

基于 FPGA 多变量模糊神经网络控制器设计

靖 固， 刘璐杨， 于晓洋

(哈尔滨理工大学 计算机科学与技术学院, 黑龙江 哈尔滨 150080)

摘 要: 在对基于 FPGA 模糊神经网络控制的原理和设计中使用的算法进行分析后, 对模糊神经网络控制器进行分层设计, 设计模块主要包括模糊化模块、控制规则模块、归一化模块、权值-参数计算模块和去模糊化模块. 在设计过程中遵循自顶向下的设计原则, 并对各个模块设计进行优化处理, 最后对整个系统进行设计综合. 实验表明: 该设计方案是可行的, 不仅可以提高控制器的运行速度, 还可以改善传统模糊神经网络控制器的控制性能.

关键词: FPGA; 模糊神经网络控制; 设计综合
中图分类号: TP237 文献标志码: A 文章编号: 1007-2683(2011)02-0044-05

The Research of Multivariable Fuzzy Neural Network Controller Based on FPGA

JING Gu, LIU Lu-yang, YU Xiao-yang

(School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

Abstract In analysis of the control principle of fuzzy neural network based on FPGA and the design algorithm, hierarchal design of the fuzzy neural network controller is carried out. The design includes fuzzification module, control rule module, normalization module, weight parameter module and defuzzification module. The principle of top-down design is followed in the design process, each module design has been optimized, and then the whole system design and synthesis is carried out. The experiment results show the feasibility of the method. Meanwhile, this design has a much better performance than the custom fuzzy neural network controller and the speed is quick.

Key words FPGA; fuzzy neural network control; design synthesis

0 引 言

随着 EDA (electronic design automation) 技术的发展, FPGA (field programmable gate array) 在数字逻辑系统中发挥着越来越重要的作用. FPGA 最明显的优势在于集成度高, 体积小, 功耗低, 可靠性高和电路设计比较灵活, 因此应用 FPGA 设计各种各样的芯片的技术应运而生^[1]. 本文设计了一种基于

FPGA 的多变量模糊神经网络控制器, 并将此控制器应用于二关节机械手臂的被控对象中.

1 模糊神经网络控制

在实际的工业过程中, 往往很难得到被控对象的精确的数学模型, 采用模糊控制可以很好的解决这一问题, 模糊控制虽然具有反应快和处理模糊及不确定性问题的能力, 但传统的模糊控制系统在很

收稿日期: 2010-04-08
基金项目: 黑龙江省自然科学基金 (F200805)
作者简介: 靖 固 (1961—), 女, 博士, 教授, E-mail: jinggu@hrbust.edu.cn
刘璐杨 (1984—), 女, 硕士研究生.

大程度上依赖于专家或操作人员的经验和知识, 当专家或操作人员缺乏这种经验或经验不足时, 其控制效果较难令人满意^[2]; 神经网络控制虽然具有自学习和非线性逼近能力, 但它缺乏人脑的推理功能并需要大量的训练数据, 其控制效果也是差强人意的^[3]. 因此将神经网络和模糊系统相结合, 各取所长, 优势互补, 使设计出的模糊神经网络控制器在复杂的过程控制中进行合理有效地控制^[4].

1.1 模糊神经网络控制原理和算法分析

模糊神经网络的原理是将人工神经网络的自学习功能和模糊推理系统的模糊语言表达能力有机结合起来, 进行优势互补, 其模糊隶属度函数及模糊规则是通过大量已知数据的学习完成的, 而不是基于经验或直觉任意给定的^[5]. 基于此原理, 可采用自适应网络形式, 将具有归一化功能的 RBF 网络结构与模糊推理系统进行功能等价^[6], 得到 RBF 模糊神经网络结构, 如图 1 所示.

