



HPM6000 系列 MCU Flash 使用指南(三)

-----片外 Flash 从 XPI0 CB 端口启动

先楫半导体

HPM6000系列MCU Flash使用指南(三) -----片外Flash从 XPI0 CB 端口启动

目录

1 简介.....	4
2 OTP 及启动镜像配置的修改.....	6
3 Flash 程序烧写.....	16
4 总结.....	19

版本:

日期	版本号	说明
2023-10-30	V1.0	初版

1 简介

HPM6000 系列微控制器是来自上海先楫半导体科技有限公司的高性能实时 RISC-V 微控制器，为工业自动化及边缘计算应用提供了极大的算力、高效的控制能力及丰富的多媒体功能。

HPM6000 系列微控制器现有以下子系列:

- HPM6700 系列双核高性能微控制器
- HPM6400 系列单核高性能微控制器
- HPM6300 系列高性能高性价比微控制器
- HPM6200 系列高实时性混合信号微控制器

本文主要介绍了 HPM6000 系列以上型号微控制器使用一款片外 Flash 时的注意事项:

- 使用 XPI0 CB 端口启动时 OTP 如何配置
- 使用 XPI0 CB 端口启动时启动镜像如何配置

如果用户想使用片内的 Flash 的时候，可以参考《HPM6000 系列 MCU 片上 Flash 使用指南》。

本文介绍的测试环境如下所示:

- Flash 型号: GD25Q40C 系列
- MCU 芯片型号: HPM6750IAN2
- 软件环境: SDK v1.3.0, HPMcicro Manufacturing Tool.

相关的软硬件资料，可以在先楫的官网上查询。

本文使用 BGA196 封装的 HPM6750IAN2 芯片，相比 BGA289 封装的 HPM6750IVM2 芯片，并没有引出 XPI0 CA 端口的引脚，如下图 1 部分引脚复用功能所示，最后需要选择 XPI0 的 CB 端口用于连接 NOR Flash，XPI 部分电路原理图如下图 2 所示:

封装		PIN 名称	数字功能	模拟功能	IO 电源	速度
BGA_289	BGA_196					
J15	H12	PE07	GPIO_E_07(ALT0) UART13_RXD(ALT2) SPI1_DAT3(ALT5) XPI0_CB_SCLK(ALT14) PWM3_P_1(ALT16) ETH0_RXD_3(ALT18) UTMI_D_2(ALT31)	-	VIO_B07	高速
J16	-	PE08	GPIO_E_08(ALT0) UART12_CTS(ALT3) CAN2_RXD(ALT7) XPI0_CA_CS0(ALT14) SDC0_DATA_1(ALT17) ETH0_MDC(ALT19) UTMI_RX_DM(ALT31)	-	VIO_B08	高速
J17	-	PE09	GPIO_E_09(ALT0) UART12_DE(ALT2) UART12_RTS(ALT3) CAN2_TXD(ALT7) XPI0_CA_SCLK(ALT14) SDC0_DATA_0(ALT17) ETH0_MDIO(ALT19) UTMI_RX_DP(ALT31)	-	VIO_B08	高速
K16	-	PE10	GPIO_E_10(ALT0) UART15_CTS(ALT3) I2C1_SDA(ALT4) CAN0_STBY(ALT7) XPI0_CA_D_2(ALT14) SDC0_CMD(ALT17) ETH1_MDC(ALT19) UTMI_RX_RCV(ALT31)	-	VIO_B08	高速
K17	-	PE11	GPIO_E_11(ALT0) UART15_DE(ALT2) UART15_RTS(ALT3) I2C1_SCL(ALT4) CAN1_STBY(ALT7) XPI0_CA_D_0(ALT14) SDC0_CLK(ALT17) ETH1_MDIO(ALT19) UTMI_TX_SE0(ALT31)	-	VIO_B08	高速



图 2

2 OTP 及启动镜像配置的修改

在 HPM6750EVK2 评估板中使用了 XPI0 CA 端口作为连接串行 NOR Flash 的引脚，(芯片的默认配置)，在芯片上电启动的过程中，芯片内部的 BootROM 会根据 OTP 配置初始化 XPI 控制器，从 Flash 0x400 的地址读取 XPI NOR 配置选项，并按照读取的配置选项将 XPI 配置为指定的工作模式，然后重新检测 Flash 是否存在，从 Flash 中加载镜像完成剩余启动步骤。XPI NOR 启动流程如图 3 所示：

