

三等奖

基于FPGA的智能型控制应用于 感应电机设计

大学院校： 元智大学电机工程学系

参赛队员： 锺招名 林明宏 陈奕隆

指导教师： 林志民

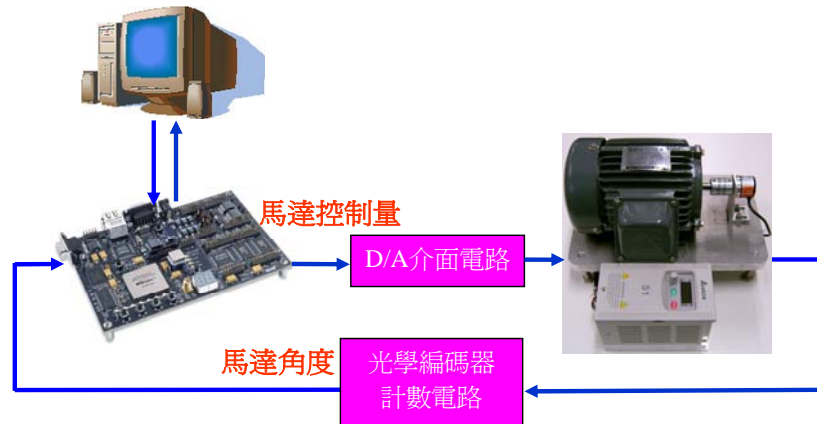
一. 设计概述

若以控制观点而言，直流电机具有电刷和整流子所以需要时常保养维修，如果电机安装在不易维修的场所，容易造成保养维修的困难或无法维修，此时就突显了直流电机的缺失，而交流电机则改善了上述直流电机的缺失，此外，交流电机尚有体积小、重量轻、低转动惯量和价格较便宜等优点。一般而言，因感应电机的动态特性为非线性时变且耦合的系统，所以感应电机的控制器设计往往要复杂许多。一般在解决控制问题时，常常有各式各样的控制理论被提出，例如比例-积分-微分控制、滑动模式控制或适应性控制等，上述之方法都是为了系统参数变动和各种外来的干扰情况下仍可使系统的行为合乎设计的要求。这些设计方法大都基于需要得知全部或部分受控系统的状态方程式，然而，在实际运用上受控系统的状态方程式往往并不容易精确获得。

如此一来，探求具有自我学习能力的智能型控制方法，以达到更好的控制效能为相当重要的研究课题之一。类神经网络被广泛的应用在面对系统具有不确定性、时变性和非线性等复杂状态时皆能提供传统控制器无法达到的效果，其成功的关键在于类神经网络的近似特

性。目前所提出的常见学习法有倒传递算法、李亚普诺夫稳定法和基因演算法等学习方式。在使用上，倒传递算法虽直接且易懂，但该方法却难保证闭回路系统的稳定性和强健性。李亚普诺夫稳定性理论为基础的控制结构，虽然系统的稳定性将可确保，但计算过程较为复杂。基因演算法虽可获得全域优化的结果，但其计算量过大且不适合实时控制运用上。因此，本参赛队伍提出一适应性模糊类神经网络控制器演算法于感应电机位置控制上，整个控制系统参数依据李亚普诺夫稳定定理和倒传递演算法则推导出来的学习法则实时自我调整参数，同时所设计开发的演算法则将可得到最快的参数收敛速度和易于实做实现等优点。

在实际运用上，数字控制器和模拟控制器相比较下，数字控制器较稳定、输出可预期，且抗噪声能力较强，尤其近年来微电子半导体技术迅速发展，使得单一器件的逻辑电路成为目前的设计趋势，若能将数字电路集成及控制法则IC化，必定可以减少控制系统总体的复杂度及增加系统的可靠度，如此不仅可以大大的缩小硬件的体积，降低设计的成本，且有执行速度快、灵活度高的优点。现场可编程阵列十分适合经济效益和研究进度的考虑。基于以上想法，我们利用Altera公司所推出之Nios II系统开发板来实现所设计开发的控制法则，而程序的撰写上选择以Verilog HDL程序语言来实现硬件部分，再以Nios core来实现控制法则的软件部分，并结合一些硬件外围电路设计建置完成全部实验环境，如图(一)所示为所提出的基于FPGA的智能型感应电机设计图。



