write</code>	The ALSA <code>msm-pcm</code> path oopses. The DSP refuses <code>AUDIO_START</code> (EIO) unless both buffers are prefilled.	Direct <code>msm7k</code> DSP devices; speaker endpoint 0, handset 6; mic at 16 kHz mono.
Accelerometer	<code>sensors.aloha.so</code>	KR3DH is silent until a flag is set on <code>/dev/akm8973_aot</code> (0x4002A113 flag, 0x4002A118 delay); <code>akmd2</code> then reports through the compass device, ~720 counts/g.	Built, then removed: the slide alone sets orientation. Flag cleared, 0 CPU ticks.
Camera	<code>mm-qcamera-test</code> , ~220 procedures	Control codes, <code>msm_ctrl_cmd</code> layout, <code>pmem</code> registration, preview seeding, snapshot/JPEG order, six callback variables set with <code>dsym</code> .	<code>camhal</code> : our C linked to stock bionic, stock <code>/system/bin/linker</code> as <code>PT_INTERP</code> .
Camera crash	faulting stack, return addresses	SIGSEGV at <code>liboemcamera+0x3dbac</code> : <code>vfe_util_update_asf_5x5</code> reads the Bayer tuning pointer at <code>cfgctrl+0x4a0</code> , NULL for a YUV sensor, whenever byte <code>cfgctrl+0x1cf8</code> is set by <code>zoom_vfe()</code> .	<code>camhal</code> clears the byte after the VFE start returns, ~130 ms before the first frame.
Flash	<code>libcamera.so</code> ; AAT1270 <code>flash_enable</code> in the kernel	Magic values on <code>timed_output/flash/enable</code> : 4660 on, 17185 off, 800 = 800 ms flash. Brightness 255 = flash current, 127 = seven ENSET pulses (torch); every other value is ignored. The UI had sent 100.	Torch always 127, after an "off" to reset the pulse count.
DSP wedge	kernel <code>vfe_7x_release()</code>	On last close it stops a running VFE and waits only 200 ms for an ack that never comes (the acking thread is gone), then gates the DSP clocks mid-stop. ADSP queue 25 never drains again; no reset path exists.	Made impossible: ordered stop, <code>sigwait</code> thread, <code>PDEATHSIG</code> parent, 3 s watchdog. 35/35 stress cycles clean.
Modem	<code>libcm</code> , <code>liboncrpc</code> , <code>libril-qc-1</code>	Call Manager over ONC RPC on <code>rpcrouter 30000000:892a3c1c</code> ; callback service 0x31000000 registered with <code>ioctl 0xc004c102</code> ; <code>cm_ph_cmd_oprt_mode</code> is <code>proc 0x14</code> .	<code>ally-modem</code> requests LPM (6) at boot; see below.
Bluetooth	LG kernel headers, HCI traces	The kernel never sets the HCI event mask, so SSP events are lost ("Authentication Failure"). LG's <code>hci_conn_info</code> is 28 bytes, not 16: BlueZ's layout corrupted the heap.	<code>ally-bt</code> sets the mask; MAC from Wi-Fi with the local bit set.
Framebuffer	disassembly of the ARM build	<code>musl</code> 's <code>memcpy</code> reached 36 MB/s into the framebuffer; the toolchain's compiler-rt copy loaded words and stored them as four <code>strb</code> bytes.	<code>ldmia/stmia</code> 8-register loop: 151.6–166 MB/s.

Table 5 What was recovered, from where, and what it changed.

Audio playback order

```
open("/dev/msm_pcm_out")
ioctl(AUDIO_GET_CONFIG 0x80046103)  buffer size, count
ioctl(AUDIO_SET_CONFIG 0x40046104)  44.1 kHz, channels
write(buf 1); write(buf 2)          prefill BOTH
ioctl(AUDIO_START      0x40046100)  only now
routing: /dev/msm_snd SND_SET_DEVICE 0x40047302,
        SND_SET_VOLUME 0x40047303
```

The modem result. `ally-modem` runs once from `inittab`, waits up to 90 s for Wi-Fi, makes at most two attempts in a forked worker with a watchdog, and can only request 5 (online) or 6 (LPM); it never sends power-off, offline or reset. It logs `oprt_mode=6 lpm (was 0 pwroff) rc=0` at 11.35 s. The "was 0" is the punchline: the modem had never been online, its searchers never voted against sleep, and equal 120 s screen-off windows look the same before and after (SMD interrupts 216/254 vs 215/250, `rpc_server` 60 vs 60, sleep-voter mask

Modem low-power mode

ONC RPC CALL	PROC
cm_client_init / release	0x53 / 0x54
cm_client_ph_reg	0x57
act / deact	0x69 / 0x6a
get_ph_info	0x1a
cm_ph_cmd_oprt_mode (6 = LPM)	0x14

Table 6 The modem calls, recovered with Ghidra; `libcm` itself is not used.

0x1031 in both). LPM ships as insurance. Charging, RTC, Wi-Fi, audio, camera, slide sleep and two reboots were rechecked with it applied.

The camera stack today. `ally-camera` (static musl) takes a session lock, creates `/dev/msm_camera/*` from sysfs and starts camhal with the stock libraries. Preview frames come back on stdout as `FRM0` + width + height + RGB565, converted from NV21 with fixed-point BT.601 tables; `snap` on stdin takes a photo in-session, answered by an `EVT0` JSON event; errors are `failed`, `busy`, `nodev` or `wedged`.

CAMERA MEASURE	RESULT
Start → first frame	0.96–1.57 s, median 1.50 s (0.8–1.3 s is the sensor's I ² C init)
Preview 400 × 240	18.5 fps in light, ~9.3 in the dark; 17 % CPU, 33 % rotated
Photo	1.40 s in session, 1.9–2.0 s with flash; 3.1–3.5 s cold; 1600 × 1200 JPEG q85
Stress	35/35 random cycles, 29/29 in-session snapshots

Table 7 Camera measurements on the phone.

