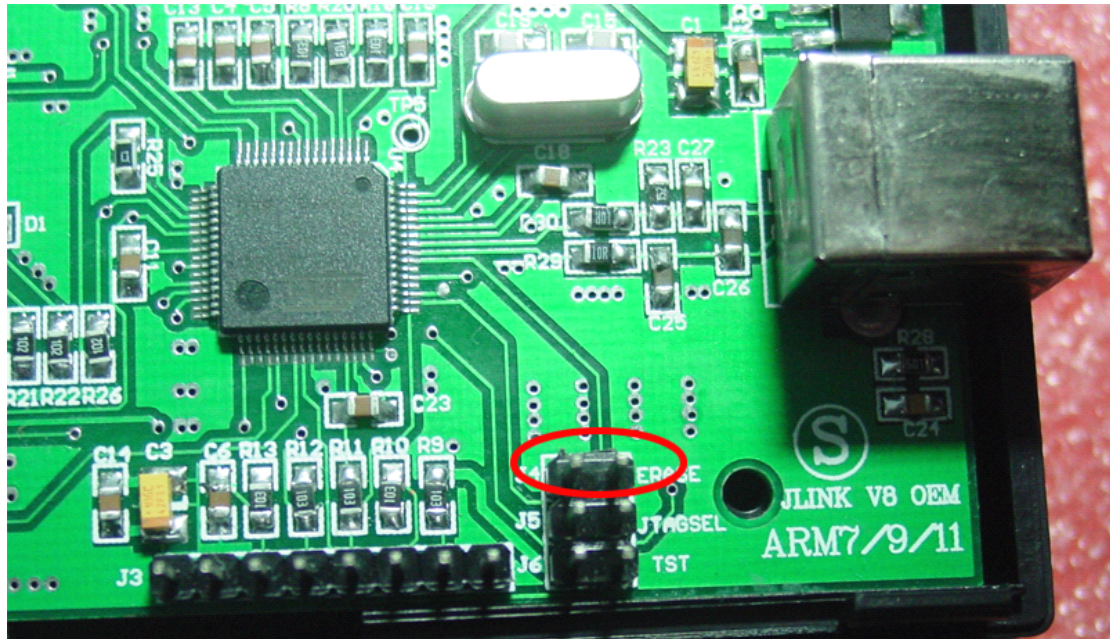


JLINK 烧写步骤

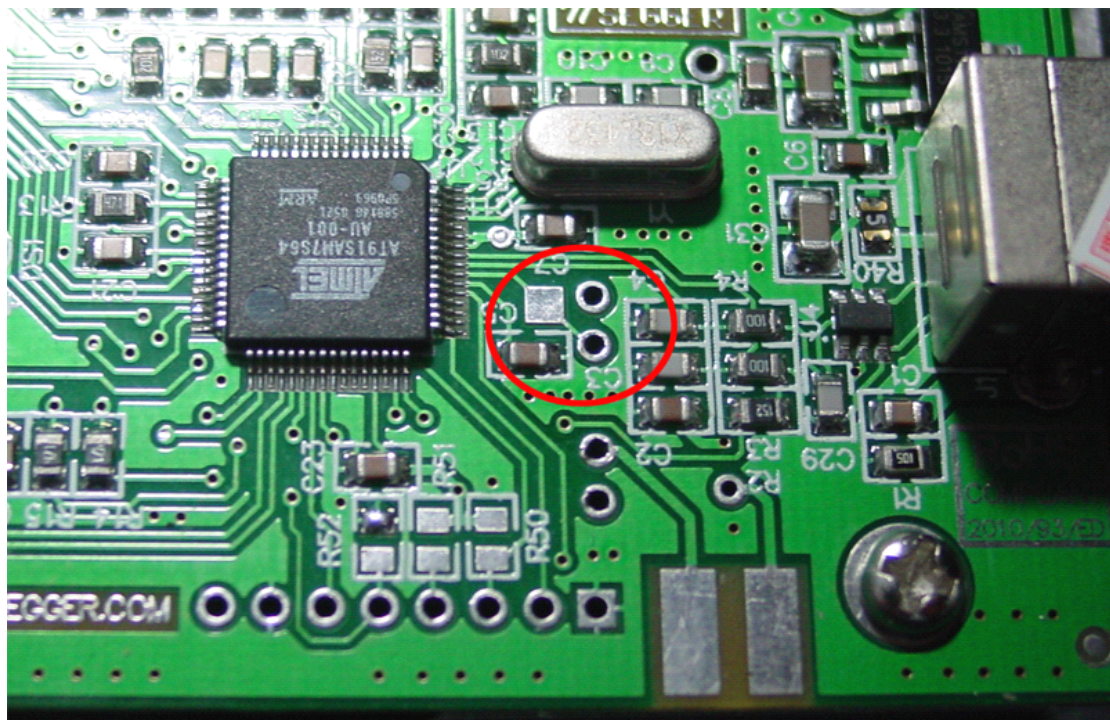
步骤 1 擦除 FLASH (需要先擦除 FLASH 后 JTAG 才能连接得上)

上电-->拉高 ERASE(短接跳线)-->等待 20 秒-->断电-->恢复 ERASE(移除跳线).

