

基于 FPGA 的电力电子变换器的精仿真研究

于 飞, 刘彦志

(青岛科技大学 自动化与电子工程学院, 山东 青岛 266042)

摘要: 在电力电子仿真领域, 基于现场可编程门阵列 (FPGA) 高速并行运算的特点, 为了达到实时效果, 对系统模型通常采用较为方便的建模; 提出了使用 FPGA 对 Boost 变换器进行复杂建模以及算法研究; 基于这一思想, 首先对 boost 变换器进行建模以及算法研究; 其次完成基于 FPGA 的 Boost 变换器的建模并利用 XILINX 的 ML605 开发板进行仿真实验, 将仿真波形与 MATLAB 进行对比, 证实了在一定误差范围内, 基于 FPGA 复杂建模仿真的有效性及准确性。

关键词: Boost 变换器; FPGA; 数学模型; 龙格库塔

FPGA-based Simulation of Power Electronic Converter

Yu Fei, Liu Yanzhi

(College of Automation & Electronic Engineering, Qingdao University of Science & Technology, Qingdao 266042, China)

Abstract: In the field of power electronics simulation based on field programmable gate array (FPGA), the system model is usually modeled more convenient in order to achieve real-time simulation using FPGA's characteristics of high-speed parallel computing. In this paper, establish complex model and algorithm research of boost converter based on FPGA. Based on this idea, the boost converter is modeled and studied firstly. Secondly, the boost converter is modeled based on XILINX ML605 Evaluation Kit and simulated and then the simulation waveform is compared with the waveform based on MATLAB. Form the simulation result, it can be seen the efficiency and accuracy of complex modeling and simulation of the boost converter based on FPGA.

Keywords: boost converter; FPGA; mathematical model; Runge-Kutta

0 引言

在电力系统仿真领域, 随着新一代的电网以及输配电系统发展, 大量电力电子器件的引入以及更高的频率, 对模型仿真计算性能的要求越来越高, 并行计算是目前解决单处理器速度瓶颈的最好方法之一^[1]。当前主流的实时仿真系统中, 如 RTDS、Hypersim、dSPACE 都采用多块 CPU 或 DSP 并行方式运行, 但其并行度有限; 而 FPGA 具有高度并行性特性以及可重构计算方式, 特别适合用来对具有高度并行特点的电力电子模型进行仿真。对于复杂的系统模型, 虽然采样率高, 但只要是并行运算, 模型的复杂程度并不会影响速度^[2]。

随着电力电子模块化结构以及分布式电源接入配电网的不断发展, FPGA 凭借其高速并行处理能力、流水型架构和内存分布性, 使其越来越多的应用于电力电子分布式控制、模块化电力电子设备的建模仿真中^[3-6]。文献 [7] 提出了利用 FPGA 完成 PLC 体系复杂算法并行处理的方法, 将 PLC 的运算速度提高到了 50 ns 级别。

解析法是基于变换器特性的建模方法, 而状态空间平均法则是解析法中最具有代表性的方法之一。为了提高仿真的精度以及快速分析的能力。文献 [8] 使用全局隐式梯形积分法, 利用后向欧拉法对开关动作点进行线性插值计算, 以此来提高精度和计算速度。文献 [9] 采用了一种 2 阶隐式龙格库塔法建立等值模型, 并在此数值积分方法上运用了开关自校正插值

的方法。龙格库塔是求解常微分方程的数值方法, 虽然 4 阶龙格库塔法具有较高的精度, 但是对步长有限制, 使用 4 阶龙格库塔法必须保证步长在绝对稳定域之内, 因此对高频开关器件电路, 只要保证步长满足要求, 使用 4 阶龙格库塔法也可以获得较高精度的仿真波形。

