

文章编号: 1005-0523(2009)04-0087-04

基于 Matlab/RTW 的单相桥式逆变器 SPWM 目标代码自动生成研究

胡文华^{1,2}, 韦宝泉²

(1. 华中科技大学 电气与电子工程学院, 湖北 武汉 430074; 2. 华东交通大学 电气与电子工程学院, 江西 南昌 330013)

摘要:提出了一种基于 Matlab/RTW 的系统设计方法和开发流程, 结合此集成一体化的控制平台, 提出了一种基于嵌入式目标代码生成工具的单相桥式逆变器正弦脉宽调制波产生方法。利用 Matlab/Simulink 工具建立算法模型, 仿真验证算法后, 给出了以 TMS320F2812 定点 DSP 为目标的 SPWM 控制模型, 并自动生成代码, 编译、下载到目标 DSP 中运行, 产生波形与理论吻合。这种基于模型的设计流程, 实现了工程开发过程从算法设计到最终实现的所有阶段, 从而提高了产品开发效率, 缩短了开发时间, 降低了开发成本。

关键词:TMS320F2812 DSP; Matlab; 实时代码生成工具; 正弦脉宽调制

中图分类号:TM921.5

文献标识码:A

传统的嵌入式系统开发方法将系统设计与代码实现分开, 开发周期长, 已经不能满足市场的要求。本文结合基于 TMS320F2812DSP 的单相桥式逆变器的开发, 利用 Matlab/RTW 工具集将 Simulink 仿真模型自动转换为嵌入式代码并自动编译下载到微控制器中, 从而实现系统设计和代码实现的统一, 可缩短开发周期, 降低开发成本^[1]。

文献[2]利用 RTW, 采用五段法 SVPWM, 实现了基于 TMS320F2812 定点 DSP 的目标代码生成, 证实了自动代码生成技术在加快系统开发进程中发挥的重要作用。文献[3]采用 PID 调节器, 实现了直流电机控制的 DSP 程序直接代码生成。然而目前国内外文献虽然为实现嵌入式系统的一体化开发提供了新思路, 但对目标模型的设计方法涉及得较少, 在此背景下, 本文利用 Matlab 嵌入式直接目标代码生成工具, 对 TMS320F2812 定点 DSP 在控制系统一体化设计中的应用作了初步探索, 提出了一种基于实时代码生成工具的单极倍频 SPWM 波形产生方法, 以 TI 公司的 TMS320F2812 为控制芯片, 利用 Target Support Package TC2 模型库, 建立了 SPWM 控制模型, 在 MATLAB/Simulink 环境中分别进行算法仿真和自动代码生成, 并编译、下载到目标 DSP 中运行。从控制思想的产生到控制的实现, 真实的体现了利用 MATLAB 进行系统设计与代码生成的简易, 体现了直接目标代码生成在电力电子装置中的应用前景。

1 主电路拓扑及 SPWM 产生原理

单相桥式逆变器主电路如图 1(a)所示, 对于该电路的 SPWM 控制方法有多种, 其中单极倍频法应用最为广泛, 它可以获得两倍的开关频率, 有利于滤波器的设计。该方法采用一对反相的正弦信号作为调制波, 与三角波比较, 产生两路独立的 PWM 脉冲, 控制功率管的开通与关断, 其 SPWM 原理图如图 1(b)所示。

v_m 与 v_m^- 是一对反相的正弦调制波, v_{cr} 为三角载波, PWM 输出原理表述如下

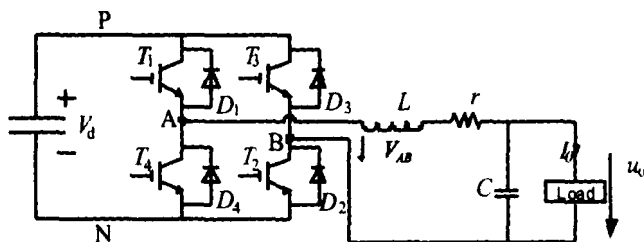
$$\begin{cases} v_m > v_{cr} \Rightarrow v_{g1} = 1 \\ v_m^- > v_{cr} \Rightarrow v_{g3} = 1 \end{cases} \quad (1)$$