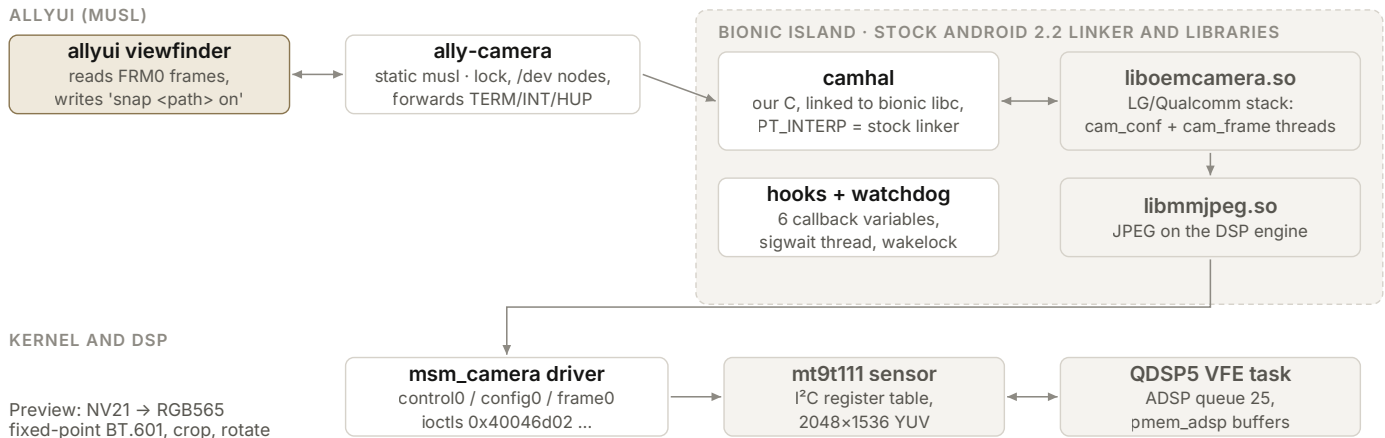


Figure 3 The camera stack: a musl front end, the "bionic island" running LG's libraries under LG's own linker, the kernel driver and the DSP's VFE.

4

The operating system

Buildroot builds a musl toolchain and a fully static userland for the exact core. Static linking keeps the root free of shared-library concerns and lets the stock bionic pieces coexist with it.

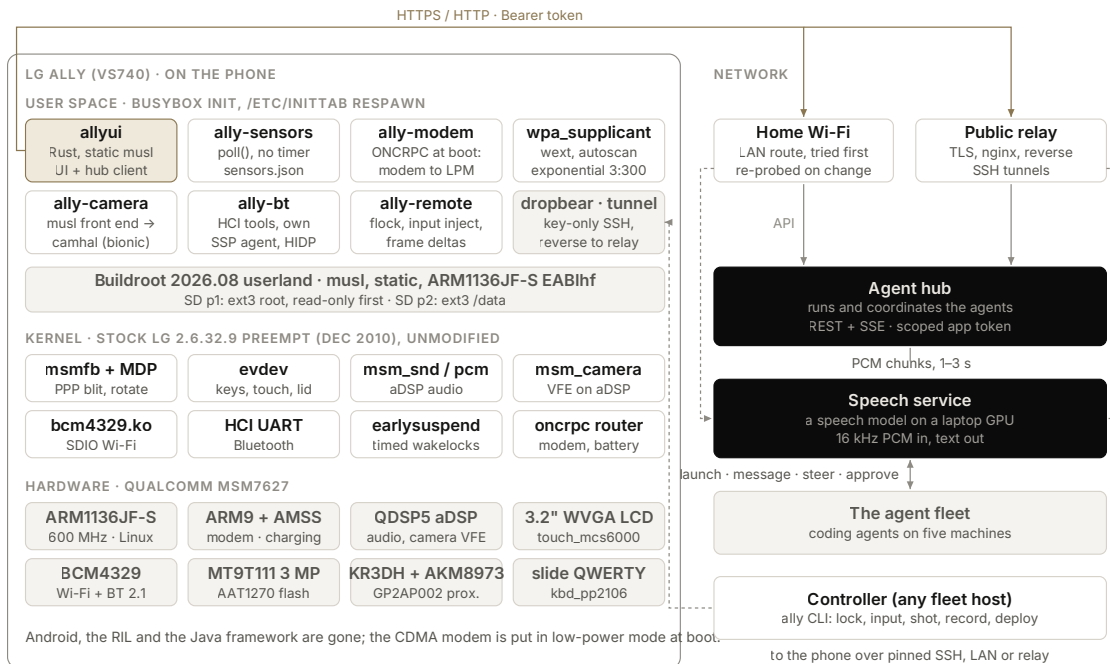


Figure 4 Processes, kernel interfaces and hardware on the phone, and what it reaches over the network.

Target	arm1136jf-s, EABIhf, VFPv2, ARM mode, musl, static only, -Os
Init	busybox 1.38 init + mdev; /etc/inittab is the only service manager
Packages	busybox, wpa_supplicant 2.12 (wext), wireless-tools, dropbear, e2fsprogs, alsa-utils, CA certificates, strace, evtest
Stock pieces kept	wireless.ko + firmware + NVRAM, akmd2, abdb, the BCM4329B1 patch, the camera's linker and nine libraries
Reproducible	os/build-rootfs.sh assembles Buildroot output, stock blobs (from the checksummed backup, never committed) and an overlay; rootfs-parity.sh confirms a fresh build is byte-identical to the phone

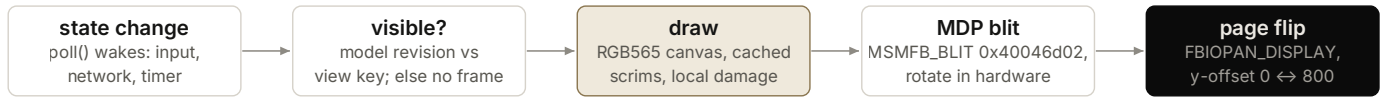
PROCESS	BUILT FROM	ROLE AND COST
allyui (+ run-ui)	Rust, static musl, 3.6 MB stripped	UI, hub client, audio, power policy. run-ui supervises it, resets the CPU floor before start and after exit (SIGKILL skips app cleanup), and persists the newest 64 KiB of log on exit.
ally-sensors	C, static	/run/ally/sensors.json: slide, headset, dock, proximity, battery. poll() only, no timer; 56 kB RSS, 0 ticks idle.
ally-modem	C, static	Once at boot: ONC RPC to put the modem in LPM (\$3).
wpa_supplicant, udhcpc	Buildroot	autoscan=exponential:3:300; DHCP 3 × 3 s then 60 s pause; hooks publish SSID/BSSID/IP to RAM for the route prober.
dropbear, tunnel, ally-remote	Buildroot; C	Key-only SSH, a reverse tunnel to the relay (372 KiB RSS), and the control helper (\$7).
ally-camera · camhal, ally-bt	C; static BlueZ 5.86 HCI tools	On demand only. Bluetooth is off by default (rfkill 0, GPIO 82 low); a scan takes 8.96 s.
akmd2, abdb, housekeeping	LG; stock; shell	akmd2 blocked in the driver (0 ticks); abdb for USB rescue; log trimming every 30 min.

