

# Updates of Android on RISC-V

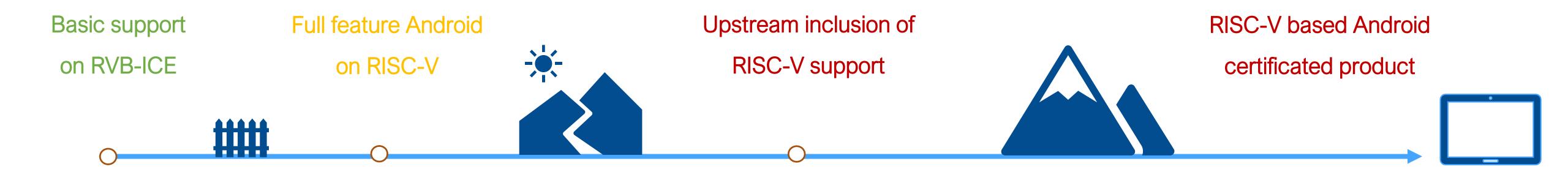
Mao Han

April. 19th, 2022





## Milestone



#### Challenge

- Poor IP support
- Incompatibility
- Lack of verification

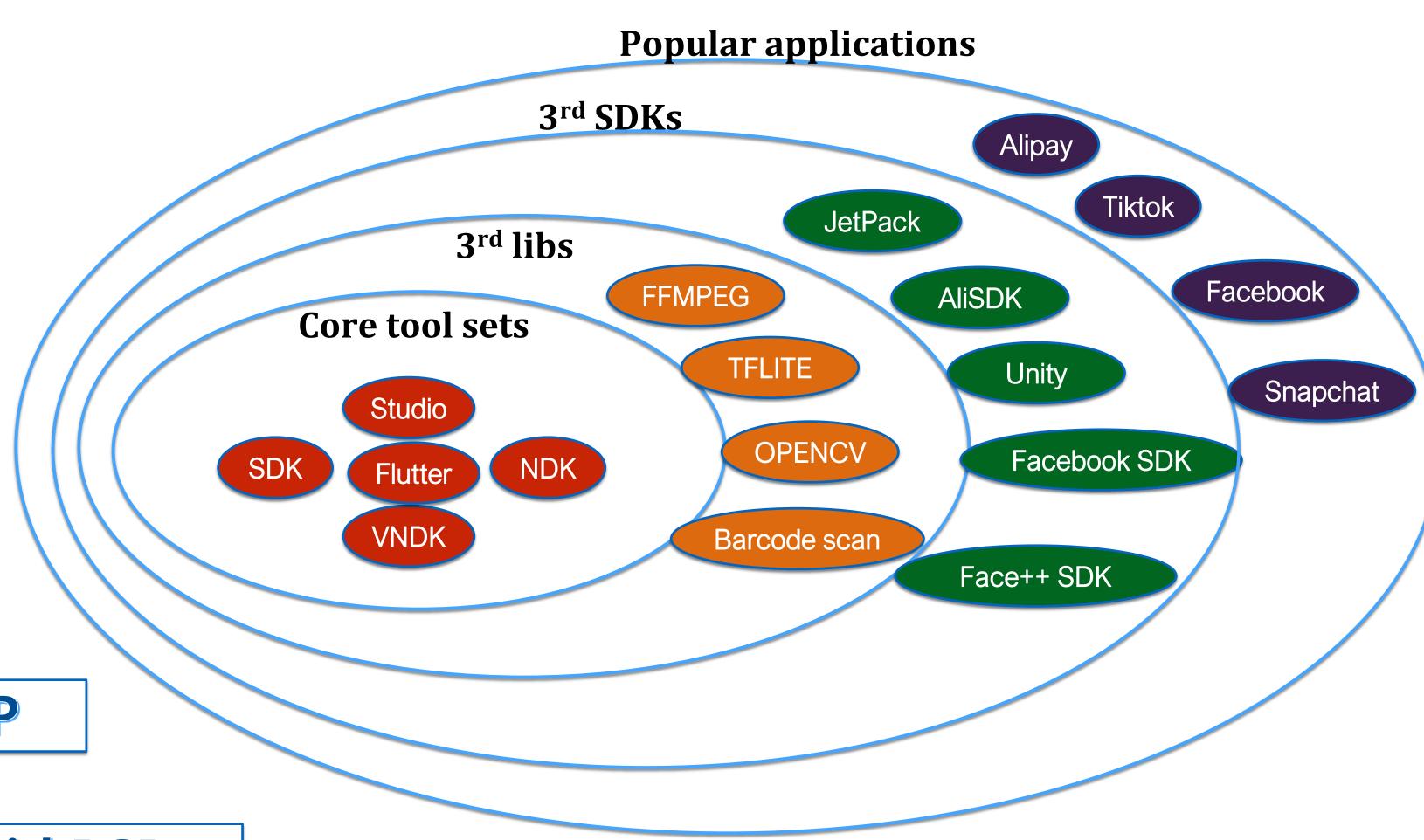
- Poor code quality
- Failing compliant test