收稿日期: 2009-02-12

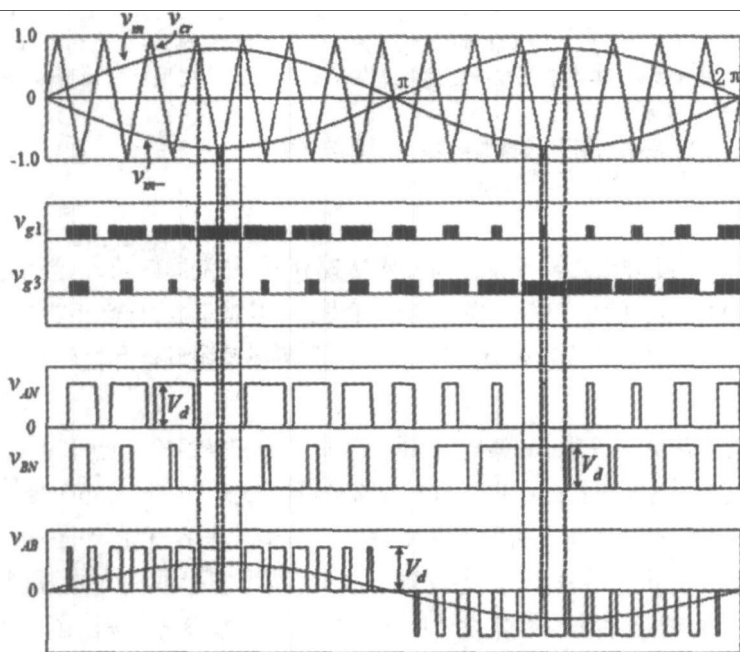
基金项目: 国家自然科学基金(50607020)

作者简介: 胡文华(1972-), 男, 江西南昌人, 博士研究生, 研究方向为电力电子传动及其控制技术。

如式(1)所述,产生的两路独立 PWM 信号 v_{g1}, v_{g3} , 分别为 T_1, T_3 的驱动信号, 另外两路信号与前者互补, 对应规则如下: 在不考虑死区的情况下, 同一桥臂的两个开关的信号互补, 即 T_1 导通时, T_4 关断; T_1 关断时, T_4 导通。 T_3 与 T_2 的关系和 T_1 与 T_4 的关系类似。



(a) 单相桥式逆变器拓扑图



(b) SPWM 原理图

图 1 主电路拓扑图及 SPWM 原理图

2 控制电路与模型实现

本系统的控制电路采用基于 TI 公司的定点控制芯片 TMS320F2812 的最小系统, 目标板通过 JTAG 仿真口与 PC 机相连, 提供 4 路 PWM 信号输出端口和各种接口功能给功率单元。

TMS320F2812 DSP 具有 150 MHz 的高速处理能力, 具备 32 位定点处理单元, 有多达 16 路的 PWM 输出, 12 位 16 通道 ADC 转换器, 特别适用于控制逆变器、交流感应电机等。与前一代 DSP 相比, 平均性能提升 50%, 并与定点 C24x 控制器软件兼容。

由于本文主要用到外设事件管理器模块, 这里对事件管理器模块作详细介绍。TMS320F2812 有二个事件管理器模块 EVA、EVB, 每个事件管理器模块包括: 两个 16 位通用定时器; 8 个 16 位的脉宽调制 (PWM) 通道。它们能够实现: 三相反相器控制; PWM 的对称和非对称波形; 可编程的 PWM 死区控制以防止上下桥臂同时输出触发脉冲^[4]。

根据 TMS320F2812 芯片的特点, 产生 SPWM 信号的三角载波由事件管理器模块内部硬件电路产生, 不能为负值, 更不能对其进行加减运算, 因此, 需要对图 1(b) 所示的调制波进行适当变换才能符合要求, 考虑到图 1(b) 中的与 v_m, v_{m-} 对应的三角载波 v_{cr} 的最大值为 1, 最小值为 -1, 调整后的比较关系如下:

$$\begin{cases} v_m/2 + 0.5 > v_{cr}' \\ v_{m-}/2 + 0.5 > v_{cr}' \end{cases} \Rightarrow \begin{cases} v_{g1} = 1 \\ v_{g3} = 1 \end{cases} \quad (2)$$