Table 8 What runs. Against stock Android 2.2: 37 services in `init.rc` (zygote and the whole Java framework) and 128.5 MB of `/system` with 52 APKs, versus 8 programs under init and about 25 MB of root filesystem.

5

allyui internals

About 24.8k lines of Rust in one static ARMv6 binary. No toolkit, no async runtime, no GPU: the UI thread sleeps in `poll()` on input devices and a wake pipe, worker threads do the network, an audio thread owns the DSP, and every pixel is computed on the CPU into an RGB565 canvas.



Landscape frames are rotated by the display processor's PPP engine instead of the CPU. Changes that are not visible request no frame; status chrome and toasts repaint only their own rectangles; dialogs reuse one cached dimmed copy of the screen behind them. Scrolls shift existing pixels and draw the exposed strip. While a finger is down the CPU floor is raised for the gesture, then reset to 122.88 MHz.

Figure 5 One frame. Since round 2 the display processor rotates landscape frames; since round 3 invisible changes draw nothing and chrome repaints locally.

Present: rotation moved to the MDP

Landscape is the panel rotated, and rotating 768,000 bytes per frame on the CPU was the largest single cost. allyui now uses the stock PPP blit ABI: `MSMFB_BLIT 0x40046d02` with this kernel's 88-byte request (it has `sharpening_strength`, not `mdp_img.priv`). Changed rows are copied into a cached `MAP_SHARED` mapping of `/dev/pmem` (770,048 bytes); the blit writes the back page with rotation flags `R90 = 4`, `R180 = 3`, `R270 = 7`; `FBIOPAN_DISPLAY` flips. The MDP is used only for rotated updates of at least 32,768 px: for portrait and small updates a straight CPU copy is faster than staging plus a PPP transaction. An `ioctl` failure switches it off and redraws the frame on the CPU.

FULL-FRAME PRESENT	R0	R90	R180	R270
CPU copy	6.01	14.76	14.64	15.11
MDP blit	9.11	8.97	8.91	8.96

Table 9 Milliseconds per 480 × 800 present on the phone (fast-copy build). Every rotation's MDP output matches the CPU's pixel for pixel.

Round 2: present path and CPU floor

MOCK WORKLOAD	PRESENT MS	TOTAL MS
Landscape conversation	18.1 → 9.7	26.0 → 15.4
Landscape shade	26.6 → 17.6	43.4 → 29.6
Landscape fleet	18.6 → 14.1	24.7 → 19.6
Portrait fleet	15.6 → 12.6	23.2 → 19.1
Card swipe (regressed)	8.1 → 16.6	—

Table 10 Round 2 on the phone. The card swipe got slower and still tracks the 60 Hz touch rate.

CPU floor. The floor had been stuck at 600 MHz: 82.54 % of all residency over 9 h of uptime. Input now raises it for a 250 ms hold, released on idle, sleep, exit and child death; `run-ui` resets it either side of the process. Touch move to pan, median: 17.26 → 4.53 ms. Idle at 122.88 MHz: 98.2–98.4 %.

Round 3: composition

WORKLOAD (MOCK DATA, PHONE)	RENDER MS	TOTAL MS	PROCESS CPU %
Landscape Wi-Fi panel	73.42 → 8.19	96.98 → 31.17	10.12 → 4.26
Portrait Wi-Fi panel	32.60 → 10.37	52.86 → 29.45	11.05 → 4.02
Portrait Rename dialog	66.16 → 19.31	88.27 → 38.07	9.57 → 2.79
Landscape End dialog	69.58 → 15.93	90.33 → 37.61	9.58 → 3.00
Portrait fleet fling	6.60 → 5.08	19.40 → 17.01	31.61 → 22.48
Landscape conversation fling	5.53 → 5.11	19.58 → 19.30	26.01 → 23.87
Idle, clock ticking	24.10 → 26.50	51.90 → 55.50	0.59 → 0.64

Table 11 Frame-count-weighted means of the logged aggregates over two complete 254 s runs under the same phone lock; counts include empty render attempts, so these are not fps. The idle pair shows no gain. All 35 simulator screens are pixel-identical before and after.

- **Retained scrims.** Overlays keep the raw and dimmed background (768,000 bytes each) and reuse them until a model revision or view key changes; blending skips opaque panel interiors.
- **Packed blending.** `fill_alpha` returns at alpha 0, fills at 255, and otherwise blends two RGB565 pixels per aligned u32. Tested equal to the scalar path for all 65,536 destinations × alpha 1–254.
- **Local damage** for the clock, battery, headers, reconnect and toasts; concurrent changes fall back to full composition to keep layering.
- **Visible invalidation.** Unchanged polls, offscreen sessions and hidden history request no frame.
- **Coverage cache** for rounded corners, discs, rings and lines: 512 entries / 512 KiB, masks over 16 KiB not kept.
- **Borrow, don't clone.** Lists, catalogs and attention items are drawn by reference; clipped rows are skipped.

Memory, layout and decoding

MEASURE (PHONE)	BEFORE	AFTER
Fleet idle VmRSS	10,684–10,824 kB	4,540–5,300 kB
After opening a 250 KB conversation	14,252 kB	5,320 kB
While streaming 32 → 64 KB	12,124 kB	5,712 kB
Process → first frame	600–718 ms	191–231 ms
Fonts + first-frame text (host)	5,353 KB, 59,107 allocs	826 KB live
Thumbnail, 3264 × 2448 JPEG	24 MB	1.3 MB

Table 12 Round 3 layout and memory work.