图(一) 基于FPGA的智能型感应电机初步设计图

本设计最重要也最难以实现的部分就是在智能控制演算法实现部分，因为所提出控制法则所使用到的计算量较多，同时需要使用到正负数以及浮点等复杂运算，而 Verilog HDL 是利用二进制的观念，故当采用 Verilog HDL 来撰写的话，虽然可以分别利用补码及固点的方式来实现，但对于不熟悉 Verilog HDL 语法的人来说，将会需要花费较多的时间来撰写，同时程序在维护上也显得困难重重。然而，当采用 Nios 嵌入式核心处理器来设计时，除了可采用较为熟悉的 C 语言撰写外，也不需要考虑到数值的正负数以及浮点问题，可以直接以十进制的观念来撰写，且如果将控制法则利用 Nios 嵌入式核心处理器来撰写的话，会因为 Nios core 的编译时间较硬件快很多，所以在参数调整时，而可以节省许多时间。另外，因 Nios 嵌入式核心处理器也提供了浮点的定制化指令，所以当设计者加入此定制化指令后，将可以大大的降低硬件在处理浮点运算所需的时间，进而提升系统的效能。

二. 功能描述

一般而言, 感应电机服务驱动系统可简化成

$$J\ddot{\theta} + B\dot{\theta} + T_l = T_e \quad (1)$$

其中 J 为转动惯量、 B 为阻尼系数、 θ 为电机转动角度、 T_l 为额外负载干扰及 T_e 为电磁转矩可定义如下

$$T_e = K_t i_{qs}^* \quad (2)$$

$$K_t = (3n_p/2)(L_m^2/L_r) i_{ds}^* \quad (3)$$

其中 K_t 为转矩常数、 i_{qs}^* 为转矩电流命令、 i_{ds}^* 为流出电流命令、 n_p 代表及对、 L_m 为气隙磁通量及 L_r 为转子电感。综合以上所言, 感应电机的动态方程式可以重写如下

$$\ddot{\theta} = -\frac{B}{J}\dot{\theta} + \frac{K_t}{J}i_{qs}^* - \frac{1}{J}T_l \triangleq A_p\dot{\theta} + B_p u + D_p T_l \quad (4)$$

其中 $A_p = -B/J$ 、 $B_p = K_t/J > 0$ 、 $D_p = -1/J$ 及 $u(t) = i_{qs}^*(t)$ 为控制命令。整个感应电机控制系统之目的为设计一控制法则, 使电机角度可以准确地追随到控制命令, 在此定义一追踪误差如下

$$e = \theta_c - \theta \quad (5)$$

首先假设感应电机的系统动态函数 A_p 和 B_p 可明确获得且外来干扰 T_l 亦可获得, 则利用回授控制理论可以获得一理想控制法则

$$u^* = B_p^{-1} \left[-A_p\dot{\theta} - D_p T_l + \ddot{\theta}_c + k_1\dot{e} + k_2e \right] \quad (6)$$

将理想控制法则(6)式带入系统动态方程式(4)后可得

$$\ddot{e} + k_1\dot{e} + k_2e = 0 \quad (7)$$

如果上式中 k_1 和 k_2 适当选取使得方程式(7)为一Hurwitz多项式(其根皆在左半平面), 则可以达到 $\lim_{t \rightarrow \infty} e = 0$ 的控制目的。然而, 在实际运用上, 系统动态函数和外来干扰往往无法获得, 即理想控制法则(6)往往并无法实际实现。

为了解决上述系统动态函数和外来干扰无法获得, 造成理想控制器无法实现的问题, 本设计拟提出一适应性模糊类神经网络控制器, 该控制器系统方框图如图(二)所示, 包括一个类神经网络控制器和一个补偿控制器, 其数学公式可表示

$$u = u_m + u_{cp} \quad (8)$$

其中类神经网络控制器 u_m 主要利用一模糊类神经网络来学习理想控制器 u^* , 和补偿控制器 u_{cp} 用来克服因类神经控制器的学习误差。首先, 定义一PID形式之滑动表面如下

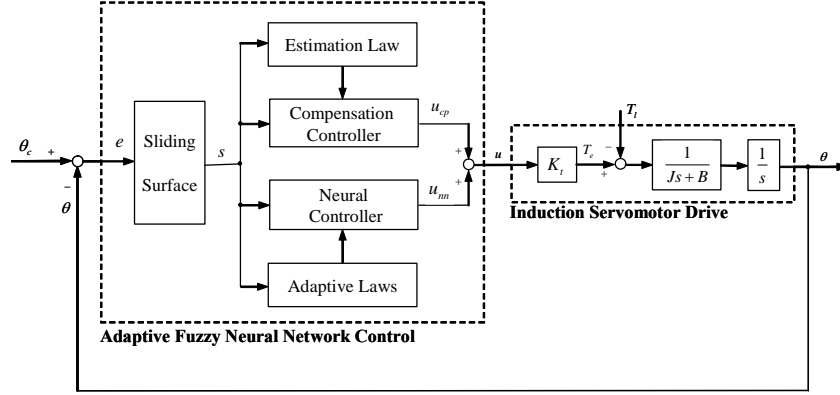
$$s = \dot{e} + k_1e + k_2 \int_0^t e(\tau) d\tau \quad (9)$$

