

基于双 NIOSII 的脱丁烷塔控制系统(二) ——稀释合成釜控制系统

牛思文, 牛爱杰

(北京化工大学 信息科学与技术学院 北京 100029)

摘要: 介绍了基于 ALTERA 公司 FPGA 的 NIOSII 双软核处理器在脱丁烷塔工业流程——稀释合成釜系统的应用。两颗高性能 32bit 的 CPU 与全部的配置模块同处于 Altera Cyclone II FPGA 芯片中, 分别负责外设控制与算法控制, 使得该控制系统与同类解决方案相比, 在各项设计成本显著降低的同时, 安全性和抗扰动能力大幅提升。

关键字: NIOSII; FPGA; mutex; 有限长单位脉冲响应; 内模控制算法

The control system of Polymer Caldron with Dual-NIOSII

NIU Si-wen, NIU Ai-jie

(Beijing University of Chemical Technology, Information school, Beijing 100029, China)

Abstract: We described the application of Altera's NIOSII (Dual) Soft-Core CPU (Based on CycloneII FPGA) on the chemical industrial equipment—polymer caldron. Since multi-NIOSII CPUs lie in one chip of FPGA, and have implemented general and arithmetic process separately, the whole system can consume lower budget while gaining more safety and stability, comparing other similar solutions.

Key Word: NIOSII; FPGA; Mutual Exclusion Unit; Finite-Duration Impulse Response (FIR) Algorithm; Internal Modal Control (IMC) Algorithm

0 引言

在工业塔釜系统和其他工业设备控制环节中, 控制系统的可靠性与抗扰动性是最重要的评价指标。作为脱丁烷塔系统的重要下游应用环节, 稀释聚合釜设备将成品导入装有稀释溶剂的容器中进行稀释, 并加入稳定剂进行初步聚合反应。这个过程的控制重点是浓度指标。使用昂贵的 PLC 控制器则造成了资源的浪费; 若使用 8 位单片机抑或 32 位的 arm 芯片, 由于设计中引入了大量的校正、容错算法, 其单芯片有限的运算速度会成为算法运行中的瓶颈。而传统的双芯片策略如 ARM+DSP 等, 由于工业现场相对恶劣的运行环境, 其跨芯片信号传输产生的随机扰动、延迟等问题都限制了控制性能的提高。本设计充分考虑到了以上问题, 应用 ALTERA 公司最先进的 SOPC 技术, 提出了一种运算处理能力和工业安全性兼得的解决方案。通过高性能 IMC-PID (内模 PID) 算法的仿真测试, 证明该方案具有稳定的实时控制效能。

1 塔釜结构分析与 NIOSII 软核处理器简介

在脱丁烷产业链中, 丁烷 (及其异构体) 的稀释聚合是一个非常重要的下游加工环节, 也是本设计的主要控制对象。其设备组成与控制模型如图 1 所示。图右侧的稀释聚合单元中, 浓度控制为主, 另辅有液位控制, 温度控制等安全辅助控制部分。长时间的非室温操作, 控制器的离地安置, 粉尘及物料挥发等客观因素, 都对控制器芯片提出了严峻的要求。本设计选用 Altera 的 Cyclone II 可编程逻辑器件, 除了可以显著降低控制系统制造成本外, 其工业级的工艺标准也是一个重要的考虑因素。

