

# Altera 的 FPGA 的 AS、PS 和 Jtag 配置模式的区别

有些人对于 FPGA 下 JTAG 的下载方式有些迷惑，为什么出现配置芯片了，为什么要用不同的下载电缆，不同的下载模式?通过在网上查阅相关资料做了总结和整理如下：

1 FPGA 器件有三类配置下载方式：主动配置方式（AS）和被动配置方式（PS）和最常用的(JTAG)配置方式。

AS 模式(active serial configuration mode):FPGA 器件每次上电时作为控制器，由 FPGA 器件引导配置操作过程，它控制着外部存储器和初始化过程，从配置器件 EPCS 主动发出读取数据信号，从而把 EPCS 的数据读入 FPGA 中，实现对 FPGA 的编程配置数据通过 DATA0 引脚送入 FPGA，配置数据被同步在 DCLK 输入上，1 个时钟周期传送 1 位数据。

PS 模式(passive serial configuration mode): 则由外部计算机或控制器控制配置过程。通过加强型配置器件（EPC16，EPC8，EPC4）等配置器件来完成，EPCS 作为控制器件，把 FPGA 当作存储器，把数据写入到 FPGA 中，实现对 FPGA 的编程。该模式可以实现对 FPGA 在线可编程。在下载配置的时候对于 Cyclone II 的器件，如 EP2C8，在 JTAG 下载方式对应.sof，AS 下载方式对应.pof。

JTAG: JTAG 是直接烧到 FPGA 里面的 由于是 SRAM 断电后要重烧，AS 是烧到 FPGA 的配置芯片里保存的 每次上电就写到 FPGA 里。

一般情况下，Cyclone II 开发板上应该有两种下载模式，AS 和 JTAG。

AS 就是下载.pof 文件到 EPCS 中。而 JTAG 就是下载.sof 文件到 FPGA 中的 RAM 中。

关于 JTAG 的原理，大家可以参考 OPEN-JTAG 开发小组写的《ARM JTAG 调试原理》。

JTAG、BDM 都比较类似，实际上是将仿真功能嵌入到芯片内部，接上比较简单的调试工具就可以进行开发了，省掉了高价的仿真器。

JTAG 调试用到了 TCK、TMS、TDI、TDO 和 TRST 这几个脚。其中 TRST 是用来对 TAP [controller](#) 进行复位的，它不是必须的。通过在 TMS 脚也可以使 TAP [controller](#) 复位。

Test Clock Input (TCK)

TCK 为 TAP 的操作提供了一个独立的、基本的时钟信号，TAP 的所有操作都是通过这个时钟信号来驱动的。TCK 在 IEEE 1149.1 标准里是强制要求的。

Test Mode Selection Input (TMS)

TMS 信号用来控制 TAP 状态机的转换。通过 TMS 信号，可以控制 TAP 在不同的状态间相互转换。TMS 信号在 TCK 的上升沿有效。TMS 在 IEEE 1149.1 标准里是强制要求的。

Test Data Input (TDI)

TDI 是数据输入的接口。所有要输入到特定寄存器的数据都是通过 TDI 接口一位一位串行输入的（由 TCK 驱动）。TDI 在 IEEE 1149.1 标准里是强制要求的。

Test Data [Output](#) (TDO)

TDO 是数据输出的接口。所有要从特定的寄存器中输出的数据都是通过 TDO 接口一位一位串行输出的（由 TCK 驱动）。TDO 在 IEEE 1149.1 标准里是强制要求的。

#### Test Reset Input (TRST)

TRST 可以用来对 TAP [Controller](#) 进行复位（初始化）。不过这个信号接口在 IEEE 1149.1 标准里是可选的，并不是强制要求的。因为通过 TMS 也可以对 TAP [Controller](#) 进行复位（初始化）。

FPGA 在正常工作时，它的配置数据存储在 SRAM 中，加电时须重新下载。在实验系统中，通常用计算机或控制器进行调试，因此可以使用 PS。在实用系统中，多数情况下必须由 FPGA 主动引导配置操作过程，这时 FPGA 将主动从外围专用存储芯片中获得配置数据，而此芯片中 FPGA 配置信息是用普通编程器将设计所得的 .pof 格式的文件烧录进去。

专用配置器件：[EPC](#) 型号的存储器

常用配置器件：EPC1，EPC2，EPC4，EPC8，EPC1441(现在好像已经被逐步淘汰了)等，对于 Cyclone II 系列器件，ALTERA 还提供了针对 AS 方式的配置器件，EPCS 系列。如 EPCS1，EPCS4 配置器件也是串行配置的。注意，他们只适用于 cyclone 系列。

