

32 位微控制器

XH-LINK 调试器



Rev1.01 2024年01月



适用对象

调试功能:

产品系列	产品型号
小华半导体 Cortex-M 全系列 MCU	小华半导体 Cortex-M 全型号 MCU

在线编程功能:

产品系列	产品型号	产品系列	产品型号
HC32F120	HC32F120F6TA HC32F120F8TA HC32F120H6TA HC32F120H8TA	HC32M120	HC32M120F6TB HC32M120J6TB

声明

- ★ 小华半导体有限公司(以下简称: "XHSC")保留随时更改、更正、增强、修改小华半导体产品和/或本 文档的权利,恕不另行通知。用户可在下单前获取最新相关信息。XHSC 产品依据购销基本合同中载明 的销售条款和条件进行销售。
- ★ 客户应针对您的应用选择合适的 XHSC 产品,并设计、验证和测试您的应用,以确保您的应用满足相应 标准以及任何安全、安保或其它要求。客户应对此独自承担全部责任。
- ★ XHSC 在此确认未以明示或暗示方式授予任何知识产权许可。
- ★ XHSC 产品的转售,若其条款与此处规定不同,XHSC 对此类产品的任何保修承诺无效。
- ★ 任何带有"[®]"或"[™]"标识的图形或字样是 XHSC 的商标。所有其他在 XHSC 产品上显示的产品或服务名
 称均为其各自所有者的财产。
- ★ 本通知中的信息取代并替换先前版本中的信息。

©2024 小华半导体有限公司 保留所有权利



目 录

适用	团家	₹		2
声	明			3
目	录			4
1	简介	ì		5
	1.1	概览…		5
	1.2	特性…		5
	1.3	板载硬	件与接口	6
		1.3.1	板载硬件说明	6
		1.3.2	接口说明	7
	1.4	串口驱	动安装	8
2	功能	论尔绍	1	4
	2.1	CMSIS	·DAP 调试1	4
		2.1.1	接口连接1	4
		2.1.2	应用样例1	4
	2.2	串口通	信1	8
	2.3	ISP 在约	戋编程2	0
3	ХH	·LINK 调词	式器固件升级2	2
	3.1	固件拖	拽方式升级2	2
	3.2	固件烧	写方式升级2	3
		3.2.1	准备工作	3
		3.2.2	硬件设置2	3
		3.2.3	烧写过程2	4
版本	修订	「记录	2	7

1 简介

XH-LINK 是为小华半导体的 Cortex-M 系列 MCU 提供的一款调试器,支持小华半导体旗下所有的 Cortex-M 系列 MCU 产品的调试功能以及部分 MCU 产品的在线编程功能,目的是为用户提供一款小巧便携、安全可靠、操作简单的调试工具。

1.1 概览

XH-LINK 连接框图如下图 1-1 所示,主要由 XH-LINK 和通信设备类(Communication Device Class, CDC)驱动软件组成。XH-LINK 通过 USB 数据线连接至 PC 端,PC 端上的 IDE(如 IAR、MDK、GCC 等)可通过 XH-LINK 对目标板下载调试代码。XH-LINK 与目标板之间可通过 SWD 排线相连。



图 1-1 XH-LINK SWD 连接框图

1.2 特性

XH-LINK 调试器 1.0 具有如下特点:

- 1) 在线调试的功能
 - 支持目标板 3.3V 或 5V 自供电环境;
 - 可直接在 Keil、IAR Embedded Workbench 等集成开发环境(Integrated Development Environment, IDE)下调试;
 - 支持串行线调试(Serial Wire Debug, SWD)接口。
- 2) USB 转串口功能
 - 采用 USB CDC 组合设备支持虚拟串口;
 - 只需一根 USB 线即可完成调试和虚拟串口的功能;
 - 可配置 USB 虚拟串口的波特率、奇偶校验、数据位和停止位。
- 3) ISP 在线编程功能