- **Lazy fonts** borrow static font data with a 256 KB least-recently-drawn glyph cache.
- **Incremental Markdown.** A streaming answer re-lays out from a checkpoint, not from the top (it was $O(n^2)$); long conversations lay out lazily from the bottom with scroll anchoring.
- **Bounded images.** One loader thread, newest request wins; JPEG decodes at DCT scale 1/1–1/8; the header is parsed before allocation; sources over 8.4 MP and progressive coefficient storage over 24 MiB are refused; the viewer decodes within 2.4 MP. QA found one more: a decoder freed after releasing its permit could briefly hold two ~24 MB allocations. Fixed in the shipped build.
- **Typed hub decode (host):** sessions 0.520 → 0.147 ms (3.54×), attention 0.283 → 0.031 ms (9.16×).

Scrolling and gestures

One scroller, pure arithmetic over a caller-supplied clock: fling decay 0.998/ms, start above 0.12 px/ms, stop below 0.02, 90 ms velocity window, 6 px/ms cap, rubber band 0.55, critically damped spring $\omega = 0.016/\text{ms}$ (settles in ~350 ms). Gestures: edge-back from the first ~22 px; swipe between agents (commits past 40 % or on a flick); tap the status bar to glide to the top; pull down to dismiss any panel (past ~1/3 or a flick); swipe a card for Rename/End; long-press to quote; pinch to 4×, double-tap 2.5×; one haptic tick at each commit point. The touch tracker follows two contacts on a protocol-A device.

Build. Dependencies: fontdue (vendored), serde/serde_json, libc, ureq + rustls, miniz_oxide, base64, png, zune-jpeg. cargo `zigbuild --release --target arm-unknown-linux-musleabihf`, LTO, one codegen unit. The same code builds for a laptop as a simulator: `--sim` renders the identical RGB565 canvas, replays scripted keys, touches, slides and rotations, and writes PNGs against recorded fixtures (`--mock`). Every screenshot in these notes comes from it.

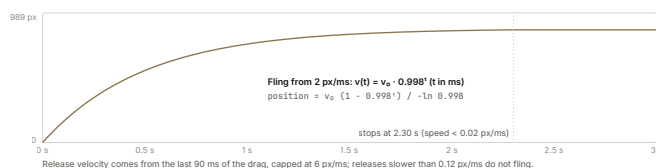


Figure 6 Offset over time for a 2 px/ms fling, computed from the scroller's constants.

6

Screens

Native-resolution captures from the simulator on a public fixture set (invented agents, no real accounts), current master. Keyboard shut is portrait; keyboard out is landscape; nothing else rotates the screen.

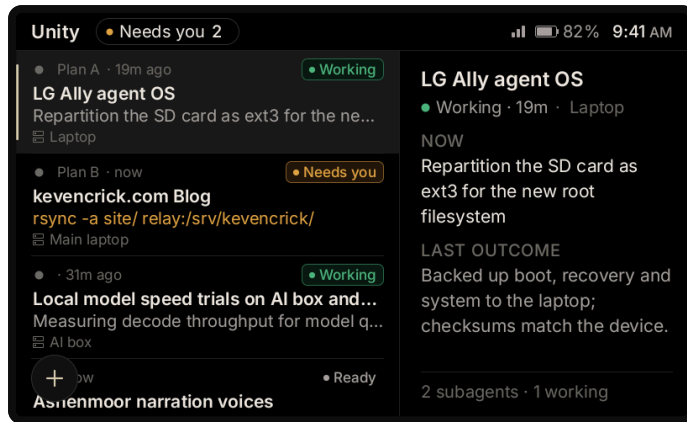


Figure 7 Fleet, keyboard out: list and the selected agent's detail. Agents that need you sort first; rows never jump as status changes.

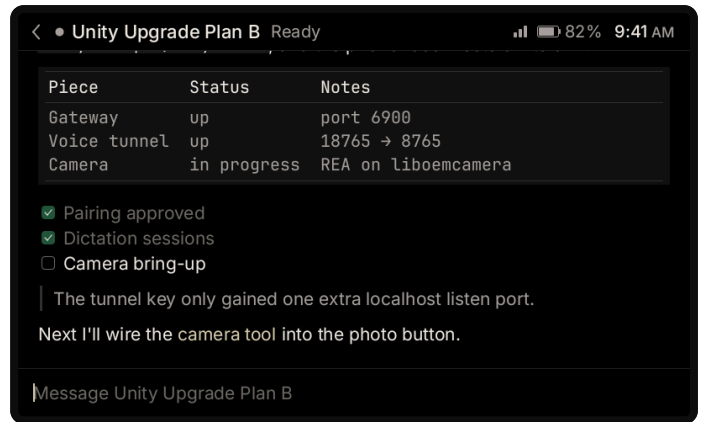


Figure 8 A conversation: Markdown, monospace tables, tool runs with diff counts, plans with step status. Output streams over SSE.

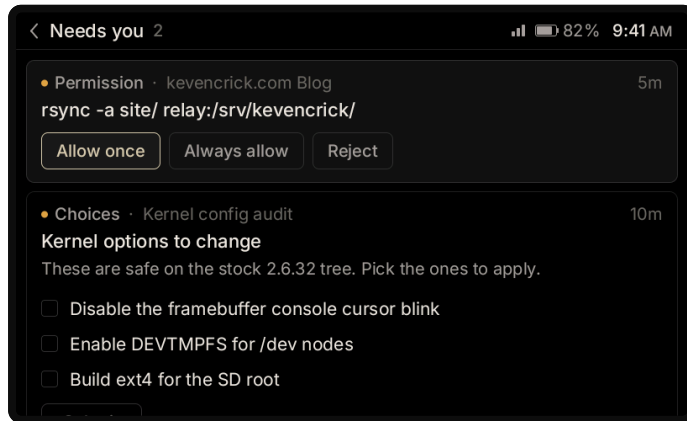


Figure 9 Needs you: every pending permission, choice and question across the fleet, answered in place.