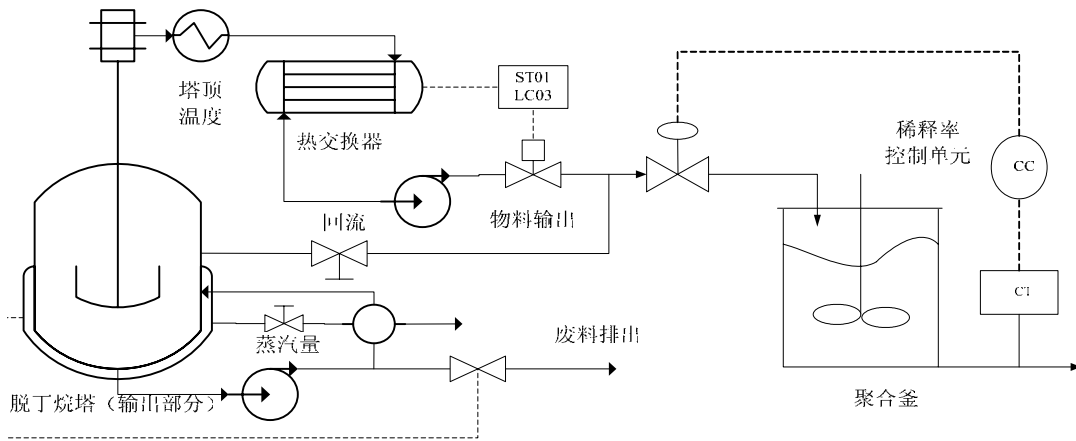


图1 脱丁烷塔下游——稀释聚合系统辨识用建模图

NIOSII是由硬件描述语言编写的基于FPGA的软核MCU，是Altera公司SOPC策略的优秀成果^[1]。在本设计中，NIOSII嵌入式处理器倚仗先进的CycloneII器件，不仅提供高于普通32位MCU近2倍的运算性能、更低的逻辑资源占用，还提供了高度开放的SOPCBuilder人机界面，使其在针对具体设备进行扩展时拥有同类产品不具备的系统级灵活性与可重用性。它拥有32位指令集，32位数据线宽度，32个通用寄存器，大量片上RAM资源；硬件断点，数据触发，以及自定义逻辑等特性亦可随需要在SOPCBuilder中便捷地加入。其结构示意图见图2。

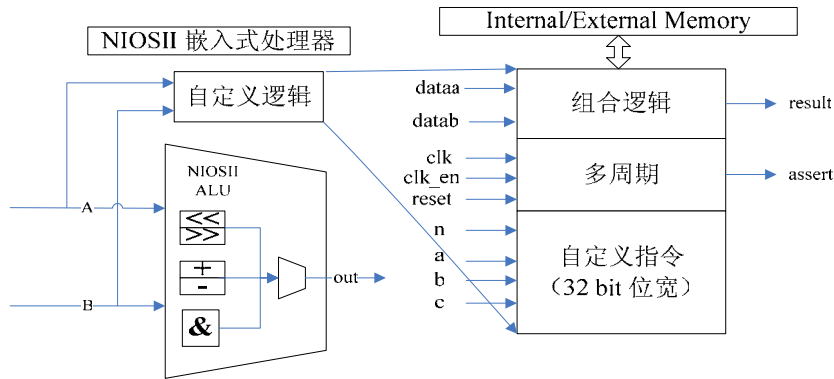


图2 NIOSII 软核CPU 结构图

2 系统总体设计方案

本控制系统采用双 NIOSII 软核处理器来对设备进行控制。由于两个 CPU 都是基于同一片 FPGA，因此其信号传输的延迟等于内部路径寻址的耗时 (<5 ns)，可以认为无延迟。双 CPU 担当的任务分别是人机交互和算法实现环节。负责算法的 CPU1 不承担其他的附加任务；另一个 CPU2，承担算法以外的部分，如数字键盘和对主机的 485 传输，数码管显示等，并应用 UC/OSII 操作系统进行任务的调度^[2]。当发生操作不当或通过 RS485 传输信息不当等情况，系统内置的校验环节和纠错单元会立即开始运作；而系统出现严重异常时，CPU1 可以在时钟上升沿到来时跳转到紧急状态，按照编制好的程序将危险因素尽可能地消除或延迟。这样就使得控制系统具备了相当的抗扰动能力。芯片布局图见图 3。