- 4) 固件可升级更新
 - U 盘拖拽方式更新固件
 - 烧写方式更新固件
- 5) 硬件复位按键功能
- 6) LED 电源及工作状态指示

1.3 板载硬件与接口

1.3.1 板载硬件说明



1	Micro-USB	2	USB VBUS 指示灯
3	XH-LINK 状态指示灯	4	XH-LINK 主芯片模式跳针
5	Level shifter 去使能跳针	6	XH-LINK 复位按键
7	ISP 模式跳针	8	Target 电源跳针(3V3)
9	Target 调试/串口通信接口	10	XH-LINK 主芯片调试接口

■ 串口

XH-LINK 提供串口功能,可以在目标 MCU 和上位机之间进行串口通信。

∎ 按键

XH-LINK 提供一个复位按键,可在需要更新固件、更换工作模式等需要 DAP 复位的情况下使用。

- LED 指示灯
 - 1) XH-LINK 提供一个 VBUS 电源指示灯(LED1)和状态指示灯(LED2)。
 - 2) 当 USB 接口接入上位机时,VBUS 电源指示灯亮起。
 - 3) 当 XH-LINK 运行在不同工作状态时,状态指示灯呈现呼吸状态或闪烁状态。

XHSC小华半导体 Xiaohua semiconductor

- 供电
 - 1) 当需要 XH-LINK 为目标板供电时,可以选择使用 J8 的 5V0 或 3V3 引脚与目标板连接。
 - 当 XH-LINK 仅有 5PIN SWD 排线(即 J8 的 VCC、GND、CLK、DIO、RST)与目标板连接时, 将 Target 电源跳针(J2)使用跳帽短接,XH-LINK 即可为目标板提供 3.3V 电源。这时 XH-LINK 的供电能力受限于 USB VBUS 的电源能力。如果需要 5V 供电,可将连接 VCC 的排线调换到 5V0 引脚即可。

1.3.2 接口说明

XH-LINK 通过双排针 J8 与目标进行调试和串口通信,接口说明如下表 1-1 所示。

编程仿真接口 引脚标号	功能	备注				
5V0	5.0V 电源输出	输出 5.0V 电压,调试时可悬空该引脚,也可用作目标 MCU 系统供电				
3V3	3.3V 电源输出 输出 3.3V 电压,调试时可悬空该引脚,也可用作目标 MCU 系统供电					
RX	UART 数据接收引脚	连接目标板 MCU UART 数据发送引脚				
ТХ	UART 数据发送引脚	连接目标板 MCU UART 数据接收引脚				
RST	目标板 MCU 复位控制引脚	连接目标板 MCU 复位引脚				
CLK	SWD 接口时钟信号引脚	连接目标板 MCU SWD 串行线时钟引脚				
DIO	SWD 接口数据信号引脚	连接目标板 SWD 串行线数据输入/输出引脚				
GND	地	连接目标板 MCU 接地引脚				
		J2 断开时,VREF 输入,需外接电源作为参考电压,此时该管脚需外接目				
VCC	目标板电源参考输入	标芯片 VCC,电压范围 3V3~5V。				
		J2 仅在需要对目标板提供 3V3 电源时短接。(注*)				

表 1-1 接口说明

*注: J2 短接时, VCC 为目标板供电,这时该引脚不可从外部输入电压,以免损坏板子。

XHSC小华半导体 Xiaohua semiconductor

1.4 串口驱动安装

通过 XH-LINK 实现串口或在线编程功能时,若电脑操作系统为 Win7,则需要先安装虚拟串口驱动 (Win10 可忽略此步骤)。请联系相关技术支持人员获取虚拟串口驱动 vcom_driver_xhsc 文件,在打 开设备管理器后,按以下步骤安装:





● 更新驱动程序软件 - mbed Serial Port	×
浏览计算机上的驱动程序文件	
在以下位置搜索驱动程序软件: F:\VCOM_DRIVER_HDSC ▼ 浏览(R) ⑦ 包括子文件夹(I)	
第三步:选择此项	
→ 从计算机的设备驱动程序列表中选择(L) 此列表将显示与该设备兼容的已安装的驱动程序软件,以及与该设备处于同一类别下的 所有驱动程序软件。	
下一步(N) 取消	