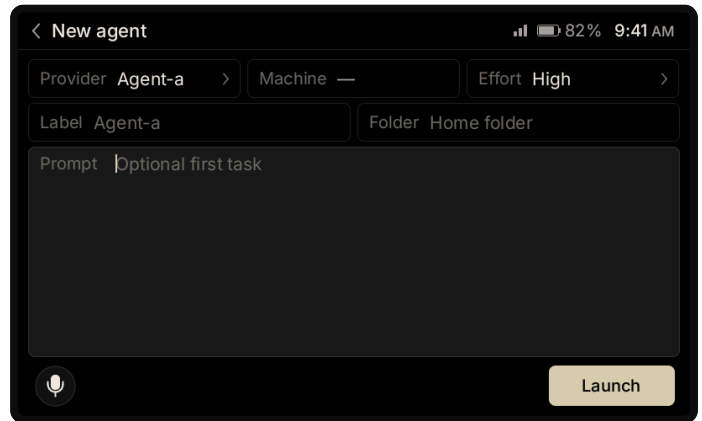


Figure 10 New agent: provider, machine, plan, model, effort, label, folder, prompt, all from the hub's live catalog.

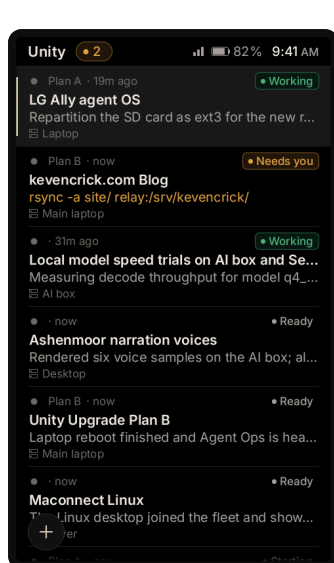


Figure 11 Fleet, keyboard shut.

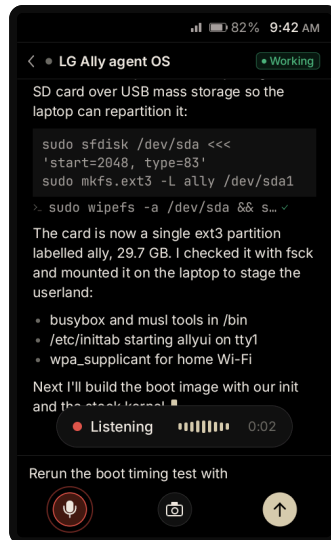


Figure 12 Dictation: Call to start, Call to finish; nothing sends before Enter.

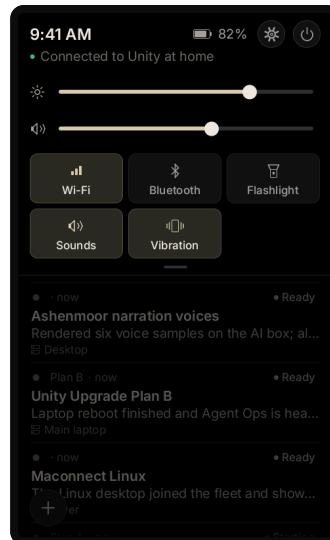


Figure 13 The shade: brightness, volume, radios, torch.

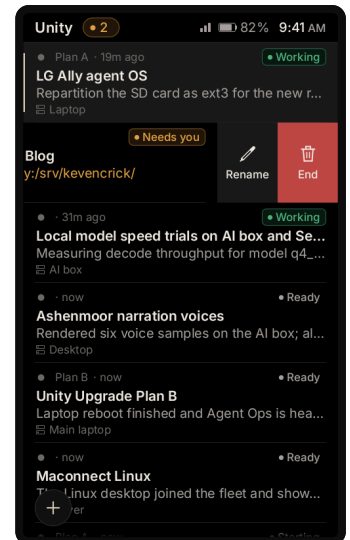


Figure 14 A card swiped left: Rename and End.

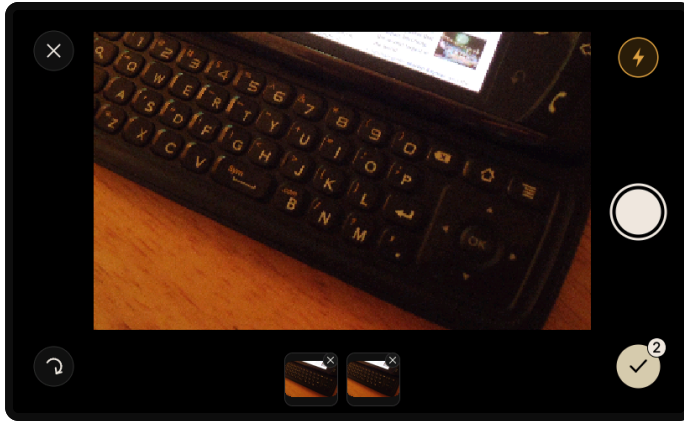


Figure 15 Viewfinder in a conversation: up to six photos per message, flash armed, per-orientation rotation fix.

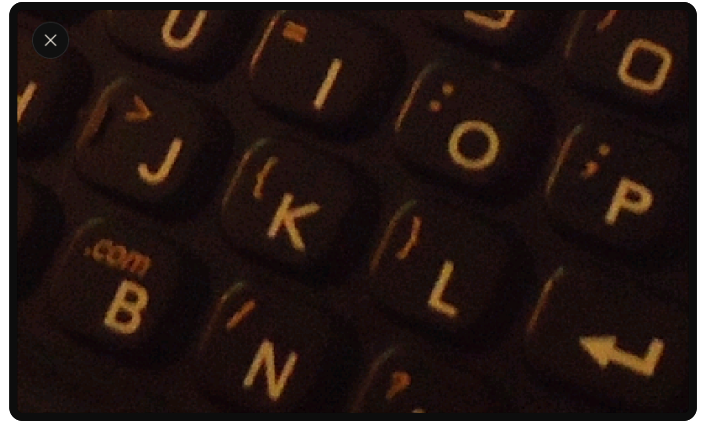


Figure 16 Picture viewer, pinched in: decoded within 2.4 MP, pull down to dismiss.