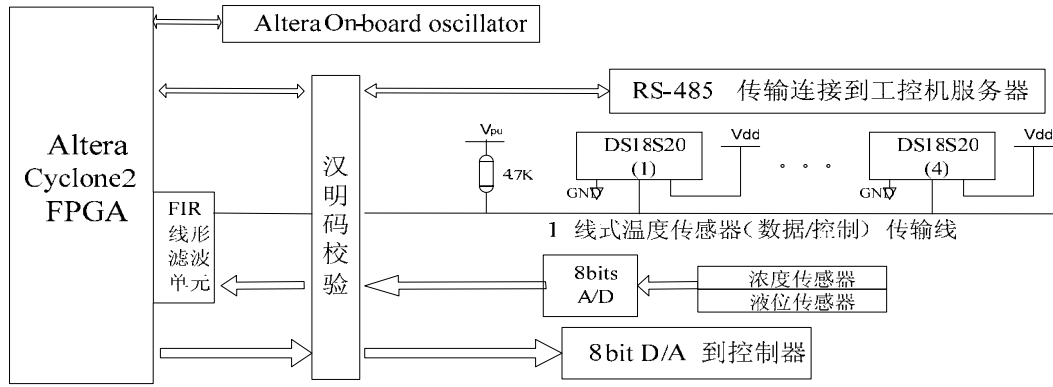


图3 FPGA 外部板面布局图

3 硬件电路设计

3.1 NIOSII 处理器与 Mico Bus 总线连接图

NIOSII 处理器与外部逻辑的 Avalon 总线接口已经存在,但专就工业应用中多核 MCU 的控制信号指令调度而言,设计一条跨 MCU 的专用指令线是十分必要的。因此,尝试开发了一条专用指令存取与传输总线(图 4),这样可以满足多核之间,MCU 与 RAM 之间共享、协作的各种需求,并为未来的发展打下基础。

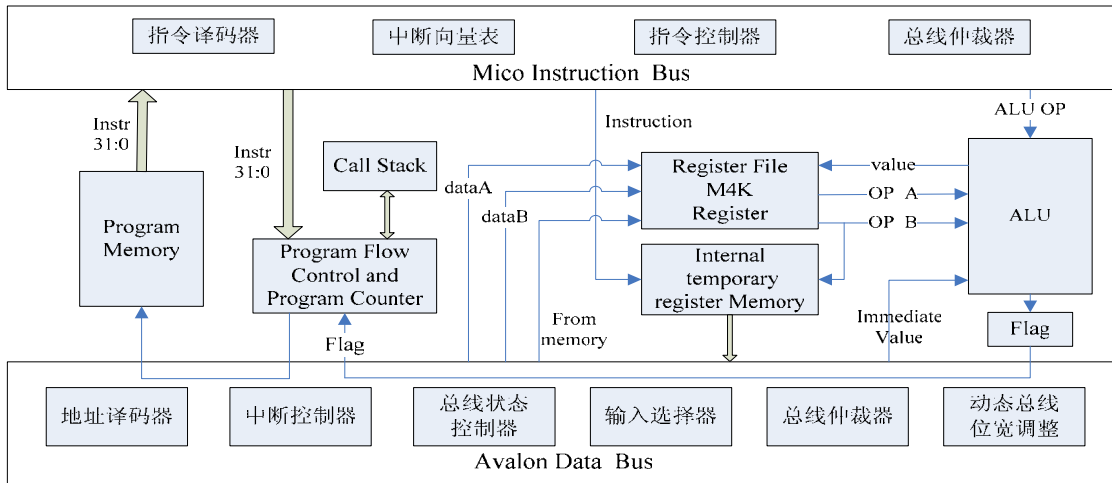


图4 NIOSII 处理器与 MicoBus 指令总线连接图

3.2 FPGA 内部模块布局及传感器收发单元设置

算法以及工作代码存储在内部配置器件中,在上电时被读入到 RAM 中执行。控制系统和主机服务器之间采用高稳定性的 RS-485 传输方式,温度传感器采用 Dallas 公司的 1 线式温度传感器。485 协议采用集成在 FPGA 内部的 IP 核,进一步降低了布线的扰动。内部模块配置图见下图 5。Avalon、Mico 两条总线,分别担任控制数据(data)传输和控制指令 (instr) 的传输。

双处理器分工明确,所有外设都通过 Avalon 总线与 CPU 进行交互。传感器和 A/D 转换器,以及 D/A 转换器由 CPU (P1) 管理,这样当 CPU2 (P2) 出现故障时,可以断开两个 CPU 之间的联络通道,由处理器 CPU1 独立运行,在一定时间内保证设备的安全。