- Poor Optimization
- Poor SDK & APP support
- No reference design





## **Android Ecosystem**



**AOSP Android** 

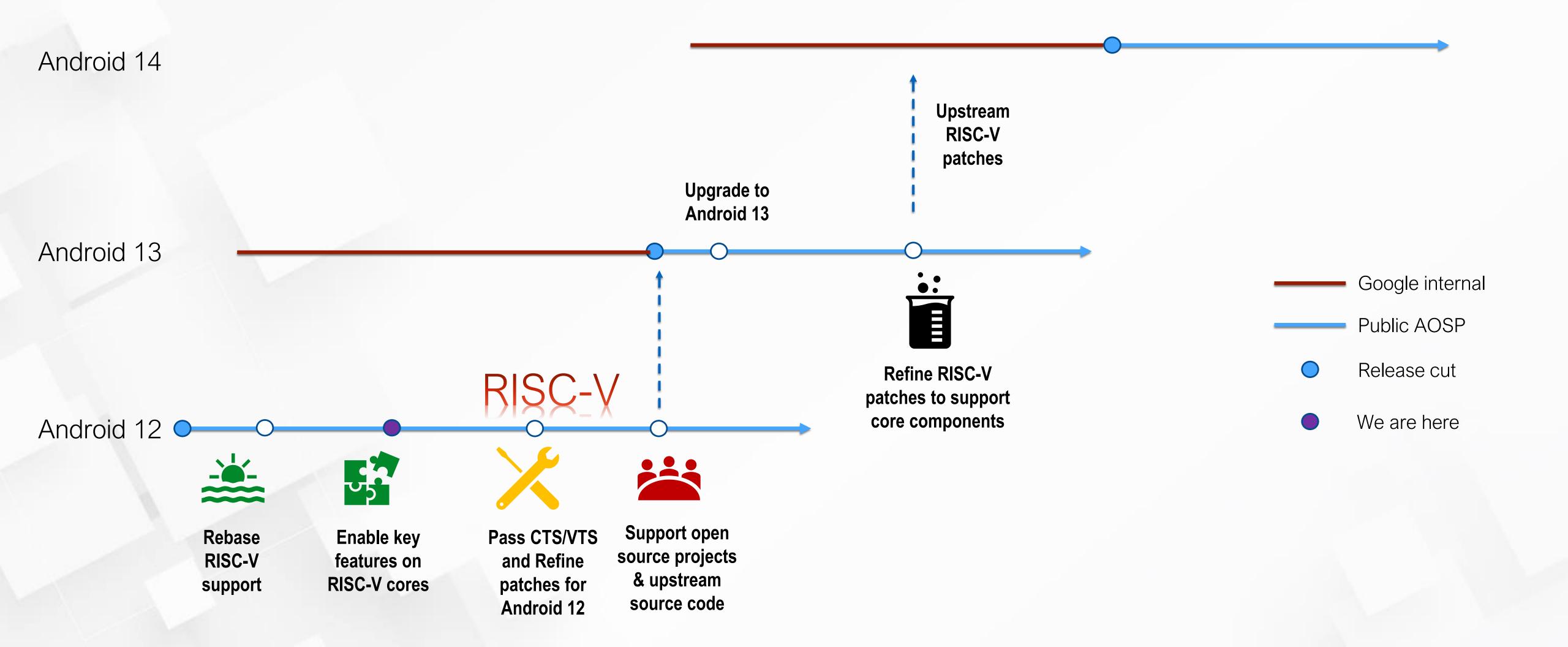
SoC Android BSP

**Device Android BSP** 





# **AOSP Upstream Plan: Why Android 12 NOW**





## Hardware Requirements for Android Development

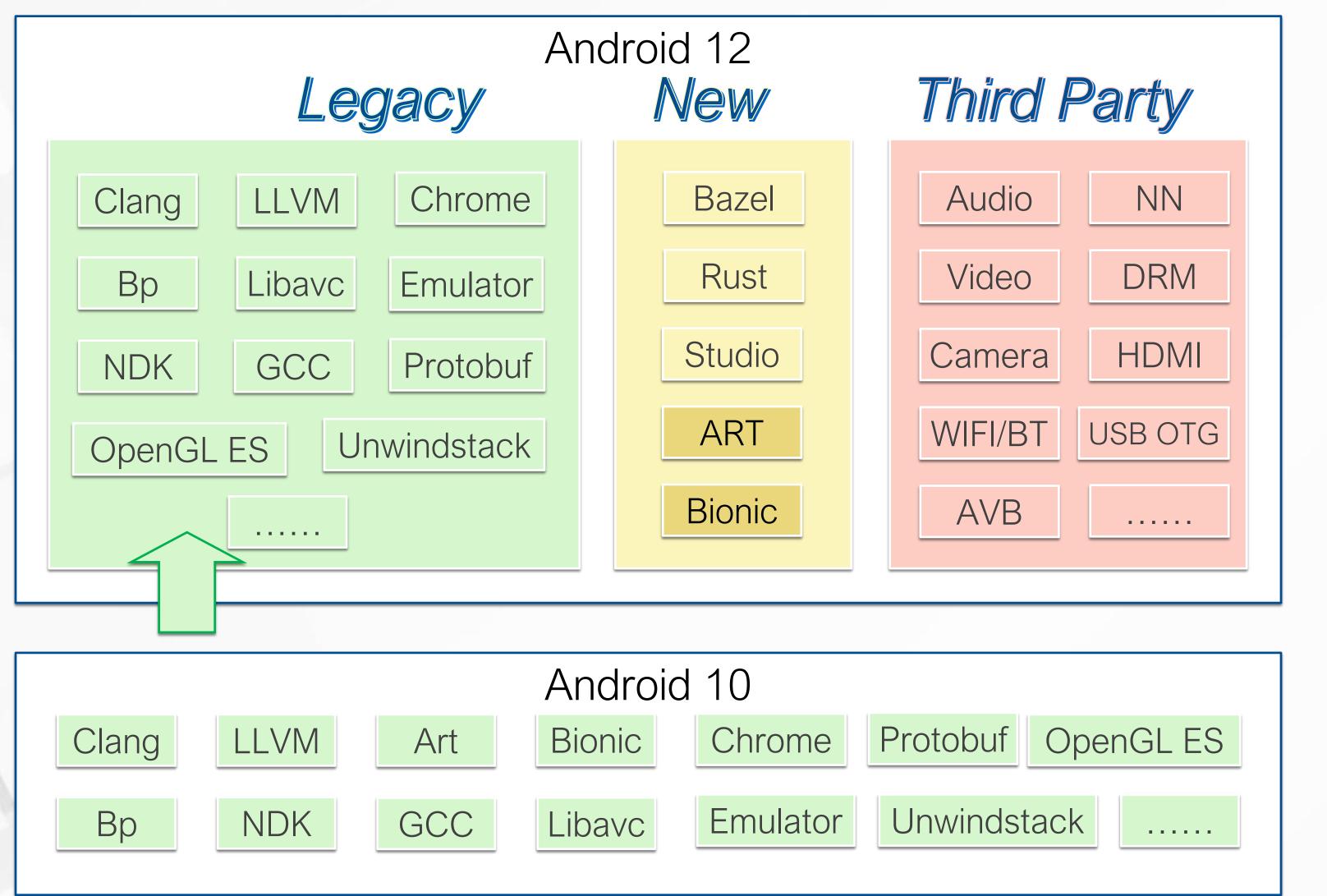
- 2+ Core at XuanTie C910 (rv64imafdcv) level
- 4G+ DDR Memory
- GPU (OpenGL ES, OpenCL)
- Display (MIPI/HDMI)
- USB
- Multi-Channel Audio output & input
- HW Video/Picture codec
- ISP, multiple MIPI lanes
- Neural Network Accelerator