1 Boost 变换器的数学建模

Boost 变换器电路原理如图 1 所示, Boost 变换器的工作模式分为两种: 电感电流连续工作模式和电感电流断续工作模式。

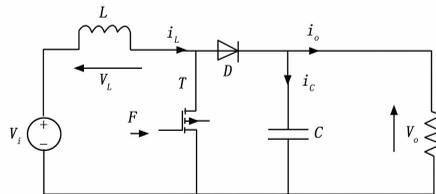


图 1 Boost 变换器的电路原理图

图 2 所示的电路工作状态为理想状态下的电路变化, 即开关管和二极管导通状态电阻为零, 开关管和二极管关断状态电阻无限大。其中图 2 (a) 和图 2 (b) 所示为电流连续工作模式的电路工作状态, 图 2 (a)、(b)、(c) 所示为电流断续工作模式的电路工作状态, 电流断续工作模式是由于采用小电感, 导致多出一个电感为零的工作状态。本文主要对电流连续工作模式进行相关建模以及仿真研究。

为了简单直观地反映对 Boost 变换器建模不同所导致的仿真波形的差异, 采用最常用的开环控制策略对开关管进行控制, 开关频率通过控制电压 v_m 与三角波 v_c 进行比较得来, 开

收稿日期: 2017-01-19; 修回日期: 2017-02-27。

作者简介: 于 飞 (1961-), 男, 河南郑州人, 教授, 研究生导师, 主要从事电力电子技术、智能控制方向的研究。

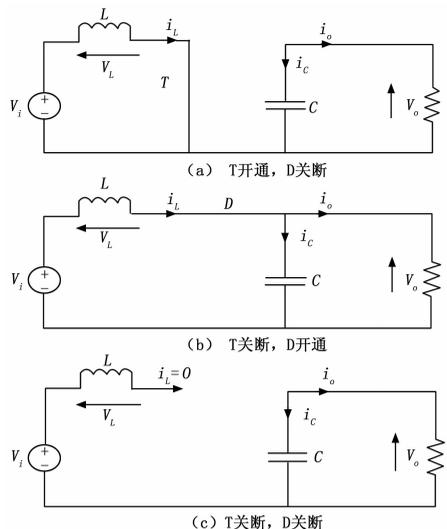


图 2 T 开通, D 关断

关管的控制 F 由 $v_m - v_t$ 决定。

由图 2 所示 Boost 变换器电路工作状态的变化, 可以对 Boost 变换器建立理想数学模型, 列出电感电压 $v_L(t)$ 和电容电流 $i_c(t)$ 的表达式:

$$\begin{aligned} v_L(t) &= v_t(t) * F + (v_t(t) - v_0(t) * F * \text{sign}(i_L)) \\ i_c(t) &= -i_0(t) * F + i_L(t) * F * \text{sign}(i_L) \end{aligned} \quad (2)$$

式中, F 是一个逻辑变量, 当时 $v_m \geq v_t$, $F=1$, 当时 $v_m < v_t$, $F=0$, 表示开关管的开通与关断, $\text{sign}(i_L)$ 是一个逻辑变量, 当 $i_L(t)$ 为正时, $\text{sign}(i_L) = 1$; 当 $i_L(t)$ 为 0 时, $\text{sign}(i_L) = 0$ 。

对 Boost 变换器进行状态空间表达式建模, 状态空间表达式描述如下:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} u \end{bmatrix} \quad (3)$$

利用串联电感的等效电流表达式 $v(t) = L \frac{di(t)}{dt}$ 和并联电容的等效电压源表达式 $i(t) = C \frac{dv(t)}{dt}$, 采用龙格库塔法将状态空间方程组转换为离散状态空间方程, 4 阶龙格库塔公式如下:

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(k_1 + 2 * K_2 * + K_3 + K_4) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2} * K_1) \\ K_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2} * K_2) \\ K_4 = f(x_n + h, y_n + h * K_3) \end{cases} \quad (4)$$

要保证 4 阶龙格库塔法能稳定, 需要对其绝对稳定域进行分析计算, 假设参考模型为:

$$\frac{di(t)}{dt} = \frac{1}{L}(v + i * R) \quad (5)$$

令 $\rho = \frac{R}{L}$, 则其稳定域由 $|1 + h * \rho + \frac{(h * \rho)^2}{2!} + \frac{(h * \rho)^3}{3!} + \frac{(h * \rho)^4}{4!}| < 1$ 得到。