将方程式(8)代入方程式(4)后可得

$$\ddot{\theta} = A_p\dot{\theta} + B_p(u_m + u_{cp}) + D_p T_l \quad (10)$$

经由方程式(6)和方程式(10)两式相减后再利用方程式(9), 得到动态方程式如下

$$\dot{s} = B_p(u^* - u_m - u_{cp}) \quad (11)$$



图(二) 适应性模糊类神经网络控制系统方框图

依据近似定理得知存在一最佳类神经网络控制器可以十分逼近理想控制器(6)，即

$$u^* = \mathbf{w}^{*T} \Theta + \varepsilon \quad (12)$$

其中 \mathbf{w}^* 为最佳值模糊规则参数矢量、 Θ 为模糊规则激发参数矢量和 ε 为网络学习近似误差且假设 $|\varepsilon| \leq E$ 。在实际运用上最佳网络参数 \mathbf{w}^* 往往是无法直接获得的或是非唯一解的，所以利用一估测器来学习估测整个最佳网络参数，即

$$u_m = \hat{\mathbf{w}}^T \Theta \quad (13)$$

其中 $\hat{\mathbf{w}}$ 为 \mathbf{w}^* 的估测参数矢量。如此可得网络输出的学习误差可定义为

$$\tilde{u} = u^* - u_m = \tilde{\mathbf{w}}^T \Theta + \varepsilon \quad (14)$$

其中 $\tilde{\mathbf{w}} = \mathbf{w}^* - \hat{\mathbf{w}}$ 。将(14)式带入(11)式可简化为

$$\dot{s} = B_p (\tilde{\mathbf{w}}^T \Theta + \varepsilon - u_{cp}) \quad (15)$$

为了可以在线学习所需的控制器参数，并确保整个闭回路系统之稳定性，本设计将依据李雅普诺夫稳定定理推导出所需的学习法则，在此选定一李雅普诺夫函数如下

$$V = \frac{1}{2} s^2 + B_p \left(\frac{\tilde{\mathbf{w}}^T \tilde{\mathbf{w}}}{2\eta_1} + \frac{\tilde{E}^2}{2\eta_2} \right) \quad (16)$$

其中 η_1 和 η_2 为学习速度，且 $\tilde{E} = E - \hat{E}$ 。将(16)式对时间微分并将(15)式代入后，可得

$$\begin{aligned} \dot{V} &= s\dot{s} + B_p \left(\frac{\tilde{\mathbf{w}}^T \dot{\tilde{\mathbf{w}}}}{\eta_1} + \frac{\tilde{E} \dot{\tilde{E}}}{\eta_2} \right) \\ &= s[B_p (\tilde{\mathbf{w}}^T \Theta + \varepsilon - u_{cp})] + B_p \left(\frac{\tilde{\mathbf{w}}^T \dot{\tilde{\mathbf{w}}}}{\eta_1} + \frac{\tilde{E} \dot{\tilde{E}}}{\eta_2} \right) \\ &= B_p \tilde{\mathbf{w}}^T \left(s\Theta + \frac{\dot{\tilde{\mathbf{w}}}}{\eta_1} \right) + sB_p (\varepsilon - u_{cp}) + B_p \left(\frac{\tilde{E} \dot{\tilde{E}}}{\eta_2} \right) \end{aligned} \quad (17)$$

选择学习法则如下

$$\dot{\hat{\mathbf{w}}} = -\dot{\tilde{\mathbf{w}}} = \eta_1 s \Theta \quad (18)$$

选择补偿控制器如下

$$u_{cp} = \hat{E} \operatorname{sgn}(s) \quad (19)$$

且

$$\dot{\hat{E}} = -\dot{\tilde{E}} = \eta_2 |s| \quad (20)$$

即(17)可简化如下

$$\dot{V} = \varepsilon s B_p - E |s| B_p \leq -(E - |\varepsilon|) |s| B_p \leq 0 \quad (21)$$