🕒 🛛 🗉	新驱动程序软件 - mbed Serial Port	×
选择	译要为此硬件安装的设备驱动程序	
La	请选定硬件设备的厂商和型号,然后单击"下一步"。如果手头有包含要安装的驱动程序的 鱼,请单击"从磁盘安装"。	磁
(正花	王检索所有设备的列表)	
	第五步:选择"从磁盘安装" ————————————————————————————————————	•
	下一步(N) 取	消

🕞 🛯 更新	区动程序软件 - mbed Serial Port	83
选择要	为此硬件安装的设备驱动程序	
1	请选定硬件设备的厂商和型号,然后单击"下一步"。如果手头有包含要安装的驱动程序的磁 A. 法单:"以送来办法" 从磁盘安装	
(正在检	插入制造商的安装盘,然后确定已在下面选定正 确定 确的驱动器。 取消 即消	7
	第六步:点击"浏览" 制造商文件复制来源 (C):	
	从磁盘安装(H)	
	下一步(N) 取消	



查找范围 (I): 🚺 vcom_driver_xhsc 🖌 🔽 🕝 🤣 📂 🖽 -	
	-
第八步:双击此文件 桌面	
《 》 计算机	
文件名(0): 」xhsecac.inf 」 打开(0) 文件类型(T): 安装信息 (*.inf) ▼ 取消	

〕更新驱动程序较	欠件 - ∍1	bed Serial Port		×
🅘 🗋 更新驱动	动程序软件	牛 - mbed Serial Port		
选择要为此	硬件安装	的设备驱动程序		
」 して 一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一	选定硬件 / 请单击	设备的厂商和型号,然后单击"下一步"。如果手头: "从磁盘安装"。	有包含要安装的驱动程序的磁	
	人磁盘安全	装	×	
	4	插入制造商的安装盘,然后确定已在下面选定正 确的驱动器。 ————————————————————————————————————		
厂商			42/11	
%StdMf		第九步:点击"确定"		
(Generic				
(Standar		制造商文件复制来源(C):		
		D:\vcom_driver_xhsc	浏览 (8)	
📮 这个躯	动程序已	经过数字签名。	从磁盘安装(H)	
<u>告诉我</u>	为什么贩	动程序签名很重要		
			下一步 (8) 取消	





驱动开始安装,几秒后显示如下画面即表示安装正确:



2 功能介绍

2.1 CMSIS-DAP 调试

2.1.1 接口连接

调试器使用 SWD 接口与目标芯片连接进行调试。如果目标芯片有模式引脚,请按照目标芯片用户手册 要求来进行引脚配置,接线方式如下图 2-1 所示:



2.1.2 应用样例

本章节主要介绍如何连接 XH-LINK,如何配置 IAR Embedded Workbench 集成开发环境,完成调试。 步骤如下所示:

以 MCU HC32F460PETB, EV_F460_LQ100_Rev2.0 为例,连接图如下图 2-2 所示。请将 XH-LINK 板上 J2 的 3V3 与 VCC 用跳线帽短接,这样 XH-LINK 即可为目标板提供电源。



图 2-2 HC32F460 与 XH-LINK 连接图



实物图连接如下图 2-3 所示:



图 2-3 实物连接图

 IDE 这里以 IAR 为例,打开 IAR Embedded Workbench IDE 工程,点击"Project" -> "Options" -> "Debugger" -> "Setup", "Driver"设置项选择 CMSIS DAP。

efm_base - IAR Embedded Workbench IDE	- Arm 8.50.9	
File Edit View Project CMSIS-DAP Tools	Window Help	
1 🗅 🗋 🖬 🗐 🔚 🕹 🛍 🗂 1 つ C	- < Q, > \$ HE < 📮 > 🕢 🗈 🛯 🗮 🗢 📼 🔘 > 📲 👪 📮	
Workspace 👻 🖡 🗙	main.c x	
Debug	main()	
Files	Options for node *efm_base* Category: General Options Static Analysis Runtime Checking C/C++ Compiler Assembler Output Converter Quiston Build Build Actions Unker Debugger Simulator CADI GBS Server	Factory Settings
Debug Log	I-jet J-Link/J-Trace	
Log Fri Nov 03, 2023 14:15:43: IAR Embedded Workbench 8:50.9 (C:\ Program Files (x86)\/AR Systems\ Embedded Workbench 8:4\arm\bin\ armproc.dll) Fri Nov 03, 2023 14:15:44: Loading the CMSIS-DAP driver	TI Stellaris Nu-Link PE micro ST-LINK Third-Party Driver TI MSP-FET TI XDS	x
Build		
Messages Building configuration: efm_base - Debu Updating build tree	g DK Cancel	