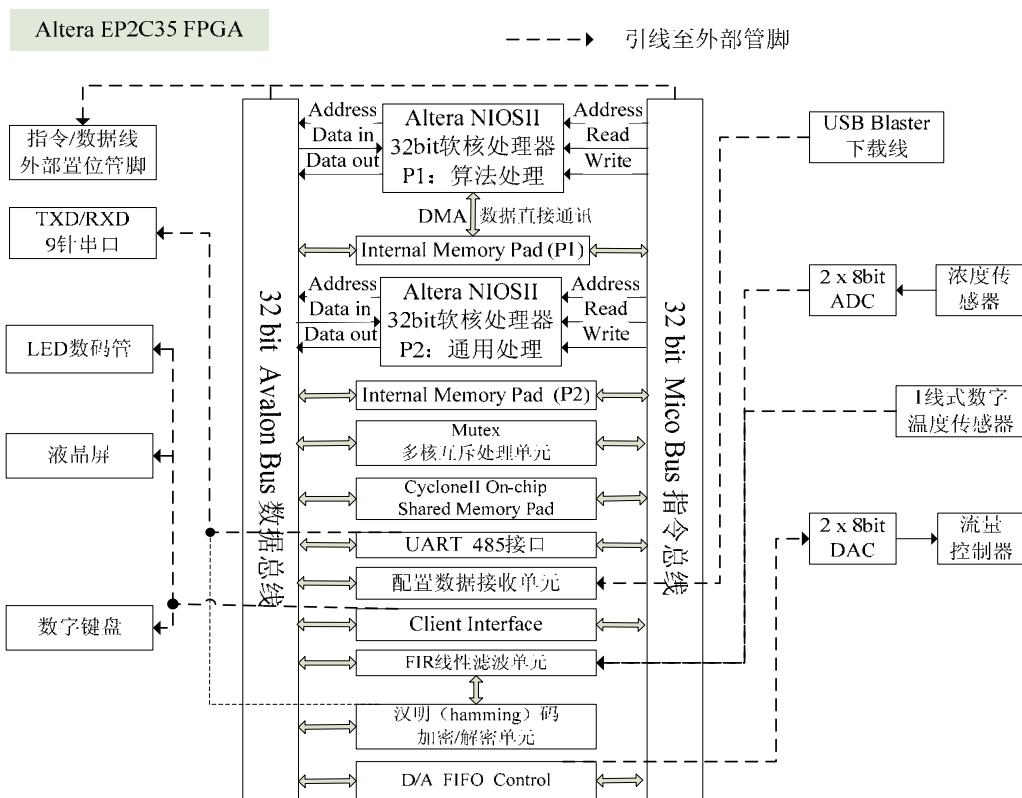


图 5 双 NIOSII 处理器的 FPGA 内部模块配置图

在数据传输环节，加入了 FIR 缓冲校验模块。这样做是因为工业塔釜在运行时，由于物料含有杂质等原因，传感器常常会在瞬时出现偏差较大的测量值，在设计中应对此进行及时处理。校验缓冲模块是由 Verilog 编写的 IP 核，每当有检测变量输入或控制变量输出时，检测该数据是否和历史数据波动相差过大；如果是，则应用有限长单位脉冲相应（FIR）算法进行线性滤波（见 4.1 节详述）。

3.3 双 NIOSII 处理器数据/地址指令的交互模块与 mutex 互斥单元

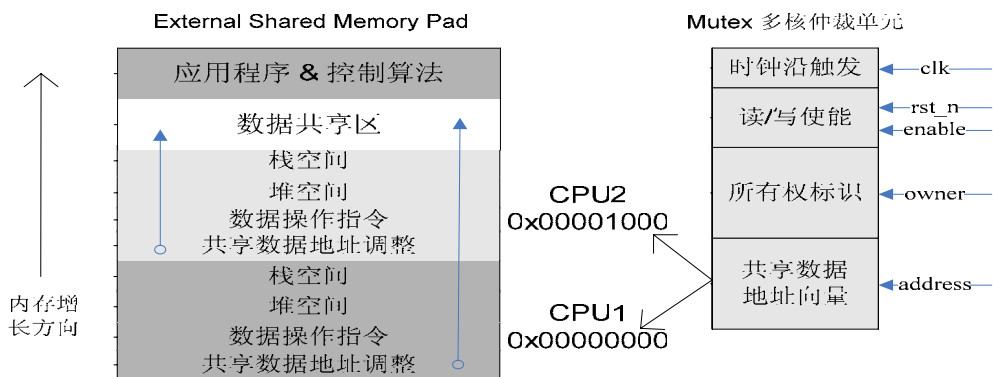


图 6 双处理器对内存空间的使用

本设计充分利用了 Cyclone II 芯片大容量 M4K 内存的巨大优势，应用 on-chip M4K 作为共享内存。双 CPU 均与此 RAM 交互。处理器间的信息交互采用非常稳定的共用交换区内存 (Shared Memory) 的方式 (见上图 6)。RAM 被划分成 4 片可用区间，即两个处理器由地址偏移开始的空间和应用程序被载入时的存储区和共享内存区。由于 CPU1 专门负责算法，本设计将其在内存中的起始位置放在了 0 偏移处，以保证最简便的寻址路径。

