

Xilinx FPGA 的快速启动

作者: Joachim Meyer
博士生
卡尔斯鲁厄技术研究所
joachim.meyer@kit.edu

Juanjo Noguera
高级研究工程师
Xilinx公司
juanjo.noguera@xilinx.com

Rodney Stewart
汽车系统架构师
Xilinx公司
rodney.stewart@xilinx.com

Michael Hübner
工程博士
卡尔斯鲁厄技术研究所
michael.huebner@kit.edu

Jürgen Becker
教授, 工程博士
卡尔斯鲁厄技术研究所
becker@kit.edu

在众多当代应用中，嵌入式系统必须满足极其苛刻的时序要求。其中之一就是启动时间——即上电后电子系统进入可操作状态所需要的时间。PCI Express® 产品或汽车应用中基于CAN的电子控制单元（ECU）就是具有严格时序要求的电子系统的应用实例。

标准PCI Express®（PCIe）系统上电后仅100毫秒，系统的根组件就开始扫描总线以便弄清拓扑结构，并在这一过

程中进行初始化配置。如果PCIe设备没有准备好响应配置请求，根组件就找不到此PCIe设备，并认为它不存在。该设备将无法加入PCIe总线系统。

汽车应用中的情形与此大体相仿。在基于CAN的网络中，ECU进入睡眠模式，这时它们将停止运行并切断电源连接。只有很少部分电路仍然保持戒备状态，以便探测唤醒信号。一旦出现唤醒事件，ECU就会重新连接电源并开始引导。虽然在唤醒事件后的头一个100毫秒

内允许错过一些消息，但在此之后所有ECU都必须在网络(如CAN网络)上做好充分的运行准备。

Xilinx Automotive、Xilinx研究实验室以及德国卡尔斯鲁厄技术研究所之间合作展开了深入的研发工作，正在设法通过一种FPGA两步配置方法解决这一问题。

半导体行业的科技趋势已经能使FPGA厂家大大增加其器件中的资源。但比特流规模也在成比例地增长，配置器件需要的时间也是如此。因此，即使对

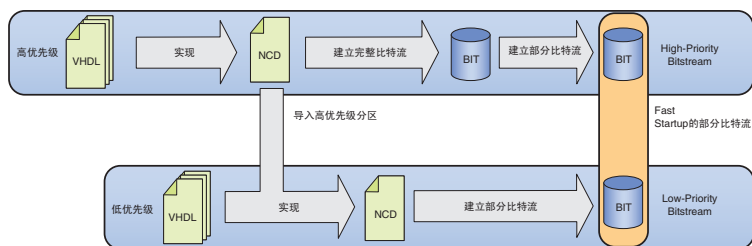


图3 Fast Startup工具流程

FPGA设计的实现。从以前的实现中导入高优先级分区，从而保证其实现方法与原来的实现方法相同。

对于Virtex-6器件而言，部分重配置（PR）流程可用于所有上述的实现。这样，就会自动得到针对低优先级分区的部分比特流。由于Spartan-6器件系列不支持PR流程，在实现针对Spartan-6设计的Fast Startup时，我们使用了针对差异化的部分重配置的BitGen选项，以获得低优先级分区的部分比特流。图3

给出了该工具流程的高层概览。

实验与结果

为了在硬件中验证Fast Startup配置方法，我们的研究小组在一块Virtex-6 ML605板和一块Spartan-6 SP605板上实现了这种方法。

Virtex-6实现的应用背景源自视频领域。用户接通视频系统电源时，他们总是希望立刻就看到系统有所响应，而不用等待数秒。因此，在图4所示的系统中，一个配备了TFT控制器的高优先

级子系统可以迅速点亮TFT屏幕。对于其他低优先级应用，即第二个设计提供了对以太网内核、UART和硬件计时器的控制和访问。

对于这种演示来说，我们使用带BPI的外置闪存作为配置接口。一旦初始高优先级比特流配置完该处理器子系统，在BRAM外运行的软件将初始化TFT控制器，并将数据写入DDR内存中的帧缓冲器。这样就能确保启动时屏幕迅速显示在TFT上面。之后，从BPI闪存中读出第二个比特流，并配置低优先级分区，这样处理器子系统可以运行其他应用程序，如Web服务器。

