

## LAT1115

Local Application Tips

#### STM32CubelDE 实用技巧之仿真带外扩 FLASH 的工程

关键字: STM32CubeIDE, 仿真, 外扩 FLASH

### 1. 前言

在开发 STM32 时,往往在很多种场景下都需要外扩 FLASH,典型场景就是开发图形应用时,需要耗费大量来存放图形资源的存储器,很多客户会选择将图形资源放在外扩的 FLASH 中,那么在开发阶段也就难免需要对这样的工程进行下载和调试,当选择 STM32CubeIDE 时,我们又应当如何设置呢?

### 2. 准备

由于 STM32CubeIDE 集成了 STM32CubeProgrammer,这使得调试和下载程序到外 扩的 FLASH 变得十分简单。

本次我们选择 STM32G071RB NUCLEO 板+X-NUCLEO-GFX01M1 显示扩展板,借助 X-CUBE-DISPLAY 软件包中的范例程序,为大家演示在 STM32CubeIDE 中,如何设置调试带外部 FLASH 的工程。

本次实验所用到的资源链接,如下所示:

G071RB NUCLEO 板的链接: <u>NUCLEO-G071RB - STM32 Nucleo-64 development</u> <u>board with STM32G071RB MCU</u>, <u>supports Arduino and ST morpho connectivity -</u> STMicroelectronics

显示扩展板的链接: X-NUCLEO-GFX01M1 - Display expansion board for STM32 NUCLEO - STMicroelectronics

X-CUBE-DISPLAY 软件包的链接: X-CUBE-DISPLAY - Display module software expansion for STM32Cube - STMicroelectronics

在调试之前,我们只需要将"X-CUBE-DISPLAY"软件包中,对应的"External Loader"文件复制到 STM32CubeIDE 的安装目录中,如下的路径:

"C:\ST\STM32CubeIDE\_1.7.0\STM32CubeIDE\plugins\com.st.stm32cube.ide.mcu.exte rnaltools.cubeprogrammer.win32\_2.0.100.202110141430\tools\bin\ExternalLoader" 。

对于如何制作 External Loader 文件,大家可以参考

"UM2237<u>STM32CubeProgrammer software description - User manual</u>

#### 图1. X-CUBE-DISPLAY 软件包中的 External Loader 文件



X-CUBE-DISPLAY > 2.0.0 > Utilities > PC_Software > ExternalLoader
Name
MX25L6433F_GFX01M1_STM32G0.stldr
MX25L6433F_GFX01M1_STM32L4.stldr
MX25L6433F_GFX01M2_STM32G0.stldr
MX25L6433F_GFX01M2_STM32WB.stldr
MX25L6433F_GFX02Z1_STM32L4.stldr
MX25L6433F_GFX02Z1_STM32U5.stldr
ReadMe.txt

#### 图2. 需要添加 External Loader 到 STM32CubeIDE 中的路径

C\ST\STM32CubelDE_1.7.0\STM32CubelDE\plugins\com.st.stm32cube.ide.mcu.externaltools.cubeprogra					
Share View					
- This PC > Windows (C:) > ST > STM32CubelDE_1.7.0 > STM32CubelDE > plugins > com.st.stm32cube.ide.mcu.externaltools.cubeprogrammer.win32_2.0.100.202110141430 > tools > bin > ExternalLoader					
Name	^	Date modified	Туре	Size	

## 3. 调试

在 STM32CubeIDE 中打开范例程序,工程编译后,在 "Debug Configurations"中, 勾选 "External Loader"选项,选择 STM32G0 NUCLEO 板对应的 External Loader 文件。

Debug Configurations		-	□ ×
Create, manage, and run configurations			<b>À</b>
🖹 🔮 🕞 🚺 🐻 🕈 🗕	Name: GFX01M1_HelloWorld Debug		
type filter text C/C++ Application C/C++ Attach to Application	Main * Debugger > Startup > Source Common Debug in low power modes: Suspend watchdog counters while halted: No configuration		
C (V++ Postmorten Debugger C (V++ Postmorteniation C GDB Hardware Debugging S Launch Group ✓ Inf STN32 Cortex-M C/C++ Application Inf GFX01M1_HelloWorld Debug	Serial Wire Viewer (SWV)	RTOS Kernel Awareness     Enable RTOS Proxy     Driver settings     Driver: ThreadX     Port: cortex_m0     Cort sumbar: 150000	
	Misc	D.Projects\NUCLEO-6071RB\Applications\GFX01M1_HelloWorld\STM32CubeIDE\Debug\st-link_g Browse	
	External Loader: MX25L6433F_STM326071B_NUCLEO     Shared ST-LINK     Max halt timeout(s): 2	6x9000000, SPLFLASH, MX25L6433F, GFX01M1_STM3260.stldd ∨ Scan ■ Initial	lize
Filter matched 8 of 9 items			PP0
0		<u>D</u> ebug	Close