工作状态 1 时, 对式 (1) 和 (2) 使用 4 阶龙格库塔离散化状态空间方程, 如下:

$$\begin{cases} v_L(t) = v_t(t) \\ K_1 = \frac{1}{L} * (v_L(t)) \\ K_2 = \frac{1}{L} * (v_L(t) + \frac{h}{2}) \\ K_3 = \frac{1}{L} * (v_L(t) + \frac{h}{2}) \\ K_4 = \frac{1}{L} * (v_L(t) + h) \\ i_L(t+1) = i_L(t) + \frac{h}{6} * (K_1 + 2 * K_2 + 2 * K_3 + K_4) \end{cases} \quad (6)$$

$$\begin{cases} i_c(t) = -\frac{v_0(t)}{C} \\ K_1 = \frac{1}{C} * (v_c(t)) \\ K_2 = \frac{1}{C} * (v_c(t) + \frac{h}{2}) \\ K_3 = \frac{1}{C} * (v_c(t) + \frac{h}{2}) \\ K_4 = \frac{1}{C} * (v_c(t) + h) \\ v_0(t+1) = v_0(t) + \frac{h}{6} * (K_1 + 2 * K_2 + 2 * K_3 + K_4) \end{cases} \quad (7)$$

同理, 工作状态 2 时, 离散化 4 阶龙格库塔理想状态空间方程如下:

$$\begin{cases} i_c(t) = i_L(t) - \frac{v_0(t)}{C} \\ K_1 = \frac{1}{C} * i_c(t) \\ K_2 = \frac{1}{C} * (i_c(t) + \frac{h}{2}) \\ K_3 = \frac{1}{C} * (i_c(t) + \frac{h}{2}) \\ K_4 = \frac{1}{C} * (i_c(t) + h) \\ v_0(t+1) = v_0(t) + \frac{h}{6} * (K_1 + 2 * K_2 + 2 * K_3 + K_4) \\ v_L(t) = v_0(t) - v_0(t) \\ K_1 = \frac{1}{L} * v_L(t) \\ K_2 = \frac{1}{L} * (v_L(t) + \frac{h}{2}) \\ K_3 = \frac{1}{L} * (v_L(t) + \frac{h}{2}) \\ K_4 = \frac{1}{L} * (v_L(t) + h) \\ i_L(t+1) = i_L(t) + \frac{h}{6} * (K_1 + 2 * K_2 + 2 * K_3 + K_4) \end{cases} \quad (8)$$

为了建立更准确的数学模型, 本文将开关器件用二值法等效, 即将开关开通状态和关断状态分别等效成小电阻和大电阻, 然后利用节点法分析电路, 获得高维度的导纳矩阵方程。在对电感和电容的离散化过程中, 为了使结果更具有对比性, 同样使用 4 阶龙格库塔法进行数值计算。

2 FPGA 仿真系统设计

2.1 FPGA 仿真环境和计算时序

FPGA 与传统的 CPU 仿真相比, 具有并行计算优势和更高的设计灵活性, 满足高速精仿真需求。传统的 FPGA 代码编写流程十分繁杂, 工作量大, 而且需要对具体的 FPGA 芯片有较深入的了解, System Generator 的出现极大的降低了 FPGA 设计开发难度, 即使对硬件不算了解, 也能进行程序设计。本文利用 System Generator 进行 Boost 变换器的模型搭建, System Generator 是 MATLAB/Simulink 环境下一款理想的 FPGA 开发软件, 也可以认为是 Simulink 中的工具包, 可以自动的将设计模型转化为硬件实现, 而无需掌握复杂的 VHDL 或 Verilog 语言。FPGA 作为一种并行计算设备, 可以快速有效地进行数据计算处理, 为高速电力电子仿真提供了合适的环境。FPGA 通过将串行计算中的并行计算变为并行计算, 将极大地减少运算的时间。假设式 (6)、(7) 中每一个式子都只有一步运算, 如果采用串行计算, 将进行 5 步运算, 而并行运算只需要 2 步, 即在运算中 K_1 、 K_2 、 K_3 、 K_4 可以同步计算。

变换器模型的并行计算方法步骤如下图 3 所示。由于变换器模型较为简单, 解算规模较为适中, 本文以控制器模型的开关周期为一个控制模型步长, 以变换器模型 $\Delta t = h$ 为变换器模型步长, 仅考虑控制模型为变换器模型解算步长整数倍的情况。

