

应用笔记

Application Note

文档编号: **AN1125**

G32R5xx SDK 快速上手指南

版本: **V1.2**

1 引言

G32R5xx SDK (Software Development Kit) 是专为 G32R5xx 系列实时控制 MCU 设计的一套开发与文档, 旨在减少软件开发时间。从器件驱动与库到外设示例, G32R5xx SDK 为评估与开发提供基础。

本应用笔记中涉及资料均可从极海官方渠道获取下载。

快速评估 G32R5xx SDK, 建议准备以下环境或内容:

- Windows 10/11
- Keil MDK v5.40 或更新版本
- IAR Embedded Workbench for Arm (G32R501 参考 9.60.2 或更新版本; **G32R502 系列必须 9.70.4 或以上**)
- Eclipse 4.35 或更新版本
- LLVM-ET-Arm 19.1.1 或更新版本
- Arm GNU Toolchain 14.2 或更新版本
- xpack-windows-build-tools 4.4.1-3 (Windows 构建辅助工具; 请安装并配置 PATH, 供 Eclipse / GNU 工具链构建使用)
- G32R501 EVAL 和/或 G32R502 评估板

注: G32R5xx SDK 在同一软件包内提供 device_support/g32r501 与 device_support/g32r502 (以及对应 driverlib、libraries 等路径)。G32R501 与 G32R502 的首次运行步骤见第 10 章与第 11 章。

目录

1	引言	1
2	SDK 结构.....	3
3	关于 boards	4
4	关于 device_support.....	5
5	关于 docs	6
6	关于 driverlib	7
7	关于 examples	8
8	关于 libraries	9
8.1	Calibration	9
8.2	DSP	9
8.3	Flash API	9
8.4	math	9
8.5	其他.....	9
9	关于 utilities.....	10
9.1	geehy_tool.....	10
9.2	keil_dbg_tool.....	10
9.3	G32R5xx_AddOn	10
10	G32R501 首次运行（摘要）	11
11	G32R502 首次运行.....	12
11.1	环境与工具链参考	12
11.2	安装芯片支持	12
11.3	打开首个例程（LED）	12
11.4	MDK 调试初始化文件（r502_dbg.ini）	13
11.5	编译与下载要点.....	13
11.6	工具链已知限制	14
12	版本历史	16

2 SDK 结构

G32R5xx SDK 主要层次结构如下:

- boards: 板卡硬件设计资料
- device_support: 器件支持 (含 g32r501、g32r502)
- docs: 用户文档 (应用笔记等 PDF)
- driverlib: 驱动库与评估板例程 (含 g32r501、g32r502)
- examples: 跨模块示例
- libraries: DSP / Math / Flash API / SFO 等库
- kernel: FreeRTOS 等
- package: CMSIS-Pack、SVD 等
- utilities: 调试工具、geehy_tool、IAR Add-On 等

更细的目录说明以本 AN 及各专题 AN PDF 为准。

3 关于 boards

该目录包含 G32R5xx 系列板卡的硬件设计原理图等相关资料。

注意：本章描述的 boards 内容位于 G32R5xx_SDK/boards。

4 关于 **device_support**

该目录包含器件特定支持文件（含 Arm 内核相关文件）、位域头文件与器件开发相关说明。按子目录区分器件：

注意：本章描述的 `device_support` 内容位于 `G32R5xx_SDK/device_support`。

- `device_support/g32r501/`: G32R501 器件树（CMSIS、`startup`、链接脚本母版、`examples` 等）
- `device_support/g32r502/`: G32R502 器件树（CMSIS、`startup`、链接脚本母版、`examples` 等）

CMSIS 相关目录包含与 Arm Cortex-M52 内核相关的头文件与启动支持。Geehy 器件目录下典型文件包括：

- 设备头文件与寄存器定义
- `startup_g32r501.` / `startup_g32r502.` 等启动文件
- `system_c/.h` 等系统初始化文件
- `include/` 下的 `device.h` 等公共头文件
- `icf/`、`sct/`、`ld/` 等链接脚本目录
- `examples/`: `device_support` 风格例程

