



Dineej, Srinivas Murthy

摘要

AFE79xx 器件的启动通常使用微控制器通过 SPI 完成。部分应用不允许在系统中使用微控制器。针对此类应用，需要基于 HDL 的器件启动来配置 AFE。德州仪器 (TI) 推出集成到 FPGA 的 BRINGUP_AFE HDL 模块以及存储器（用于存储启动指令），助力配置 AFE79xx 器件。

内容

1 简介..... 1

2 使用 BRINGUP_AFE 模块的硬件设置..... 2

3 存储器与 BRINGUP_AFE 模块之间的接口..... 3

4 BRINGUP_AFE 模块中控制寄存器和状态标志的配置..... 4

 4.1 指令集..... 5

5 使用 BRINGUP_AFE 模块启动..... 7

6 支持错误解码的启动..... 8

7 总结..... 9

8 参考资料..... 10

商标

所有商标均为其各自所有者的财产。

1 简介

射频收发器用于通过无线通信介质（如电磁波）传输数据、语音或视频信息。收发器广泛部署在电信（基站）、雷达、航空航天研究、弹药等系统中。通常情况下，这些高复杂度器件使用 FPGA 或 ASIC 中的微控制器通过 SPI 进行配置。部分国防应用不允许在系统中使用微控制器。针对此类应用，可使用 FPGA（配备状态机）内部实现的 HDL 模块来配置器件，无需微控制器介入。

2 使用 BRINGUP_AFE 模块的硬件设置

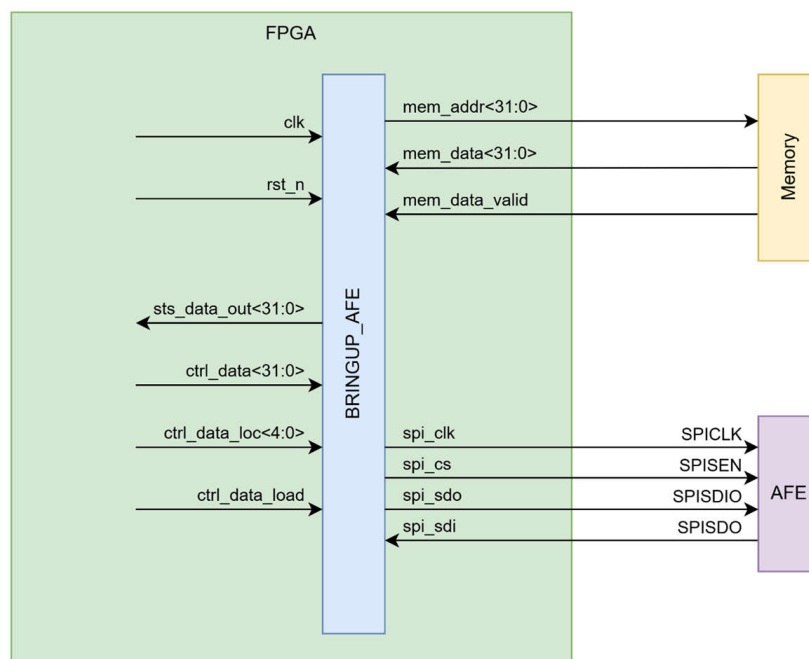


图 2-1. BRINGUP_AFE 模块的硬件连接

表 2-1. BRINGUP_AFE 模块 - 端口说明

端口	I/O	位宽度	说明
clk	输入	1	模块时钟。SPI 时钟采用了相同时钟。最大时钟速度为 10MHz
rst_n	输入	1	低电平有效复位
mem_addr	输出	32	启动期间需要从中加载指令的外部存储器地址。 内存地址从 0 开始，每执行一条指令递增 1
mem_data	输入	32	存储在 <i>mem_addr</i> 中的启动指令
mem_data_valid	输入	1	<i>mem_data</i> 端口中的指令仅在 <i>mem_data_valid</i> = 1 时执行 当 <i>mem_addr</i> 更改时，清除 <i>mem_data_valid</i> 在 <i>mem_data</i> 反映存储在 <i>mem_addr</i> 位置的指令后，置位 <i>mem_data_valid</i>
ctrl_data	输入	32	<i>ctrl_data</i> 用于配置 BRINGUP_AFE 模块寄存器
ctrl_data_loc	输入	5	需要加载 <i>ctrl_data</i> 的寄存器地址。 <i>sts_data_out</i> 反映此寄存器地址中的数据
sts_data_out	输出	32	寄存器地址 <i>ctrl_data_loc</i> 中的状态信息
ctrl_data_load	输入	1	当 <i>ctrl_data_load</i> = 1 时， <i>ctrl_data</i> 将加载到 <i>ctrl_data_loc</i> 寄存器
spi_clk	输出	1	SPI 串行接口时钟。连接到 AFE 的 SPICLK 引脚
spi_cs	输出	1	SPI 低电平有效串行数据使能。连接到 AFE 的 SPISEN 引脚
spi_sdo	输出	1	MOSI。连接到 AFE 的 SPISDIO 引脚
spi_sdi	输入	1	MISO。连接到 AFE 的 SPISDO 引脚