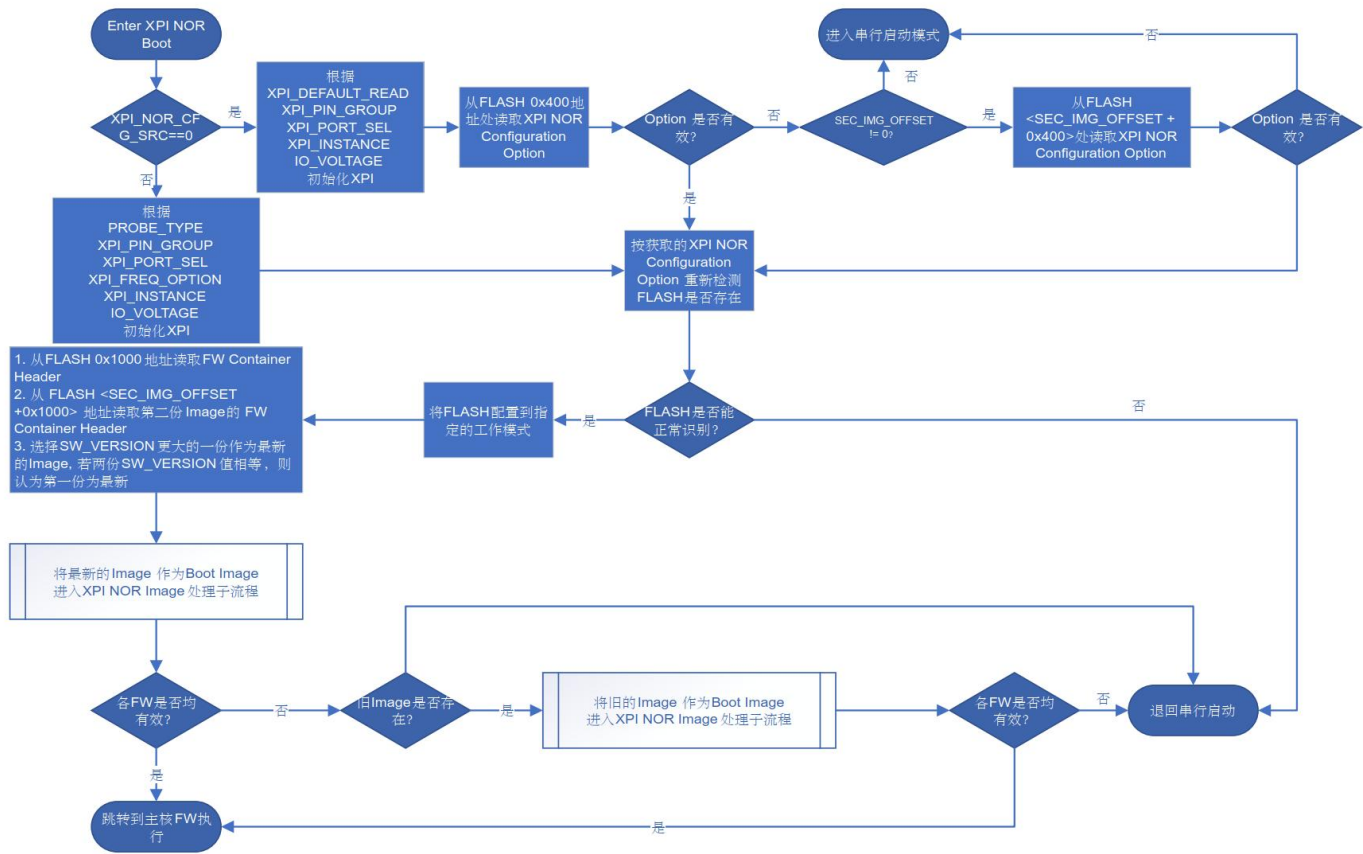


图 3

因此为了实现从 XPI0 CB 端口启动，用户需要修改 OTP 中的对应的字段以及启动镜像中的 XPI NOR 配置选项。

2.1 OTP 的烧写

通过查阅 HPM6700 系列的参考手册，发现控制 BootROM 启动时指定 Flash 连接方式的为以下几个字段，如图 4 所示：

- 根据 XPI_INSTANCE 决定 XPI 的实例
- 根据 XPI_PIN_GROUP 选择默认的 XPI 引脚分组
- 根据 XPI_PORT_SEL 决定 XPI 默认的连接端口

OTP 字索引	位偏移	字段宽度 (位)	字段	描述
1	0	4	LIFE_CYCLE_A	Life Cycle(生命周期)
	8	8	PUBK_REVOKE	Public Key REVOKE(公钥撤销) 低 4 位有效, bit_n 代表撤销第 n 个公钥
	28	4	LIFE_CYCLE_B	Duplicate Life Cycle(重复的生命周期) 实际的生命周期值为 LIFE_CYCLE_A LIFE_CYCLE_B
3	0	32	SW_VER	软件最低版本号
24	0	4	XPI_FREQ_OPTION	XPI 频率选项 详见 XPI 频率选项
	4	1	XPI_INSTANCE	XPI 实例 0 - XPI0 1 - XPI1
	5	1	XPI_PIN_GROUP	XPI 引脚分组 0 - 第一组 1 - 第二组
	6	2	XPI_PORT_SEL	XPI 端口选择 0 - XPI_CA (仅支持 CS0) 1 - XPI_CB (仅支持 CS0)

图 4

因此在本文测试中，仅需修改 XPI_PORT_SEL 字段为 1，让 BootROM 从 CB_CS0 端口启动即可。

用户可以通过 HPMicro 量产工具来烧写 OTP 字段，将 BOOT 拨码开关调至 BOOT_MODE[1:0]= 0b10，以 ISP 模式启动，将 USB0 连接至 PC，然后打开 HPMicro Manufacturing Tool.