# What We need to Support Android 12 on RISC-V





Android 12 on RISC-V

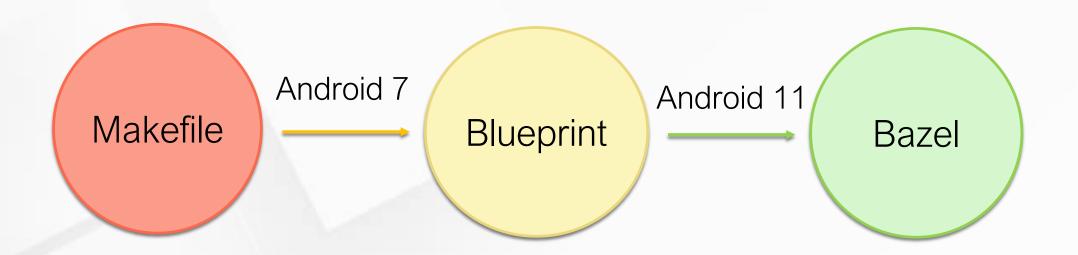


### **New Features**

# **Build system**

- Slow
- Inflexible
- Complex

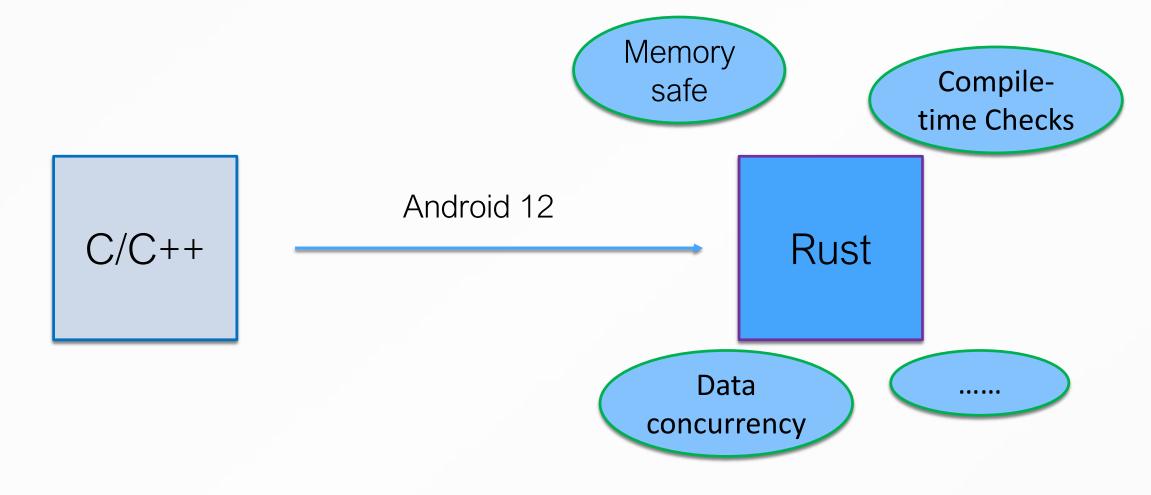
- Faster
- Flexible
- Simpler



#### RISC-V support:

- Configurability
- Setting & constraints

### Rust



#### RISC-V support:

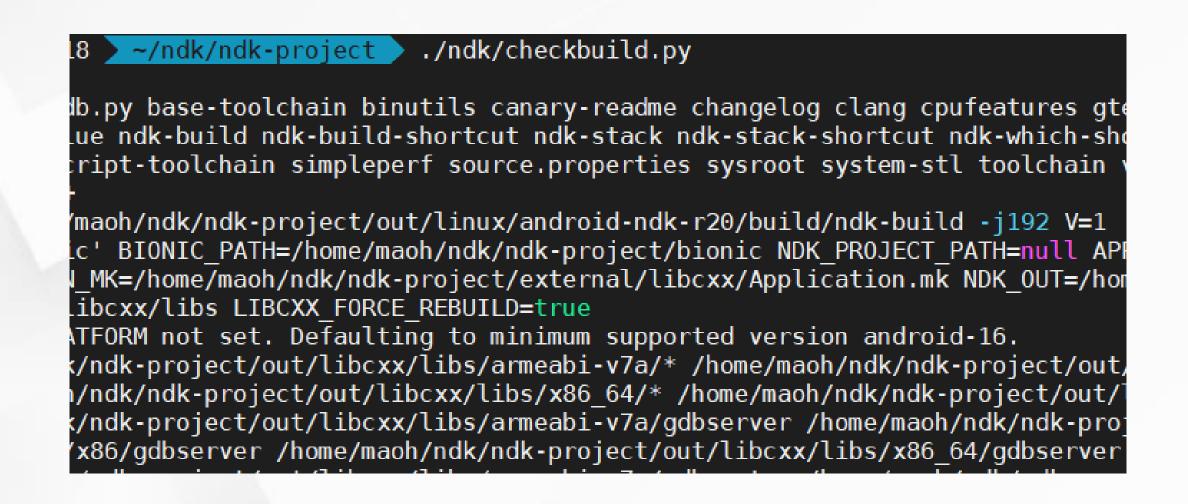
- Target triplets & search path
- ●LIBC API
- Crates

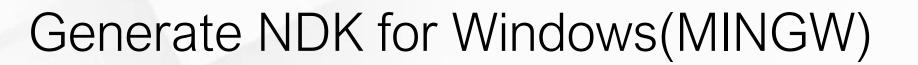
New prebuilt project repo, changes in 5+ architecture related gits, 20+ files