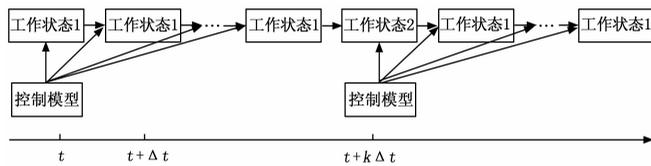


图 3 变换器模型求解计算时序

图 3 可知给出了控制模型解算步长为变换器模型解算步长倍的时序, 即控制模型计算一次时, 变换器模型计算次。变换器模型由时刻积分到时刻的一个步长中, 采用时刻的值作为工作状态 1 的输入, 当变换器模型完成一个解算步长之后, 需要将上一个状态的输出变量用作此次积分计算的输入值。每个工作状态里开关的等效值会根据状态的不同进行改变, 在到时刻的积分计算过程中, 一个工作状态的求解结果会更新到另一个工作状态中, 依次进行计算求解。在一个开关周期内即在第一个控制模型步长内, 根据工作状态 1 中的式 (6)、(7) 解算出电路的各个变量状态值以及等效值等输出值, 作为输入值更新到工作状态 2 中; 在第二个控制模型步长中, 根据式 (8)、(9) 对变换器模型重新进行积分计算, 完成一个工作周期。

2.2 基于 FPGA 的模型设计

首先根据变换器原理图建立器件模型, 然后利用 MATLAB 对建立的数学模型进行编程仿真, 最后在 MATLAB 的基础上, 使用 System Generator 对变换器模型进行搭建, 模型的搭建过程按照模块化搭建方式分步进行, 对可重复利用的子模块进行搭建, 将可以减少搭建的复杂程度、节约时间, 同时理解方便增加了设计的可读性。

2.2.1 PWM 控制模型

由于对 Boost 变换器采用的是简单的开环控制,

PWM 控制信号为固定占空比的方波, 由参考电压与一个固定频率与峰值的三角波相比较实现, 其中参考电压, 占空比为。图 4 为 PWM 控制模型, 使用两个 counter 计数器相差 2 倍的关系来产生三角波, 再通过三角波与参考电压对比得到控制信号, 当参考电压大于三角波时输出高电平, 反之输出低电平。系统时钟为 10 MHz, PWM 频率为 50 kHz, 计数器的计数值分别设为 100 和 200。

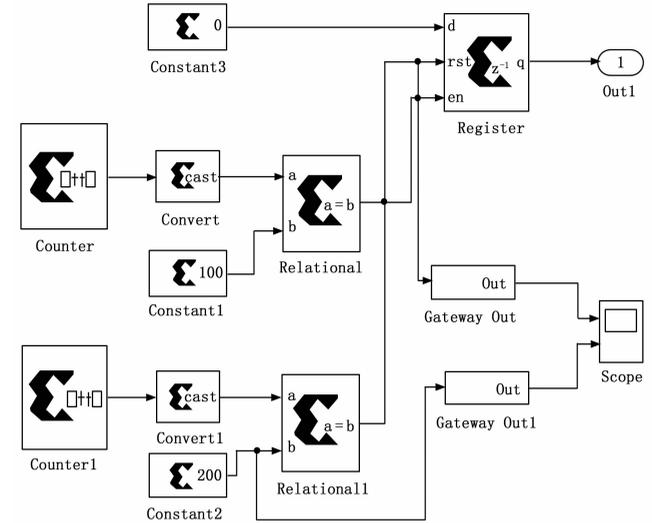


图 4 PWM 模型

2.2.2 4 阶龙格库塔模型

按照建立的 Boost 变换器的数学模型, 用 XILINXBlockset 库中的模块进行模型的搭建设计。如图 5 所示, 为搭建的 4 阶龙格库塔模型。在进行整体模型设计的过程中, 可以将 4 阶龙格库塔模型作为一个子模块, 重复使用子模块以减少工作的重复性。从图 5 所示的模型的搭建过程中可以看出, 该模型是按照并行的方式进行搭建的。

