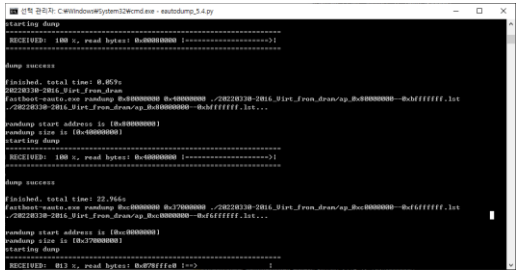
Ramdump & Debugging

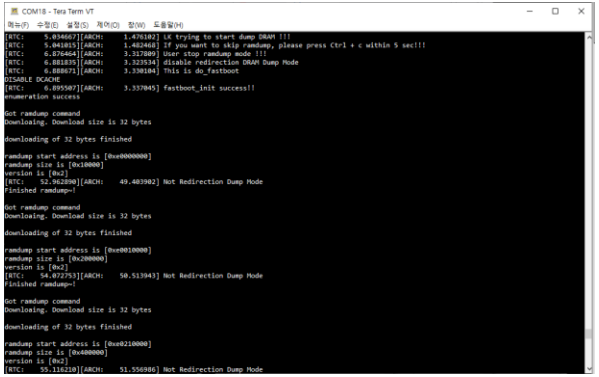
2022.04. | Samsung

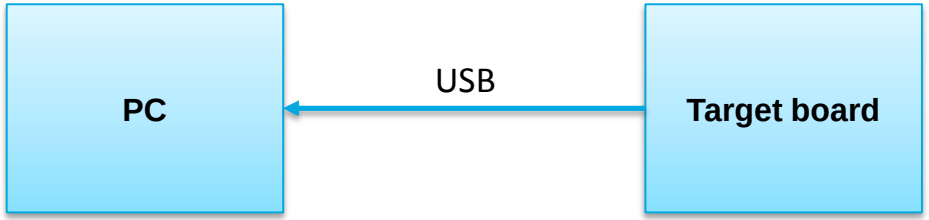
RAMDUMP

- **Exynos Auto V9 supports ramdump tool to dump entire DRAM area**

ap_0x8c0000000--0x8fffffff.lst
ap_0x9c0000000--0x9fffffff.lst
ap_0x80000000--0xbfffffff.lst
ap_0x880000000--0x8bfffffff.lst
ap_0x900000000--0x93fffffff.lst
ap_0x940000000--0x97fffffff.lst
ap_0x980000000--0x9bfffffff.lst
ap_0xa00000000--0xa3fffffff.lst
ap_0xa40000000--0xa7fffffff.lst
ap_0xa80000000--0xabfffffff.lst
ap_0xac0000000--0xaffffffff.lst
ap_0xb00000000--0xb3fffffff.lst
ap_0xb40000000--0xb7fffffff.lst
ap_0xb80000000--0xbbfffffff.lst
ap_0xbc0000000--0xbffffffff.lst
ap_0xc0000000--0xf6fffffff.lst
ap_0xfb600000--0xffffffff.lst
BOOT_bootlog_0xe5c00000--0xe5cfffff.lst
HYP_DSS_0x9fa580000--0x9fa5fffff.lst
HYP_log1_0x9fa401000--0x9fa4fffff.lst
HYP_log2_0x9fa500000--0x9fa57fffff.lst
HYP_pram_0x9fa400000--0x9fa7fffff.lst
IVI_header_0xe3c00000--0xe3c0fff.lst
IVI_kernel_0xe3c10000--0xe3e0fff.lst
IVI_kevents_0xe4450000--0xe4a4fff.lst
IVI_platform_0xe3e10000--0xe420fff.lst
SFI_log1_0xfb800000--0xfb8fffff.lst
SFI_log2_0xfb900000--0xfbbfffff.lst
SYS_arrdump_0xe1a50000--0xe254fff.lst
SYS_header_0xe0000000--0xe000fff.lst
SYS_kernel_0xe0010000--0xe020fff.lst
SYS_kevents_0xe2b50000--0xe314fff.lst
SYS_platform_0xe0210000--0xe060fff.lst
SYS_s2d_0xe0f10000--0xe470fff.lst