选择对应的 USB 设备点击 连接，连接成功后如下图 5 所示：

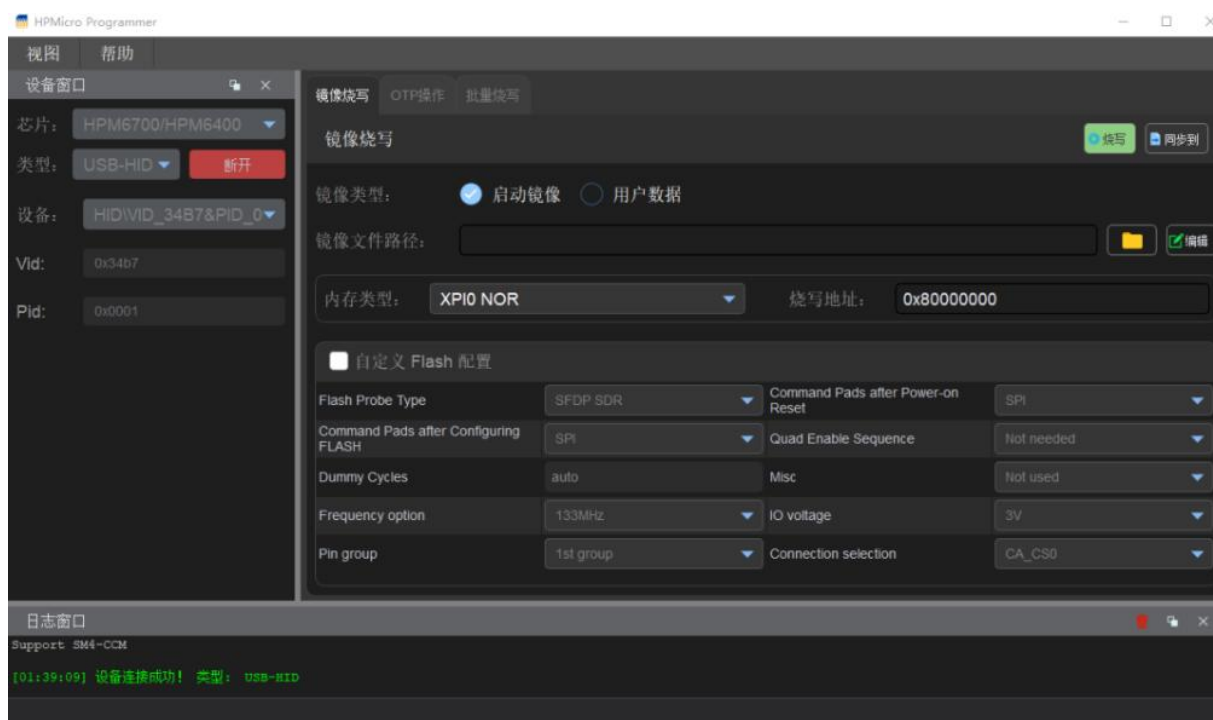


图 5

需要用户注意，OTP 的操作由于是**不可逆的**，每一位只能够烧写一次，所以 OTP 操作需要非常谨慎。切到 OTP 操作页面中，点击刷新读取当前芯片中 OTP 的数据，默认 Word 24 的值为 0x00000000，也就是从 XPI0 CA 端口的第一组引脚启动，如下图 6 所示：