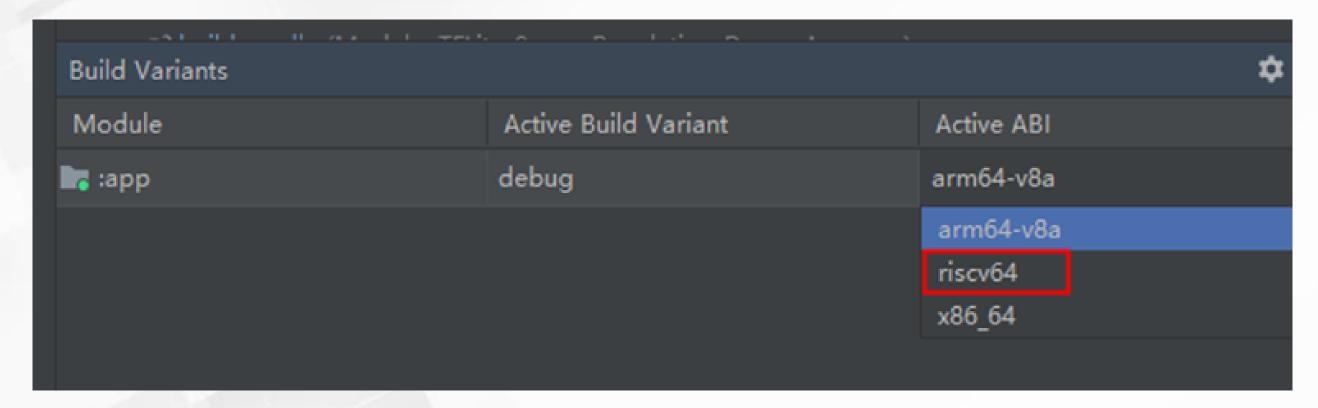




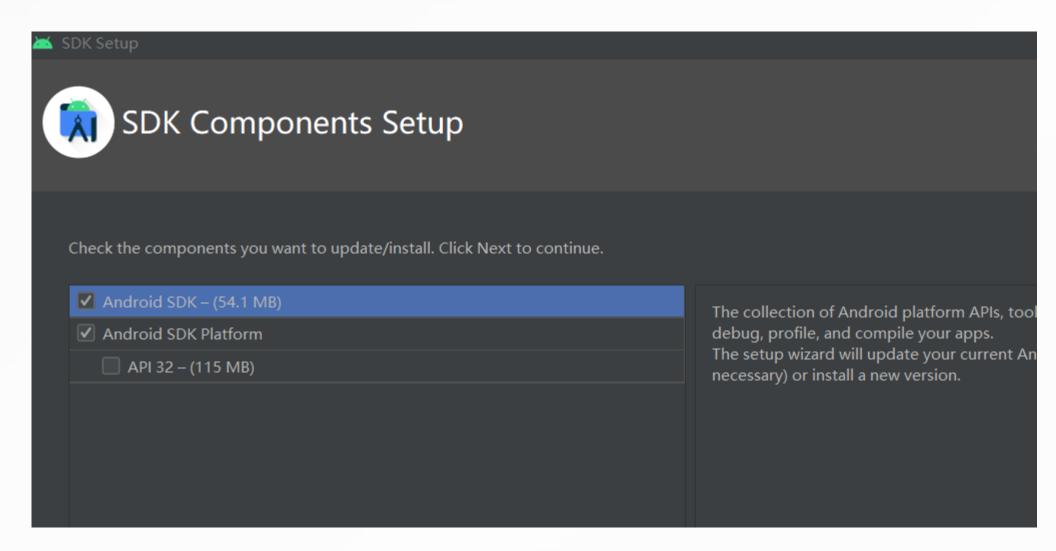
## **Support Android Studio**







Build APP for RISC-V



Import SDK with RISC-V NDK support

#### Must-have:

- Emulator in Window SDK
- Profiling tools
- Other missing dependent packages/libraries





## Running TF lite on RISC-V

## NN HAL support



Build TF lite demo APK



Install and execute

```
🔼 File Edit View Navigate Code Analyze Refactor Build Run Tools Git Window Help TFLite Super Resolution Demo App - build.gradle (:app)
                                                                                                                                                                                                                                                                                             🔺 app 🔻 📗 📑 unknown AOSP on Xuan
  android > app > A build.gradle
                                                                                                                           🕀 👱 🕏 🗘 ocal.properties × 🗬 build.gradle (TFLite Super Resolution Demo App) × 🛕 CMakeLists.txt × 🚓 SuperF
       ▲ Android ▼
      🗡 📑 app
                                                                                                                                                                   You can use the Project Structure dialog to view and edit your project configuration

∨ ■ manifests

                                                                                                                                                                                  apply plugin: 'com.android.application'
                      # AndroidManifest.xml
                                                                                                                                                                                  apply plugin: 'de.undercouch.download'

✓ iava

▼ Image: various of the various
                                                                                                                                                                                  android {
                           AssetsUtil
                                                                                                                                                                                           compileSdkVersion 31
                           MainActivity
            cpp
                                                                                                                                                                                           defaultConfig {
                > 🛅 includes
                                                                                                                                                                                                    applicationId "org.tensorflow.lite.examples.superresolution"
                                                                                                                                                                                                    minSdkVersion 29
                      ## SuperResolution.cpp
                                                                                                                                                                                                    targetSdkVersion 31
                      ## SuperResolution.h
                                                                                                                                                                                                    versionCode 1
                      ## SuperResolution_jni.cpp
                                                                                                                                                                                                    versionName "1.0"
           > 📭 assets
                                                                                                                                                                                                    externalNativeBuild {
            > 📭 jniLibs
      arguments '-DANDROID_STL=c++_shared'
                 build.gradle (Module: TFLite_Super_Resolution_Demo_App.app)
                ndk {
                 gradle-wrapper.properties (Gradle Version)
                                                                                                                                                                                                             abiFilters 'x86_64', 'arm64-v8a', 'riscv64'
                 proguard-rules.pro (ProGuard Rules for TFLite_Super_Resolution_Demo_App.app)
                 gradle.properties (Project Properties)
                 local properties (SDK Location)
                                                                                                                                                                                          buildTypes {
                                                                                                                                                                                                    release {
                                                                                                                                                                                                            minifyEnabled false
                                                                                                                                                                                                            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-android.txt'), 'proguard-android.txt')
                                                                                                                                                                                   android{} → defaultConfig{} → ndk{}
      Build: Build Output × Build Analyzer

✓ ▲ Build: finished At 2022/4/13 16:27 with 1 warning

                                                                                                                                                                                                   Clean SuperResolution arm64-v8a,lib_tensorflowlite arm64-v8a
                         ▲ [CXX5106] NDK was located by using ndk.dir property. This method is deprecated and will be remo
                                                                                                                                                                                                    Clean SuperResolution x86_64.lib_tensorflowlite x86_64
```