表 1 mutex 互斥单元结构

偏移量	寄存器名称	读/写	位描述符 (31 ~ 16)	位描述符 (15 ~ 1)	位描述符 (0)
0	mutex	RW	OWNER	VALUE	VALUE
1	reset	RW	N/A	N/A	RESET

互斥信号量 (mutex) 是解决多处理器争用存储器的优秀解决方案 (见表 1)。Altera 文档^[1]中的推荐范例操作如下：首先将 VALUE 值清零，RESET 置位，使能 mutex。CPU1 在时钟沿的激励下，发出对共享内存的读写需求，这时如果 mutex 标识为可用，并且其 OWNER 值等于 CPU1 的 ID，则获使用权；操作完成后将 OWNER 变成 CPU2 的 ID，再将 VALUE 清零。下一个时钟到来时 CPU2 将自己的 ID 与 OWNER 相比较，相同则获得使用权，然后依据地址寄存器对共享内存进行寻址。在双内核的应用中，使用 mutex 方式进行数据的共享是最为有效率的，因为数据在共享内存中没有任何移动，移动的是地址标识寄存器，显然这个开销是十分小的。在多核 (大于等于 3) 应用中，就必须使用 mailbox，即多 mutex 进行操作。

4 各模块详细设计说明

4.1 有限脉冲响应 (Finite Impulse Response - FIR) 滤波器算法^[3]

FIR 算法是应用较广泛的滤波算法，常常用于数字系统设计中，对输入信号进行线性化调整。这里应用 FIR 算法对接收的测量值进行调整，以避免瞬时采样的剧烈变化使监控单元发生误报^[4]。一个 FIR 系统的脉冲响应是有限的周期；系统的输出是目前输入的加权以及先前输入的加权和的总和。而输入经过有限个 N 步后，将不再输出。公式为：

$$y_{FIR}[n] = \sum_{k=0}^N a(k)x[n-k]$$

Verilog 核心实现代码如表 2，FIR 算法的仿真测试结果如图 7。

表 2 FIR 硬件算法 (Verilog)

代码 (变量声明)	代码 (算法)
<pre> module FIR_Gaussian_Lowpass (Data_out, Data_in, clock, reset); parameter order = 8; parameter word_size_in = 8; parameter word_size_out = 2*word_size_in + 2; output [word_size_out - 1: 0] Data_out; input [word_size_in - 1: 0] Data_in; input clock, reset; parameter b0 = 8'd7; // Filter coefficients parameter b1 = 8'd17; parameter b2 = 8'd32; parameter b3 = 8'd46; parameter b4 = 8'd52; parameter b5 = 8'd46; parameter b6 = 8'd32; parameter b7 = 8'd17; parameter b8 = 8'd7; reg [word_size_in - 1: 0] Samples [1: order]; integer k; </pre>	<pre> assign Data_out = b0 * Data_in + b1 * Samples[1] + b2 * Samples[2] + b3 * Samples[3] + b4 * Samples[4] + b5 * Samples[5] + b6 * Samples[6] + b7 * Samples[7] + b8 * Samples[8]; always @ (posedge clock) if (reset == 1) begin for (k = 1; k <= order; k = k+1) Samples [k] <= 0; end else begin Samples [1] <= Data_in; for (k = 2; k <= order; k = k+1) Samples [k] <= Samples[k-1]; end endmodule </pre>

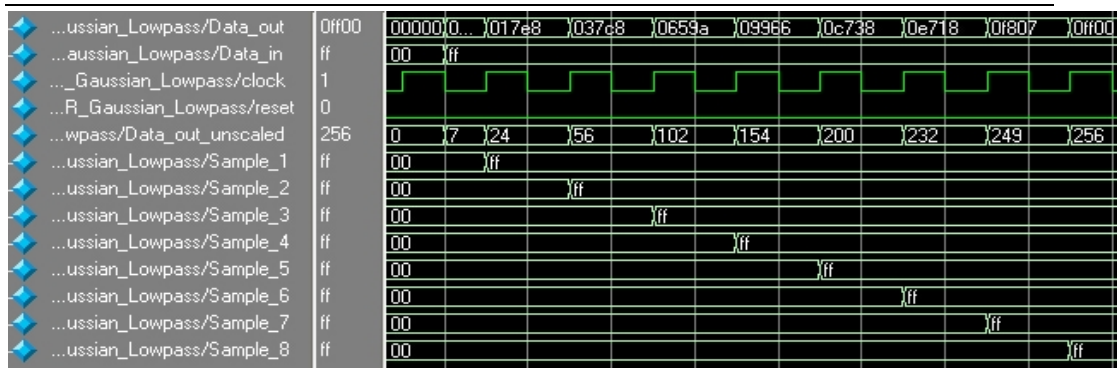


图 7 FIR 算法的仿真测试结果 (ModalSim 6.1d)