3 存储器与 BRINGUP_AFE 模块之间的接口

BRINGUP_AFE 模块支持设置 *mem_addr* 端口，以指示存储器可在 *mem_data* 端口上填入数据的地址。在复位 (*rst_n* = 0) 或 *start_bringup* = 0 时，*mem_addr* 设置为 0。当器件退出复位 (*rst_n* = 1) 且 *start_bringup* = 1 时，将执行启动指令。每执行一条指令后，*mem_addr* 递增 1。每当 *mem_addr* 值发生变化时，存储器就可以在一个 BRINGUP_AFE 时钟周期内将 *mem_data_valid* 端口设置为 0。在存储器将 *mem_addr* 处的数据填入到 *mem_data* 端口后，*mem_data_valid* 即可设置为 1。

备注

如果存储器中 *mem_addr* 处的数据在 BRINGUP_AFE 模块的一个时钟周期内反映到 *mem_data* 端口上，则 *mem_data_valid* 端口可始终设置为 1。

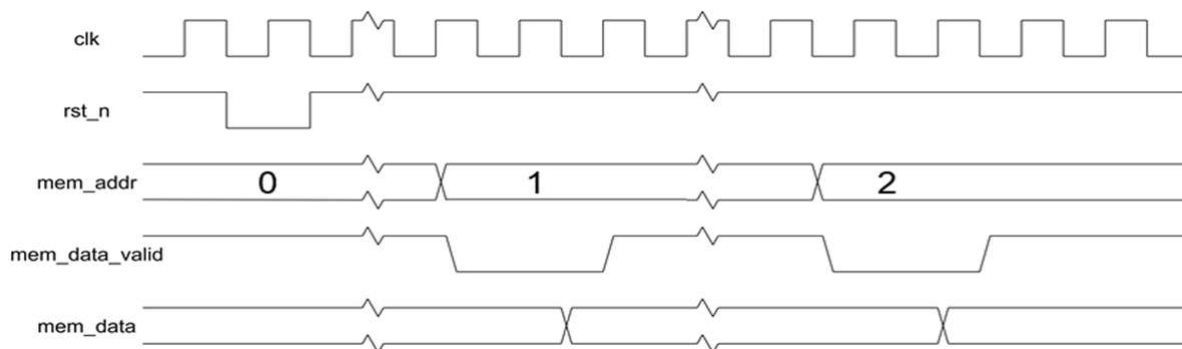


图 3-1. BRINGUP_AFE 模块和存储器之间的数据传输时序

4 BRINGUP_AFE 模块中控制寄存器和状态标志的配置

BRINGUP_AFE 模块使用 32 位寄存器来控制模块，并存储启动状态。每个 32 位寄存器均可使用 `ctrl_data_loc`、`ctrl_data`、`ctrl_data_load` 和 `sts_data_out` 端口访问。`Ctrl_data_loc` 用于选择要修改或读取的 32 位寄存器。在设置 `ctrl_data` 后，对 `ctrl_data_load` 执行 0-1-0 切换操作，即可将 `ctrl_data_loc` 的 32 位数据修改为 `ctrl_data`。`Sts_data_out` 实时反映 BRINGUP_AFE 模块中 `ctrl_data_loc` 的 32 位数据。