具体文件名以当前 SDK 包内实物为准。

5 关于 docs

该目录包含 G32R5xx SDK 用户指南与软件包说明 (PDF)。

注意：本章描述的 docs 内容位于 G32R5xx_SDK/docs。

6 关于 driverlib

该目录包含特定器件的驱动程序库及基于驱动的外设示例。按器件分子目录，例如：

注意：本章描述的 driverlib 内容位于 G32R5xx_SDK/driverlib。

- driverlib/g32r501/
- driverlib/g32r502/

各器件树下通常包含：

- driverlib/：外设驱动 API 头文件与源文件（ADC、CAN/CANFD、GPIO、PWM、UART、Flash 等）
- examples/：评估板例程（如 eval/）

例程说明文档可参考《G32R5xx SDK Examples 用户手册》（AN1127）。

7 关于 examples

该目录用于存放跨外设/应用类示例。正式使用前建议：

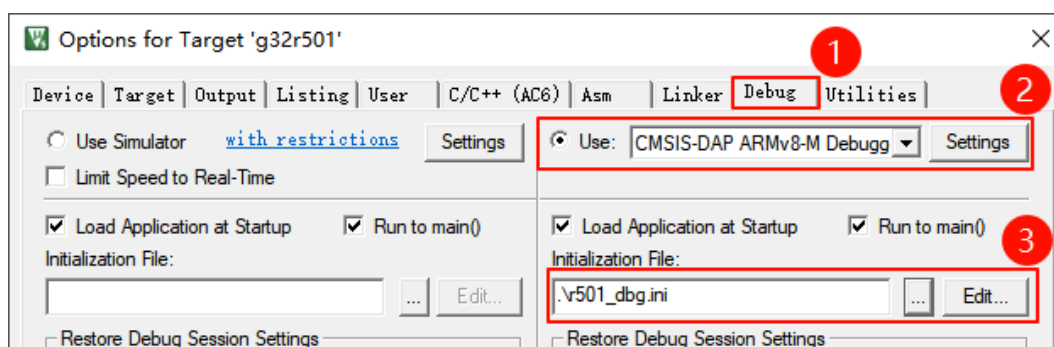
- 阅读对应例程说明与 AN1127。

如需运行例程，请按下列准备：

- 软件环境（编译器、调试器、IDE）。
- 硬件环境（调试器如 GEEHY-LINK / J-Link，以及对应评估板）。
- 在 MDK 环境进行调试时，请先编译一次程序，再在 **Options for Target**→**Debug** 下选择合适的调试器及调试脚本，如图 1 所示。

说明：以下界面以 G32R501 例程（目标名常见为 g32r501）为例。开发 G32R502 时，请在 Device / Target 中选择对应 G32R502 型号，将路径中的 g32r501 换为 g32r502，MDK 调试初始化文件使用 r502_dbg.ini。

图 1 选择合适的调试器及调试脚本



8 关于 **libraries**

该目录包含 DSP、数学、Flash API、校准等库。多数库按 g32r501 / g32r502 分子路径。常见分类如下:

注意: 本章描述的 **libraries** 内容位于 G32R5xx_SDK/libraries/。

8.1 **Calibration**

包含 HRPWM 相关校准库 (SFO)。使用说明可参考 AN1134。

8.2 **DSP**

通常包含 Fixed Point、FPU、VCU 等子库, 提供 FFT、滤波、向量运算与 CRC 等能力。

8.3 **Flash API**

提供 Flash 擦写等 API; 使用说明见 Flash API 用户手册。

8.4 **math**

通常包含 Fix32math、FPUfastRTS、FastIntDiv 等数学库。

8.5 **其他**

包内还可能包含 CoreMark、PMBus、Secure ROM APIs 等; 以当前 SDK 树为准。

9 关于 utilities

该目录包含开发辅助工具，当前包内主要包括：

注意：本章描述的 utilities 内容位于 G32R5xx_SDK/utilities。

9.1 geehy_tool

路径: utilities/geehy_tool/。目录内主要程序见下列列表；完整命令行参数与示例见 AN1131 《Tool 用户手册》。

目录内主要程序包括：

- Geehy_Bin.exe: 固件格式转换与 CMAC 相关处理（支持 boot / UART / SPI / I2C 等选项，以及主安全启动、扩展安全启动等模式；可通过 --cmac 指定密钥文件）
- Geehy_Serial_flash_Programmer.exe / Geehy_Serial_flash_Programmer_appln.exe: 串口 Flash 下载工具
- hex2bin.exe: 十六进制与二进制转换辅助
- key.txt、.cmd: 示例密钥与命令脚本

说明：使用前可在同目录对可执行文件运行 --help 查看参数，或直接按 AN1131 中的示例命令操作。

9.2 keil_dbg_tool

MDK 调试辅助工具（说明见 AN1129）。G32R502 侧 r502_dbg.ini 配置要点见第 11 章。

9.3 G32R5xx_AddOn

IAR 芯片支持安装包（说明见 AN1126）。

10 **G32R501 首次运行（摘要）**

1. 安装 CMSIS-Pack / IAR Add-On（按 AN1126）。
2. 打开 driverlib/g32r501/examples/ 下评估板 LED 或同类入门例程的 MDK/IAR/Eclipse 工程。
3. 确认器件选型与调试脚本后编译、下载到 G32R501 EVAL。
4. 详细 IDE 步骤与工具链说明见 AN1126；例程清单见 AN1127。

11 G32R502 首次运行

本章依据 G32R502 器件树与评估板路径整理，供在 G32R5xx SDK 包内首次跑通 502 例程。

11.1 环境与工具链参考

SDK 例程目标器件为 **G32R502MC** (Cortex-M52)。建议对齐表格 1 中的参考工具链版本（更高版本可能可用；遇问题时请先对齐参考版本）。

表格 1 参考工具链版本

IDE / 工具链	参考版本
Keil MDK (uVision)	v5.40 及以上（推荐 v5.43）
Arm Compiler for Embedded 6	v6.24
CMSIS-Pack	Geehy.G32R5xx_DFP（以包内 package/ 实际 pack 为准）
IAR Embedded Workbench for Arm	G32R502 系列必须 9.70.4 及以上
IAR Add-On	G32R5xx_AddOn（以 utilities/G32R5xx_AddOn/ 实际版本为准）
Eclipse Embedded CDT	2023-12 及以上
Arm LLVM (Eclipse)	19.1.x

11.2 安装芯片支持

- Keil MDK:** 用 Pack Installer 安装 package/ 下 G32R5xx DFP pack（含 SVD 与 Flash 算法）。若直接安装卡住，可将扩展名改为 .zip 解压后，用 **Import Folder** 导入。
- IAR Embedded Workbench for Arm:** 关闭 IAR 后运行 utilities/G32R5xx_AddOn/ 下安装程序，选择 **IAR 安装根目录**（不要选安装目录下的 config 等子目录）。安装程序中选择 **IAR 安装根目录**（不要选 config 等子目录），安装完成后重新打开 IAR。开发 **G32R502 系列时，IAR 版本必须为 9.70.4 或以上**。
- 调试（可选）:** J-Link Devices 安装与 pyOCD 注册的完整步骤见 **AN1126**（J-Link 器件支持安装与第 7 章 pyOCD）。

11.3 打开首个例程（LED）

推荐路径:

```
driverlib/g32r502/examples/eval/led/led_ex1_blinky/project/  
└─ MDK/project.uvprojx
```

```
└─ IAR/project.ewp
└─ Eclipse/.cproject
```

以 MDK 为例:

1. 用 Keil 打开 project.uvprojx。
2. 确认 Target 名称为 **G32R502**。
3. 确认已安装 G32R5xx DFP, Device 为 G32R502 系列。
4. 按 11.4 节确认本工程 MDK 目录下的 r502_dbg.ini 及调试选项配置正确。
5. 编译并下载到 G32R502 评估板。

说明: 本例程点亮评估板 **LD2 (GPIO49)**, 不占用 JTAG TDI。详见例程源文件头部说明。

串口打印示例路径: driverlib/g32r502/examples/eval/uart/uart_ex5_printf/project/。

11.4 MDK 调试初始化文件 (r502_dbg.ini)

现行 SDK 中, G32R502 各 MDK 例程工程目录通常已附带 r502_dbg.ini (例如 driverlib/g32r502/examples/eval/adc/adc_ex1_soc_software/project/MDK/r502_dbg.ini)。工程按相对路径引用该文件。首次用 Keil 打开 502 例程时, 请核对下列配置。

1. 确认本工程 MDK 目录下存在 r502_dbg.ini。若缺失, 可从模板复制并命名为 r502_dbg.ini:

源: utilities/keil_dbg_tool/r502_dbg.ini.template

目标: <例程>/project/MDK/r502_dbg.ini

2. 在 **Options for Target → Debug → Initialization File** 确认填入 .\r502_dbg.ini。
3. 确认 **User → After Build/Rebuild** 已配置调用 keil_dbg_tool (以工程内现有命令为准), 以便编译后更新 MSP/PC。
4. 执行一次完整 Build, 确认工具输出成功后再 Debug / Download。
5. 每个 MDK 例程工程目录各用一份 r502_dbg.ini, 勿多工程共用同一物理文件。

更完整的工具选项见 AN1129。最小核对: 将 utilities/keil_dbg_tool/r502_dbg.ini.template 复制为 <例程>/project/MDK/r502_dbg.ini, 并在 Options for Target → Debug → Initialization File 填入 .\r502_dbg.ini。

11.5 编译与下载要点

- **Target 名称:** 各 IDE 工程统一为 G32R502。

- **链接脚本:** eval 例程默认使用 Flash 运行 + RAM 向量表布局。
- **编译参数:** 须包含 -mcpu=cortex-m52+nomve.fp+nofp.dp+cdec0 (以工程现有选项为准)。