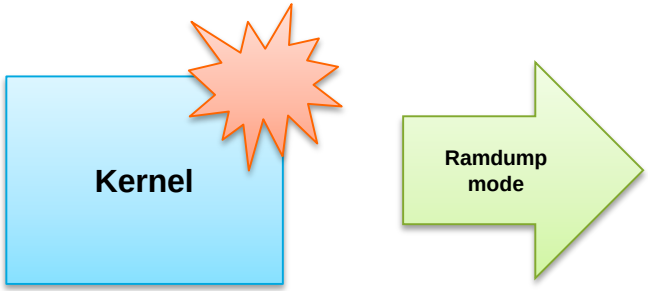


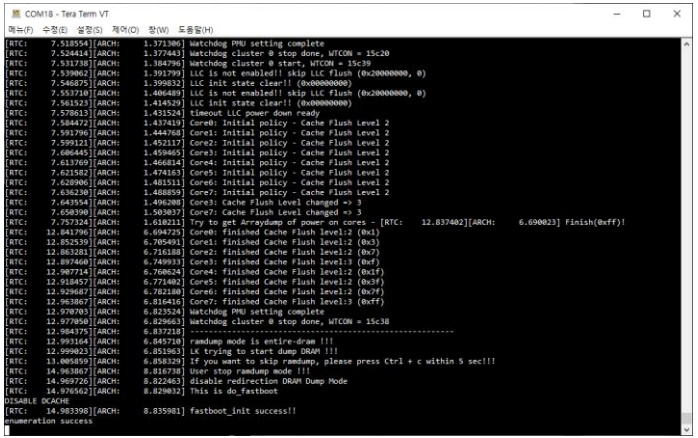




ENTERING RAMDUMP MODE

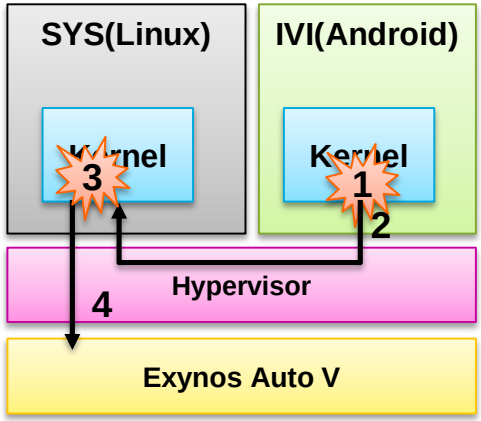
- **LK enters ramdump mode when**
  - Warm Reset
  - Watchdog Reset
  - Reset after Kernel Panic

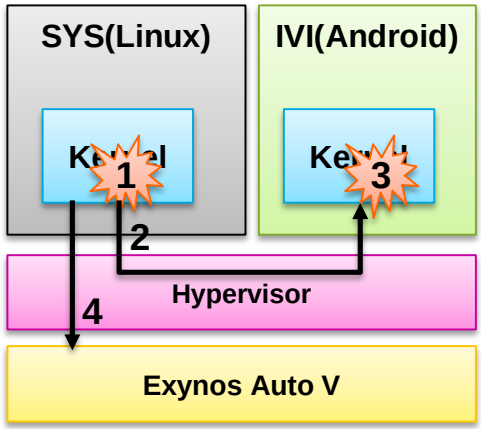




VM CASE

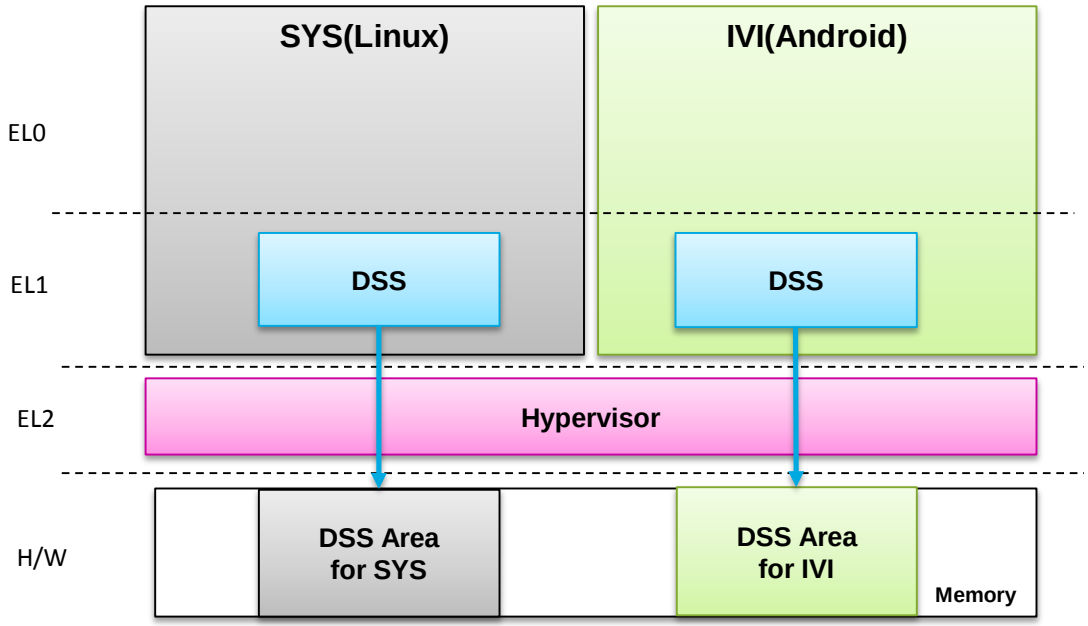
- **Hypervisor supports “Panic Propagation” to enter ramdump mode**
  - When IVI domain panic,
    - Hypervisor propagates panic to IVI domain
    - After SYS domain gets “Panic propagation”, SYS domain panic and reset whole system to enter ramdump mode
  - When SYS domain panic,
    - Hypervisor propagates panic to IVI domain, and SYS domain waits for IVI domain panic
    - After IVI domain panic, SYS domain resets whole system to enter ramdump mode