表 4-1. 状态和控制寄存器映射

地址	位	字段名称	访问	默认值	说明
0x0	0:0	bringup_finished	R	0x0	"1" 表示启动已完成
0x0	1:1	无效	R	0x0	无效字段。数据不相关。
0x0	2:2	bringup_err_sticky	R	0x0	如果置位，表示发生了启动错误。 粘滞标志。切换到 <code>clear_bringup_err</code> 以清除此标志
0x0	3:3	bringup_halt	R	0x0	指示 BRINGUP_AFE 是否处于停止状态。 启动完成或在 <code>stop_at_bringup_fail</code> = "1" 时发生启动错误
0x0	4:4	unknown_instruction	R	0x0	模块无法解析当前指令。 预计不会发生
0x0	5:5	module_ready_flag	R	0x0	状态标志指示模块是否准备好启动
0x0	6:6	bu_cycles_ovr	R	0x0	如果 <code>bu_cycle</code> 大于 $2^{**}32$ ，则设置溢出标志
0x0	7:7	give_sysref_now	R	0x0	置位 <code>give_sysref_now</code> 标志后，在单脉冲 <code>sysref</code> 的情况下发送 <code>sysref</code> 。 将 <code>sysref</code> 发送至 AFE 后，切换到 <code>continue_bringup</code> 。 对于连续 <code>sysref</code> ，则忽略
0x0	10:8	无效	R	0x0	无效字段。数据不相关。
0x0	11:11	data_valid	R	0x0	用于调试。 <code>Mem_data_valid</code> 端口的值
0x0	12:12	无效	R	0x0	无效字段。数据不相关。
0x0	15:13	无效	R	0x0	无效字段。数据不相关。
0x0	23:16	spi_data	R	0x0	用于调试。从器件读回当前指令的 Spi 数据
0x0	31:24	无效	R	0x0	无效字段。数据不相关。
0x1	31:0	curr_instr	R	0x0	用于调试。指示当前指令
0x2	31:0	无效	R	0x0	无效字段。数据不相关。
0x3	31:0	无效	R	0x0	无效字段。数据不相关。
0x4	31:0	bu_cycle	R	0x0	启动所需的时钟周期数。当 <code>start_bringup</code> = 0 时复位
0x5	31:0	mem_addr	R	0x0	用于调试。与 <code>mem_addr</code> 端口的值相同
0x6	31:0	mem_data	R	0x0	用于调试。与 <code>mem_data</code> 端口的值相同
0x7	31:0	BRINGUP_AFE_version	R	0x100	指示 BRINGUP_AFE 模块型号。
0x8	31:0	current_bu_err_code	R	0x0	发生任一启动错误时的错误代码。
0x10	0:0	clear_bringup_err	R/W	0x0	切换以清除 <code>bringup_err_sticky</code>
0x10	1:1	stop_at_bringup_fail	R/W	0x1	如果置位，模块会在启动错误时停止。否则忽略启动错误并继续
0x10	2:2	continue_bringup	R/W	0x0	切换到此选项，以便在模块因启动错误而停止后继续启动
0x10	4:3	RSVD	R/W	0x0	保留。请勿更改默认值。
0x10	5:5	ovr_instr	R/W	0x0	覆盖指令。是否覆盖当前指令
0x10	6:6	ovr_instr_trig	R/W	0x0	0-1-0 切换操作触发执行 <code>ovr_val_instr</code> 。仅在 <code>bringup_halt</code> 为 1 时可行。

表 4-1. 状态和控制寄存器映射 (续)

地址	位	字段名称	访问	默认值	说明
0x10	7:7	RSVD	R/W	0x0	保留。请勿更改默认值。
0x10	8:8	start_bringup	R/W	0x0	在启动期间，始终将此标志维持在高电平。 提供一个 0 --> 1 脉冲来触发启动。 在变更到 1 之前，0 可至少保持在 5 个时钟周期
0x10	31:9	RSVD	R/W	0x0	保留。请勿更改默认值。
0x11	31:0	RSVD	R/W	0x100000	保留。请勿更改默认值。
0x12	31:1	RSVD	R/W	0x0	保留。请勿更改默认值。
0x13	31:2	ovr_val_instr	R/W	0x0	覆盖当前指令的值。 仅当 <i>ovr_instr</i> = "1" 时有效

控制寄存器的更新步骤

1. 将 *ctrl_data_loc* 设置为需要加载 *ctrl_data* 的寄存器地址。
2. 设置 *ctrl_data*。
3. 将 *ctrl_data_load* 置位为 1。
4. 等待 2 个 BRINGUP_AFE 时钟周期。
5. 将 *ctrl_data_load* 置位为 0。