4.2 改进的高速卷积 (Convolution) 控制算法

卷积算法是本设计中的另一个重要算法，是一种应用广泛的信号处理方法，在大信噪比条件下，它可以获得很高的测量精度和较高的有效信号分辨率。计算机离散化后的公式为：

$$y[j] = \sum_{k=0}^j x[k]h[j-k], \text{ 其中 } j=0,1,2 \dots 2N-2. \quad (1)$$

如果使用 C 语言进行软件实现，当 N=3 时，则要使用 25 个运算周期来完成一次卷积运算。为了提高控制系统的实时性，决定使用 FPGA 的硬件逻辑来实现并优化该算法，并作为 IP 核集成到系统中。借助 Altera 的高性能 FPGA，我们设计了全新的高速硬件架构。利用 3 个并行的 ALU，在控制逻辑下，将 25 步处理周期缩短为 9 周期，效率提升了 280%；而对于大容量的新型 Cyclone II 器件，资源消耗的增加则完全可以忽略不计。

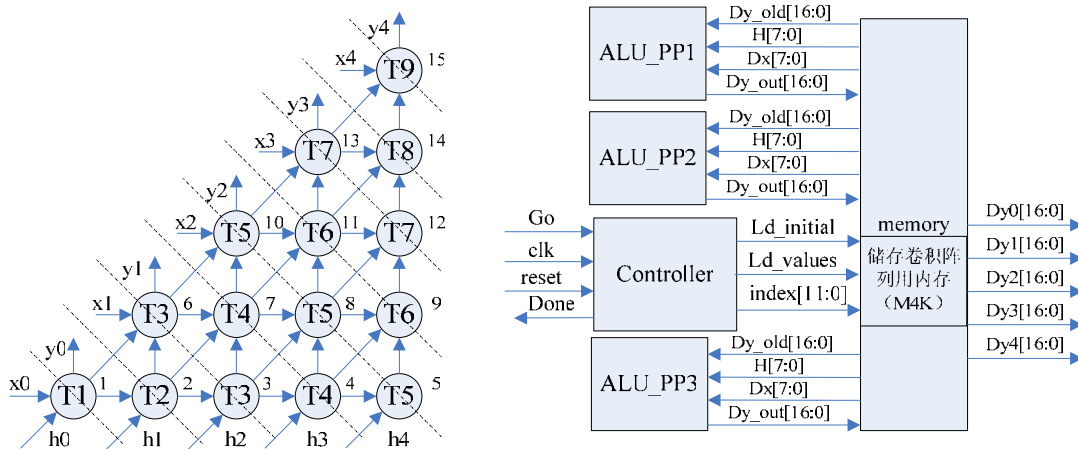


图 8 告诉卷积算法示意图 (左) 和硬件结构图 (右)

图 8 (左) 中所示的圆形为节点，内部的数字指示将在哪一周期的执行运算；外面的数字为该节点在整个节点阵列中的序号。输入 x 在左侧，脉冲响应 h 在下侧。图 8 (右) 为该模块的结构示意图。M4K 存储单元响应指令信息 Ld_initial 和 Ld_values, 以及配置信号 index。