如此, 所设计的控制系统可以依据李亚普诺夫稳定定理保证系统稳定。为了增加网络的学习效能, 本研究接著引入最陡坡降法以调整更多的参数, 首先定义一能量函数如下

$$E = \frac{1}{2} e^2 \quad (22)$$

依据最陡坡降法可得模糊规则的调整量为

$$\Delta w_k = -\eta_w \frac{\partial E}{\partial w_k} = \left[-\eta_w \frac{\partial E}{\partial y_o^4} \frac{\partial y_o^4}{\partial net_o^4} \right] \left[\frac{\partial net_o^4}{\partial w_k^4} \right] = -\eta_w \delta_o^4 x_k \quad (23)$$

其中 $\delta_o^4 = \frac{\partial E}{\partial net_o^4}$ 。将(18)式和(23)式系数比较可获得

$$\delta_o^4 = s \quad (24)$$

此为整系统之Jacobian项, 接著即可进行归属函数参数调整如下

$$\Delta m_{ij}^2 = -\eta_m \frac{\partial E}{\partial m_{ij}^2} = \left[-\eta_m \frac{\partial E}{\partial y_o^4} \frac{\partial y_o^4}{\partial net_o^4} \frac{\partial net_o^4}{\partial y_k^3} \frac{\partial y_k^3}{\partial net_k^3} \frac{\partial net_k^3}{\partial y_j^2} \frac{\partial y_j^2}{\partial net_j^2} \frac{\partial net_j^2}{\partial m_{ij}^2} \right] \quad (25)$$

$$\Delta \sigma_{ij}^2 = -\eta_\sigma \frac{\partial E}{\partial \sigma_{ij}^2} = \left[-\eta_\sigma \frac{\partial E}{\partial y_o^4} \frac{\partial y_o^4}{\partial net_o^4} \frac{\partial net_o^4}{\partial y_k^3} \frac{\partial y_k^3}{\partial net_k^3} \frac{\partial net_k^3}{\partial y_j^2} \frac{\partial y_j^2}{\partial net_j^2} \frac{\partial net_j^2}{\partial \sigma_{ij}^2} \right] \quad (26)$$

本设计所提出的基于FPGA的感应电机智能型控制系统方框图如图(三)所示, 其中硬件电路部分包括了除频电路(Divider)、感应电机角度计数模块(Theta_Acc)以及两组的数字转模拟信号模块(DAC_1, DAC_2)。软件部份则为Nios嵌入式核心处理器(Nios II CPU), 各模块详细描述如下:

(a) 除频电路(Divider)

由于FPGA芯片的输入频率为50MHz, 所以需要将其除频为符合所需的频率, 首先设计两个频率(clk, clk1), 一个是用来控制DAC Converter的Chip Select(CS)信号以及更新DAC数据信号(LDAC), 并控制系统能够每1msec输入及输出数据一次, 另外一个则用来控制DAC Converter的数据选择信号(A0、A1), 利用比clk还慢的clk1来控制, 以确保输出数据在完全输入至Input Latch后才更换数据选择信号, 防止输出不正确的数据。

(b) 感应电机角度计数模块(Theta_Acc)

本模块将12比特的光学编码器计数电路所计算出的电机角度(en), 利用累加器增加为15比特。因为光学编码器计数电路输出只有12比特(0~4095), 所以利用一个15比特的寄存器(Theta)将光学编码器的输出做累加的动作, 使得电机旋转的角度可以大于4095, 并有分正转和反转。

另外设计一判断条件, 使得程序有能力判断当前一次和后一次的角度超过光学编码器12比特的4095以及小于0后的角度是正转还是反转, 以计算出正确的旋转角度。但前提是要前一次和后一次的旋转角度不得大于2047度, 然后在判断完旋转角度后将结果以累加器来做累加的动作, 并将其输出。

(c) 函数生成器模块

此模块包含一个存储正弦函数值的内存，并利用查表法的方式读取内存的内容值，以生成电机追踪命令的正弦函数 (θ_{ref})。其中正弦函数内存内容值设计成可输出振幅2V，频率为 $1/2\pi$ ，并在5.5秒后增加为 $5/3\pi$ 之正弦函数。