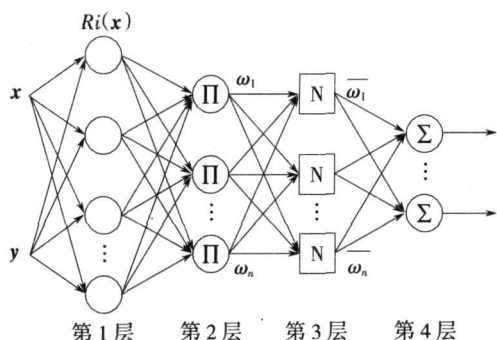


图 1 模糊神经网络结构图

第 1 层 输入变量模糊化

输入变量模糊化处理是模糊规则建立的第一步, 图中 $\mathbf{X} = [x_1, x_2, \dots, x_n]^T$ (或 $\mathbf{Y} = [y_1, y_2, \dots, y_n]^T$) 为输入变量, $R_i(x)$ (或 $R_i(y)$) 为模糊隶属度函数, 采用高斯径向基函数实现, 即

$$R_i(x) = \exp\left[-\frac{\|\mathbf{X} - \mathbf{C}_i\|^2}{2b_i^2}\right], \quad i = 1, 2, \dots, m \quad (1)$$

第 2 层 模糊规则激励强度 ω_i 的计算, 用 Π 表示, 其输出为

$$\omega_i = R_i(x)R_j(y) \quad (2)$$

第 3 层 激励强度归一化处理, 即第 i 个节点计算第 i 条规则的激励强度与所有规则激励强度之和的比值, 用 N 表示, 其输出为

$$\bar{\omega}_i = \frac{\omega_i}{\sum_{i=1}^m \omega_i} \quad (3)$$

第 4 层 去模糊化处理, 即采用加权平均判别

法来实现, 用 Σ 表示, 其输出为

$$u = \sum \bar{\omega}_i w_i = \frac{\sum_i \omega_i w_i}{\sum_i \omega_i} \quad (4)$$

关于模糊神经网络中涉及的网络权值及计算隶属度函数时所需的相关参数均可根据梯度下降法获得, 其迭代算法为

$$w_i(t) = w_i(t-1) + \eta(u(t) - u_m(t))h_i + \alpha(w_i(t-1) - w_i(t-2)) \quad (5)$$

$$\Delta b_i = (u(t) - u_m(t))w_i h_i \frac{\|\mathbf{X} - \mathbf{C}_i\|^2}{b_i^3} \quad (6)$$

$$b_i(t) = b_i(t-1) + \eta \Delta b_i + \alpha(b_i(t-1) - b_i(t-2)) \quad (7)$$

$$\Delta c_{ij} = (u(t) - u_m(t))w_i \frac{x_j - c_{ij}}{b_i^2} \quad (8)$$

$$c_{ij}(t) = c_{ij}(t-1) + \eta \Delta c_{ij} + \alpha(c_{ij}(t-1) - c_{ij}(t-2)) \quad (9)$$

其中: $w_i(t)$ 为网络权值; $u_m(t)$ 为网络输出; h_i 为高斯基函数; b_i 为节点 i 的基宽度参数; $c_{ij}(t)$ 为基函数中心矢量; η 为学习速率; α 为动量因子.

1.2 模糊神经网络控制的仿真

取二关节机械手臂为被控对象, 其运动方程可由以下二阶非线性微分方程描述^[7]:

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \Delta(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} \quad (10)$$

式中: $\mathbf{q} \in \mathbf{R}^2$ 为关节位置向量; $\mathbf{H}(\mathbf{q}) \in \mathbf{R}^{2 \times 2}$ 为机械手臂的正定质量惯性矩阵; $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbf{R}^2$ 表示离心力和哥式力; $\Delta(\mathbf{q}, \dot{\mathbf{q}})$ 为建模误差; $\boldsymbol{\tau} \in \mathbf{R}^2$ 为关节驱动输入控制力矩. 式 (10) 具体可表达为

$$\begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} -2m_2 l_1 s_2 \dot{q}_2 \sin q_2 & -m_2 l_1 s_2 \dot{q}_2 \sin q_2 \\ -2m_2 l_1 s_2 \dot{q}_2 \sin q_2 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \Delta(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (11)$$