表 3 三个 ALU 的任务分配表

	T1	T2	T3	T4	T5	T6	T7	T8	T9
ALU_PP1	1	2	3	4	5	9	12	14	15
ALU_PP2			6	7	8	11	13		
ALU_PP3					10				

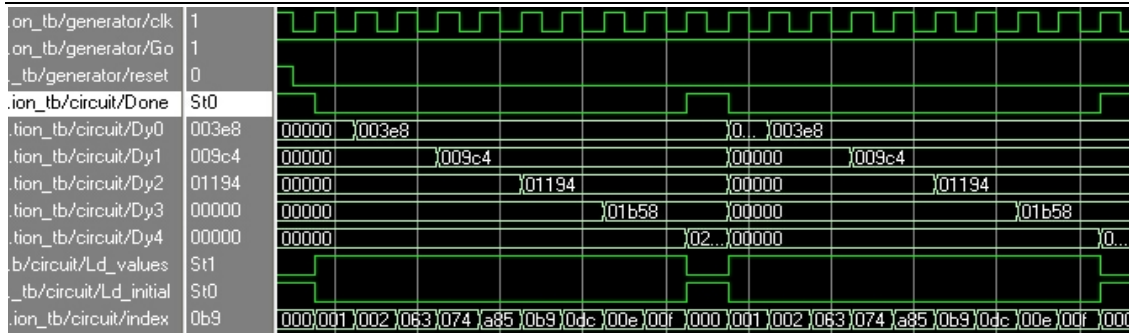


图 9 高速卷积算法仿真图

表 3 为 9 周期内三个 ALU 的任务分配表。三个 ALU 为纯组合逻辑。控制器在时钟源的激励下设置 12 位指令总线 index。总线每 4bit 标识一个 ALU，以确定某个 ALU 是否使用和将哪个节点分配给它进行运算。例如，当 $index = \{4'd0, 4'd0, 4'd0\}$ ，3 个 ALU 处于空闲状态； $index = \{4'd0, 4'd0, 4'd1\}$ ，第一个节点赋给了 ALU_PP1，而另两个处于空闲状态；而当 $index = \{4'd10, 4'd8, 4'd5\}$ 时，三个 ALU 均在执行任务（节点 5，节点 8 和节点 10）。图 9 是在 ModalSim6.1d 下的仿真结果。从图 9 中可以明显看出，指令信号(index)在每个时钟的上升沿进行调整，而整个处理过程在 9 周期内处理完毕，并得到了预期的结果。

4.3 汉明码(hamming code)安全校验算法^[3]

考虑到工业现场的复杂性，为了防止数据线由于振动、辐射、导电粉尘和温度骤变等外部扰动导致的部分比特位异常翻转，本设计采用了高安全性的汉明码校验算法。汉明码是一个错误校验码集，利用奇偶校验位的概念，通过在数据位后面增加冗余比特来验证数据的有效性。汉明码不仅可以验证数据是否有效，还能在部分数据出错的情况下自动修复错误位。

推求汉明码时的基本考虑是确定所需最少的监督位数 k。长度为 m 位时，若附加了 k 个监督位，则所发送的总长度为 m+k。在接收器中要进行 k 个奇偶检查，每个检查结果或是真或是伪。这个奇偶检查的结果可以表示成一个 k 位的二进制数，它可以确定最多 2k 种不同状态。这些状态中必有一个其所有奇偶测试都是真的，它便是判定信息正确的条件。于是剩下的 (2^k-1) 种状态，可以用来判定误码的位置。于是导出如下关系： $2^k-1 \geq m+k$ 。汉明码纠错算法在 Synplify 中的综合结果（顶层模块）如图 10 所示， ModalSim 仿真结果如图 11：



图 10 汉明码(hamming code) 加密/解码单元综合图 (Synplify 8.4)

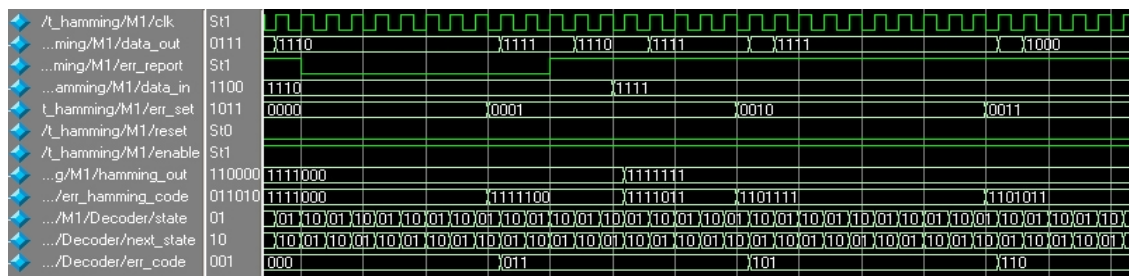


图 11 汉明码(hamming code) 加密/解码单元仿真测试 (ModalSim 6.1d)

本设计中对于 A/D 和 D/A 环节大量运用了这种高效的纠错算法，再次突出了安全型设计原则。以稀释度检测单元为例，传感器电路将测定数据加密后进行传输，NIOSII 控制器收到数据后立即通知由 Verilog 定制好的 IP 核进行解码。图 10 中数据位为 4bit，因此增加了 3bit 校验码。在测试中人为对数据施加了扰动，如图中左边栏 err_set 所示。data_in 为传感器输出，hamming_out 为加密后输出，data_out 为解码后的最终数据。经测试，当 err_set 只有 1 位置 1 时，data_out 与 data_in 完全相同，即对 1bit 错误数据进行了自动修复。在 50MHz 的频率下，最长修复时间仅为 0.06μs；同时，对 2 位数据出错可以完全正确报告，对多位数据出错的报告精度高达 90%。这大大提高了数据传输的安全性和容错能力。