式中 v_{cr}' 为最大值为 1, 最小值为 0 的三角载波。

经过变换后的调制波, 对应的载波均是最大值 1、最小值 0, 频率为 10 kHz 的三角波 v_{cr}' 。考虑到事件管理器模块内部三角波是由内部定时计数器产生, 设载波频率为 f_c , 计数时钟频率为 f_{con} , 计数周期 P 为

$$P = \frac{f_{con}}{f_c} > 0$$

则有 $v_m/2 + 0.5 > v_{cr}' \Leftrightarrow P * (v_m/2 + 0.5) > P * v_{cr}'$ (3)

由于比较值 $P * (v_m/2 + 0.5)$ 有可能小于 0, 也有可能大于周期寄存器的值, PWM 模块要求输入的比较值必须非负, 且不能超出周期寄存器所能表示的范围, 而 $0 \leq P * v_{cr}' \leq P$, 于是需要对式(3)作限幅处理。需注意的是, 最终输入到 DSP 模块的值数据类型必须符合 DSP 寄存器的定义, 这里为 16 位无符号整型。

根据以上分析, 结合 Embedded Target for TI C2000 提供的模块, 建立单相单极倍频 SPWM 控制模型如图 2 所示。

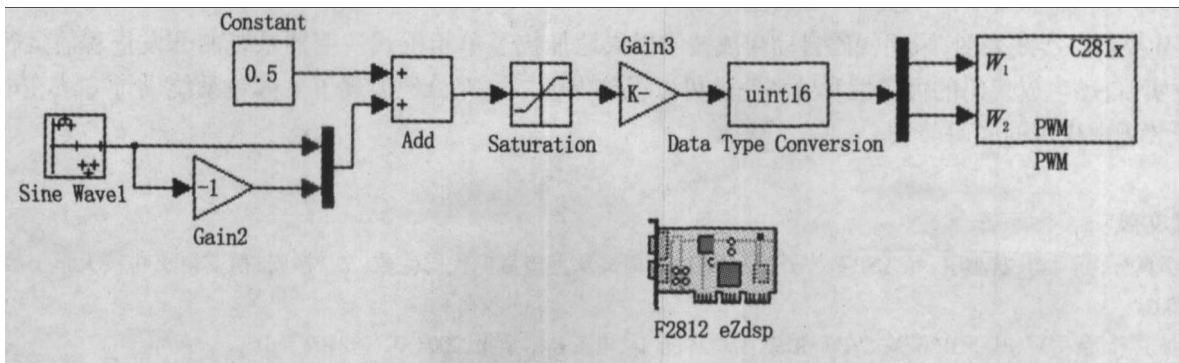


图 2 基于嵌入式目标代码的控制模型

图 2 中 sine wave1 为幅值 0.4, 频率为 400 Hz, 相位为 0 的正弦调制信号, 正弦调制信号 v_m 、 v_{m-} 在做上述处理后分别作为事件管理器模块 EVA 的输入, 至此, 算法模型已搭建完成。

建立模型的另一个重点内容就是事件管理器模块的设置:

(1) 在参数设置对话框中设置 Waveform type 为 Symmetric, 使模块输出对称的 PWM 波形。

(2) 设置 Duty cycle source 数值来源为端口输入, 完成设置后会在模块旁边出现一个输入端口用来装载 Duty cycle source 数值, 这里将其分别与两路比较值产生子模块输出相连, 根据设置, 一旦计数器到零, 事件管理器模块就会从该端口读入新的比较值, 用来更新比较寄存器。

(3) 根据设计的载波频率 f_c , 选择事件管理器模块的基准时钟频率 f_{con} 与计数周期 P , 通过计算, 设计开关频率为 10 KHz, 取 f_{con} 为 75 MHz 时, 应该选择计数周期 P 为 7 500, 对应的设置 TB clock 预分频为 2, 计数周期值内部指定, 初始值为 0。

(4) 设置 PWM 控制逻辑, PWM1 为低有效, PWM2 为高有效, PWM3 为低有效, PWM4 为高有效。其中 PWM1、PWM2、PWM3、PWM4 分别对应 v_{g1} 、 v_{g4} 、 v_{g3} 、 v_{g2} 。

(5) 最后, 设置前沿死区时间为 1 微秒。

控制模型中的 F2812 eZdsp 模块主要用于完成用户定制开发板的设置, 是建立 MATLAB \ Simulink 与目标板的重要纽带。通过该模块用户可以访问 building、linking、compiling、targeting 的设置, 从而可以根据需要配置实时工作站产生的代码, MATLAB 将根据其设置, 产生适合目标板的 C 语言程序。