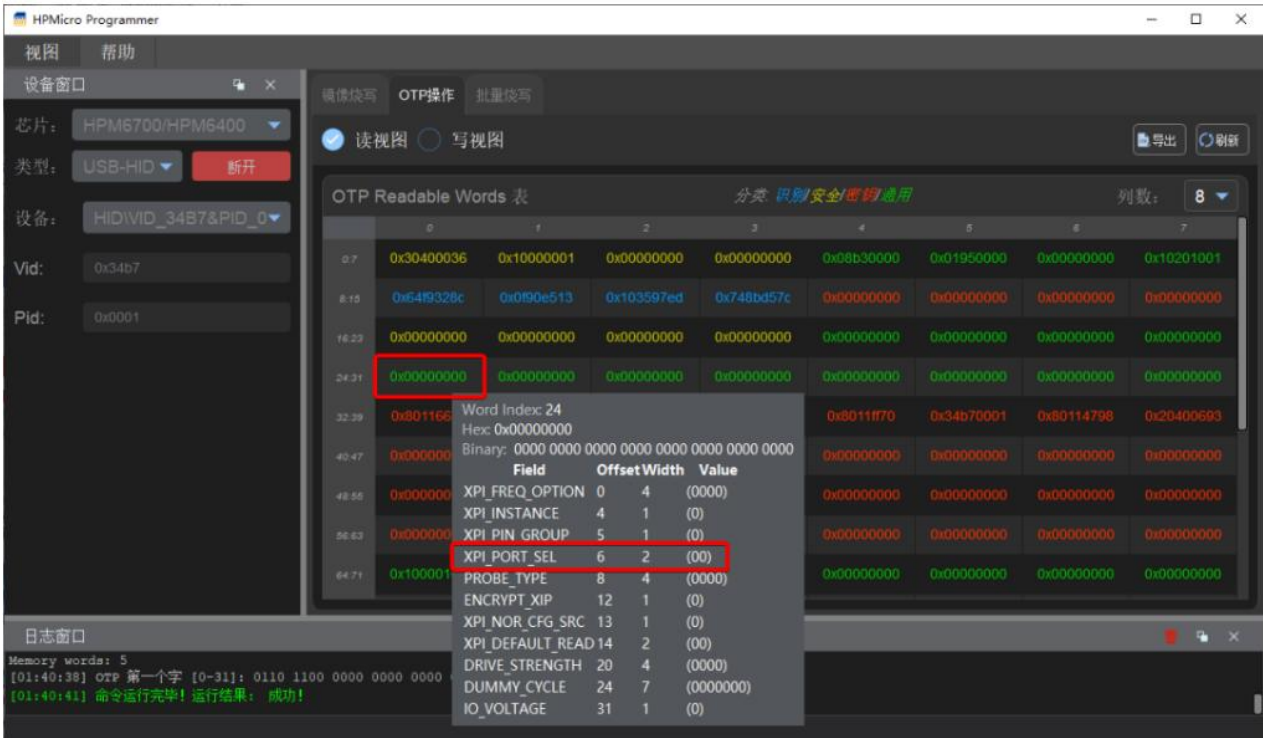


图 6

切换至写视图，点击 添加 Words，如下图 7 所示：

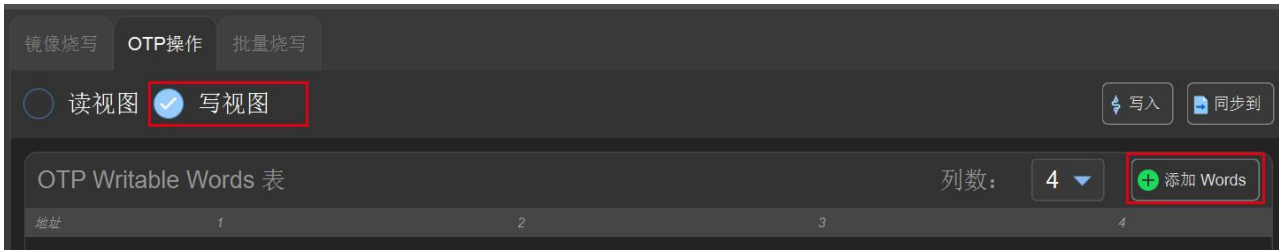


图 7

在弹出的对话框中按照下图 8 输入，即：将 Word 24 的 bit 6 置 1，从 CB_CS0 端口启动。点击确定后，再次确认数据是否有误，确认无误后点击写入。修改内容如下图所示：