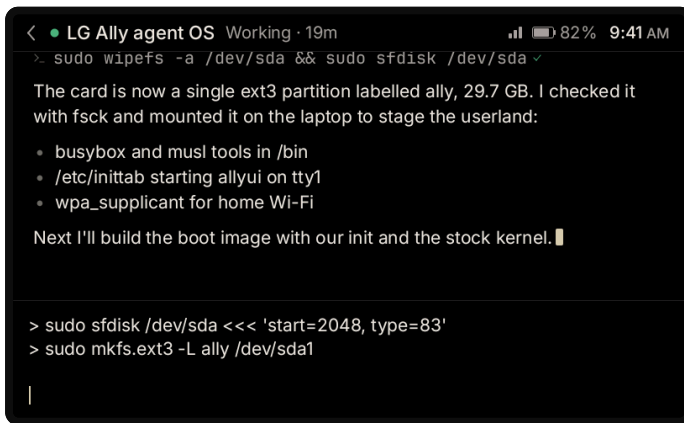


Figure 17 Long-press a message to quote it into the draft.

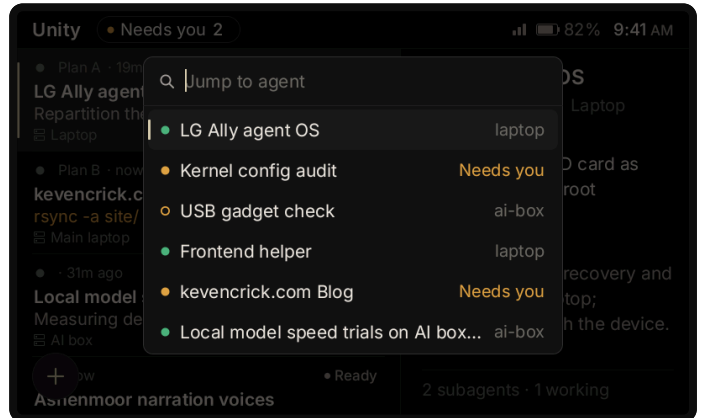


Figure 18 Search opens a quick switcher that filters by name, machine and provider.

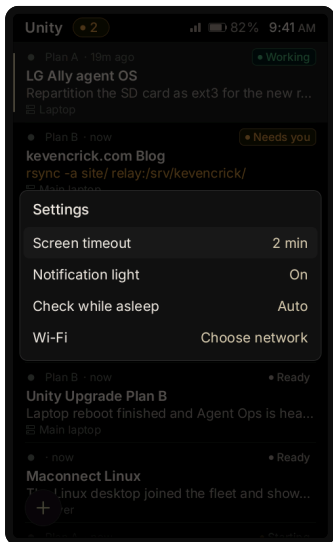


Figure 19 Settings: check while asleep (Auto: 3 → 6 → 12 min).

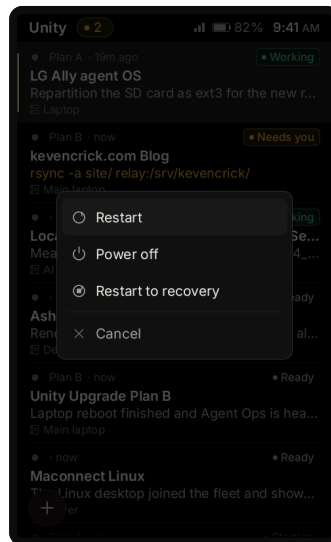


Figure 20 Hold power: Restart is first and selected.

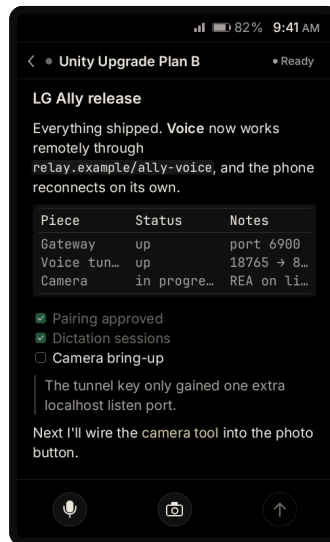


Figure 21 A conversation, keyboard shut.

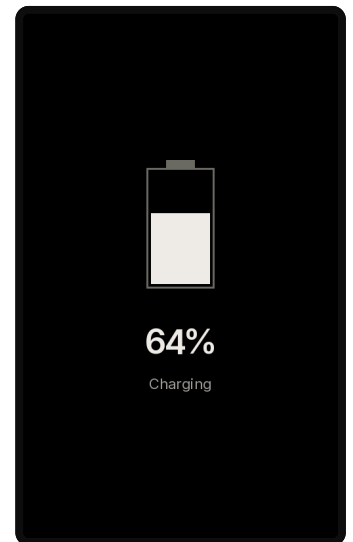


Figure 22 Off-charging screen, drawn by ally-boot.

KEY	ACTION	KEY	ACTION
Call	Start / finish dictation	Camera	Viewfinder, shutter; hold for torch (3 min auto-off)
Enter / Alt+Enter	Send follow-up / send as steer	End, Power	Screen on/off; hold Power for the menu
Search · Home · Menu	Switcher · fleet · shade	Fleet keys	1-9 jump, n new, a needs you, s settings, type to filter

Table 13 Keyboard-first: Shift and Alt are sticky as on the stock phone, SYM opens a picker. Notifications: a two-note chime for a finished turn, a three-note cue and longer buzz for a request, both synthesized at 44.1 kHz; the LED shows amber (needs you) or green (unread) with the screen off.

7

Connectivity and the control plane

The phone is an ordinary external app to the agent hub: it pairs once, holds a scoped token and uses the public API, at home or through a public relay. A second, separate plane lets any fleet machine hold, drive, capture and update the phone over pinned SSH.