其中: m 表示关节质量; l 表示关节长度; s 表示关节质心距轴心距离; I 表示关节转动惯量; $d_{11} = m_1 s_1^2 + m_2 (l_1^2 + s_2^2 + 2l_1 s_2 \cos q_2) + I_1 + I_2$, $d_{12} = d_{21} = m_2 (s_2^2 + l_1 s_2 \cos q_2) + I_2$, $d_{22} = m_2 s_2^2 + I_2$.

根据式 (10)、(11) 所列出的运动方程, 将给定的机械臂各关节运动轨迹坐标数据作为训练样本, 利用式 (1) ~ (9) 给出的模糊神经网络控制算法计算得到的运动轨迹如图 2 所示.

在训练样本中,以机械手臂关节位移向量为网络输出值,以关节位移向量对应的关节末端坐标值为网络输入,对网络进行训练并求出关节向量,由所求的关节向量对机械手臂进行操作,从而得到机械手臂关节末端所达到的实际位置坐标数据与给定的目标位置坐标数据的对照表(由于篇幅有限,进列出部分对比数据),如表 1 所示.由表 1 可以看出,模糊神经网络通过对样本数据进行学习,完成对网络的训练后,所达到的实际位置基本满足机械手臂关节的控制精度要求,并能完成给定的操作动作.如想达到更高要求,则需要更多的样本数据对网络进行训练.

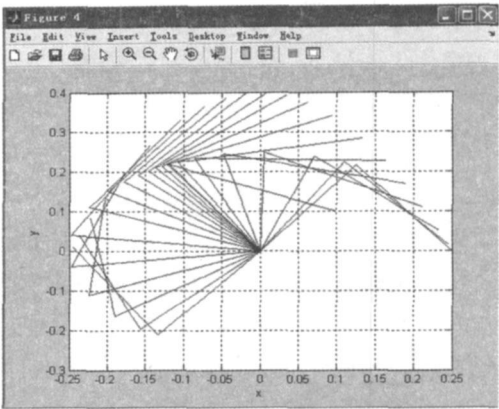


图 2 模糊神经网络控制器仿真结果图

表 1 目标位置坐标与实际位置坐标比较

序号	目标位置	实际位置	序号	目标位置	实际位置
0	(- 0.245 9, 0.013 2)	(- 0.247 0, 0.010 6)	16	(0.011 9, 0.399 1)	(0.010 7, 0.399 3)
1	(- 0.234 6, 0.047 7)	(- 0.234 5, 0.048)	17	(0.039 1, 0.390 2)	(0.039 7, 0.389 9)
2	(- 0.218 6, 0.094 2)	(- 0.218 7, 0.093 7)	18	(0.068 3, 0.370 2)	(0.067 0, 0.371 3)
...	...	...	...	...	...

2 模糊神经网络控制器的 FPGA 设计

模糊神经网络控制器的设计是采用自顶向下的设计方法,在顶层进行系统的模块划分和设计,

然后用 Verilog HDL 语言和原理图结合的方法进行电路设计^[8].最后下载到 Altera 公司的 Cyclone II 系列中的 EP2C35F672C6 芯片的电路板上进行仿真实验.模糊神经网络控制器的顶层结构图如图 3 所示.