A 型 JLINK 的 ERASE 所在的位置:

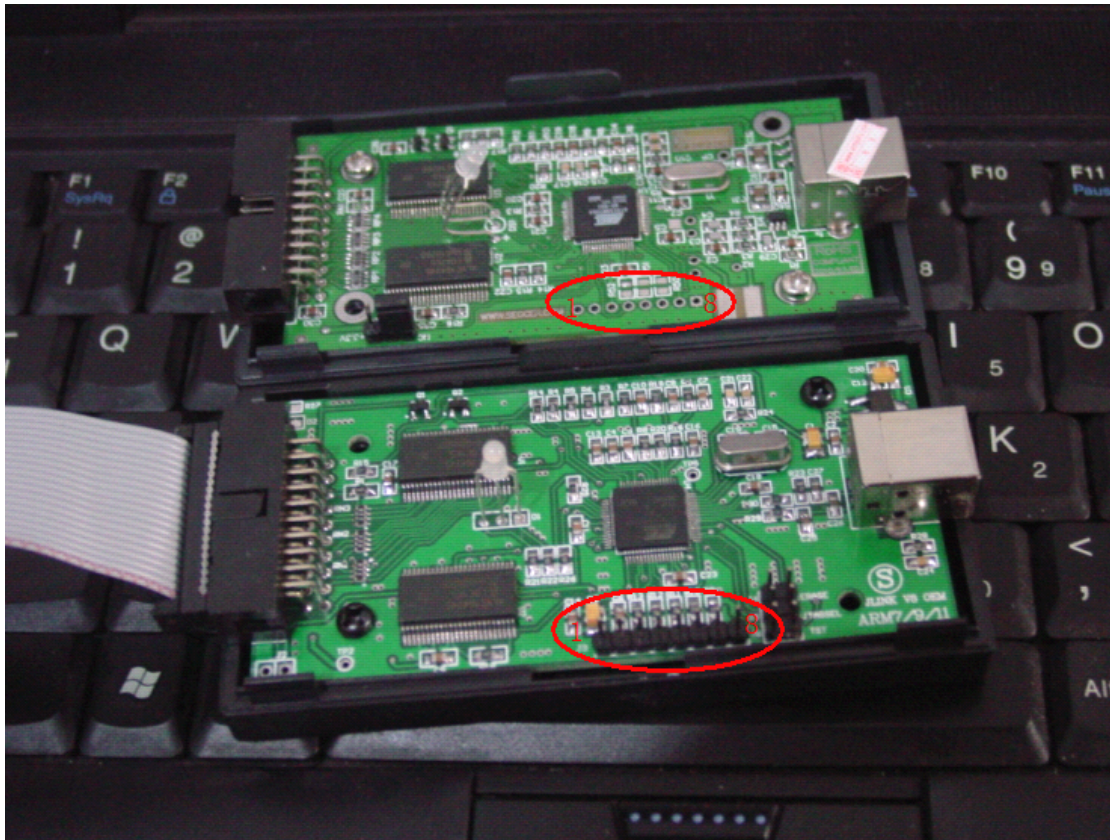


B 型 JLINK 的 ERASE 所在的位置:

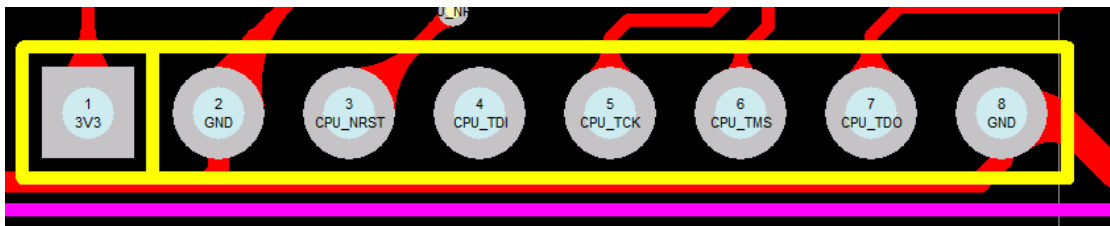


步骤 2 恢复使用另外的 JLINK 连接需要烧写固件的 JLINK

1.JLINK 上面的 JTAG 接口图:(左边为第一脚)



JLINK 的 JTAG 接口定义图:



从左到右分别为: **VCC**, **GND**, NRST, TDI, TCK, TMS, TDO, **GND**.