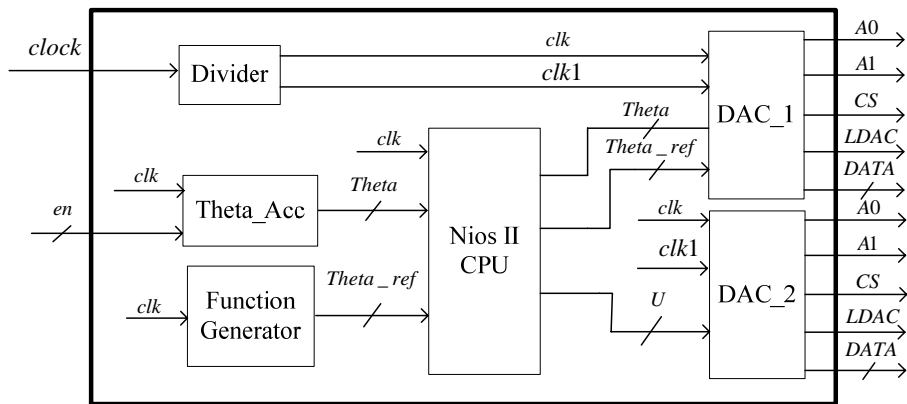
(d) Nios嵌入式核心处理器

Nios嵌入式核心处理器(Nios II CPU)主要用来撰写所提出的感应电机智能型控制法则，并利用中断来控制其计算周期为1msec。因此程序开始执行并遇到中断生成时，先将硬件电路的感应电机角度计数模块以及函数生成器模块所输出的电机角度以及追踪命令输入至Nios II CPU，然后再利用此CPU内所撰写的控制法则计算电机的控制量。由于计算后的电机控制量范围在-5V~5V之间，但是因为感应电机的控制电压范围为0V~10V，其中5V为停止，大于5V为正转，小于5V为反转，所以必需要用位移的方式加上5V，使得控制电压范围可以在0V~10V之间，最后才将此控制量输出到下一个模块。

(e) 数字转模拟信号控制模块(DAC_1, DAC_2)

第一组之数字转模拟信号模块用来将追踪命令及电机角度输出，输出电压范围为-5V~5V。由于DAC IC可以输出两组信号，但一次只能接收8比特的数据，而输出信号为12比特，因此需要分4次将追踪命令及电机角度输出。首先将欲输出的12比特追踪命令及电机角度分别拆开成8个低比特数据，以及4个高比特数据，然后利用数据选择信号(A1, A0)来分别输出数据给DAC Converter的Input Latch，并利用更新信号(LDAC)来将Input Latch的数据传送到DAC Latch以输出更新后的数据。而输出数据顺序为先输出追踪命令的8个低比特，然后是追踪命令的4个高比特，再来是电机角度的8个低比特，最后是电机角度之4个高比特。

第二组的数字转模拟信号模块用来输出电机控制量，因为DAC IC可以输出两组信号，同样也是将两组12比特的电机控制量拆开成8个低比特数据及4个低比特数据，并分4次输出。而其中一组电机控制量输入电压范围为0V~10V，用来控制感应电机，另一组电机控制量输入电压范围为-5V~5V，用来接示波器观察用。



图(三) FPGA感应电机控制系统硬件方框图

三. 性能参数

本设计因为采用了FPGA来实现，所以感应电机角度取样周期以及感应电机控制频率皆可达到1kHz，相较于传统采用计算机或者是单芯片等硬件实现之控制效能，有非常显著的提升。尤其因为利用了Nios嵌入式核心处理器来计算所提出的智能型控制法则，不仅仅能够简化了原本采用Verilog HDL撰写程序时的复杂度，同时也大大的减少了程序设计所需要花费的时间。另外由于外围装置的部分可以快速且自由的添加，以及因为有许多IP core可以使用，所以使得整个控制系统可以容易的依设计者的需求迅速的作调整，进一步能够更快速的修改控制法则的运算参数和演算法则。