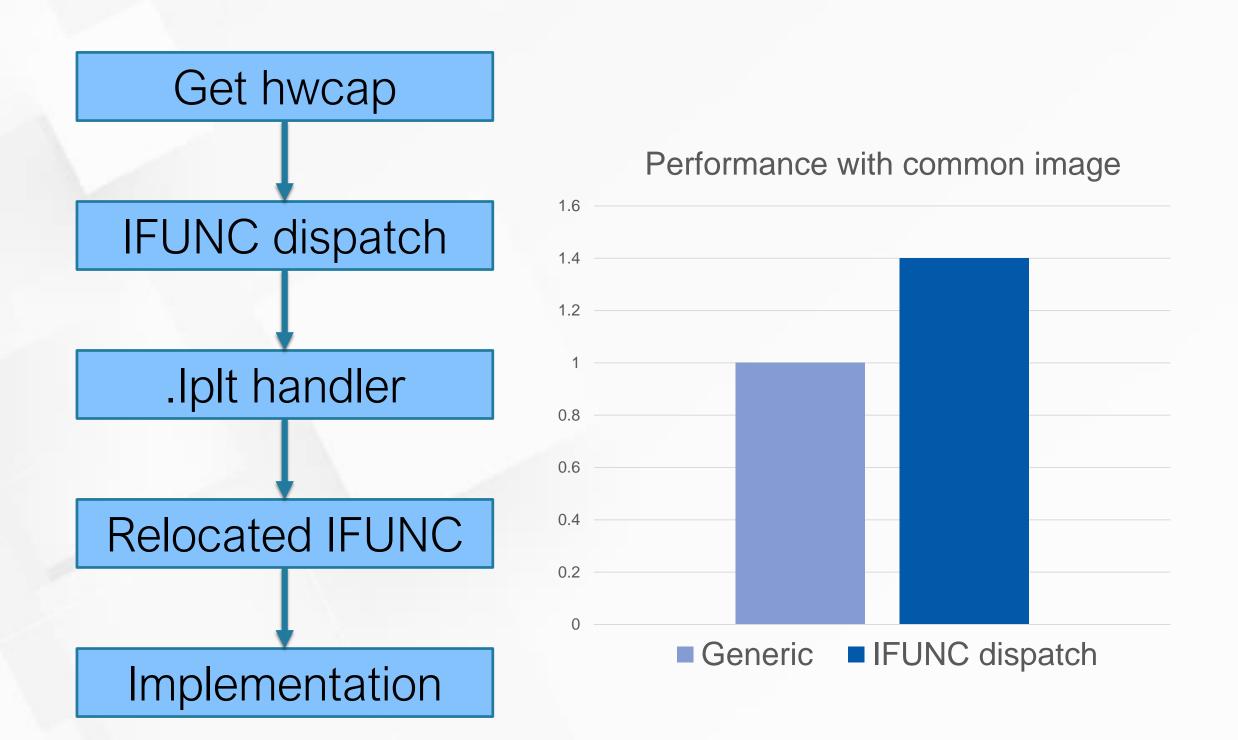




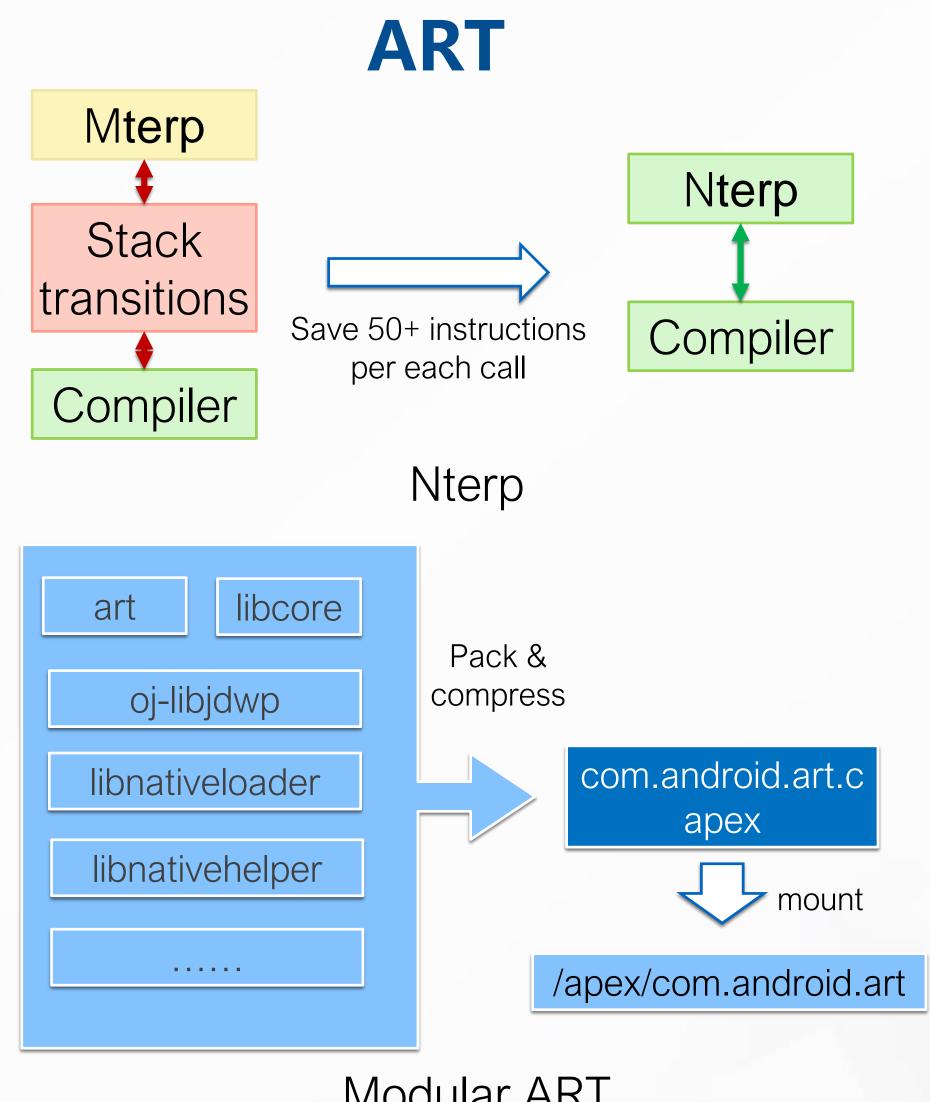


## Features Require Major Changes

## Bionic



- Dynamic syscalls generation
- Kernel UAPI update
- New common relocation