用来烧写的 JLINK 的 JTAG 定义如下:

VTref	1 ●	● 2	NC	VCC	<--> VTref
nTRST	3 ●	● 4	GND	GND	<--> GND
TDI	5 ●	● 6	GND	CPU_NRST	<--> RESET
TMS	7 ●	● 8	GND	CPU_TDI	<--> TDI
TCK	9 ●	● 10	GND	CPU_TCK	<--> TCK
RTCK	11 ●	● 12	GND	CPU_TMS	<--> TMS
TDO	13 ●	● 14	GND	CPU_TDO	<--> TDO
RESET	15 ●	● 16	GND	GND	<--> GND
DBG RQ	17 ●	● 18	GND		
5V-Supply	19 ●	● 20	GND		

步骤3 使用 J-FLASH 来烧写 JLINK

在烧写 FLASH 前,我们需要确认 JLINK 已经连接上待烧写的 JLINK.

运行 jlink.exe 如果已经连接上.会显示如下信息:

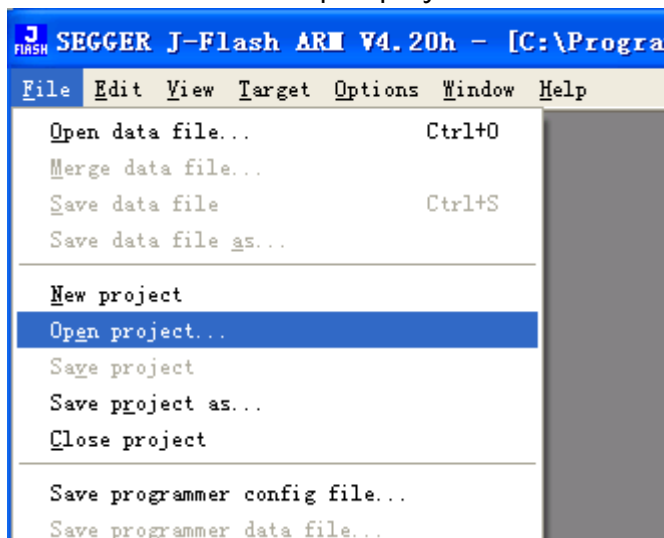
```
J-Link ARM V4.20h
SEGGER J-Link Commander V4.20h <'?' for help>
Compiled Oct  5 2010 19:11:57
DLL version V4.20h, compiled Oct  5 2010 19:11:41
Firmware: J-Link ARM V8 compiled Oct  5 2010 08:59:59
Hardware: V8.00
S/N: 20100214
Feature(s): RDI,FlashDL,FlashBP,JFlash,GDBFull
UTarget = 3.319U
Info: TotalIRLen = 4, IRPrint = 0x01
Found 1 JTAG device, Total IRLen = 4:
#0 Id: 0x3F0F0F0F, IRLen: 04, IRPrint: 0x1, ARM7TDMI Core
Found ARM with core Id 0x3F0F0F0F <ARM7>
JTAG speed: 100 kHz
J-Link>
```

如果显示没有发现芯片.则可能:

- 1.擦除不成功.
- 2.连接线没有接对或接反.

连接上芯片以后,我们就可以使用 J-FLASH 来对目标芯片编程.

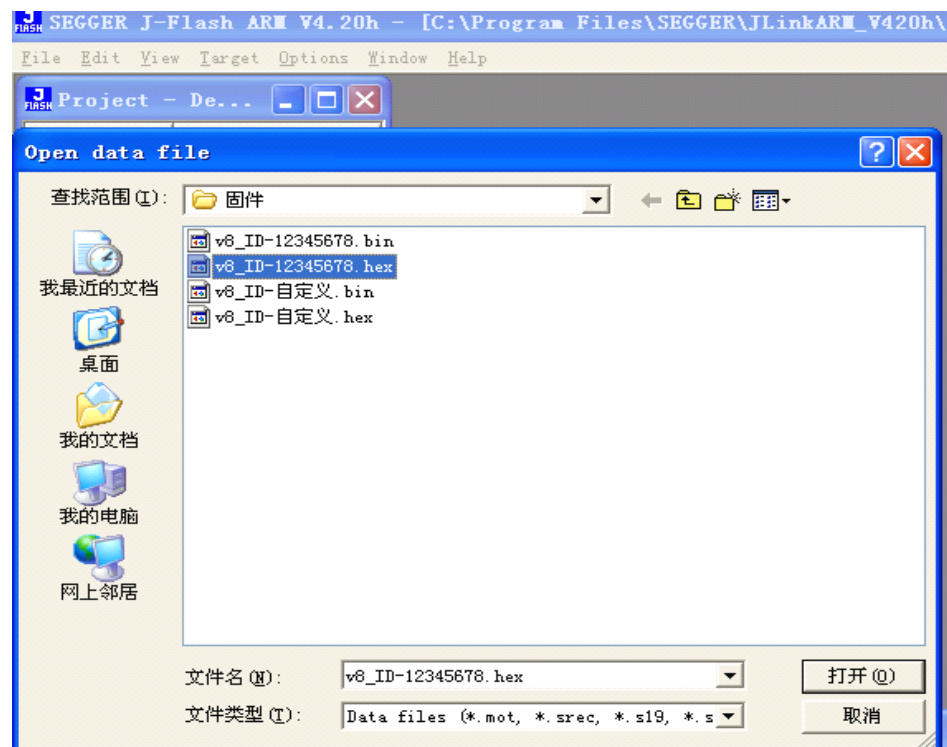
J-FLASH 是 JLINK 的 FLASH 下载工具.使用前需要对目标芯片进行设定,这里可以使用设定好的工程模块. File --> Open project