控制或状态寄存器的读取步骤

1. 将 *ctrl_data_loc* 设置为所需地址。
2. 从 *sts_data_out* 读取数据。

4.1 指令集

BRINGUP_AFE 模块使用 32 位指令集来配置器件。位 <31:29> 用作标识指令类型的操作码。



图 4-1. 32 位指令字段

表 4-2. 32 位指令中的操作码解释说明

Opcode<31:29>	解释说明
0	RSVD
1	AFE 寄存器读取和寄存器读取校验
2	AFE 寄存器写入和 AFE 寄存器读-改-写
3	AFE 寄存器轮询
4	等待
5	突发写入
6	继电器 - 用于子操作码
7	RSVD

操作码 6 用作中继操作码以实现其他功能。

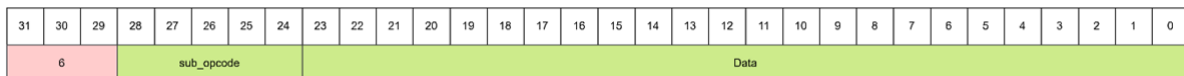


图 4-2. 子操作码映射

表 4-3. 32 位指令中的子操作码解释说明

Sub_opcode<28:24>	解释说明
0	RSVD
1	BRINGUP_AFE 模块型号校验
2	设置启动错误代码
3	立即发送 sysref - 当连续 sysref 禁用时有效
4	启动完成
5	设置 AFE 复位引脚
6-31	RSVD

启动指令的生成步骤：

1. 在 AFE79xx Latte 启动脚本中设置 logDumpInst.logFormat=0x4000。
2. 运行启动文件。
3. 在 ≈\Texas Instruments\AFE79xxLatte\lib 文件夹中会生成包含启动指令的文本文件。
4. 使用文本文件中的 32 位指令加载到存储器。

5 使用 BRINGUP_AFE 模块启动

在存储器加载器件启动指令后，可以按照以下步骤使用连续 `sysref` 进行启动。启动完成后，检查 `bringup_err_sticky` 状态确认是否存在启动错误。