Modular ART

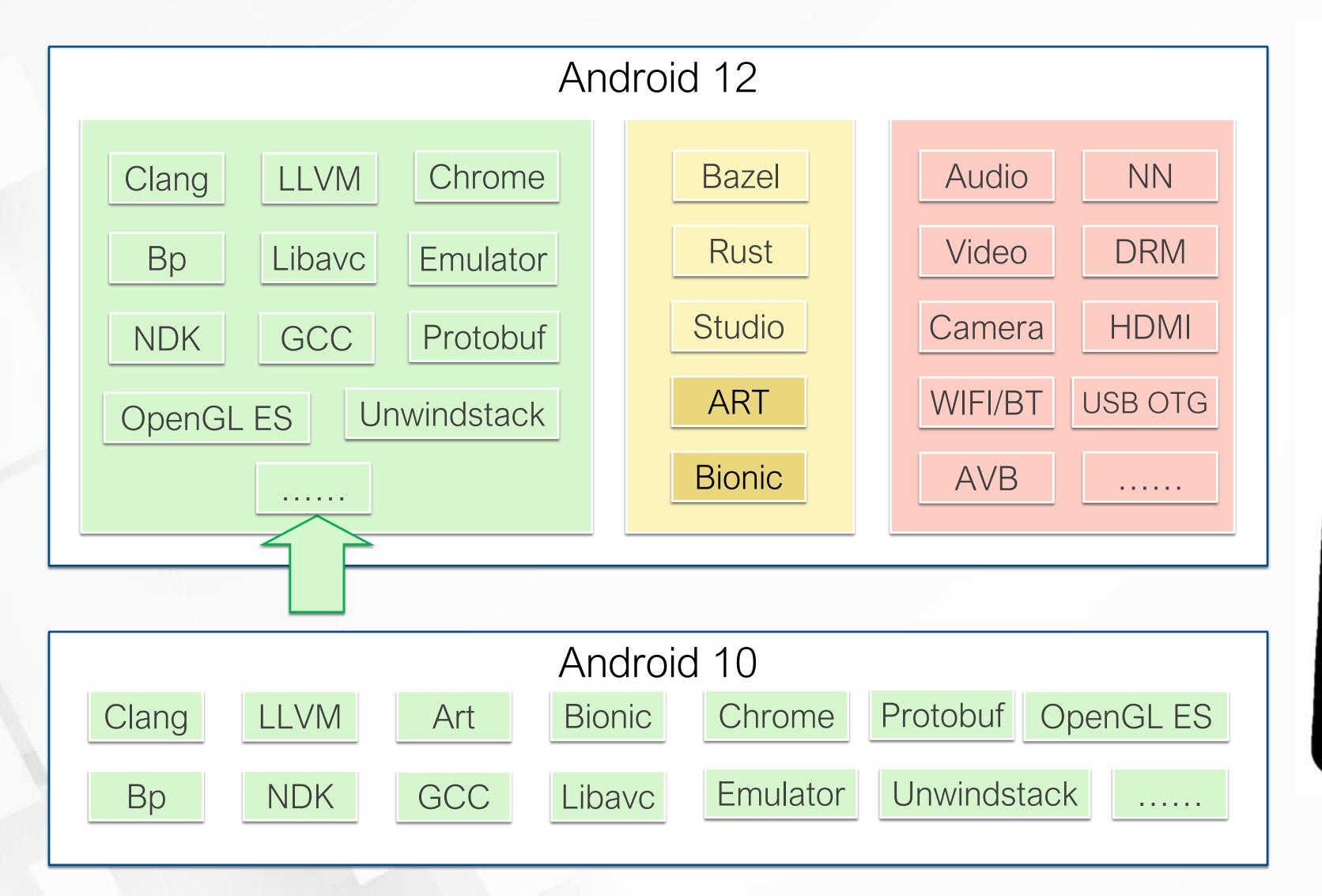
~6k lines changes, 30+ commits

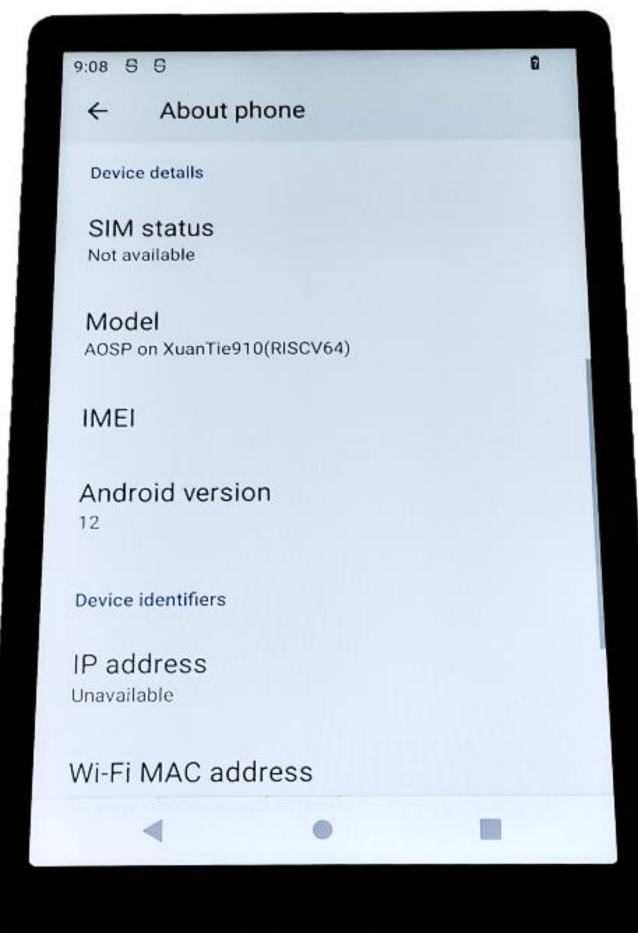
~10K lines changes, 60+commits: JNI, New API, Functionality changes