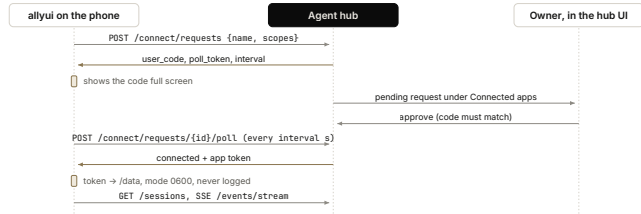
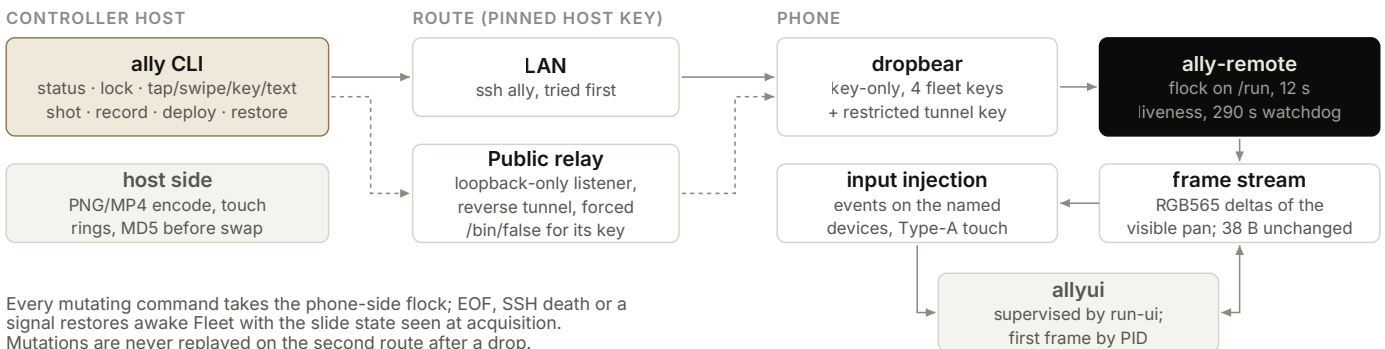


Figure 23 Device-code pairing. Scopes: read, launch, message, steer, control, approvals. The hub needed no change.

NEED	CALL (UNDER /AGENT-OPS/V1)
Fleet	GET /sessions?view=ops, typed decode, tree from parent ids
Transcript	GET /sessions/{id}/events, backward cursors; live SSE resumed with Last-Event-ID
Send, steer	POST /sessions/{id}/messages, follow_up or steer, photos as base64
Launch	GET /catalog (cached by hash, retried 5–300 s), POST /agents
Requests	GET /attention only when attention fields change, or 60 s safety

Table 14 Every mutating call carries a stable command id per user action, reused on retry, so a resend after a drop can never launch or send twice.

Remote control: ally



Every mutating command takes the phone-side flock; EOF, SSH death or a signal restores awake Fleet with the slide state seen at acquisition. Mutations are never replayed on the second route after a drop.

Figure 25 The control plane. Agents use it to test, record and update the phone without anyone touching it.

PROPERTY	MECHANISM
One controller	flock on /run/ally-control.lock, 30 s acquire timeout; nested commands share the parent's session through a private Unix socket
Dead controller	4 s heartbeat; 12 s liveness deadline; 290 s hard watchdog; controller wakelock expires at 300 s even after SIGKILL

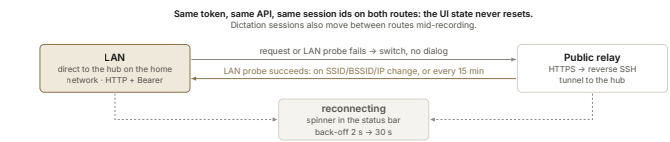


Figure 24 Route selection. Since round 3 the LAN is re-probed when Wi-Fi identity changes (supplicant and DHCP hooks), with a 15-minute fallback, instead of every 60 s.

BACKGROUND POLLING	REQ/H BEFORE	AFTER	MB/H AFTER
Awake, active (5 s poll)	1,440	780	100.17
Awake, idle (15 s poll)	480	300	35.19
Asleep, 3 min fixed	60	40	3.61
Asleep, quiet, Auto at 12 min	60	10	0.90

Table 15 Model, for a captured 135,365-byte session list and 45,104-byte attention list. Auto backs off 3 → 6 → 12 min while nothing changes; any change, working agent, request or wake resets it.

PROPERTY	MECHANISM
Always restored	EOF, SSH death, HUP/INT/TERM → awake Fleet, slide state as snapshotted by EVIOCGSW
Input	named devices: Back 248, Home 102, Center 232; keys a device doesn't advertise are refused
Routes	pinned host key; LAN first, relay loopback-only listener; auth failures stop, only network failures fall back; mutations never replayed
Relay key	restricted: forced <code>/bin/false</code> , a shell attempt returned 1 and ran nothing

DEPLOY AND CAPTURE	MEASURED
Recording	visible-pan RGB565 deltas on a second SSH stream; 30 frames in 6.000 s at 5.0 fps on LAN and relay (≈187 KB); unpaced LAN 8.91 fps
Frame size	unchanged 38 B; incompressible up to 768,032 B

Table 16 The ally CLI: 28 passing tests including rollback, copy-truncated logs and touch lift. Why the gate matches by PID: a valid deploy was once rolled back because log rotation moved the first-frame line above a line-count cutoff.

8

Dictation

A 600 MHz ARM11 cannot run speech recognition, so the phone records and a speech model on a laptop GPU transcribes (fp16, about 2.2 GB of VRAM, warmed once; a CPU fallback model without CUDA). The service is one Python file on the standard library's HTTP server.



Nothing reaches the agent until Enter. A second Call press posts the final body to /finish (a full-beam pass). Retries reuse the same seq and body; the server de-duplicates them and replays a completed finish.

Figure 26 Live dictation. Each response carries committed text, which only grows, and a tentative tail the phone replaces every time.

- **Wire.** Raw S16LE mono at 16 kHz (32,000 B/s); bearer token compared in constant time before any body is read; 15-minute sessions, two per token, 10-minute idle expiry.
- **Cheap re-transcription.** Only the uncommitted tail is re-transcribed (greedy). Segment timestamps commit text at a ≥ 0.8 s pause or once the tail passes 20 s; 200 characters of context carry wording across. A silence gate and a hallucination filter keep quiet audio out; `finish` runs a full-beam pass.
- **Round 3.** Chunks adapt 1–3 s to round-trip time and coalesce up to 10 s pending; samples are freed per request (before: up to 28.8 MB retained per 15 min); the mic opens before a cached cue; `TCP_NODELAY` on the server (isolated median 40.99 → 0.24 ms per response).
- **Retries.** Same seq and body; duplicates ignored, out-of-order rejected, a completed finish replayed for 120 s. QA found a retry of a long final body (> 2.048 s) getting 413; fixed.