3. 点击"CMSIS DAP" -> "Interface",设置 Interface 为 SWD。

Category: General Options	1						Factory Settings
Runtime Checking	Cature II	atorfaco	Due alue aliata				
C/C++ Compiler Assembler	Setup	literrace	Breakpoints				
Output Converter	Probe co	onfig	Probe confi	guration file			
Custom Build	Auto		Override	default			
Build Actions							
Linker	○ From	n file					
Debugger	○ Expli	cit	CPU:		Seleo	ct	
CADI	· · ·						
CMSIS DAP	Interfac	e	Explicit pro	be configuration			
GDB Server			🗆 Multi-tar	aet debug system			
I-jet	0,170	,	Tourse		A defension of	0	
J-Link/J-Trace	• SWD)	Target	number (TAP or N	nuitiarop	0	
Nu-Link			Targe	t with multiple CPU	ls		
PE micro			CB	L number on	0		
ST-LINK			CF	o number on			
Third-Party Driver	Interface	speed					
TI MSP-FET	Auto de	tect ~					
11 ADS							

4. 点击主菜单"CMSIS DAP"->"Memory Configuration...",根据目标 MCU 存储器映射,配置"Memory Configuratoin..."

File Edit View Project	CM	SIS-DAP Tools Window Help	
i 🗅 🗅 🔛 🕋 I 🖴 I 🗴	⇔,	Memory Configuration	< C
Workspace	~	Disable Debugger Cache	
Debug		Disable Interrupts When Stepping	
Files		Leave Target Running	(LL_OK
🗆 🌒 efm_base - Debug		ETM Trace Settings	BSP_LE
l—⊕ ≣ bsp		ETM Trace Save	id) EFM_
⊢⊕ ■ cmsis	1	ETM Trace	Ret2 =
	Z₽.	Function Trace	Addr +=
		Vector Catch	@ <i>rogra</i> 1 (LL_OK
- ⊕ i Output		Timeline	BSP_LE
		Function Profiler	id)EFM_ Ret3 =
		Session Overview	
		Breakpoint Usage	_FWMC_(Registe
		158 LL 159	PERIPH