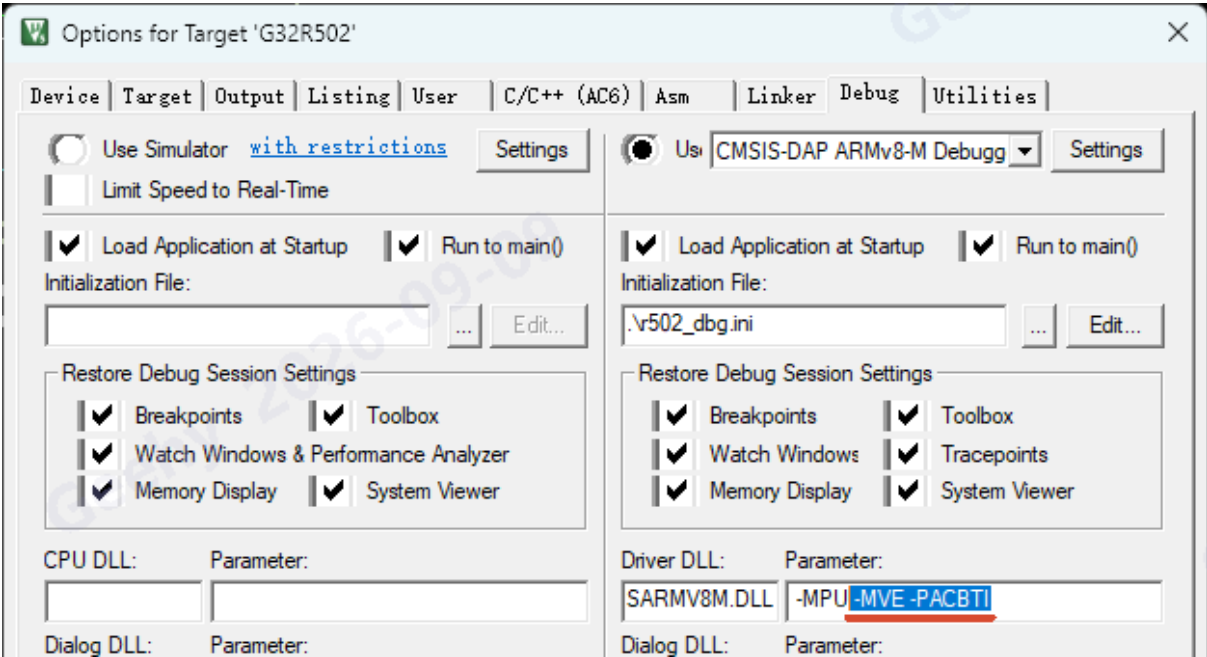
11.6 工具链已知限制

- **IAR (G32R502):** G32R502 系列芯片的 IAR 开发环境**必须**使用 IAR Embedded Workbench for Arm**9.70.4** 及以上版本; 低于该版本不满足本系列开发要求。
- **Keil MDK 寄存器显示:** MDK 5.40 上 Debugger 对 16 位寄存器字段显示通常正常; MDK 5.43.1 上同类字段可能显示异常。若须以寄存器窗口读数为准, 可暂用 MDK 5.40, 或改用 Memory 窗口核对。
- **Keil MDK 源码跳转:** MDK 5.43 以下在 Device 选用 Cortex-M52 时, F12 (转到定义) 可能不可用; 需要源码跳转时请使用 MDK 5.43 或更高版本。
- **Keil MDK + J-Link V12 (Cortex-M52):** MDK 配合 J-Link V12 对 Cortex-M52 做 Debug 时, 可能出现断点无法按预期命中、单步位置异常或变量查看结果异常。发生此类现象时, 应先排除优化等级过高、下载镜像与源码不一致、探针连线不稳以及目标复位状态异常等基础原因。现有信息表明, J-Link 驱动与 MDK 针对 M52 的 Debug 配置 (如 MVE、PACBTI 等) 可能存在不兼容; 该判断仅针对该组合, 不表示所有 J-Link、所有 MDK 版本或所有 M52 工程必然出现相同问题。

在保留工程副本或已记录原始选项的前提下, 可临时删除 Driver DLL 参数中的“-MVE”、“-PACBTI”等可选项后重新构建并下载, 以判断异常是否缓解。操作要点如下:

1. 在 MDK 中打开目标工程, 确认当前目标为 Cortex-M52 (如 G32R501 / G32R502)。
2. 打开 **Options for Target**, 进入 **Debug** 页。
3. 记录当前调试器、接口、下载设置, 以及 Driver DLL Parameter 中的“-MVE”、“-PACBTI”等可选参数, 便于恢复与对照。参数所在位置如图 2 所示 (图中调试器类型仅为界面示例; 使用 J-Link 时同样在该 Parameter 框中查看与修改)。
4. 在已记录原值后, 从 Parameter 中删除“-MVE”、“-PACBTI”等可选参数 (可保留与本次排查无关的其它参数, 例如样张中的“-MPU”)。
5. 保存工程, 执行 **Rebuild**, 下载新生成的镜像。

图 2 MDK Debug 页 Driver DLL 参数中的 MVE、PACBTI



12 版本历史

文件版本历史见表格 2。

表格 2 文件版本历史

日期	版本	变更说明
2025.1	1.0	● 新建
2025.4	1.1	● 勘误描述
2026.8	1.2	● 适用产品列为 G32R501、G32R502 ● 增补 G32R502 首次运行 ● 明确 G32R502 使用 IAR 须 9.70.4 及以上。