四. 设计结构

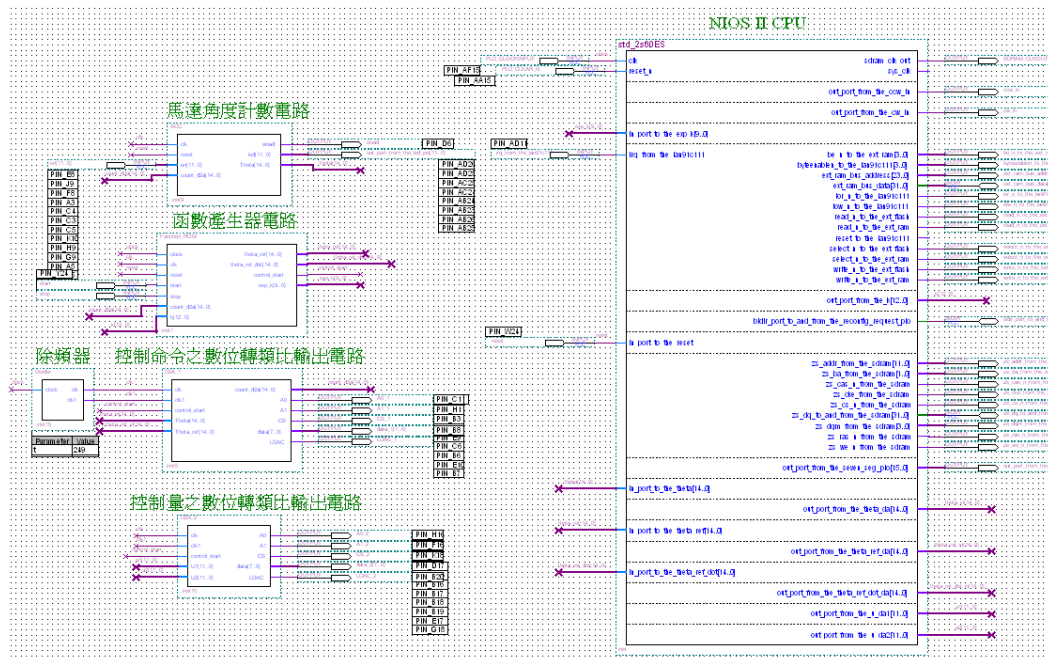
本设计的外围电路如图(四)所示，其中包含光学编码器计数电路及两组可调整输出电压范围之数字转模拟信号电路，整组感应电机定位控制实验环境如图(五)所示，其中利用Quartus II软件以及Nios II核心所设计完成的系统设计电路图则如图(六)所示。最后，图(七)的软件流程图说明由Nios II IDE所撰写的C语言程序流程。