🔮 efm_base - IAR Embedded Workbench IDE - Arm 8.50.9



ctory ranges elected device d :\xiaoHua\Projec specifies the follo	description file in Project Opti cct/gitlab/hc32_ddf/xxxx_dd	ions: \projects\ev_hc32f460_lqfp1					
elected device d ::\xiaoHua\Projec specifies the follow	description file in Project Opt ct\gitlab\hc32_ddl\xxxx_dd	ions: Il\projects\ev_hc32f460_lqfp1					
:\xiaoHua\Projec	ct\gitlab\hc32_ddl\xxxx_dd	ll\projects\ev_hc32f460_lqfp1					
pecifies the follow			00_v2\examples\e	efm\efm_base\EWARM	/////drivers/cmsis/D	De	
	owing default memory ranges	E.					
lone	Name	Start	End	Type	Size	Extra	
ed ranges							
ed ranges is is the memory Cone	configuration that will be us Start End	ed. You can modify this as ne 1 Cache 1	eded. Type Si	.ze Ext	ra Comment		New
id ranges is the memory ione lemory	configuration that will be us Start End 0x0000000 0x	ed. You can modify this as ne 1 Cache 1 0007FFFF ROM/F1a	eded. Fype Si ash 51	ize Ext	ra Comment Flash	1	New
d ranges s is the memory one emory emory	configuration that will be us Start End 0x00000000 0x1 0x03000C00 0x1	ed You can modify this as ne 1 Cache 1 0007FFF ROM/F1a 03000FFB ROM/F1a	eded. Fype Si ash 51 ash 10	ize Ext 12 kbytes 120 bytes	ra Comment Flash OTP]	New Edt
d ranges s is the memory one emory emory emory emory	configuration that will be us Start End 0x00000000 0x1 0x15FF8000 0x1	ed. You can modify this as ne 1 Cache 1 0007FFF ROM/F1a 3000FFB ROM/F1a 20026FFF RAM	eded. Type Si ash 51 ash 10 18	ze Ext 2 kbytes 20 bytes 18 kbytes	ra Comment Flash OTP SRAM]	New Edt Pemove
ed ranges is is the memory Cone Lemory Lemory Lemory Lemory Lemory	configuration that will be us Start End 0x0000000 0x1 0x1FFF800 0x1 0x200F0000 0x1	ed. You can modify this as ne d Cache 1 0007FFF ROM/Fla 0000FFF ROM/Fla 20026FFF RAM 200F0FFF RAM	eded. Fype Si ash 51 ash 10 18	ze Ext 2 kbytes 20 bytes 8 kbytes kbytes	ra Comment Flash OTP SRAM RET_SRAM]	New Edit Remove Use Factor
ed ranges is is the memory Cone (encry lemcry (encry (encry (encry	configuration that will be us Start En 0x0000000 0x1 0x1FF8000 0x2 0x200F0000 0x1 0x200F0000 0x1	ed You can modify this as ne d Cache 1 0007FFFF ROM/F1a 0000FFF ROM/F1a 20026FFFF RAM SFFFFFFFF RAM SFFFFFFFF Uncach	eded Type Si ash 51 ash 10 18 4 ed/SFR 51	ze Ext 12 kbytes 120 bytes 18 kbytes 12 kbytes 12 kbytes 12 Mbytes	ra Conment Flash OTP SRAM RET_SRAM Peripheral		New Edit Bemove Use Factor
ed ranges is is the memory Zone Aemory femory femory femory femory femory	Start Enc 0x0000000 0x1 0x03000000 0x1 0x1FFF8000 0x2 0x40000000 0x1 0x4000000 0x1 0x40000000 0x1	ed You can modify this as ne <u>Cache 1</u> 0007FFFF ROM/F12 20026FFF RAM 20070FFF RAM 200F0FFF RAM 30FFFFFFF ROM/F12 MOREFFFFF ROM/F12 MOREFFFFF ROM/F12 MOREFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	eded. Type Si ash 51 ash 10 18 ed/SFR 51 ash 64 CFR 51	ze Ext 22 kbytes 20 bytes 18 kbytes 2 Mbytes 1 Mbytes 1 Mbytes	ra Conment Flash OTP SRAM RET_SRAM Peripheral OSPI OSPI		New Edt Remove Use Factor

5. 编译代码,编译通过后点击"Download and debug",进入调试状态。

efm_base - IAR Embedded Workbench IDE - Arm 8.50.9

File Edit View Project	CMSIS-DAP Tools	Window Help
i 🗅 🗅 🖬 🕋 🔚 🛛 🗶	00 5 C	- < Q > \$= HE < Q > 2 D 🗈 🛛 📼 🛛 🗩 📜 🚛 📜
Workspace	→ ↓ ×	main.c x
Debug	~	main()
Files	* • *	<pre>58 * Function implementation - global ('extern') and local ('static') 59 60 61 → /** 62 * @brief Main function of EFM project 63 * @param None 64 * @retval int32_t return value, if needed 65 */ 66 int32_t main(void) 67 → { 68 uint32_t u32Addr; 70 uint16_t u16Data16Bit = 0x96U; 71 uint32_t u32Data32Bit = 0x5A5A5A5U; 72 uint8_t u8Read8Bit; 73 uint16_t u16Read16Bit; 74 uint32_t u32Read32Bit; 75 int32_t i32Ret1, i32Ret2, i32Ret3; 76</pre>
efm_base		<