3 实验结果

根据以上建立的目标模型,自动生成目标代码、编译、下载到目标板运行。DSP 输出 SPWM 波形如图 3 所示。

图 3 中 1、2 通道波形分别对应 v_{g1} 、 v_{g3} , MATH 为 $v_{g3} - v_{g1}$, 其为单极倍频信号。经测量,开关频率, PWM 波形输出模式均符合设计要求。

4 结束语

这种基于模型的直接目标代码生成方法,不需要熟练掌握 DSP 繁杂的寄存器位的设置,不需要在手工编程上花费过多的时间和精力,方便而简单地实现了单相单极倍频

SPWM 控制。实践表明,采用代码自动生成技术极大地加快了单相桥式逆变器系统的开发进程。运行测试证实:自动生成代码的可靠性和效率完全可以得到保证。这充分地展现了一体化系统设计方法在电力电子装置中应用的光明前景。

参考文献:

- [1] 李真芳,苏涛,黄小宇,等. DSP 程序开发: MATLAB 调试及直接目标代码生成[M]. 西安:西安电子科技大学出版社, 2003.
- [2] 张卫丰,余岳辉. 基于 RTW 的 SVPWM DSP 控制系统[J]. 电工技术学报, 2007, 22(3): 102 - 106.
- [3] Duma R, Dobra P, Abrudean M, et al. Rapid prototyping of control systems using embedded target for TI C2000 DSP[J]. Mediterranean Conference on Control and Automation, 2007, (7): 1 152 - 1 156.
- [4] 万山明. TMS320F281XDSP 原理及应用实例[M]. 北京:北京航空航天大学出版社, 2007.

Research on Automatic Code-generation of SPWM of Single-phase Inverter Based on Matlab/RTW Tools

HU Wen-hua^{1,2}, WEI Bao-quan²

(1. School of Electrical and Electronic Engineering, Huazhong University of Science and Technology, Wuhan 430074, China; 2. School of Electrical and Electronic Engineering, East China Jiaotong University, Nanchang 330013, China)

Abstract: Trying to minimize the time consumption and the cost of the whole product, a system design method and a development flow chart based on Matlab/RTW are introduced. An embedded code generation method is proposed in single-phase SPWM system. After having built and validated the algorithm models in Matlab/Simulink, a SPWM control system model whose target board is TMS320F2812 DSP is presented. Automatic code-generation is accomplished from this model, then compiled and downloaded into the TMS320F2812 DSP board. The results of the experiment are in line with theory analysis. This design process implements all the stages ranging from algorithm design to final realization, thus to improve efficiency of product developing, shorten its developing time, and reduce its developing cost.

Key words: TMS320F2812 DSP; Matlab; RTW; SPWM

(责任编辑:吴泽九)

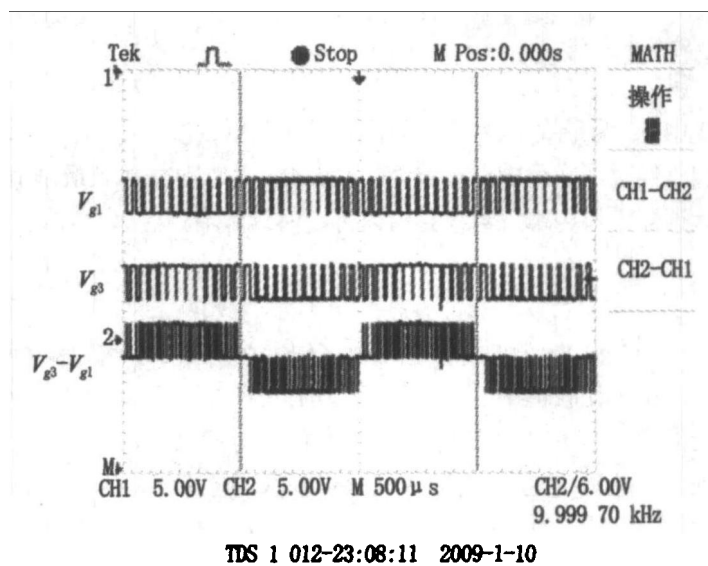


图 3 SPWM 波形输出