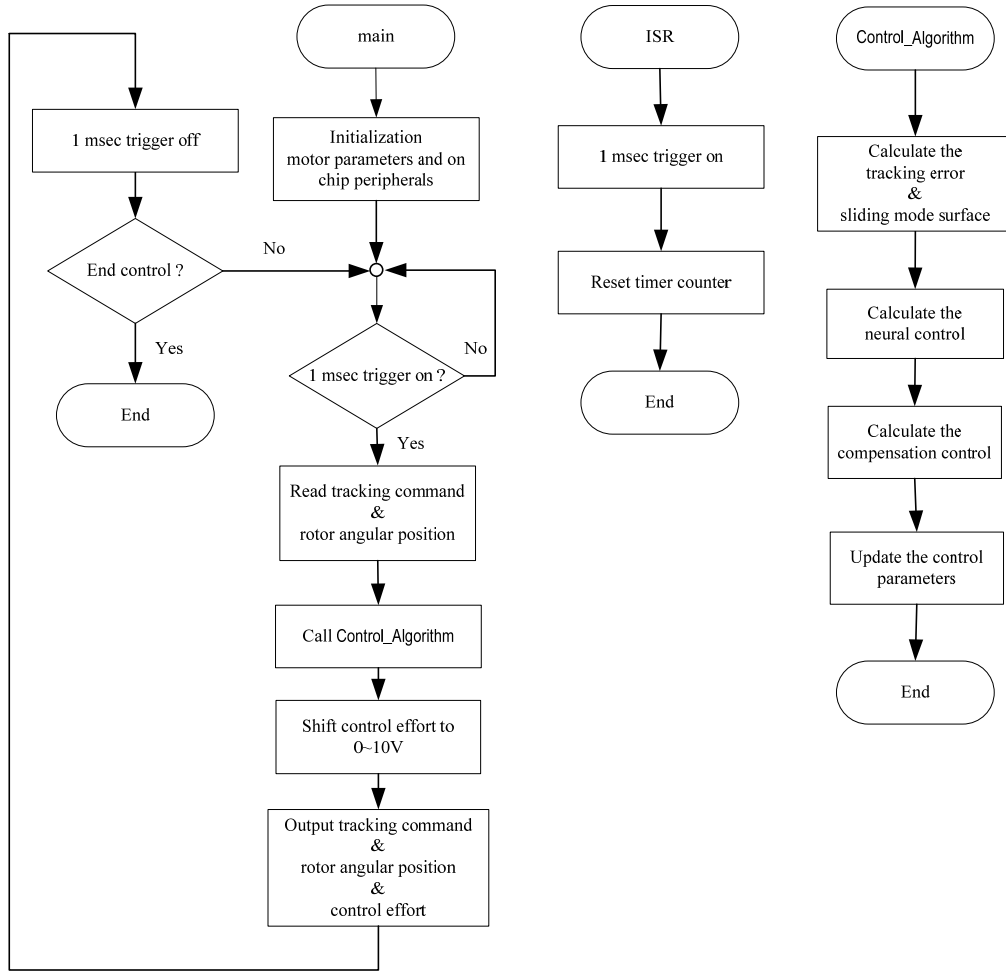
图(四) 外围硬件印刷电路板



图(五) 基于FPGA的感应电机智能型控制系统实验环境



图(六) 基于FPGA的感应电机智能型控制系统设计电路图

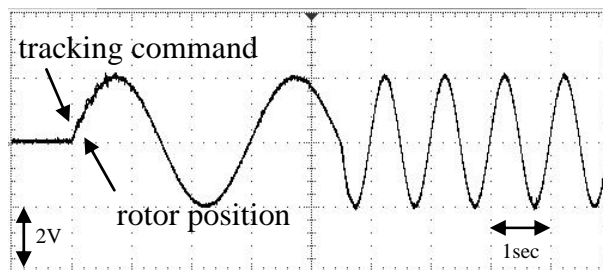


图(七) 软件流程图

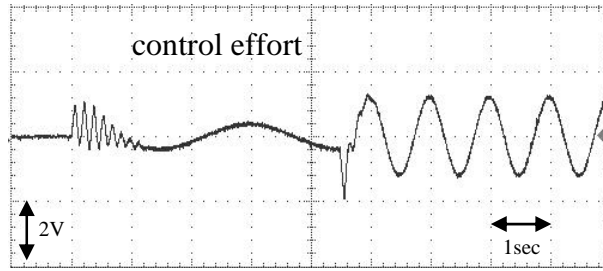
五. 设计方法

Step1: 设计感应电机系统的外围硬件电路，包含1组光学编码器计数电路以及2组数字转模拟电路，其各电路描述如下，所设计的光学编码器计数电路目的在接收光学尺编码气候计数感应电机旋转的角度，其中集成电路MC14584B主要用来将光学编码器的A相和B相信号分别延迟，造成为4倍的分辨率。集成电路SN74159 IC则利用此4倍频信号来判断感应电机为正转或是反转，并以3颗4比特的集成电路SN74193 IC来计数感应电机的旋转角度，如此即可得知电机实际角度。而在数字转模拟电路设计上，本设计选择集成电路AD7237，此集成电路有2个channel，可分别输出0~10V以及-5~5V的电压信号，并利用集成电路74245 IC来防止电流回流至FPGA板。

- Step2: 利用Quartus II软件撰写光学编码器计数电路及数字转模拟电路对应的Verilog HDL程序(电机角度计数模块, 数字转模拟信号控制模块), 并配合外围硬件电路以Quartus II软件进行仿真及实际硬件测试, 再经由波形图及嵌入式逻辑分析仪以验证其功能正确性。
- Step3: 撰写除频电路及控制命令函数生成器模块。以除频电路来控制系统的硬件部分控制周期为 1 msec 。同时先将欲追踪的命令函数值取精确度为小数点后3比特, 并存储于正弦函数内存中。
- Step4: 以SOPC Builder构建感应电机控制系统的Nios嵌入式核心处理器, 包含32比特之Nios II CPU、浮点定制化指令、Nios II 存储元件flash和s dram、Avalon TriState Bridge、system ID Peripheral、jtag uart、Nios II所需的timer、中断所需的timer、提供CPU及s dram CLOCK的PLL, 和电机角度、追踪命令及控制量的Avalon PIO等, 并与硬件程序相结合以完成感应电机控制系统硬件结构。
- Step5: 利用Nios IDE撰写软件程序, 包括外围装置、感应电机的智能型控制法则, 控制程序及主程序, 同时设计频率为 1 kHz 的中断程序。然后由中断程控 1 msec 执行一次感应电机的智能型控制法则, 以计算感应电机输出控制量。
- Step6: 集成外围硬件电路、Verilog HDL硬件程序、Nios II软件程序、及感应电机, 完成基于FPGA的感应电机智能型控制系统。
- Step7: 启动感应电机后, 首先由光学编码器计数电路得到感应电机的实际角度, 并通过FPGA硬件的电机角度计数模块增加比特数为15后和控制命令函数生成器模块所生成的电机角度追踪命令一同送至Nios II CPU, 并在程序进入中断后利用感应电机的智能型控制法则计算电机控制量, 然后传送到数字转模拟信号控制模块, 最后再输出到外部D/A芯片转换成模拟电压以控制感应电机。
- Step8: 利用数字示波器观察控制结果及验证整个感应电机智能型控制系统效能, 实验结果如图(八)所示, 其中图(a)为追踪命令和电机角度, 图(b)为感应电机控制量, 由实验结果中可以看出所提出之方法在控制参数学习过后可以获得不错的响应结果, 而且当控制命令改变时依旧可以获得不错的响应效果, 所以, 本研究所提出的方法可以有效地控制感应电机的旋转角度。