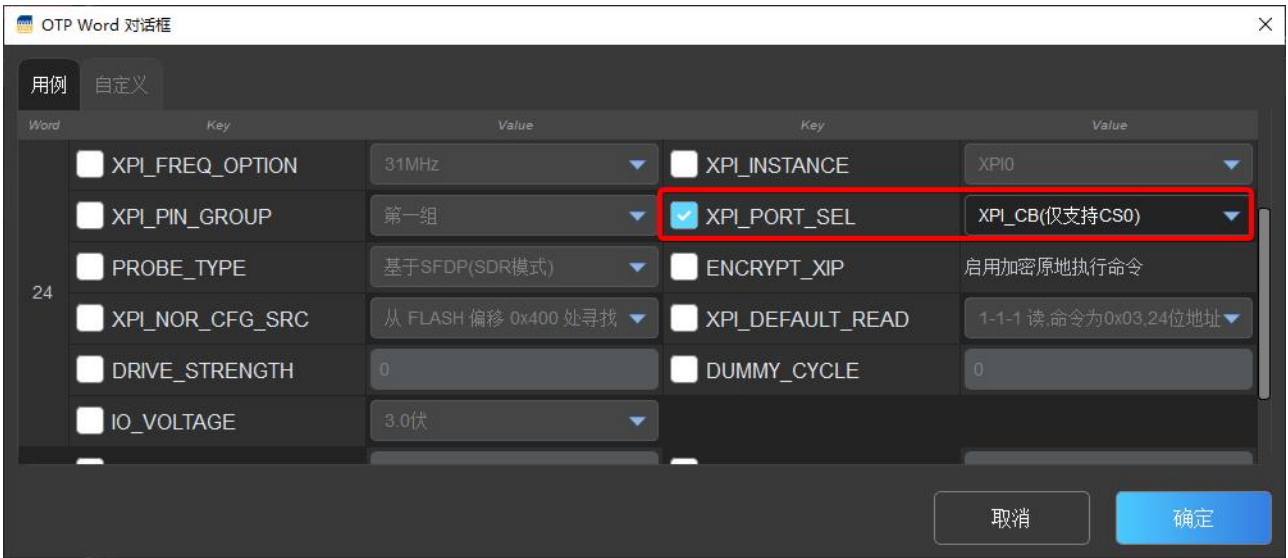


图 8

写入完成后切换回读视图，点击刷新，可以看到 Word 24 已经变为期望值，如下图 9

所示：

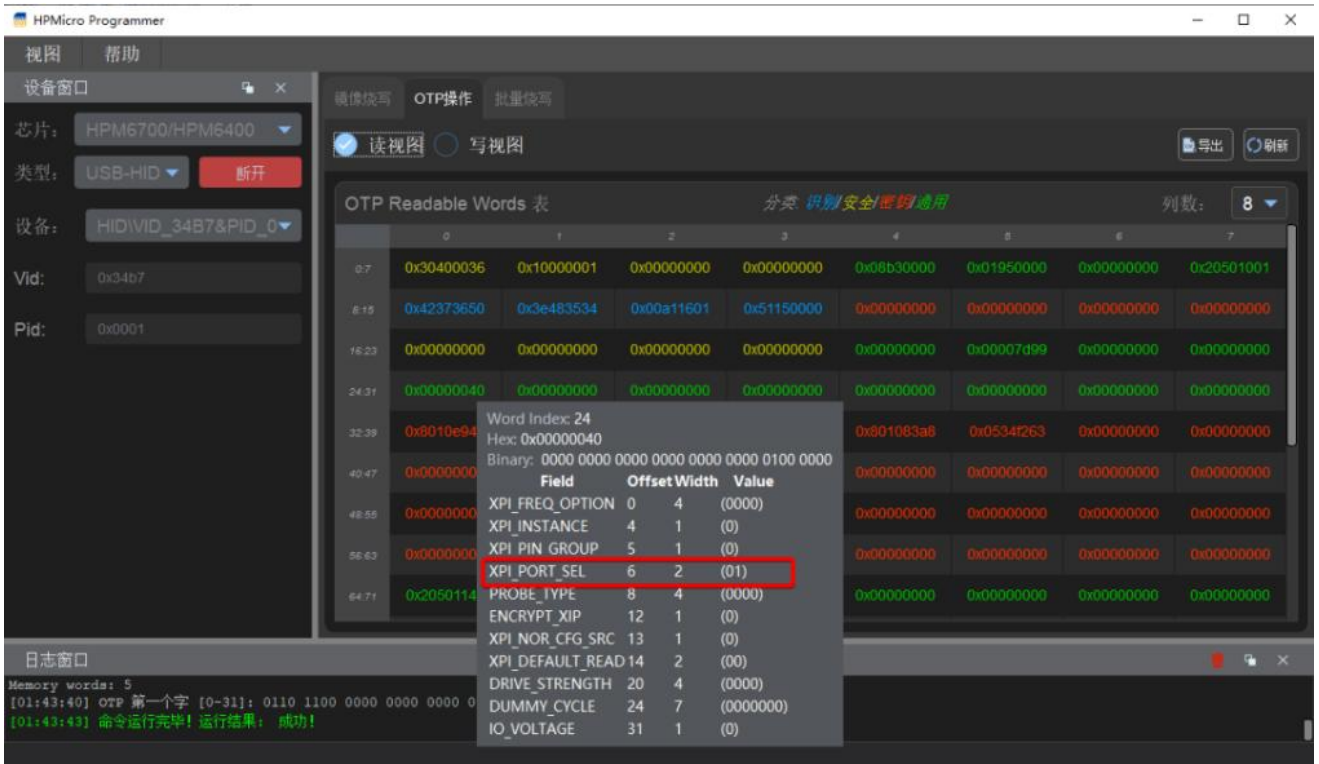


图 9

将拨码开关拨回 XPI NOR 启动后复位芯片，可使用逻辑分析仪观察 XPI0_CB_SCLK 引脚在上电时的电平，如可以观察到对应的时钟信号，证明 OTP 设置已经生效。

2.2 启动镜像的配置

在 BootROM 检测到 Flash 后，会从 0x400 读取 XPI NOR 配置选项重新配置 XPI，为了确保 XPI 重新配置后仍能检测到 Flash，需要对默认配置进行修改。根据 HPM6700 系列的参考手册，配置选项的定义如下表格：

XPI NOR 配置选项

偏移 (字节)	字段	描述
0x00	Header	Bit[31:12] - tag: 标记, 必须为 0xfcf90 Bit[11:4] - 保留 Bit[3:0] - words: 配置选项字数 (不包括 Header 本身)