efm_base - IAR Embedded	Workbench ID	Arm 8.50.9		- 🗊 X
File Edit View Project Det	Disassemi	CMSIS-DAP Tools Window Help ▼ < Q > \$ HE < Q > 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1	- 📜 j em son i 🚓 🗍	
Workspace	▼ # ×	nain.c X	·	Disassembly 🔻 🎙 :
Debug		ain()	fo	Go to Memory
Files © fine_base - Debug © fine_sp: ©	•	<pre>9 * American Applementation - global ("statut") 9 *** 9 * American Applementation - global ("statut") 9 *** 9 * American Applementation - global ("statut") 9 *** 9 **** 9 ***********************</pre>		Diessembly 0x428: 0x4000'0x82 0x428: 0x4000'0x82 0x4000'0x82 0x440: 0x2000'004 0x4000'0x82 0x441: 0x2000'004 0x4000'0x82 0x441: 0x2000'004 0x4500'0x82 0x451: 0x4001'4408 0x451'0x4001'4408 0x452: 0x4001'4408 0x451'0x4001'4408 0x451: 0x4001'408 0x451'0x400'4408 0x452: 0x4001'408 0x455'0 0x452: 0x4005'0 0x455'0 0x455: 0x465'0 0x455'0
efm_base			> ~	<pre><</pre>
Debug Log				→ 0 3
Log Thu Nov 02, 2023 16:35:3 Thu Nov 02, 2023 16:35:3 Configuring trace using ¹ N Thu Nov 02, 2023 16:35:3 ON-Chip (ETEM/TB) trac supported by this process disabled.	8: Target rese 8: INFO: (TB' setting 9: INFO: e mode is not sor - trace is			
Build				- a :
Messages Building configuration: efr Updating build tree Configuration is up-to-dat	n_base-Deb e.		File	Line

2.2 串口通信

串口通信需要连接 XH-LINK 上的 RX、TX 到目标板的 USART 的 TX、RX。还是以 MCU HC32F460PETB, EV_F460_LQ100_Rev2.0 为例,样例采用 adc_base,该样例可通过串口 USART4 打印 AD 采样值和对 应的电压值数据。

请按照下图 2-4 所示连接 XH-LINK 与目标板。



图 2-4 串口通信连接图

实物连接如下图 2-5 所示:



图 2-5 实物连接图



串口通讯步骤如下:

 打开串口调试助手,按照代码配置来设置相应的端口、波特率、数据位、检验位、停止位等,设置 完成后打开端口,等待接收数据。

文件(F) 编辑(F) 视图(V) 工具(T) 控制(C) 帮助(H)		
. E ∞ ► I = C + -	- 🛅 🅸		
串口设置			
端			
波特率 115200 💌			
数据位 8			
校验位 None ▼			
停止位 1			
流 控 None ▼			
▶ 自动换行			
□ 显示发送			
□ 显示时间			
发送设置			
ASCII C Hex			发送
□ 日利重友 1000 <u></u> ms]		
	5		-
COM14 OPENED, 115200, 8, NONE, 1, OF	Rx: 0 Bytes Tx:	0 Bytes	Alithon

2. 参考应用样例章节进行 IDE 的配置和编译下载,下载后全速运行。可观察到串口数据输出。

File Edit View Project De	bug Disassemb	oly CMSIS-DAP Tools Window Help	
10 10 🖬 🗿 🖶 🕹 🛍	DOC	- < Q, > ⇆ +표 < 📮 > d 🍺 📓 🚳 🛲 🕒 C 🔕 📮 (A-7+ r* +1 + 🕨) 🖬 + 🖕 💷 🐝 🖡	fa
Workspace	▼ ¤ ×	main.c x Readme.txt	•
Debug		main()	fo
Files	•	<pre>91 → /** 92 → ** 93 ** 94 ** 95 ** 95 ** 95 ** 96 ** 96 ** 96 ** 97 ** 96 ** 97 ** 97 ** 97 ** 98 ** 97 ** 98 ** 9</pre>	
adc base		<	> ~