# How to Support Android 12 on RISC-V





Android 12 on RISC-V





# Vendor Module Integration on RISC-V

- Audio playback
- DRM hwcomposer
- USB OTG
- WIFI/BT
- Video playback
- Camera

Poor RISC-V support



Challenges

Not yet verified on RISC-V





## Features - Video

Incompatible service bit width

Build configs not verified

Conflict in OMX header

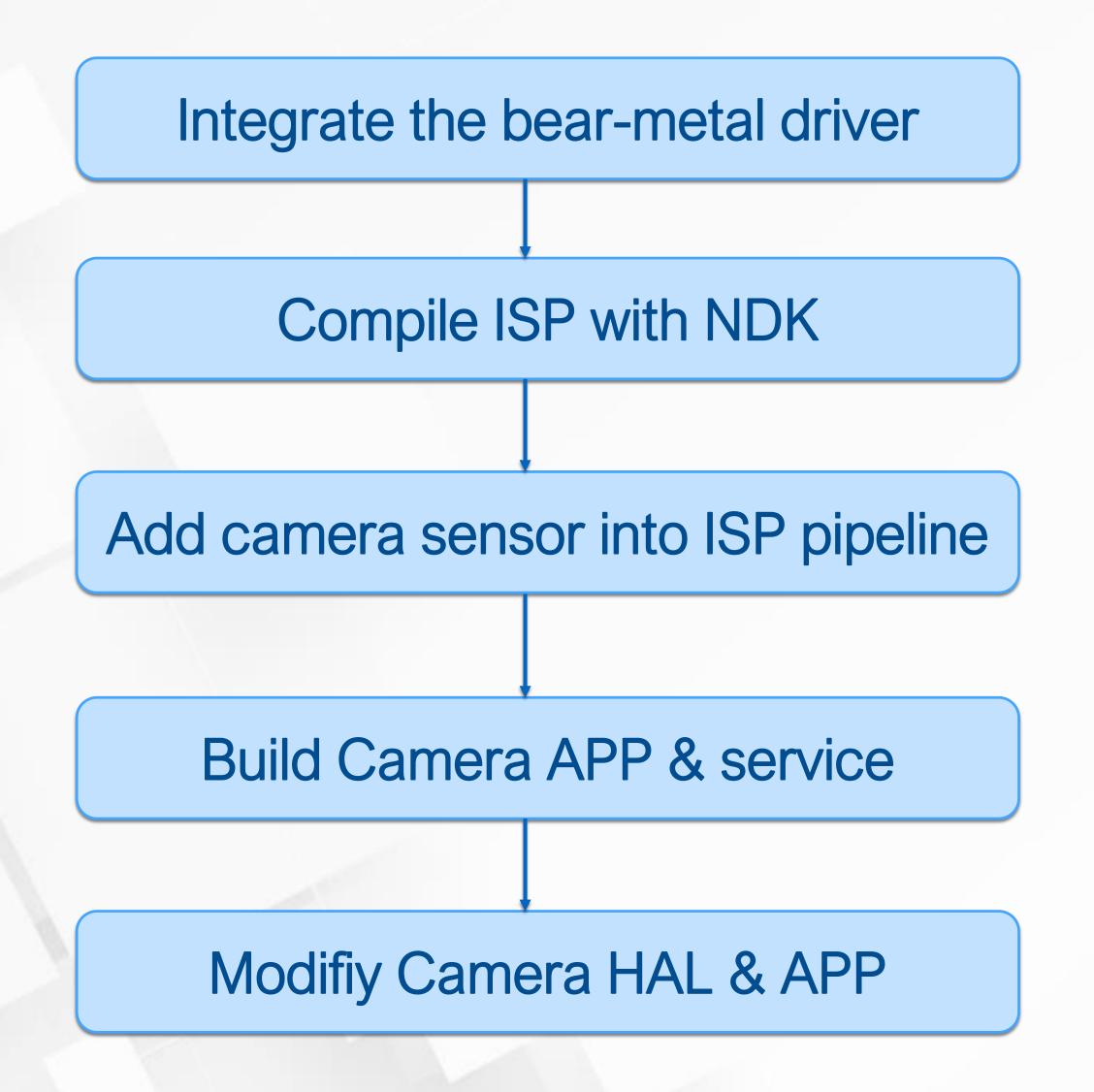
Missing NWB handler

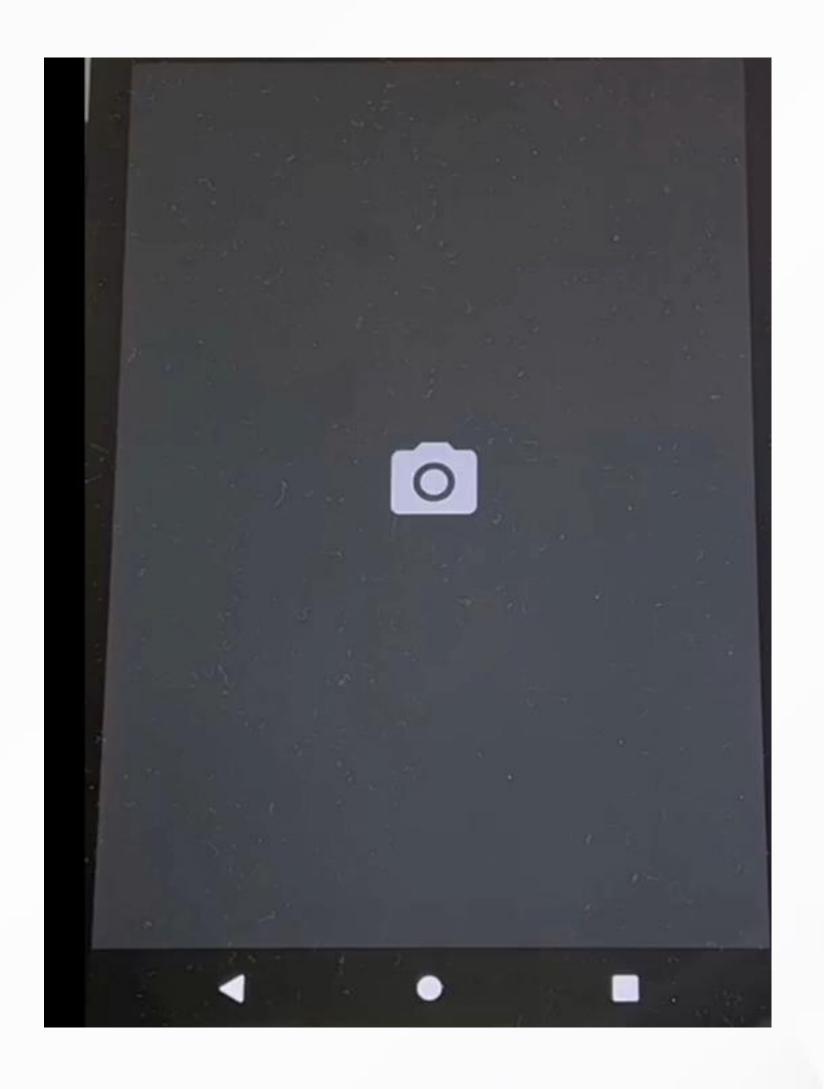






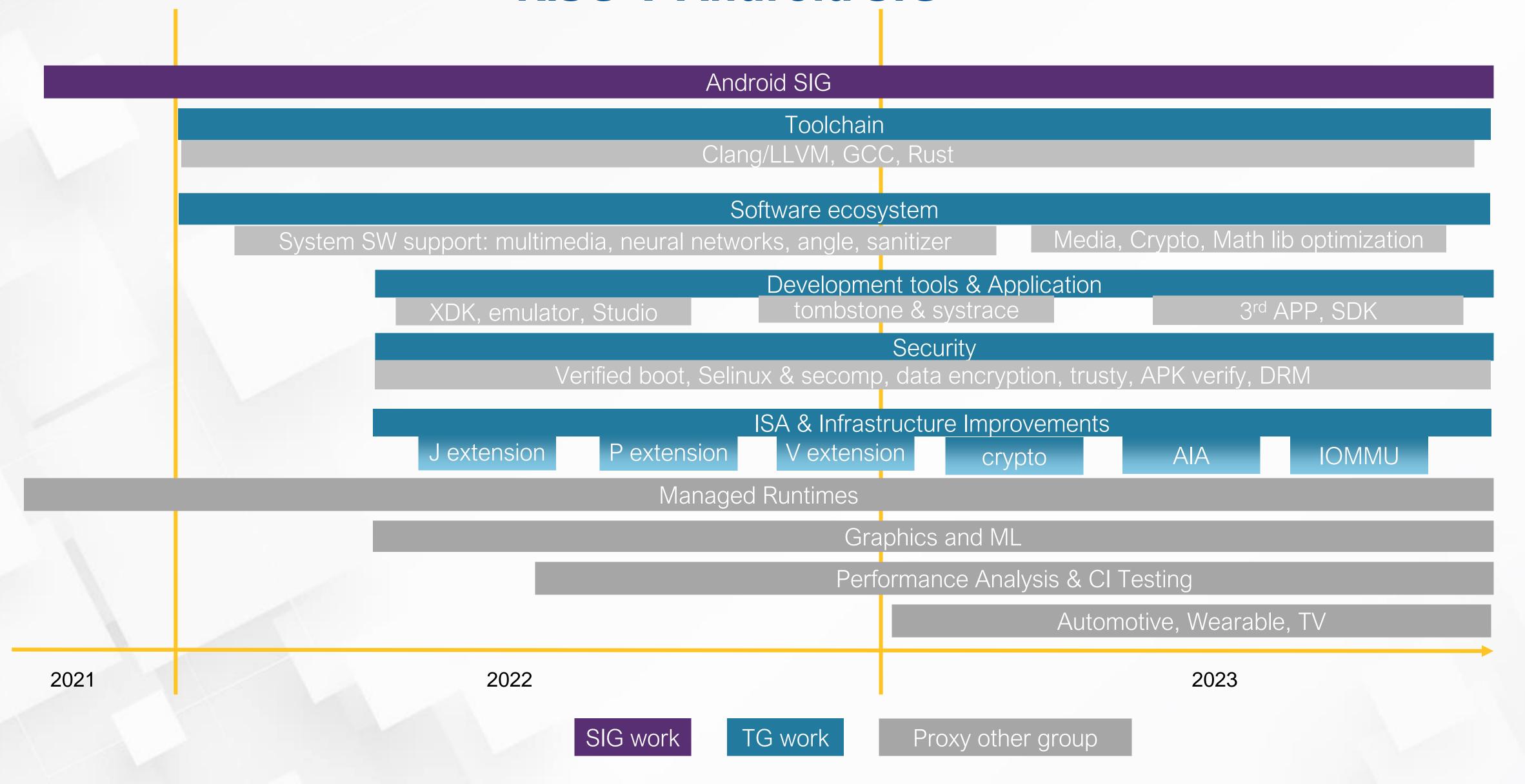
#### Features - Camera







#### **RISC-V Android SIG**

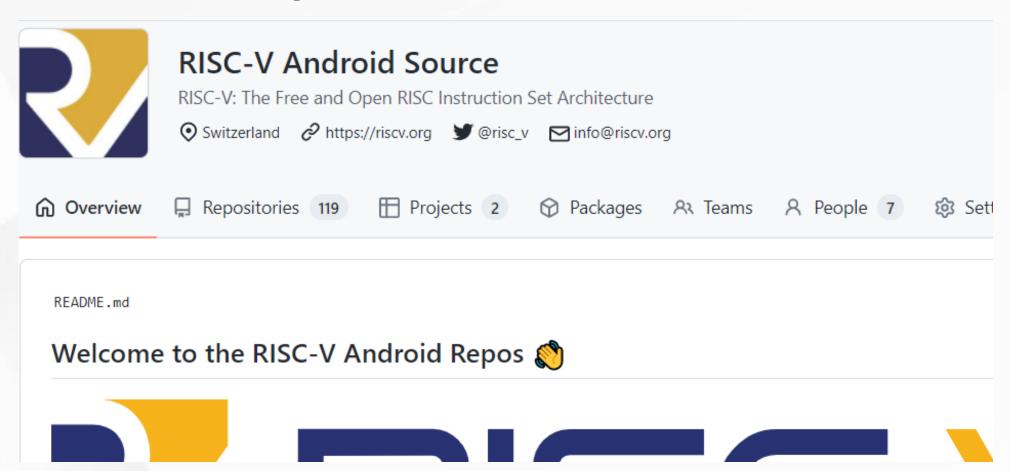




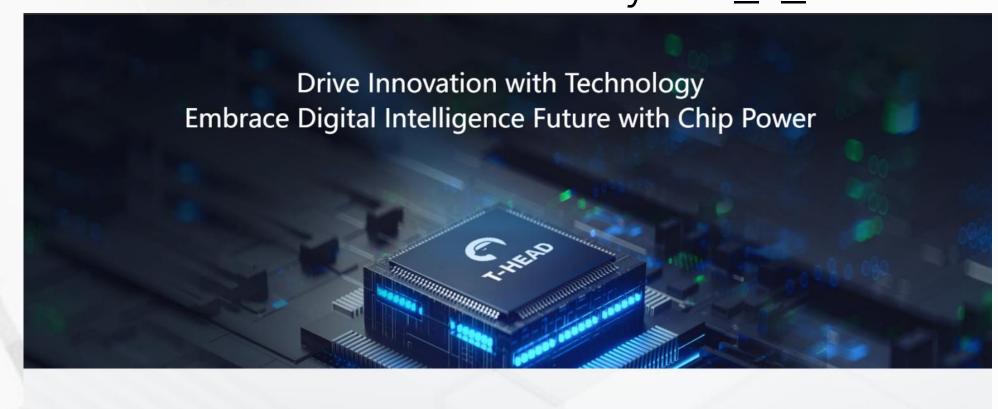


#### Look for the codes/binaries:

https://github.com/riscv-android-src



# Open chip community (English page): occ.t-head.cn/community/risc\_v\_en



#### Introduction for XuanTie Series

XuanTie processor IP covers various scenarios from terminal to cloud and is the cornerstone of intelligent, secure integrated chip architecture of edges and cloud, to provide computing center for the digital age. XuanTie persists in the self-development of core technology since its establishment. Its new series actively embraces the open-source RISC-V architecture. XuanTie processor IP is widely applied to computer vision, storage solution, industrial interconnection, network communication, smart home biological recognition, information security and other areas. By 2020, the shipment of the chips with XuanTie CPU architecture has reached 2 billion.

#### Build Android12 on RISC-V:

# https://github.com/riscv-android-src/riscv-android/blob/main/doc/android12.md

To download the RISC-V Android source tree to your working directory:
<pre>mkdir ~/riscv-android-src &amp;&amp; cd mkdir ~/riscv-android-src repo init -u git@github.com:riscv-android-src/manifest.git -b riscv64-android-12.0.0_dev repo sync cd prebuilts/rust/ git lfs pull cd - cd prebuilts/clang/host/linux-x86/ git lfs pull mkdir prebuilts/runtime/mainline/runtime/sdk/android/riscv64/include/bionic/libc/kernel/uapi/asm-riscv64 rm external/angle/Android.bp touch packages/modules/ArtPrebuilt/com.android.art-riscv64.apex</pre>
Build full emulator image with command:
source build/envsetup.sh lunch sdk_phone64_riscv64 m -j

#### Join the discussion:

https://lists.riscv.org/g/sig-android

RISCV	A Your Groups →
<b>☆</b> Home	
Subscription	A STATE OF THE STA
Messages	
# Hashtags	
New Topic	
Lill New Poll	ACCOUNT OF THE PARTY
Subgroups	Market Mark Mark Mark
<b>A</b> Directory	
	Tech: Android SIG sig-android@lists.riscv.org
	Android SIG under the Software Horizontal Committee
	The preliminary charter of the Android SIG is as follows:
	Scope:  * Improve the functionality, efficiency, robustness of RISC-V supports on Android software stack.  * Enabling RISC-V based Android device development and make RISC-V Android products a reality in the near future.  * Liaise with Google and Android community to coordinate the upstream and maintenance affairs.  * Arrange and coordinate efforts of developers from different entities willing to contribute to the implementation of AOSP on RISC-V.  Goal:
	* Maintain a stable version AOSP on RISC-V repository for device development.
	* Maintain an up-to-date version AOSP on RISC-V repository for upstream patchwork.  * Upstream the RISC-V supports patches to the AOSP projects, Linux kernel and external projects; and get them into the chunk.
	and the description of the Control Production is a complete product of the Control Co

# THANK YOU