偏移 (字节)	字段	描述
0x04	Option0	<p>Bit[31:28] - 探测类型 0 - SFDP SDR, 1 - SFDP DDR 2 - 1-4-4 Read 0xEB, 3 - 1-2-2 Read 0xBB 4 - HyperBus 1V8, 5 - HyperBus 3V0 6 - OctaBus DDR, 8 - Xccela DDR 10 - EcoXiP DDR</p> <p>Bit[27:24] - 上电复位后发送命令的数据引脚数 0 - SPI, 1 - DPI, 2 - QPI, 3 - OPI</p> <p>Bit[23:20] - FLASH 配置完后发送命令的数据引脚数 0 - SPI, 1 - DPI, 2 - QPI, 3 - OPI</p> <p>Bit[19:16] - 1-4-4 模式使能的序列 0 - 不需要或者自动 1 - QE bit is at bit 6 in Status Register1 2 - QE bit is at bit1 in Status Register2 3 - QE bit is at bit7 in Status Register2 4 - QE bit is at bit1 in Status Register2 and should be programmed by 0x31</p> <p>Bit[15:8] - dummy cycles 0 - 自动探测/默认值 其它 - 用户指定的值, 对于 DTR 读, 该值需要是实际数据手册上的 <i>dummycycle</i>×2</p> <p>Bit[7:4] - 杂项 0 - 无 1 - SPI 模式 2 - 内部回环模式 3 - 强制外部 DQS</p> <p>Bit[3:0] - 频率选项 0 - 不改变现有配置 Others - SoC 相应的值</p>

偏移 (字节)	字段	描述
0x08	Option1	该字段在“words”字段值大于 1 时有效 Bit[31:20] - reserved Bit[19:16] - IO 电压 0 - IO is 3V, 1 - IO is 1.8V Bit[15:12] -引脚分组选择 0 - 第一组, 1 - 第二组 Bit[11:8] - 连接方式选择 0 - PORTA_CS0, 1 - PORB_CS0, 2 - PORTA_CS0 +PORTB_CS0, 3 - PORTA_CS0+PORTA_CS1, 4 - PORTB_CS0+PORTB_CS1 Bit[7:0] - IO 驱动强度 0 - 默认值 (最大驱动强度) 其它 - 详见 (TBD)
0x0C	Option2	该字段在“words”大于 2 时有效 Bit[31:16] - reserved Bit[15:12] - 擦除 sector 命令选项 0 - Erase 4KB, 1 - Erase 32KB, 2 - Erase 64KB, 3 - Erase 256KB Bit[11:8] - Sector 大小选项 0 - 4KB, 1 - 32KB, 2 - 64KB, 3 - 256KB Bit[7:0] - FLASH 大小选项 0 - 4MB, 1 - 8MB, 2 - 16MB

表 70: XPI NOR 配置选项 (XPI NOR Configuration Option)

NOR_CFG_OPTION 的区域，已经在 XPI0 的基地址偏移 0x400 处存放，这就是 XPI Flash 配置选项。

%HPM_SDK_BASE%\boards\HPM6750EVK2 路径下的 board.c 中，就可以找到 NOR Flash Configuration 选项，以下是 HPM6750EVK2 的默认代码：

```

#if defined(FLASH_XIP) && FLASH_XIP
__attribute__((section(".nor_cfg_option"))) const uint32_t option[4] = {0xfc90001, 0x00000007, 0x0,
0x0};

#endif
    
```

由于是自己画的电路板，推荐用户创建一个新的 board 文件，避免修改 SDK 中的文件，本文以 HPM6750IAN2 为例。

%HPM_SDK_BASE%\boards\路径下，将 hpm6750evk2 的文件夹复制一份，并新命名为 hpm6750IAN2。

%HPM_SDK_BASE%\boards\openocd\boards\路径下，将 hpm6750evk2 的脚本文件复制一份，并新命名为 hpm6750IAN2。

%HPM_SDK_BASE%\boards\hpm6750IAN2\路径下首先修改 yaml 配置文件的名称，名称同新 board 名称一致，故改为 hpm6750IAN2。

随后打开此配置文件，修改 name，与新 board 名称一致；修改 device 为 hpm6750xANx,如下图 10 所示：

```
board:
  name: hpm6750IAN2
  soc: HPM6750
  device: HPM6750xANx
  openocd-soc: hpm6750-dual-core
  openocd-probe: cmsis_dap
```

图 10

那么此时就已经新增了一个新 board 的信息了，在 start_gui.exe 中已经可以识别到了，如下图 11 所示：

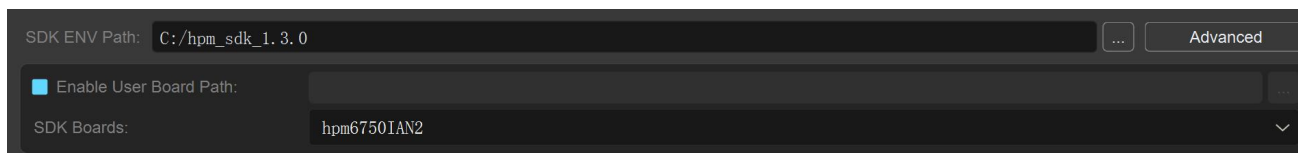


图 11

这时，就可以前往%HPM_SDK_BASE%\boards\hpm6750IAN2 路径下的 board.c 中修改 NOR Flash Configuration，本文使用 XPI0 CB_CS0 端口，所以需要将 Header[3:0] 改为 2，Option[11:8] 改为 1。即：

```
#if defined(FLASH_XIP) && FLASH_XIP
__attribute__((section(".nor_cfg_option"))) const uint32_t option[4] = {0xfc90002, 0x00000007,
0x00000100, 0x0};
#endif
```