MEASURE	BEFORE	AFTER
Call → mic open	622, 625, 641 ms	28, 29, 30 ms
First PCM	—	121–127 ms
Speech end → final text, LAN	0.917 s	0.581 s
Relay-like, 300 ms RTT	1.990 s	1.616 s
Poor link, 800 ms RTT, 48 kB/s	17.088 s	4.293 s
19.4 s clip, Σ RTT at 0.5 / 1 / 2 s chunks	11.673 / 5.978 / 3.039 s	

Table 17 Benchmarks with identical final text in every profile; the chunk sweep is round 2's isolated test.

9

Performance and power

Software that does almost nothing while nobody is looking leaves the hardware's idle states as the only real cost. What can be shown so far is that the software is no longer the cost; what cannot be shown yet is battery life.

RUNNING COST	RESULT
RAM in use, whole system	9.5 MB of 173.8 MB, awake (was 15.3 MB before round 3)
allyui resident	4.5–5.3 MB idle on the fleet; 5.3 MB after a 250 KB chat
allyui CPU, idle, screen on	0.59–0.64 % in round 3's idle windows
allyui CPU, screen off	0 ticks in 60 s; system 0.1 % (round 1)
Hub thread CPU, awake	2.76 % → 1.94 % of an awake window
CPU at 122.88 MHz	98.2–98.4 % idle; 93.9–94.9 % in an awake window
Other services	sensors, supplicant, DHCP, adbd, akmd2: 0 ticks
Wi-Fi receive, awake	133.02 → 87.06 MB/h (extrapolated; includes SSH)

Table 18 Measured on USB power.

INTERACTION	RESULT
Fleet fling	17.0 ms total per frame portrait, 17.3 landscape
Conversation fling	19.3–19.9 ms per frame
Card swipe	~19 ms, follows the finger at the touch rate
Dialogs, Wi-Fi panel	29–50 ms (were 53–97 ms)
Touch move → pan	4.53 ms median
Framebuffer copy	151.6–166 MB/s (libc: 21.9–36.1)
250 KB chat, worst frame	1,336 → 92 ms
Streaming answer frame	190–260 → 22–28 ms

Table 19 Frame totals are round 3 means; long-chat figures are from the layout work.

Sleep

BACKGROUND WAKE	BEFORE	AFTER
Wakelock held per wake	20.000 s	0.844 s (QA: 1.378 s, never expired)
Hub CPU for that wake	0.24 s	0.16 s
Forced awake, 3 min cadence	400 s/h (11.1 %)	~16.9 s/h
Forced awake, quiet Auto	—	~4.2 s/h (model)
Wi-Fi power save	constant awake mode	on in the suspend hook; LAN ping 177–465 ms on vs 5–8 ms off

Table 20 The wakelock is now owned by the poll's completion; an older completion cannot unlock a newer poll.

Asleep, allyui wakes on an RTC alarm, holds a timed wakelock for one poll and the LED update, and releases it. Logs moved to RAM (/run/allyui.log , 256 KiB cap) and the session cache is written only after changes, at most every 30 s, without fsync. The LED is a steady PMIC level: no blink timer.

Not yet proven. The battery reports whole percent and voltage only: no current or charge counter. Every measurement had USB attached, whose usb_bus_active wakelock blocked full suspend in every sample (suspend count 0). Battery drain per hour, deep-suspend residency, whether the DSP's sleep vote matters, Wi-Fi host-wake while suspended and away-from-home scan rates are all unmeasured; the bar is LG's 500 h rated standby, about 3 mA.

10

Limits, verification and sources

AREA	STATE
Battery	Battery-only suspend qualification is the next test.
Rendering	Full-screen flings take 17–20 ms of work per frame; a guaranteed 60 fps would mean drawing straight into the framebuffer in panel orientation.
Camera	Wedge prevented, not recoverable without a reboot. NV21 → RGB565 on the MDP is proposed (17–33 % of preview CPU), not built.

RELEASE GATE (ROUND 3)	RESULT
Rust release tests	179 passed, 2 intentionally ignored
Speech service	65 passed
Full Python gate	99 passed, 55 subtests
Phone, live (read-only)	16/16 steps in 67.33 s
Phone, mock (acting)	7/7 in 72.66 s
Phone, Auto polling	5/5 in 201.73 s

AREA	STATE	RELEASE GATE (ROUND 3)	RESULT
Bluetooth	Pairing and bonding verified with a laptop; a real keyboard end to end is untested; no controller sleep while on.	Review findings	4 medium, 2 low, all fixed and rechecked
Charger power-on	Tested in test mode and warm reboots; a cold power-on by the charger is unverified.	Table 21 Independent review over the integrated round 3 build. Live phone runs never send, launch, answer or approve anything; acting tests use mock data.	
GPS, auto-brightness	GPS lives in the modem (ONC RPC <code>loc_api</code>); no ambient-light sensor exists.		
Picture viewer	Pinch, double-tap and dismiss proven in tests and the simulator, not yet by hand.		

How it was built

The work ran as one project ticket and was carried out by coding agents coordinated through the same hub the phone now controls: a lead agent for the OS and boot chain, helpers for the interface, the speech service, the camera, Bluetooth, power, rendering, layout, remote control and QA, each in its own worktree and serialised on the phone by the control lock. The human part was directing, approving, and pressing Power when an early test forgot its timeout, which is why every test path now has a timeout and a remote escape. The repository holds 131 commits between 13 Oct 2026 and 14 Oct 2026.

Sources

- Repository `lg-ally`: `docs/` (hardware, sensors, power, Bluetooth, remote control), `docs/research/round2-*.md` and `round3-*.md`, `qa/review-round2.md` and `review-round3.md`, `camera/NOTES.md`, `voice/README.md`, `app/README.md`, and the source under `app/`, `os/`, `camera/`, `modem/`, `voice/`, `tools/`.
- The phone, read-only: `/proc`, `cpufreq` and `fbdev` sysfs, wakelock and interrupt counters, kernel log markers.
- "LG VS740", Wikipedia; PhoneScoop LG Ally specifications; iFixit LG Ally VS740 device page.
- REA (github.com/morluto/rea), a reverse-engineering CLI over Ghidra; Buildroot 2026.08; BlueZ 5.86; Inter and JetBrains Mono (SIL OFL).
- Photographs: "LGAlly.JPG" by Indiemon, Wikimedia Commons, public domain (cover, adapted). iFixit LG Ally VS740 device image, CC BY-NC-SA 3.0 (portrait mockups, adapted; this adaptation is shared under the same licence).