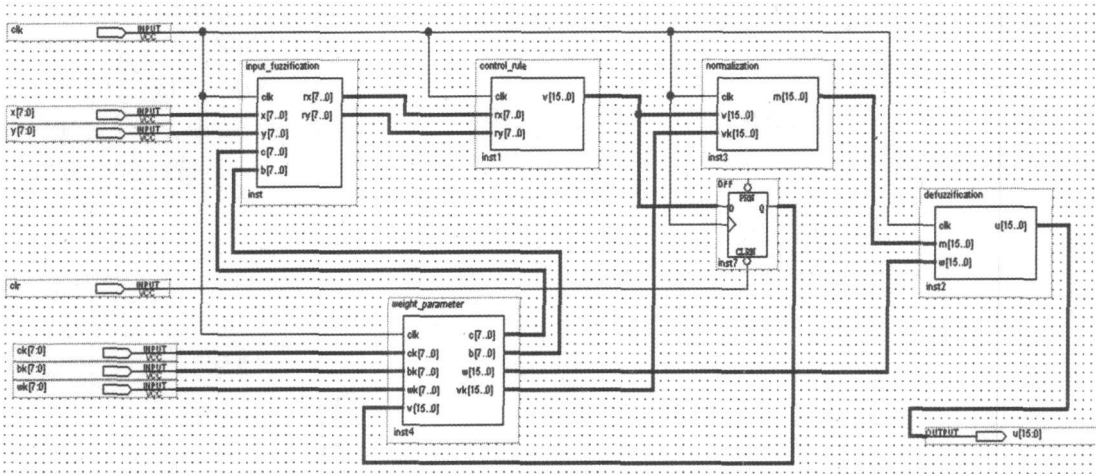


图 3 模糊神经网络控制器的顶层结构图

整个模糊神经网络控制器主要由 5 个模块构成:模糊化模块、控制规则模块、归一化模块、权值 - 参数计算模块和去模糊化模块^[9].其中,模糊化模块主要是对控制器的输入变量进行模糊量化处理,并计算隶属函数;控制规则模块主要完成控制器的控制规则表的建立,输出控制规则的激励强度;归一化模块主要对控制规则激励强度进行处理,即计算

每条控制规则激励强度与所有控制规则激励强度之和的比值,并计算出模糊控制量;权值 - 参数计算模块主要根据在线学习算法中的梯度下降法,对控制器的网络权值及参数的样本数据进行学习,完成对控制网络的训练,以满足控制器的控制精确度要求;去模糊化模块主要对计算出的模糊控制量进行清晰化处理,并将得到的精确控制量输出,完成对被控对

象的控制操作. 由式 (1) ~ (9) 分析可知, 可以由加法运算、减法运算和乘法运算来实现这五个模块. 减法器可以用加法器进行设计, 而乘法器的运算速度和占用的资源多少对整个模块的运算速度和占用的资源起着关键的作用^[10].

2.1 加法器的 VerilogHDL 语言设计和仿真

由上面的算法分析可知, 在设计模糊神经网络控制器时主要涉及的是加法运算、减法运算和乘法运算. 加法运算可以用简单的串行加法器实现, 串行加

法器虽然设计电路简单、连接方便, 但缺点是运算速度慢, 由于最高位的运算必须等到所有低位进位的运算依次结束并产生新的进位信号后才能进行, 因此运算速度受到限制. 为了提高加法器的运算速度, 本文采用一种并行加法器, 即超前进位加法器, 原理是在串行加法器基础上增加超前进位形成逻辑, 以减少由于逐位进位信号的传递所造成的延迟. 在用 VerilogHDL 语言设计超前进位时, 主要由计算进位和计算加法位的两个模块组成, 其功能仿真图如图 4 所示.