图3. STM32CubeIDE 的调试选择的设置

点击"Debug"就可以进入调试界面进行仿真调试了。





<pre>The Lab provide labor in page labor in the Labor in</pre>	CONTRACTOR AND	W181				
<pre>c c c c c c c c c c c c c c c c c c c</pre>	Ela Edit Source Refector Navigate Search D	1/App/app_	alspinyk - simskubelue - Moniema - Maha			- 0 ^
GOUDULINGUARY 10 A particular       Canado Add 10 Apple (Apple (App	Die Ton Source wenden Hendere Selere D					
Control Jokewick : Source : So		* * 4				94 at 199 th 19
# Debg 31 # procession       C max       # NumberStructure       C may difference       We dif	GFX01M1_HelloWorld > Application > User >	bispla	Y > 📴 App > 🗲 app_display.c			
	🟦 Debug 🐹 🍓 Project Explorer 👘 🖂		: 📄 STM32G071R8TX, FLASH.1 🕐 stm32gDox_hal.c 📴 Qx1HH2160 🧲 app_display.c 🕱 🛅 GFX01M1, HelloWorld.mia. 🗮 Image1.h 🗢 🗉			🚓 Reg 👥 Liv 🐹 🛲 SFR 💣 Fau 📟 🖽
<pre>&gt;</pre>	Fa 🔆 👥 :					<b>X 祝 日</b>
<pre></pre>	✓ C GFX01M1_HelloWorld Debug [STM32 Cortex-M C/C++ A		<pre>staticIO uint8_t can_move = 0;</pre>		Туре	Value
<pre>image in marker inter int</pre>	GFX01M1_HelloWorld.elf [cores: 0]			readme_file		Failed to evaluate expression
<pre>     menod at maxes (10 \$400000     move = 1;</pre>	MX DISPLAY Process() at ann display.cr291.0x8	282	if (TransferAllowed)	V Colmages	image_t [7]	
<pre>/* Insplay something */ if movor easight B31299210</pre>	main() at main.c:105 0x8000404			Images[0]	image_t	[_] 200
<pre># \$FLINK(GTLINK GDE server) 225 if (key) 226 { 226 { 227 { 228 key = 0; 228 key = 0; 229 { 229 { 229 { 220 {</pre>	📕 arm-none-eabi-gdb (8.3.1.20191211)		/* Display something */	Height	uint32 t	240
<pre>236 { 237 /* Reset key value */ 238 key = 0; 239 /* Check if we can allow scrolling up/down the picture */ 239 /* Check if we can allow scrolling up/down the picture */ 240 if ((Ico_rientation == LCO_ORIENT)    (LCO_Orientation == KEY_ORIENT) 241 /* Allow moving the picture on screen */ 243 { 244 /* Allow moving the picture on screen */ 255 can move = 1; 266 } 267 if (Tanges[image_id].Weight); 267 posy = 0) /* reload new image */ 268 { 269 BSP_LCO_Clear(0, 0, 0, 0, LCO_Width.1CO_Height); 269 bSP_LCO_Clear(0, 0, 0, 0, LCO_Width.2CO_Height); 260 posy = ((LCO_Height - Images[image_id].Weight)/2); 360 } 360 { 360 } 360 { 360 } 360 <math> 360 \\ </math></pre>	📕 ST-LINK (ST-LINK GDB server)		if(key)	Ma bpp	uint8_t	
<pre> %* Outa world; %* Conta world; %* Conta</pre>						0x900a8c00 <image1> <error: access="" cannot="" n<="" td=""></error:></image1>
<pre>key = 0; /* Check if we can allow scrolling up/down the picture */ */* Allow moving the picture on screen */ can move = 1; /* Allow moving the picture on screen */ can move = 1; /* Allow moving the picture on screen */ can move = 1; /* Allow moving the picture on screen */ can move = 0 /* reload new image */ /* BSP_LCD_Clear(0, 0, 0, UCD_Width, LCD_Height); */* allow moving image_id].Height/2); */* allow move = 0.1 * reload new image */ /* allow move = 0 /* reload new image */ /* allow move = 0 /* reload new image id].Height/2); */* allow move = 0 /* reload new image id].Heigh</pre>				14 Data	uint8_t	
<pre>239 249 249 250 250 26 mapped mapped mapped in apped in apped</pre>			key = 0;	) images[1]	image_t	
<pre>290 /* Check if we can allow scrolling up/down the picture */</pre>				5 Contrages[3]	image_t	
<pre>*201 if ((can_nove == 0) &amp;&amp; (Images[image_id].Height &lt; LCD_Height) \ 202 &amp;&amp; ((LCD_orientation == LCD_ORIENTAITON_PORTRAIT)    (LCD_orientation == KEY_ORIENT 203 { 204</pre>			/* Check if we can allow scrolling up/down the picture */	> 🥭 images[4]	image_t	
<pre>88 ((LCD_Orientation == LCD_ORIENTATION_PORTRAIT)    (LCD_Orientation == KEY_ORIENT 23 24 25 25 26 27 26 27 27 27 27 27 27 27 27 27 27 27 27 27</pre>		€291	if ((can move == 0) && (Images[image id].Height < LCD Height) \	> 🥭 images(5)		
<pre>233 { 244 /* Allow moving the picture on screen */ 255 can_move = 1; 267 } 267 } 277 if(posy == 0) /* reload new image */ 278 { 279 BSP_LCD Clear(0, 0, 0, LCD Width, LCD_Height); 400 posx = ((LCD_Width - Images[image_id]width)/2); 400 if(fimages[image_id]width( &lt; LCD_Height) //2); 400 } 302 { 403 posy = ((LCD_Height - Images[image_id].Height)/2); 304 } 305 else 306 { 307 posy = 0; 308 } 309 } 309 } </pre>			&& ((LCD Orientation == LCD ORIENTATION PORTRAIT)    (LCD Orientation == KEY ORIENT	) 🥭 Images(6)	image_t	
<pre>294</pre>				+ Rad new expression		
<pre>295 can_move = 1; 296 } 297 if(posy == 0) /* reload new image */ 298 { gsp LCD Clear(0, 0, 0, 0, LCD Width, LCD Height); 400 posx = ((tCLWidth - Images[image_id].Width)/2); 400 if(fimages[image_id].Height &lt; LCD.Height) 302 { 403 posy = ((LCD_Height - Images[image_id].Height)/2); 304 } 305 } 306 { 307 posy = 0; 308 } 308 } 309 } 309 } 309 } 309 } 309 } 309 } 309 } 300 } 30</pre>			/* Allow moving the picture on screen */			
<pre>296 297 3f(posy == 0) /* reload new image */ 298 { 299 BSP_LCD_Clear(0, 0, 0, LCD_Width, LCD_Height); 300 posy = ((LCD_Width - Images[image_id].Width)/2); 301 if(Images[image_id].Height &lt; LCD_Height) 302 303 posy = ((LCD_Height - Images[image_id].Height)/2); 304 305 alse 306 306 306 307 posy = 0; 308 309 309 309 309 309 309 309 309 309 309</pre>			can move = 1;			
<pre>297 if(posy == 0) /* reload new image */ 298 { 299 BSP_LCD_Clear(0, 0, 0, 0, LCD_Width, LCD_Wigth); 0 300 posx = ((LCD_Width - Images[image_id].Width)/2); 0 301 if(Images[image_id].Width)/2); 0 302 { 0 303 posy = ((LCD_Height - Images[image_id].Height)/2); 0 304 } 305 else 306 { 307 posy = 0; 308 } 308 } </pre>						
<pre>298 { 299 BSP_LCD_CLear(0, 0, 0, LCD_Hidth, LCD_Height); 4300 posx = ((LCD_Hidth - Images[image_id].Width)/2); 4301 302 303 304 305 alse 306 306 306 306 306 306 306 306 306 306</pre>			if(posy == 0) /* reload new image */			
<pre>299 BSP_LCD_Clear(0, 0, 0, LCD_Midth, LCD_Height);</pre>						
<pre>*300</pre>			BSP LCD Clear(0, 0, 0, LCD Width, LCD Height);			
<pre>+301 if(Images[image_id].Height &lt; LCD_Height) 302 {</pre>			<pre>posx = ((LCD Width - Images[image id].Width)/2);</pre>			
302 {			<pre>if(Images[image id].Height &lt; LCD Height)</pre>			
<pre>*303</pre>			{			
304 } 305 else 306 { 307 posy = 0; 308 } 309			<pre>posy = ((LCD Height - Images[image_id].Height)/2);</pre>			
305 <b>*1se</b> 306 { 307 <b>posy = 0;</b> 308 } 309						
306 { 307 posy ≈ 0; 308 } 309						
307 posy = 0; 308 } 399						
308 }			posy = 0;			
309						
		309				