2.2.3 缓冲寄存器模块

并行计算的过程中, 同步进行的求解计算会因为式子的复杂度而导致时间上的不一致, 这时候就需要缓冲寄存器模块进行时间上的同步, 以保证程序的正确执行与结果的正确性。例如本文中 (8)、(9) 中的 $i_c(t) = i_L(t) - \frac{v_0(t)}{C}$ 、 $v_L(t) = v_i(t) - v_0(t)$, 由于两个式子的计算所用时间不同, 将导

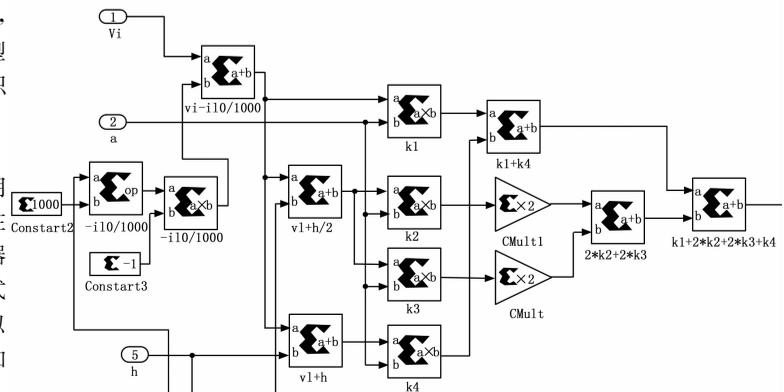


图 5 4 阶龙格库塔模型

致后面的计算时间不同，因此在式 (8)、(9) 完成计算的时候，需要对其进行不同时间的延迟补偿等待下一次积分运算，否则将会产生错误的结果。式 (8) 与式 (9) 详细的并行运算步骤如下：

式 (8) 的并行运算步骤：

- 1) 同步计算 $\frac{v_0(t)}{C}$ 、 $\frac{1}{C}$ 、 $\frac{2}{h}$ 、 $\frac{h}{6}$ 计算；
- 2) 计算 $i_c(t)$ 的值；
- 3) 计算 K_1 、 $i_c(t) + \frac{h}{2}$ 、 $i_c(t) + h$ 的值；
- 4) 计算 K_2 、 K_3 、 K_4 的值；
- 5) 计算 $2 * K_2$ 、 $2 * K_3$ 的值；
- 6) 计算 $K_1 + 2 * K_2 + 2 * K_3 + K_4$ 的值；
- 7) 计算 $\frac{h}{6} * (K_1 + 2 * K_2 + 2 * K_3 + K_4)$

的值；

- 8) 计算 $v_0(t+1)$ 的值。

式 (9) 的并行运算步骤：

- 1) 同步计算 $v_L(t)$ 、 $\frac{1}{C}$ 、 $\frac{h}{2}$ 、 $\frac{h}{6}$ 计算；
- 2) 计算 K_1 、 $v_L(t) + \frac{h}{2}$ 、 $v_L(t) + h$ 的值；
- 3) 计算 K_2 、 K_3 、 K_4 的值；
- 4) 计算 $2 * K_2$ 、 $2 * K_3$ 的值；
- 5) 计算 $K_1 + 2 * K_2 + 2 * K_3 + K_4$ 的值；
- 6) 计算 $\frac{h}{6} * (K_1 + 2 * K_2 + 2 * K_3 + K_4)$ 的值；
- 7) 计算 $v_0(t+1)$ 的值。

通过对比式 (8) 与式 (9) 详细的并行运算步骤可以看出，

$i_c(t) = i_L(t) - \frac{v_0(t)}{C}$ 、 $v_L(t) = v_i(t) - v_0(t)$ 这两个式子复杂度的不同导致了式 (8) 比式 (9) 多了一个时钟周期的运算，因此在每次运算之后，需要进行一定的时间补偿，经过补偿后的时间与变换器模型仿真步长一致。

将建好的变换器模型和控制器模型通过 JTAG 下载到 FPGA 中，本文利用 XILINX 的 Virtex-6FPGAML605 进行实物仿真。

3 仿真结果

在对 Boost 变换器数学模型研究的基础上，本文对 Boost 变换器理想状态空间模型、二值状态空间模型和 PWM 控制器模型进行建模仿真，其中控制器以及 Boost 变换器建模仿真中各个参数设置如表 1 所示。