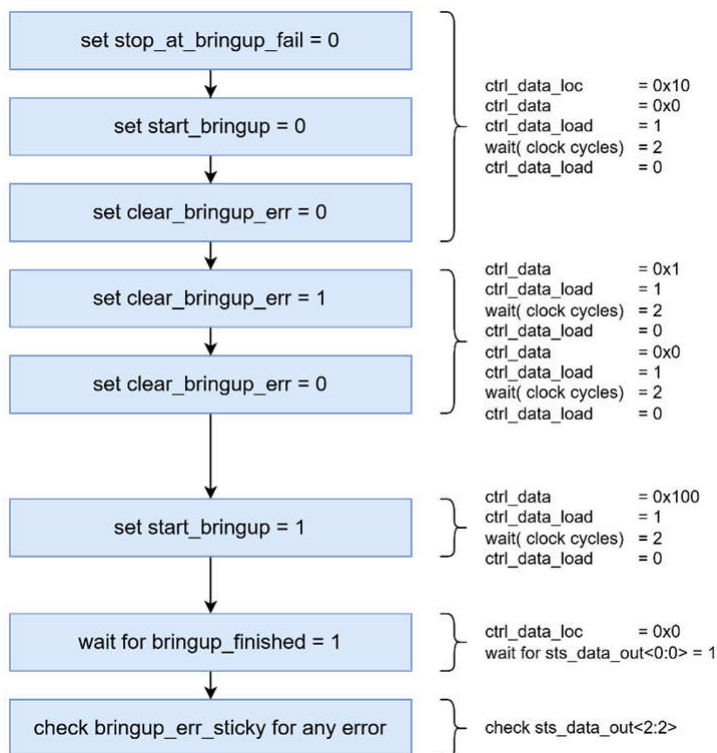


图 5-1. 使用 BRINGUP_AFE 模块启动器件

6 支持错误解码的启动

通过执行以下步骤即可在启动失败时停止器件，并可读回错误寄存器以找出可能的错误原因。

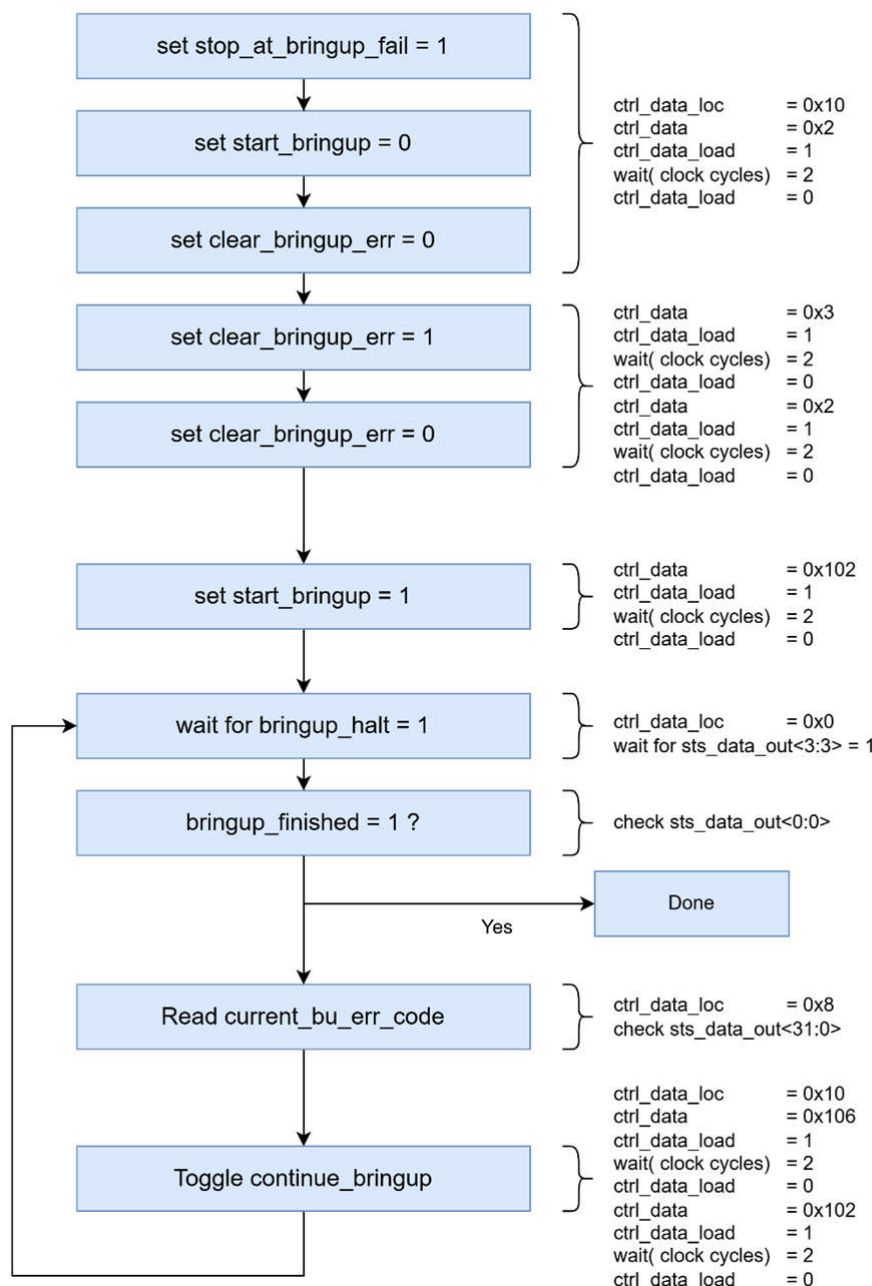


图 6-1. 支持错误解码的器件启动

7 总结

本应用手册介绍了使用集成到 **FPGA** 的 **HDL** 模块 (无需微控制器) 配置 **AFE79xx** 的过程。**BRINGUP_AFE** 模块支持多种功能, 例如支持错误报告的启动、支持错误解码的启动、使用覆盖指令的调试选项、使用单脉冲 **sysref** 的启动等。依托运行在 **10MHz** 时钟下的 **AFE79xx** 启动, 助力实现小于 **300ms** 的超快启动时间。

8 参考资料

- 德州仪器 (TI) , [AFE7900 具有 12GSPS DAC 和 3GSPS ADC 的 4T6R 射频采样 AFE](#) 数据表。
- 德州仪器 (TI) , [AFE7950 具有 12GSPS DAC 和 3GSPS ADC 的 4T6R 射频采样 AFE](#) , 数据表。

重要通知和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的相关应用。严禁以其他方式对这些资源进行复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
版权所有 © 2025，德州仪器 (TI) 公司