声明

本手册由珠海极海半导体有限公司（以下简称“极海”）制订并发布，所列内容均受商标、著作权、软件著作权相关法律法规保护，极海保留随时更正、修改本手册的权利。使用极海产品前请仔细阅读本手册，一旦使用产品则表明您（以下称“用户”）已知悉并接受本手册的所有内容。用户必须按照相关法律法规和本手册的要求使用极海产品。

1、权利所有

本手册仅应当被用于与极海所提供的对应型号的芯片产品、软件产品搭配使用，未经极海许可，任何单位或个人均不得以任何理由或方式对本手册的全部或部分内容进行复制、抄录、修改、编辑或传播。

本手册中所列带有“®”或“™”的“极海”或“Geehy”字样或图形均为极海的商标，其他在极海产品上显示的产品或服务名称均为其各自所有者的财产。

2、无知识产权许可

极海拥有本手册所涉及的全部权利、所有权及知识产权。

极海不应因销售、分发极海产品及本手册而被视为将任何知识产权的许可或权利明示或默示地授予用户。

如果本手册中涉及任何第三方的产品、服务或知识产权，不应被视为极海授权用户使用前述第三方产品、服务或知识产权，也不应被视为极海对第三方产品、服务或知识产权提供任何形式的保证，包括但不限于任何第三方知识产权的非侵权保证，除非极海在销售订单或销售合同中另有约定。

3、版本更新

用户在下单购买极海产品时可获取相应产品的最新版的手册。

如果本手册中所述的内容与极海产品不一致的，应以极海销售订单或销售合同中的约定为

准。

4、信息可靠性

本手册相关数据经极海实验室或合作的第三方测试机构批量测试获得，但本手册相关数据难免会出现校正笔误或因测试环境差异所导致的误差，因此用户应当理解，极海对本手册中可能出现的该等错误无需承担任何责任。本手册相关数据仅用于指导用户作为性能参数参照，不构成极海对任何产品性能方面的保证。

用户应根据自身需求选择合适的极海产品，并对极海产品的应用适用性进行有效验证和测试，以确认极海产品满足用户自身的需求、相应标准、安全或其它可靠性要求；若因用户未充分对极海产品进行有效验证和测试而致使用户损失的，极海不承担任何责任。

5、合规要求

用户在使用本手册及所搭配的极海产品时，应遵守当地所适用的所有法律法规。用户应了解产品可能受到产品供应商、极海、极海经销商及用户所在地等各国有关出口、再出口或其它法律的限制，用户（代表其本身、子公司及关联企业）应同意并保证遵守所有关于取得极海产品及/或技术与直接产品的出口和再出口适用法律与法规。

6、免责声明

本手册由极海“按原样”（as is）提供，在适用法律所允许的范围内，极海不提供任何形式的明示或暗示担保，包括但不限于对产品适销性和特定用途适用性的担保。

极海产品并非设计、授权或担保适合用于军事、生命保障系统、污染控制或有害物质管理系统中的关键部件，亦非设计、授权或担保适合用于在产品失效或故障时可导致人员受伤、死亡、财产或环境损害的应用。

如果产品未标明“汽车级”，则表示不适用于汽车应用。如果用户对产品的应用超出极海提供的规格、应用领域、规范，极海不承担任何责任。

用户应该确保对产品的应用符合相应标准以及功能安全、信息安全、环境标准等要求。用户对极海产品的选择和使用负全部的责任。对于用户后续在针对极海产品进行设计、使用的过程中

所引起的任何纠纷，极海概不承担责任。

7、责任限制

在任何情况下，除非适用法律要求或书面同意，否则极海和/或以“按原样”形式提供本手册及产品的任何第三方均不承担损害赔偿 responsibility，包括任何一般、特殊因使用或无法使用本手册及产品而产生的直接、间接或附带损害（包括但不限于数据丢失或数据不准确，或用户或第三方遭受的损失），这涵盖了可能导致的人身安全、财产或环境损害等情况，对于这些损害极海概不承担责任。

8、适用范围

本手册的信息用以取代本手册所有早期版本所提供的信息。

©2026 珠海极海半导体有限公司 – 保留所有权利