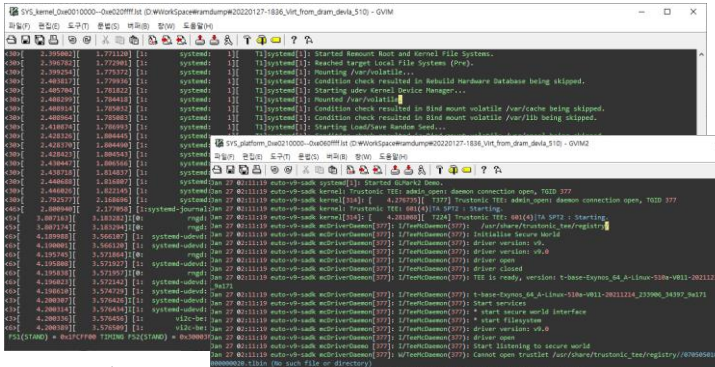
DSS

- **Debug Snapshot stores debug information to reserved DRAM Area**
  - Kernel log
  - Platform log
  - AARCH64 GPRs
    - X0~31, SP, PC
  - MMU Regs



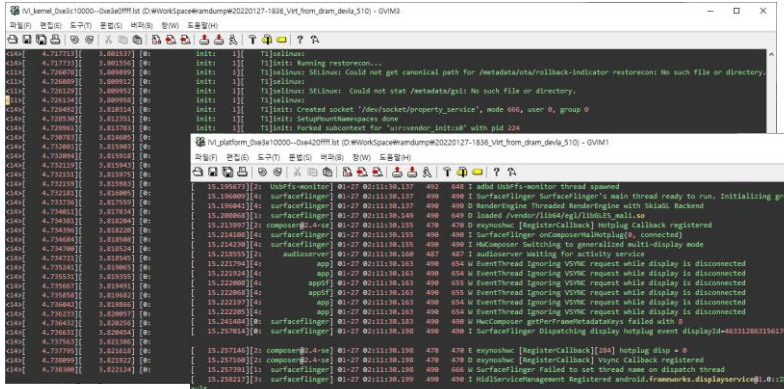
DSS

- You can get the kernel & platform logs from RAMDUMP



Linux Kernel Log

Linux Platform Log



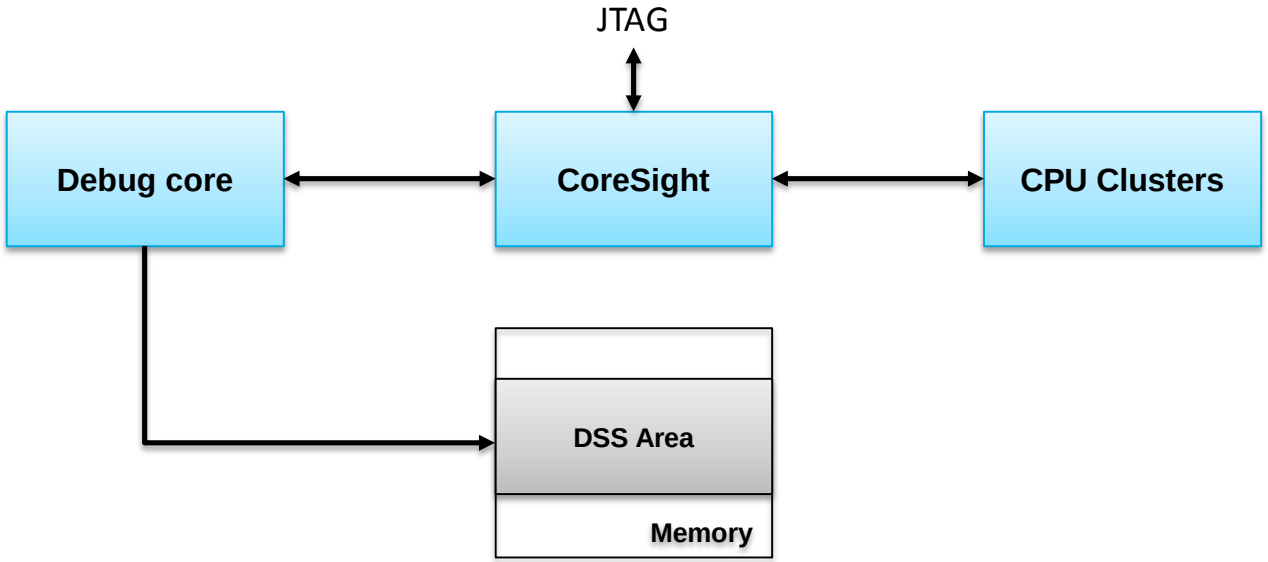
Android Kernel Log

Android Platform Log

- AARCH64 GPRs and MMU Regs is used for iTSP RAMDUMP Simulation

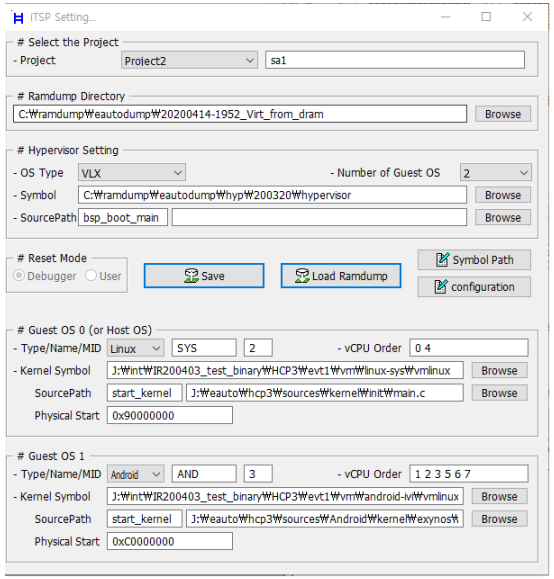