为方便扩展和清晰隔离两个分区，我们使用了AXI至AXI桥接器。这也在最大程度上减少了穿过两个设计分区边界的网络。低优先级分区与高优先级分区共享系统时钟。

表1表示FPGA资源利用情况，表2表示传统启动方法、仅有高优先级分区

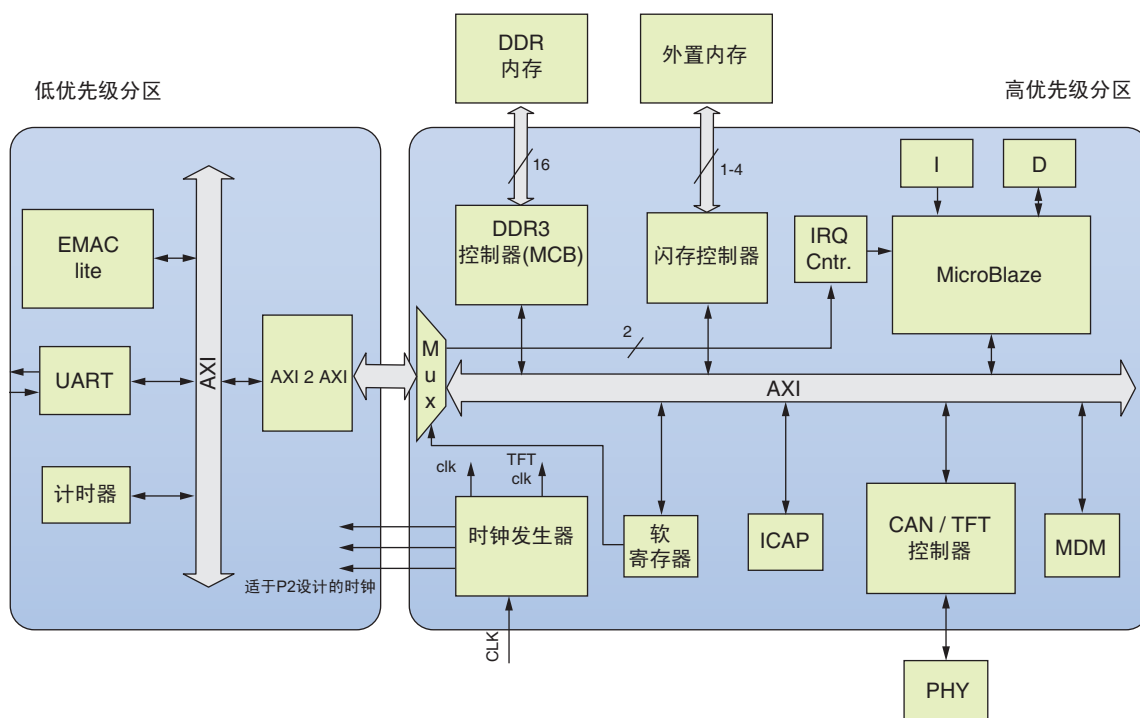


图4 Virtex-6和Spartan-6演示（Virtex-6包括TFT模块，Spartan-6仅包括CAN模块）的基本框图

压缩比特流的启动方法和Fast Startup配置方法的配置时间。每种方法都使用BPIx16配置接口，而采用的配置速率（这个选项决定了目标配置时钟频率）为2 MHz和10 MHz。我们使用一台示波器测量了该数据，捕获了FPGA的“init”和“done”信号。

表2中“压缩的”一栏表示仅有高优先级分区的压缩比特流。含有两个分区的完整FPGA设计的压缩比特流将达到3.1 Mbyte。

SPARTAN-6汽车ECU设计

为了验证针对Spartan-6的Fast Startup方法，我们选择了汽车领域的ECU应用情形。每当您在汽车电子控制单元中看到一个FPGA器件时，它一般都由ECU的主应用处理单元使用（见图5）。我们的目标是实现一种将系统处理器放入FPGA中的设计。这样我们就能避免对外置处理器的需要，从而降低整个系统的成本、复杂性、空间和功耗。