表 1 仿真参数列表

仿真参数	参数值
PWM 控制频率/kHz	50
参考电压 Vm/V	0
输入电压/V	12
电感 L/H	2e-4
电容 C/F	5e-5
电阻 R/Ω	5
仿真时间/s	7e-3
开关管导通电阻 R/Ω	1e-5
开关管关断电阻 R/Ω	1e5

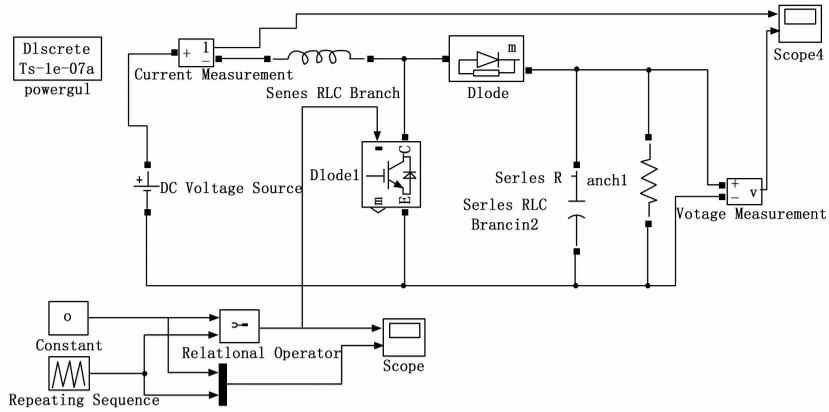


图 6 Boost 变换器器件模型及控制器模型

本文为了验证 FPGA 对复杂模型仿真的精度与仿真的可行性，并适当地提高仿真速度，因此分别对 Boost 变换器和控制器进行建模，对控制模型设置仿真步长为 $10 \mu s$ ，对器件模型仿真和复杂模型仿真设置仿真步长为 $200 ns$ ，仿真时间均为 $7 ms$ ，Simulink 系统周期与 FPGA 系统时间周期一致为 $10 ns$ 。图 6 为 Boost 变换器器件模型，图 7 为 Boost 变换器复杂模型与控制器模型。

进行仿真分析时，需要将图 7 所建立的模型通过 JTAG 下载到 FPGA 中，得到输出电压波形，将仿真波形与按照图 6 进行的 MATLAB 仿真产生的波形进行对比。图 8~10 分别为 Matlab/Simulink 器件模型的仿真、基于 FPGA 的理想以及复杂模型。

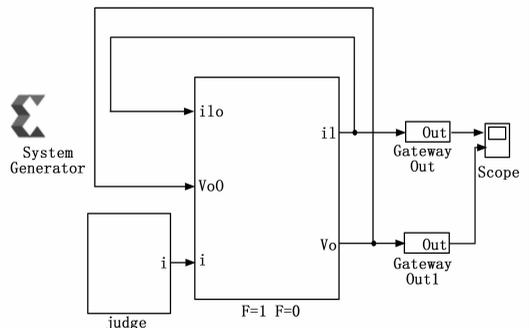


图 7 Boost 变换器模型及控制器模型

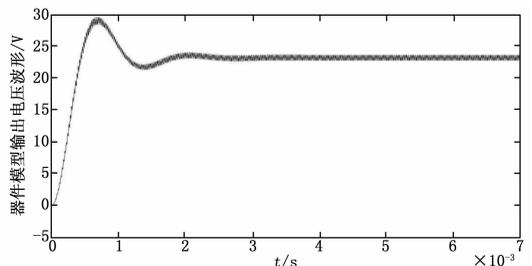


图 8 Boost 变换器器件模型输出电压波形

图 8 与图 9 对比可以看出，理想状态空间模型与器件模型相比，虽然波形响应基本一致，但是电压的幅值存在着一定的误差；图 8 和图 10 对比可以看出，随着建模精确度的提高，基于二值法的模型与器件模型相比，波形基本一致，验证了基

(下转第 154 页)