除了 AS 和 PS 等单 BIT 配置外，现在的一些器件已经支持 PPS，FPS 等一些并行配置方式，提升了配置速度。当然所外挂的电路也和 PS 有一些区别。还有处理器配置比如 JRUNNER 等等，如果需要再百度吧，至少不下十种。比如 Altera 公司的配置方式主要有 Passive Serial(PS),Active Serial(AS),Fast Passive Parallel(FPP),Passive Parallel Synchronous (PPS)，Passive Parallel Asynchronous (PPA)，Passive Serial Asynchronous(PSA)，JTAG 等七种配置方式，其中 Cyclone 支持的配置方式有 PS，AS，JTAG 三种。

2 对 FPGA 芯片的配置中，可以采用 AS 模式的方法，如果采用 EPCS 的芯片，通过一条下载线进行烧写的话，那么开始的 "nCONFIG，nSTATUS" 应该上拉，要是考虑多种配置模式，可以采用跳线设计。让配置方式在跳线中切换，上拉电阻的阻值可以采用 10K。

3 在 PS 模式下 tip:如果你用电缆线配置板上的 FPGA 芯片,而这个 FPGA 芯片已经有配置芯片在板上，那你就必须隔离缆线与配置芯片的信号。一般平时调试时不会把配置芯片焊上的,这时候用缆线下载程序.只有在调试完成以后，才把程序烧在配置芯片中，然后将芯片焊上，或者配置芯片就是可以方便取下焊上的那种，这样出了问题还可以方便地调试。

在 AS 模式下 tip: 用过一块板子用的 AS 下载，配置芯片一直是焊在板子上的，原来 AS 方式在用线缆对配置芯片进行下载的时候，会自动禁止对 FPGA 的配置，而 PS 方式需要电路上隔离。

4 一般是用 JTAG 配置 epc2 和 flex10k,然后 epc2 用 PS 方式配置 flex10k.这样用比较好.（这是我在网上看到的,可以这样用吗?怀疑中）望达人告知.

5 下载电缆,Altera 下的下载电缆分为 ByteBlaster 和 ByteBlasterMV,以及 ByteBlaster II,现在还推出了基于 USB-blaster.由于 BB 基本已经很少有人使用,而 USB-Blaster 现在又过于昂贵,这里就说一下 BBII 和 BBMV 的区别.

BBII 支持多电压供电 5.5v,3.3v,2.5v,1.8v;

BBII 支持三种下载模式:AS,可对 Altera 的 As 串行配置芯片(EPCS 系列)进行编程; PS 可对 FPGA 进行配置; JTAG, 可对 FPGA, CPLD, 即 Altera 配置芯片(EPC 系列)编程; 而 BBMV 只支持 PS 和 JTAG。

6 一般在做 FPGA 实验板,(如 Cyclone 系列)的时候, 用 AS+JTAG 方式,这样可以用 JTAG 方式调试,而最后程序已经调试无误了后, 再用 AS 模式把程序烧到配置芯片里去, 而且这样有一个明显的优点, 就是在 AS 模式不能下载的时候, 可以利用 Quartus II 自带的工具生成 JTAG 模式下可以利用的.jic 文件来验证配置芯片是否已经损坏, 方法详见附件(这是骏龙的人写的,摘自坛子,如有版权问题,请包涵)。

7 Altera 的 FPGA 可以通过单片机, CPLD 等加以配置,主要原理是满足 datasheet 中的时序即可,这里我就不多说了, 有兴趣的朋友可以看看下面几篇文章, 应该就能够明白是怎么回事了。

8 配置时, Quartus 软件操作部分:

(1).assignment-->device-->device&pin options-->选择 configuration scheme, configuration mode, configuration device

注: 意在不支持远程和本地更新的机器中 configuration mode 不可选择,而 configuration device 中会根据不同的配置芯片产生 pof 文件, 如果选择自动,会选择最小密度的器件和适合设计;

(2).可以定义双口引脚在配置完毕后的作用,在刚才的 device&pin option-->dual-purpose pins-->, 可以在配置完毕后继续当 I/O 口使用;

(3).在 general 菜单下也有很多可勾选选项,默认情况下一般不做改动,具体用法参见 Altera configuration handbook,volume2,sectionII.

(4)关于不同后缀名的文件的适用范围:

.sof(SRAM Object File)当直接用 PS 模式下将配置数据下到 FPGA 里用到,USB

BLASTER,MASTERBLASTER,BBII,BBMV 适用,Quartus II 会自动生成, 所有其他的配置文件都是由 sof 生成的。

.pof(Programmer Object File)也是由 Quartus II 自动生成的,BBII 适用,AS 模式下将配置数据下到配置芯片中

.rbf(Raw Binary File)用于微处理器的二进制文件。在 PS,FPP,PPS,PPA 配置下有用处。

rpdc(Raw Programming Data File)包含 bit stream 的二进制文件,可用 AS 模式配置,只能由 pof 文件生成。

.hex(hexadecimal file)这个就不多说了,单片机里很多。

.tcf(Tabular Text File)适用于 FPP,PPS,PPA,和 bit-wide PS 配置方式。

.sbf(Serial Bit stream File)用 PS 模式配置 Flex 10k 和 Flex6000 的。

.jam(Jam File)专门用于 program, verify, blank-check。