系统分区

对于这种情形，系统分区是显而易见的。我们把我们的ECU设计分成为高优先级分区的系统处理器部分和作为低优先级分区的应用处理部分。

资源类型	分区			
	高优先级	%	低优先级	%
触发器	8,849	2.9	1,968	0.7
查找表	7,039	4.7	2,197	1.5
I/O	135	22.5	20	3.3
RAMB36s	34	8.2	2	0.5

表1 占用的FPGA资源(针对XC6VLX240T)

XC6VLX240T	配置方法		
配置接口	传统的 8.9 MB	压缩的 2.0 MB	Fast Startup 1.4 MB
BPIx16 CR2	1,740 ms	389 ms	278 ms
BPIx16 CR10	450 ms	112 ms	84.4 ms

表2 测得的配置时间(Virtex-6视频设计)

这种设计与Virtex-6设计存在很多相似之处，但不同的是，我们用SPI取代BPI作为外置闪存的接口，因此必须用CAN控制器取代TFT控制器。上电后，系统控制器只有有限的时间引导并准备好处理第一个通信数据。由于ECU使用CAN总线用于通信，这个引导时间一般限制为100毫秒。按照传统配置方法，使用带低成本配置接口（如SPI或Quad-SPI）的大型Spartan-6很难达到如此严格的时序要求。而使用速度更快、更加昂贵的配置接口在汽车领域中是无法接受的。

测量装置

对于SP605汽车ECU演示来说，我们在实验室中进行了测量，图6给出了测量装置。图中左侧是一个基于Spartan-3的X1500汽车平台，它实现了针对CAN总线的网络发包器，该发生器能够收发CAN消息并使用硬件计时器测量CAN消息之间的时间。右侧为目标平台，它没有直接与CAN总线相连，而是使用来自额外的定制电路板的CAN收发器。除了提供一个CAN PHY之外，这个定制电路板还要控制目标电路板的电源。

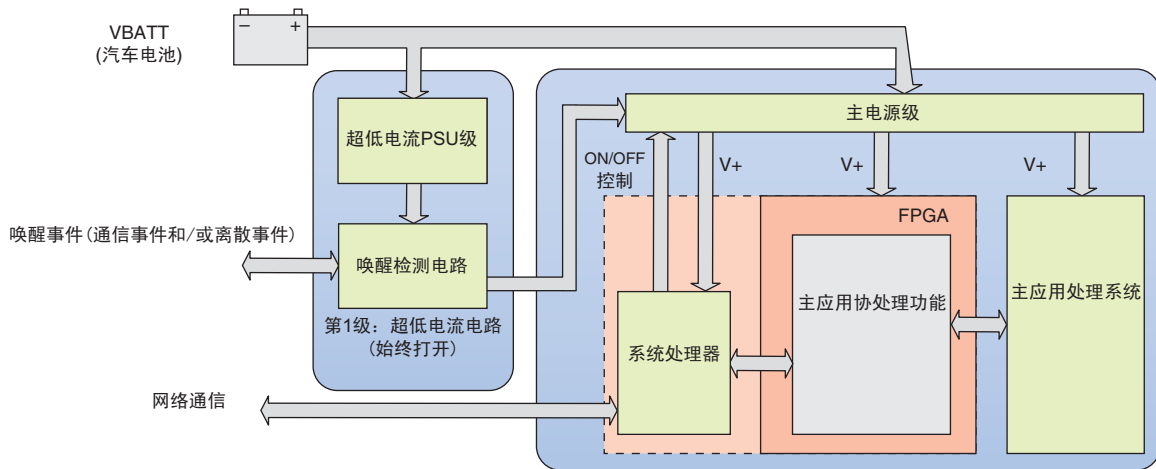


图5 现代汽车ECU中的FPGA应用，FPGA中集成了处理器(虚线)