DEBUG CORE

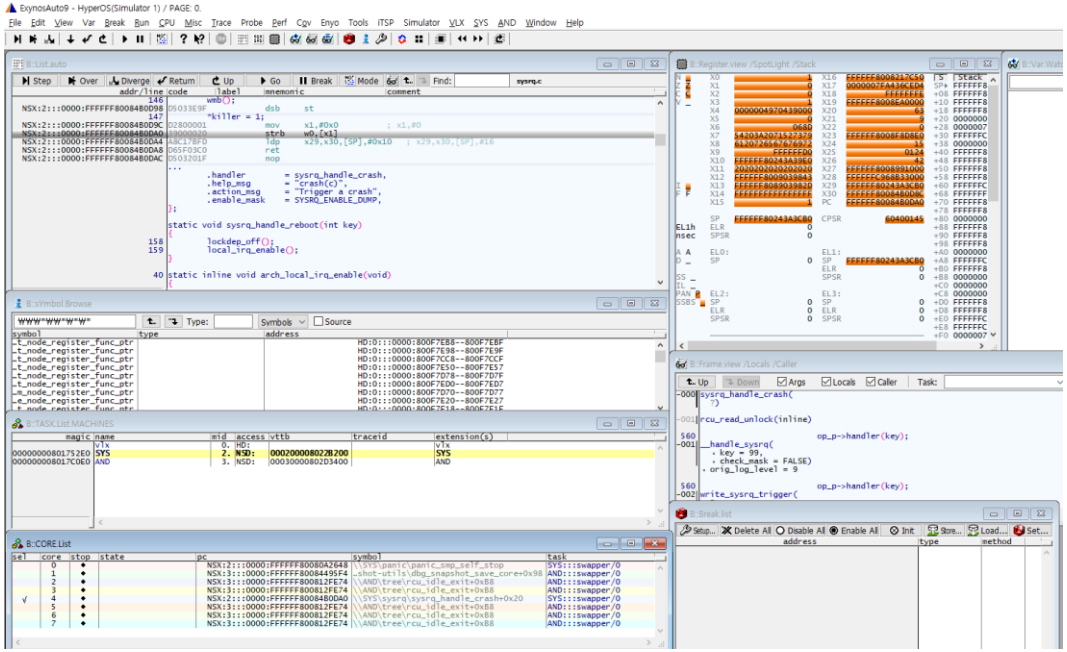
- **CoreSight technology enables on-chip debug and trace complex multi-SoCs**
  - with JTAG debugger (Ex. Trace32)
- **Debug Core**
  - can access CPU Clusters through CoreSight like JTAG interface
  - dumps each cores GPRs in each CPU Clusters except panicked cores to analyze lockup issues



RAMDUMP SIMULATION

- **You can simulate the situation of panic or lockup through iTSP simulator and Trace32**
  - You can get support iTSP for Exynos Auto V9 from “Hancom-MDS”





Q&A