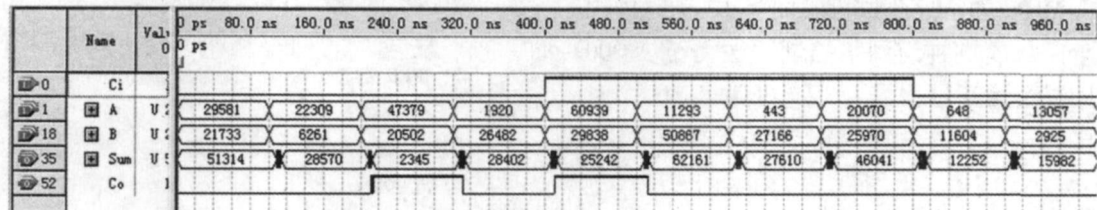


图 4 超前进位加法器的仿真图

2.2 乘法器的 VerilogHDL 语言设计和仿真

乘法器是算术单元, 数字信号处理器和其他进行算术运算的电路的重要功能单元. 乘法运算可以用组合逻辑电路或时序逻辑电路来实现. 本文采用时序逻辑电路来实现乘法运算, 主要运用状态机方法进行设计, 因为状态机的结构简单, 设计灵活, 更易构成良好的同步时序逻辑, 还可以避免电路中的毛刺现象. 本文设计状态机是主要有 6 个状态,

state0 到 state5 乘法器的执行过程是: 在初始状态 state0 下, 乘法器接收乘数 x 和被乘数 y , 但不做任何其他操作. 当启动信号 $start$ 变为高电平时, 在时钟 $clock$ 的驱动下进入状态 state1; 否则, 一直停留在状态 state0. 在 state1 到 state4 的过程中, 主要执行加法和右移操作, 求部分积. 在 state5 状态, 主要完成最后的乘积计算, 并将结果输出, 其仿真图形如图 5 所示.

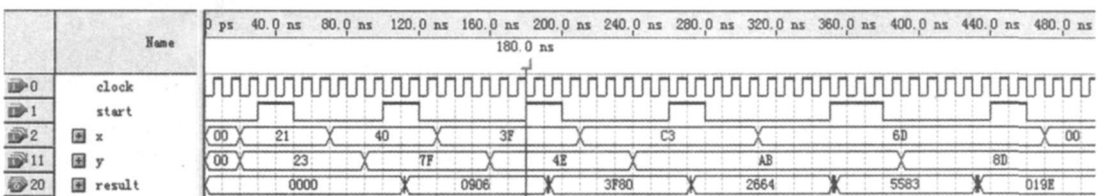


图 5 乘法器的仿真图

2.3 通用 A/D、D/A 转换器的设计

为了不失系统的通用性和可扩展性, 本文采用常用的 ADC0809 进行模数转换的设计. ADC0809 是 CMOS 的 8 位模数转换器, 片内有 8 路模拟采样开关, 可控制 8 位模拟量中的一个进入转换器中. 通过对 ADC0809 的工作时序的分析, 可以采用 Moore 状

态机的方法对 ADC0809 进行 VerilogHDL 语言的设计, 仿真结果如图 6 所示. 整个系统首先通过 ADC0809 转换得到数字信号, 然后经过模糊神经网络控制器的处理, 最后经过 DAC0832 可以得到模拟信号来实现对被控对象的控制.