# 4. 小结

大家在平日里使用 STM32CubelDE 时,对于调试可能仅用到常规的调试功能.其实 STM32CubelDE 的调试设置界面里,还有很多可以设置的功能选项,大家可以多多尝 试,多多体验。



#### 参考文献

【如有, 请注明; 否则, 请注明: 无】

文件编号	文件标题	版本号	发布尔日期
UM2237	STM32CubeProgrammer software description	Rev 17	23-Nov-2021

### 文档中所用到的工具及版本

STM32CubeIDE V1.7.0

版本历史

日期	版本	变更
2022年04月11日	1.0	首版发布



#### 重要通知 - 请仔细阅读

意法半导体公司及其子公司("ST")保留随时对 ST 产品和 / 或本文档进行变更的权利, 恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。 ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用, ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定,将导致 ST 针对该产品授予的任何保证失效。

ST和 ST 徽标是 ST 的商标。若需 ST 商标的更多信息,请参考 www.st.com/trademarks。所有其他产品或服务名称均为其 各自所有者的财产。

本文档是 ST 中国本地团队的技术性文章,旨在交流与分享,并期望借此给予客户产品应用上足够的帮助或提醒。若文中内容存有局限或与 ST 官网资料不一致,请以实际应用验证结果和 ST 官网最新发布的内容为准。您拥有完全自主权是否采纳本文档(包括代码,电路图等)信息,我们也不承担因使用或采纳本文档内容而导致的任何风险。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2020 STMicroelectronics - 保留所有权利