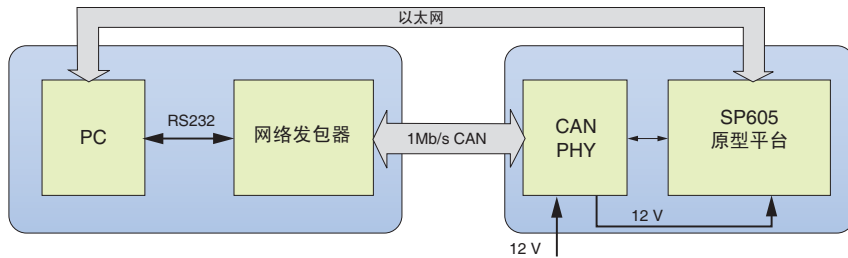


图6 汽车ECU的测量装置

测量配置时间的程序从网络发包器处于空转（空档）状态下开始，此时CAN PHY电路板上的CAN收发器也处于睡眠模式，SP605也与电源断开。接着，网络发包器启动一个硬件计时器并发送一条CAN消息。识别CAN总线上的事件后，CAN PHY被唤醒并重新让SP605接通电源。然后FPGA开始载入来自SPI闪存的初始比特流。

由于没有接收器确认由网络发包器发送的消息，这条消息立刻重复发送，直至FPGA已经完成其配置并用有效的波特率配置了CAN内核。一旦Spartan-6设计的CAN内核确认了这条消息，网络发包器的CAN内核将触发一个中断，由这个中断停止硬件计时器。该计时器现在保存着SP605设计的引导时间。测量结果包括SP605设计内一个额外的硬件计时器，此结果表明，当执行该软件来配置内置BRAM内存的CAN内核时，软件启动时间可以忽略不计。

表3表示每个分区的FPGA资源消耗。百分比信息是用来表示使用的XC6S45LXT器件的有效资源的总数量。

表4表示配置时间的测量结果。对于这些结果，我们实现并比较了完整设计的一个标准比特流和一个压缩比特流以及使用部分初始比特流的Fast Startup方法。该表列出了针对不同SPI总线带宽和不同配置速率（CR）设置的配置时间。不出所料，配置时间与比特流大小成正比。由于使用快速配置，时

钟没有影响清仓过程（housecleaning process），高CR设置的比率（按百分比）发生了变化。

资源类型	分区			
	高优先级	%	低优先级	%
触发器	3,480	6%	1,941	4%
查找表	3,507	13%	1,843	7%
I/O	58	20%	20	7%
RAMB	12	10%	2	2%

表3 Spartan-6设计中占用的FPGA资源

配置接口	配置方法		
	传统的1,450 KB	压缩的	低优先级
SPIx1 CR2	5,297 ms	3,382 ms	1,157 ms
SPIx1 CR26	292 ms	196 ms	85 ms
SPIx2 CR2	2,671 ms	1,699 ms	596 ms
SPIx2 CR26	161 ms	113 ms	58 ms
SPIx4 CR2	1,348 ms	872 ms	311 ms
SPIx4 CR26	97 ms	73 ms	45 ms

表4 测得的Spartan-6配置时间

在硬件中验证

我们开发的高级配置方法可以称为优先化的FPGA启动，因为它分两步配置器件。这种方法不仅对于解决现代FPGA中配置时间不断增加的挑战是必不可少，而且也能在很多现代应用中得到使用，如PCI Express或基于CAN的汽车系统。

除了提出高优先级初始配置方法，我们还在硬件中对这种方法进行了验证。我们使用并测试了针对Fast Startup

的工具流程和方法，以在Spartan-6评估板（SP605）上实现基于CAN的汽车ECU，以及在Virtex-6原型板上实现视频设计。通过使用这种新颖的方法，我们减小了初始比特流大小，从而使配置时间改进了84%（与标准完整配置方案相比）。

Xilinx将在针对7系列FPGA的软件中支持针对PCI Express应用的Fast Startup概念，并通过优化的实现方法简化其使用。在7系列中，新的两步比特

流方法是实现起来最简单最低成本的方法。设计FPGA时，用户可以通过一个简单的软件开关实现两级比特流。该比特流的第一级仅包含配置时序关键模块需要的配置帧。配置时，会产生一个FPGA STARTUP序列，关键模块变成活动模块，这样就可轻易满足100毫秒时序要求。当时序关键模块运行时（例如PCI Express枚举/配置系统过程正在进行），剩下的FPGA配置得以加载。两级比特流方法能够使用便宜的闪存器件存储比特流。🌈