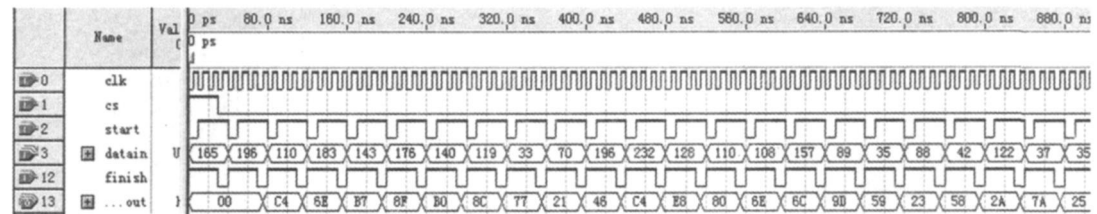


图 6 A/D 转换器的仿真图

2.4 实验对比

由于在整个硬件电路的设计中乘法器是电路中最关键的逻辑电路模块. 在仿真试验中通过对时序逻辑性能分析仪进行分析可知, 乘法器稳定运行所需的最小时钟周期是 30 ns, 最大的时钟频率可以达到 33MHz. 由于整个运算过程共需要 25 个时钟周期, 所以基于 FPGA 设计的模糊神经网络控制器的电路的最快采样处理频率为: $33\text{MHz}/25=1.32\text{MHz}$. 而传统单片机 (以 MCS-51 系列为例), 若最大时钟频率为 12MHz, 每条指令执行时间为 12 个时钟周期, 要完全完成模糊神经网络控制算法需至少 200 条指令, 则整个运算过程需要 $12\times 200=2400$ 个时钟周期, 采样处理频率为 $12\text{MHz}/2400=5000\text{Hz}$. 可见 FPGA 方法的仿真处理速度已经远远的超过了传统单片机方法的仿真处理速度, 传统仿真与 FPGA 仿真运行速度比较及仿真性能比较分别如表 2 和表 3 所示.

表 2 传统仿真与 FPGA 仿真运行速度比较

速度比较	时钟周期 /个	处理频率 /Hz
传统单片机方法仿真	2 400	5 000
FPGA 方法仿真	20	1×10^6

表 3 传统仿真与 FPGA 仿真性能比较

性能比较	调整时间 /s	最大超调 /(%)	稳态误差
传统方法仿真	9.969	30.566	0.152
FPGA 方法仿真	1.427	22.361	0.033

3 结 语

基于 FPGA 构建的模糊神经网络控制器具有设计灵活、可在线训练学习、可靠性高以及开发周期短的优点. 系统设计主要采用自顶向下的设计方法和

模块化设计思想, 将整个设计分成 5 个部分. 各个模块通过 Verilog HDL 语言和原理图结合的方法进行设计, 最后进行各个模块的整合; 在模块内部电路设计时, 采用了优化设计, 与传统的系统设计相比占用的资源少、设计灵活. 实验表明该设计方案是可行的, 不仅可以提高控制器的运行速度, 还可以改善传统模糊神经网络控制器的控制性能.

参 考 文 献:

[1] CHAN Y F, MOALLEM, WANG W. Efficient Implementation of Advanced Control Algorithm Using FPGA Technology [C] // The 43rd IEEE Conference on Decision and Control Bahamas 2004 11: 4885-4890

[2] 张慧平, 戴波, 杨薇. 现代控制理论在过程工业中的应用和发展 [J]. 北京石油化工学院学报, 2006, 14(3): 57-61

[3] STELLIOS Timotheou The Random Neural Network: a Survey [J]. The Computer Journal 2010, 11(53): 251-267.

[4] 张弘. 仿人模糊神经网络控制器的设计和实现 [J]. 微机计算机信息, 2009, 25(7): 56-58

[5] 崔涛, 杨全胜. 基于 FPGA 技术的通用模糊控制器设计 [J]. 计算机技术与发展, 2009, 19(3): 15-18

[6] LASZLO Bako Real-time Classification of Datasets with Hardware Embedded Neuromorphic Neural Networks [J]. Brief Bioinform, 2010, 7(11): 348-353.

[7] ALEXANDER Spröwitz, REO Moeckel, AUKE Jan Ijspeert Learning to Move in Modular Robots Using Central Pattern Generators and Online Optimization [J]. The International Journal of Robotics Research 2008 8(27): 423-443

[8] 李利歌, 阎保定, 侯忠. 基于 FPGA 的神经网络硬件可重构实现 [J]. 河南科技大学学报: 自然科学版, 2009, 30(1): 37-40.

[9] TONG Shaoheng, LI Changying, LI Yongning Fuzzy Adaptive Observer Backstepping Control for MIMO Nonlinear Systems [J]. Fuzzy Sets and Systems 2009 8(19): 2755-2775

[10] 吴军, 吁少峰, 来昂. FPGA 乘法器在电能集中采集系统中的应用 [J]. 江西理工大学学报, 2009, 30(6): 34-37

(编辑: 温泽宇)