选择预先设定好的 JLINK.jflash 即可.这样,目标芯片和 FLASH 等都配置好了.

再调入我们要下载的固件文件即可: File --> open data file

选择 HEX 文件就好.如果是选择 BIN 文件,则需要设置调入地址为 0x100000



再执行 Target --> Auto 即可进行自动编程.
烧写成功后,断开 JLINK 的电源,再接上 PC 即可正常使用.

Lucking....

SEGGER J-Flash ARM V4.20h - [C:\Documents and Settings\user\桌面\jlink\jlink_firmware\JLINK.jflas...

File Edit View Target Options Window Help

Project - JLINK C:\Documents and Settings\user\桌面\jlink\jlink_firmware\固件\v8_ID...

Name	Value
Connection	USB [Device 0]
Target interface	JTAG
Init JTAG speed	5 kHz
JTAG speed	Auto recognition
TAP number	<not used>
IR len	<not used>
MCU	Atmel AT91SAM7S64
Clock speed	32000 Hz
Endian	Little
Check core Id	Yes
Core Id	0x3F0F0F0F
Use target RAM	Yes
RAM address	0x200000
RAM size	16 KB
Flash memory	AT91SAM7S64
Manufacturer	Atmel
Size	64 KB
Flash Id	0x0
Check flash Id	No
Base address	0x100000
Organization	32 bits x 1 chip

Address: 0x100000 x1 x2 x4

Address	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
100000	06	00	00	EA	FE	FF	FF	EA	FE	FF	FF	EA	FE	FF	FF	EA
100010	FE	FF	FF	EA	FE	FF	FF	EA	FE	FF	FF	EA	FE	FF	FF	EA
100020	81	09	A0	E3	D3	F0	21	E3	00	D0	A0	E1	14	00	9F	E5
100030	0F	E0	A0	E1	10	FF	2F	E1	0C	00	9F	E5	0F	E0	A0	E1
100040	10	FF	2F	E1	FE	FF	FF	EA	38	01	10	00	50	00	10	00
100050	00	00	00	EB	6A	01	00	EB	2C	00	8F	E2	00	0C	90	E8
100060	00	A0	8A	E0	00	B0	8B	E0	01	70	4A	E2	0B	00	5A	E1
100070	00	00	00	1A	62	01	00	EB	0F	00	B0	E8	18	E0	4F	E2
100080	01	00	13	E3	03	F0	47	10	13	FF	2F	E1	50	08	00	00
100090	05	30	8F	E0	13	FF	2F	E1	71	07	00	00	00	00	A0	E1
100100	FE	FF	FF	EA	1E	FF	2F	E1	3F	40	2D	E9	00	E0	A0	E1
100110	01	C0	A0	E1	02	40	A0	E1	03	50	A0	E1	01	0B	4C	E2
100120	00	00	8D	E5	04	C0	8D	E5	0F	00	9D	E8	10	D0	8D	E2

J-Flash ARM V4.20h

Target erased, programmed and verified successfully - Completed after 8.993 sec

确定

LOG

- Data file opened successfully (65536 bytes, 1 range, CRC = 0xE19C5C5E)
- Auto programming target (65536 bytes, 1 range) ...
- Connecting ...
- Connected successfully
- Erasing chip ...
- Erase operation completed successfully
- Programming target (65536 bytes, 1 range) ...
- Target programmed successfully
- Verifying CRC of affected sectors ...
- CRC of affected sectors verified successfully (CRC = 0xE19C5C5E)
- Target erased, programmed and verified successfully - Completed after 8.993 sec

Ready Connected Core Id: 0x3F0F0F0F Speed: 8000 kHz