文件(E) 编辑(E) 视图(V) Ⅰ目(T) 控制	(1) 契助(4)	
串口设置	The ADC value of potentiometer is 662, voltage is 533 mV	
泄 □ COM14/USP 电行设备)	The ADC value of potentiometer is 662, voltage is 533 mV	
	The ADC value of potentiometer is 663, voltage is 534 mV	
波特率 115200 🔻	The ADC value of potentiometer is 662, voltage is 533 mV	
*1.0 12	The ADC value of potentiometer is 662, voltage is 533 mV	
剱据12 8 ▼	The ADC value of potentiometer is 662, voltage is 533 mV	
校验位 None ▼	The ADC value of potentiometer is 662, voltage is 533 mV	
	The ADC value of potentiometer is 662, voltage is 533 mV	
停止位 1	The ADC value of potentiometer is 662, voltage is 533 mV	
渣 控 None ▼	The ADC value of potentiometer is 663, voltage is 534 mV	
	The ADC value of potentiometer is 663, voltage is 534 mV	
接收设置	The ADC value of potentiometer is 662, voltage is 533 mV	
ix to Kill	The ADC value of potentiometer is 662, voltage is 533 mV	
ASCII C Hex	The ADC value of potentiometer is 662, voltage is 533 mV	
反 自动换行	The ADC value of potentiometer is 662, voltage is 533 mV	
	The ADC value of potentiometer is 663, voltage is 534 mV	
□ 显示发送	The ADC value of potentiometer is 662, voltage is 533 mV	
□ 显示时间	The ADC value of potentiometer is 662, voltage is 533 mV	
	The Abe value of potentionneter is doz, vortage is 555 mv	-
┌发送设置		
ASCII C Hex		
□ 白动重发 1000 ····		发送
1 日初重次 1000 - ms		
	r.	
	>	<u> </u>
COM14 OPENED, 115200, 8, NONE, 1, 0	F Rx: 1,682 Bytes Tx: 0 Bytes	Alithon

2.3 ISP 在线编程

XH-LINK 具有 USB 转串口功能,可配合 CM ISP 软件来进行在线编程,支持 HC32F120、HC32M120 系列芯片。使用在线编程功能前,需要将 XH-LINK 的 J6 用跳帽短接。



对于 HC32F120 系列 MCU,接线方法如下图 2-6 所示:



图 2-6 HC32F120 系列连接图



对于 HC32M120 系列 MCU, 接线方法如下图 2-7 所示:



图 2-7 HC32M120 连接图

具体编程方法请参考《Cortex-M 在线编程器用户手册》。

3 XH-LINK 调试器固件升级

调试器固件升级有两种方式。一种是 PC 机通过 USB 连接 XH-LINK,U 盘拖拽的方式直接更新;一种是通过烧写的方式下载固件到 XH-LINK。下面章节将分别介绍这两种方式。

3.1 固件拖拽方式升级

用户可通过 USB 接口来进行固件更新,请联系相关技术支持人员获取最新的固件。

请按照以下步骤进行操作来进行固件拖拽更新:

1. 将 J1 的 GND 引脚和 UBOOT 引脚用跳帽短接。



2. 将 XH-LINK 使用 USB 连接线接入 PC,此时状态灯 LED2 呈红色。



3. 将新的固件复制到 PC 上新增的 U 盘内。



复制完成后,移除 J1 跳帽,并按复位按键或重新上电。此时状态灯 LED2 呈呼吸灯状态,表示固件更新完成。

3.2 固件烧写方式升级

- 3.2.1 准备工作
 - 1. 硬件:
 - 1块 XH-LINK 板;
 - 2根 Micro USB 数据线;
 - 1个 USB 转串口模块;
 - 2. 软件
 - 固件烧写 hex 文件;
 - 烧录工具 XHSC ISP;

*注:请联系我司技术支持人员获取最新的烧写文件。

3.2.2 硬件设置

1. 将 J1 的 MD 和 GND 用跳帽短接。



2. 将串口模块与 XH-LINK 的 J5 连接,接线方法如下图所示:



图 3-1 串口模块与 XH-LINK 接线图

3. 最后使用 2 根 USB 数据线分别将串口模块和 XH-LINK 接入 PC。



图 3-2 整体连接框图

3.2.3 烧写过程

本次烧写的固件 hex 文件以 DAPLink_v0256_Integration-20221014.hex 为例,演示正确的烧写过程, 步骤如下所示。

- 1. 按下 XH-LINK 上的复位按键,并打开 xhsc.exe,设置如下:
 - 目标 MCU HC32F460xExx/ HC32F45xxExx
 - 通信波特率 1000000
 - Hex 文件 DAPLink_v0256_Integration-20211014.hex
 - 端口设置 选择 USB 转 UART 模块所识别的 COM 口,单击"读 hex"按钮