(a)



(b)

Step9: 由于比赛场地的限制, 因此无法展示基于FPGA的感应电机智能型控制系统的结果, 因此在作品展示时将改以控制直流无刷电机来展示硬件效能。但也因为如此, 更能显现及验证了Nios的功能性和方便性, 以及所提出的智能控制系统的适应性, 因为当控制对象不同时, 我们不需要改变任何硬件部份的设计, 而控制器参数亦在线自我学习出来, 并不需做其它调整。

六. 设计特点

本设计结合人工智能学习法则的智能型控制技术开发, 运用于各种不同的实际例子, 藉由运用于感应电机定位控制成果显示出所设计方法的优越性。本设计有如下的特点:

1. 所提出的控制法则的快速学习能力以及强健性: 由于感应电机控制较为复杂且电机参数容易因操作情形不同而改变, 如采用传统的控制方法将无法有效控制, 因此本计划提出一适应性模糊类神经网络控制器, 整个控制系统包含一类神经控制器和一补偿控制器, 其中类神经控制器利用一模糊类神经网络学习近似一理想控制器, 补偿控制器用来确保整个系统稳定性, 而且控制器参数依据李亚普诺夫稳定理论推导出来的学习法则实时自我调整参数, 以致整个闭回路系统的稳定性能够保证收敛。而集成以上两个控制器的感应电机控制系统, 将可实现感应电机控制系统的强健性和精密定位控制的目的。适应性模糊类神经网络控制器能够有可学习的能力, 且能针对外在的干扰即刻调整内部的参数值, 以达到良好的控制效能, 因此即使在一些可靠度需求和实时反应能力高的控制系统上, 也不需要担心控制系统在长久使用后参数改变或是零件故障而控制不好的问题。
2. 采用 FPGA 实现后的效能提高: 近年来, 集成电路制程技术不断的提升, 现场可编程逻辑阵列 (FPGA) 的技术已相当成熟, 近年来也常被应用于各种演算法的硬件化。此外, 系统芯片 (SOC) 的应用已愈来愈多, 因此本设计将所提出的适应性模糊类神经网络控制器以 FPGA 实现, 和以往由计算机来实现相比, 除了可缩小控制系统的体积、提高灵活度及降低成本外, 在系统的计算时间和执行速度上也可大幅的增加。更由于 FPGA 的可重复编程特性, 因此设计者可不断地改变及规划芯片以符合用户所需的功能。

3. 至于程序撰写方面，除了硬件外围电路之程序以 Verilog HDL 来设计外，在控制法则的撰写上，选择以 Nios core 来实现，除了可降低程序的复杂度，更可加快设计以及参数调整的时间。
4. 所完成之 FPGA 控制系统能符合现在业界的需求：本设计利用 FPGA 所完成的适应性模糊类神经网络控制器，可提供低成本、高性能和高可靠度的感应电机控制系统，且在经由些许修改所设计的程序后，便能应用到各种电机的控制上。而因其有体积小、灵活度高、成本低、处理速度快以及生产周期少的特点，同时由于硬件架构采用模块化设计，当应用在不同的控制系统时所需花费的修改部份将不会有太大的困难，所以本设计的成果将能够利用在各种高效的控制应用上，除了各种需在线、实时学习的系统的控制，如工业界和医学界常用的机器手臂的控制外，包括了家用机器人控制系统、汽车的倒车至定点控制、自动驾驶控制系统以及数字式轮椅控制的应用皆有很大的发展空间。因此相信在许多领域上，包括工业界及医学界等都可以有重大的贡献。

七. 总结

虽然在这次的比赛之前，参与学生已经对Verilog HDL语法及Quartus II软件有了初步的认识，但并不算是非常熟悉，而在Nios的部分则是完全不懂，但在参加此次的比赛的过程中，除了更加强学生在硬件设计的能力，同时也发觉Nios所提供的强大功能，不但能加快硬件部分设计的时间，同时也能达到更简化及方便的设计方式，使得本参赛团队可以利用许多工具，如嵌入式处理器、浮点定制化指令、内存等快速且容易的规划所需要的硬件。由此次参赛作品中可得知FPGA运用在控制的实际效能。另外，也因为Nios IDE的人性化界面，让我们能轻松的撰写、编译及执行我们的程序。这不仅仅可以对学生在之后的研究可以有非常大的帮助，同时也能增加参与学生将来往业界发展所必须具备的实力和竞争力。