3 Flash 程序烧写

修改好 board.c 的配置之后，用户有两种方式来进行程序的下载调试。

第一种方式：当用户通过 Segger Embedded Studio 在线调试时，调用的 openocd 的脚本文件(%HPM_SDK_BASE%\boards\openocd\boards\hpm6750IAN2)也要对 flash option 修改相应的配置，在相应代码旁有注释，如下图 12 所示：

```

> hpm_53evk_bringup > hpm_sdk > boards > openocd > boards > hpm6750evk2.cfg
1 # Copyright (c) 2022 HPMicro
2 # SPDX-License-Identifier: BSD-3-Clause
3
4 # openocd flash driver argument:
5 # - option0:
6 #   [31:28] Flash probe type
7 #     0 - SFDP SDR / 1 - SFDP DDR
8 #     2 - 1-4-4 Read (0xEB, 24-bit address) / 3 - 1-2-2 Read(0xBB, 24-bit address)
9 #     4 - HyperFLASH 1.8V / 5 - HyperFLASH 3V
10 #     6 - OctaBus DDR (SPI -> OPI DDR)
11 #     8 - Xccela DDR (SPI -> OPI DDR)
12 #     10 - EcoXiP DDR (SPI -> OPI DDR)
13 #   [27:24] Command Pads after Power-on Reset
14 #     0 - SPI / 1 - DPI / 2 - QPI / 3 - OPI
15 #   [23:20] Command Pads after Configuring FLASH
16 #     0 - SPI / 1 - DPI / 2 - QPI / 3 - OPI
17 #   [19:16] Quad Enable Sequence (for the device support SFDP 1.0 only)
18 #     0 - Not needed
19 #     1 - QE bit is at bit 6 in Status Register 1
20 #     2 - QE bit is at bit1 in Status Register 2
21 #     3 - QE bit is at bit7 in Status Register 2
22 #     4 - QE bit is at bit1 in Status Register 2 and should be programmed by 0x31
23 #   [15:8] Dummy cycles
24 #     0 - Auto-probed / detected / default value
25 #     Others - User specified value, for DDR read, the dummy cycles should be 2 * cycles on FLASH datasheet
26 #   [7:4] Misc.
27 #     0 - Not used
28 #     1 - SPI mode
29 #     2 - Internal loopback
30 #     3 - External DQS
31 #   [3:0] Frequency option
32 #     1 - 30MHz / 2 - 50MHz / 3 - 66MHz / 4 - 80MHz / 5 - 100MHz / 6 - 120MHz / 7 - 133MHz / 8 - 166MHz
33 # - option1:
34 #   [31:20] Reserved
35 #   [19:16] IO voltage
36 #     0 - 3V / 1 - 1.8V
37 #   [15:12] Pin group
38 #     0 - 1st group / 1 - 2nd group
39 #   [11:8] Connection selection
40 #     0 - CA_CS0 / 1 - CB_CS0 / 2 - CA_CS0 + CB_CS0 (Two FLASH connected to CA and CB respectively)
41 #   [7:0] Drive Strength
42 #     0 - Default value
43
44 # xpi0 configs
45 # - flash driver:      hpm_xpi
46 # - flash ctrl index: 0xF3040000
47 # - base address:     0x80000000
48 # - flash size:       0x20000000
49 # - flash option0:    0x7
50 flash bank xpi0 hpm_xpi 0x80000000 0x20000000 1 1 $_TARGET0 0xF3040000 0x7
51

```

图 12

本文需要使用 XPI0 CB_CS0 端口，那么应该改为：

```
flash bank xpi0 hpm_xpi 0x80000000 0x20000000 1 1 $_TARGET0 0xF3040000 0x7 0x00000100
```

用户修改此处的 Flash option 时要与 board.c 中保持一致。

此脚本文件中还有 SDRAM 的相关配置，如果用户没有外挂 SDRAM，可以将此处注释

掉。如下图 13 所示:

```
338  $_TARGET0 configure -event reset-init {  
339      init_clock  
340      #init_sdram  
341  }
```

图 13

做好修改, 调整 boot0=0, boot1=0, 确保 BOOT_MODE 是从串行 NOR FLASH 启动, 就可以正常调试下载程序了。本文以 Hello world 工程为例, 使用 start_gui.exe, 选用刚刚创建的 hpm6750IAN2 生成 flash_xip 类型工程并打开, 如下图 14 所示:

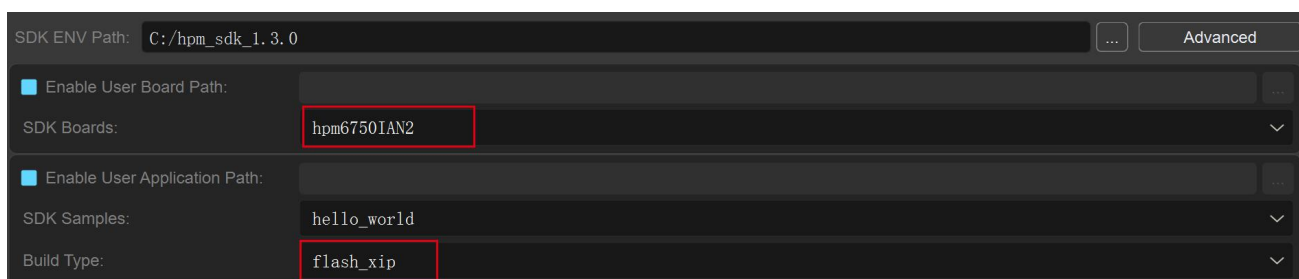
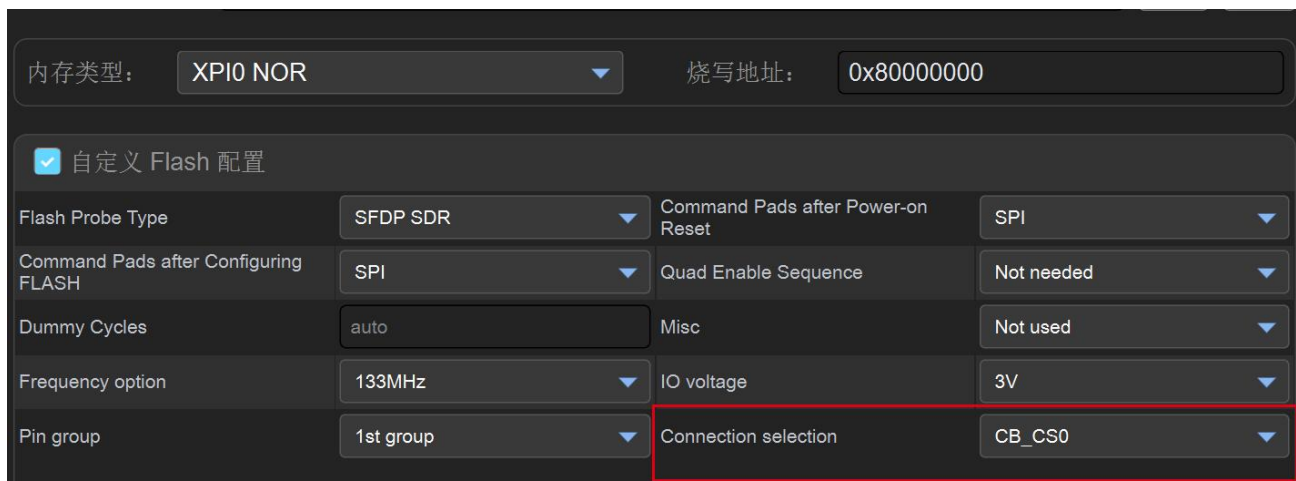


图 14

编译成功之后, 点击 DEBUG->GO, 下载并成功运行工程, 串口成功打印相应信息。

第二种方式: 如果用户通过 HPMicro Programmer 下载 Flash, boot pin 调整到 boot0=0, boot1=1, 连接成功之后, 接下来要在这里修改配置, 如图 15 所示, 选择引脚连接为 CB 端口。编译好在 board.c 中修改过 `nor_cfg_option` 的 Flash 工程。载入编译后生成的 BIN 文件, 就可以烧写了, 以 Hello world 为例, 路径地址为:

hpm_sdk\samples\hello_world\hpm6750IAN2_flash_xip_build\segger_embedded_studio\Output\Debug\Exe\demo.bin。



内存类型:	XPI0 NOR	烧写地址:	0x80000000
<input checked="" type="checkbox"/> 自定义 Flash 配置			
Flash Probe Type	SFDP SDR	Command Pads after Power-on Reset	SPI
Command Pads after Configuring FLASH	SPI	Quad Enable Sequence	Not needed
Dummy Cycles	auto	Misc	Not used
Frequency option	133MHz	IO voltage	3V
Pin group	1st group	Connection selection	CB_CS0

图 15

成功通过工具烧写了 Flash 程序之后，将 boot pin 调整回 boot0=0, boot1=0 的时候，按下 Reset 按键，程序运行成功，串口成功打印相应信息。

用户修改此处的自定义 Flash 配置时也要与 board.c 中的 NOR Flash Configuration 保持一致。

4 总结

本文选择 XPI0 的 CB 端口用于连接 NOR Flash，介绍了用户需要修改的启动镜像的配置，以及如何烧写相应 OTP。本文使用的电路以及 Flash 芯片都是验证过的，如果用户根据本文的介绍修改了引脚配置还是没有烧写成功，可参考《先楫半导体 HPM6000 系列微控制器片外 Flash 使用指南》，检查电路以及启动镜像中 Flash 芯片相关的配置是否正确。