Х岩SС小华半导体 хіаониа ѕемісолоистоя

H XHSC MCU Programmer	– 🗆 X
文件 帮助	
MCU设置 HC32F460xExx/HC32F45xxExx ~ 目标MCU HC32F460xExx/HC32F45xxExx ~ 通信波特率 1000000 ~ Hex文件 户Link_v0256_Integration-20: 端口设置 COM4 MCU Flash信息 MCU Flash信息 日劫编号 开始地址 0x 0000000 万大小 0万大小 8192 当前编号: 当前编号: 町皮编号: 町皮编号:	编号次数 编号步进
操作	XHSC V2.09
请确认目标芯片与串口模块的连接: ▲ 加密 全选 MCU. VCC <>> 串口模块. VCC 00-03 04-07 08- MCU. GND <>> 串口模块. RXD 16-19 20-23 24-3 MCU. RXD (PA14) <> 串口模块. RTS/DTR 32-35 36-39 40-4 MCU. MODE <> MCU. GND 48-51 52-55 56-4	11 12-15 27 28-31 43 44-47 59 60-63
	0%

2. 确保"擦除"、"空白检测"、"编程"、"验证"已勾选,然后单击"执行"按钮。

H XHSC MCU Programmer			– 🗆 🗙
文件 帮助			
MCU设置 HC32F460xExx/HC32F45xxExx ~ 通信波特率 1000000 ~ Hex文件 .PLink_v0256_Integration-20! 端口设置 COM4 ~	MCU Flash信息 开始地址 0x 00000000 页大小 8192 页数量 64 Flash大小 512K	 □ 自动编号 开始地址 0x 起始编号 当前编号: 历史编号: 	编号次数 编号步进
操作 ● 片擦除 ○ 页擦除	上传 ✓ 擦除 ✓ 空白检测 ✓ 编程 ✓ 验证 □ 读保护1 □ 读保护2	读Hex 执行	X#SC V2.09
请确认目标芯片与串口模块的连接: MCU. VCC 〈──〉 串口模块. VCC MCU. GND 〈──〉 串口模块. GND MCU. TXD (PA13) 〈──〉 串口模块. RXD MCU. RXD (PA14) 〈──〉 串口模块. TXD MCU. RESET 〈──〉 串口模块. RTS/DTR MCU. MODE 〈──〉 MCU. GND	▲ 加密 00-0 16-1 32-3 ↓ 48-5	 全选 3 04-07 08-11 9 20-23 24-27 5 36-39 40-43 1 52-55 56-59 	28-31 44-47 60-63
			0%



3. 等待编程结束,信息框出现"验证成功"。

HXHSC MCU Programmer				_	×
文件 帮助					
MCU设置 HC32F460xExx/HC32F45xxExx ~ 目标MCU HC32F460xExx/HC32F45xxExx ~ 通信波特率 1000000 ~ Hex文件 PLink_v0256_Integration-202 端口设置 COM4 ~	MCU Flash信息 开始地址 0x 0000 页大小 8192 页数量 64 Flash大小 5124	00000 开始地址 2 2 2 3 2 3 前编号 3 5 5 5 5 5 5 5 5 5 5 5 5 5	명 Ox 3	扁号次数 扁号步进	
操作 ● 片擦除 ○ 页擦除	上传 又 到	部余 2 白 检测 读 Hex 幸 征 気保 护 1 気保 护 2	执行	X#S V2.09	С
<		IP語 ● 全法 00-03 04-0 16-19 20-2 32-35 36-3 48-51 52-5	08-11 08-11 09 40-43 05 56-59	12-15 28-31 44-47 60-63	
					100%

 断开串口模块和 XH-LINK 之间的连接,移除 J1 跳帽,按下复位按键,此时状态指示灯应该处于绿 色由明到暗的呼吸灯状态,指示固件烧写成功。点击 WIN10"安全删除硬件并弹出媒体",应能看 到固件版本号,指示 XH-LINK 被 PC 正确识别。







版本修订记录

版本号	修订日期	修订内容
Rev1.0	2023/11/24	初版发布。
Rev1.01	2024/01/24	XHLINK、XLINK 均修改为 XH-LINK。