4.4 A/D、D/A 转换用 FIFO 缓存单元

过程设备的控制器需要进行实时的大批量数模/模数转换操作。4-20mA 工业标准电流信号转换成数字量后，必须进行缓存，然后在时钟上升沿被读入内存进行下一步操作。缓存单元的存在对系统稳定性具有重要作用，它保证了转换后的数值可以在 MCU 需要时被读入，避免出现数据到来过早产生的信息丢失或到来过晚导致的系统等待延迟。

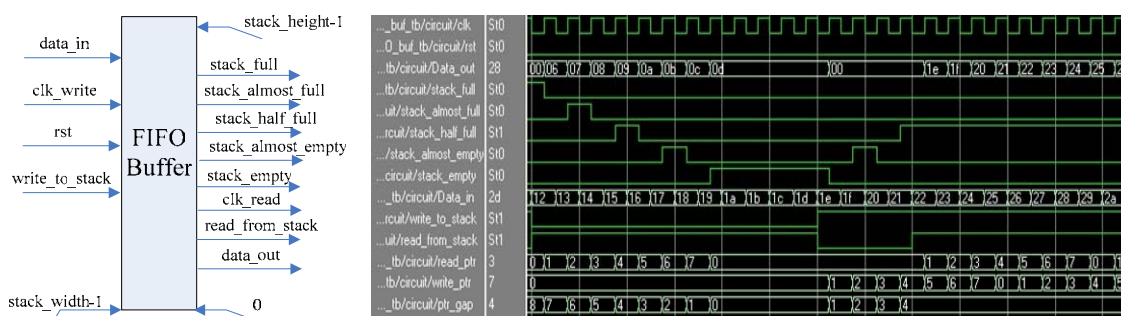


图 12 FIFO 结构图(左) 和仿真图(右)

设计中为每一个 A/D、D/A 转换单元都分配了一个 FIFO 缓存阵列，如图 12（左）所示。图 12 右，位宽为 8×8bit 时的 ModelSim6.1d 仿真图形。经海量存取测试，达到了设计要求。FIFO_buffer 的 Verilog 源代码见下表 4。

表 4 FIFO Verilog 源代码

核心代码	
<pre> assign stack_full = (ptr_gap == stack_height); assign stack_almost_full = (ptr_gap == AF_level); assign stack_half_full = (ptr_gap == HF_level); assign stack_almost_empty = (ptr_gap == AE_level); assign stack_empty = (ptr_gap == 0); integer i; //change to the following code when simulation always @(posedge clk or posedge rst) begin //always @(posedge clk or posedge rst) begin if(rst) begin Data_out <= 0; read_ptr <= 0; write_ptr <= 0; ptr_gap <= 0; for(i=0; i<stack_height; i=i+1) begin stack[i] <= 0; end end else if(write_to_stack && (!stack_full) && (!read_from_stack)) begin stack [write_ptr] <= Data_in; Data_out <= 0; write_ptr <= write_ptr + 1; ptr_gap <= ptr_gap + 1; end </pre>	<pre> else if(!write_to_stack) && (!stack_empty) && read_from_stack) begin Data_out <= stack [read_ptr]; read_ptr <= read_ptr + 1; ptr_gap <= ptr_gap - 1; end else if(write_to_stack && read_from_stack && stack_empty) begin stack [write_ptr] <= Data_in; Data_out <= 0; write_ptr <= write_ptr + 1; ptr_gap <= ptr_gap + 1; end else if(write_to_stack && read_from_stack && stack_full) begin Data_out <= stack [read_ptr]; read_ptr <= read_ptr + 1; ptr_gap <= ptr_gap - 1; end else if(write_to_stack && read_from_stack && (!stack_full) && (!stack_empty)) begin Data_out <= stack [read_ptr]; stack [write_ptr] <= Data_in; read_ptr <= read_ptr + 1; write_ptr <= write_ptr + 1; end end </pre>

5 控制程序流程图

控制程序主要分为三部分，即控制器初始化，控制程序执行和故障处理环节（见图 13）。

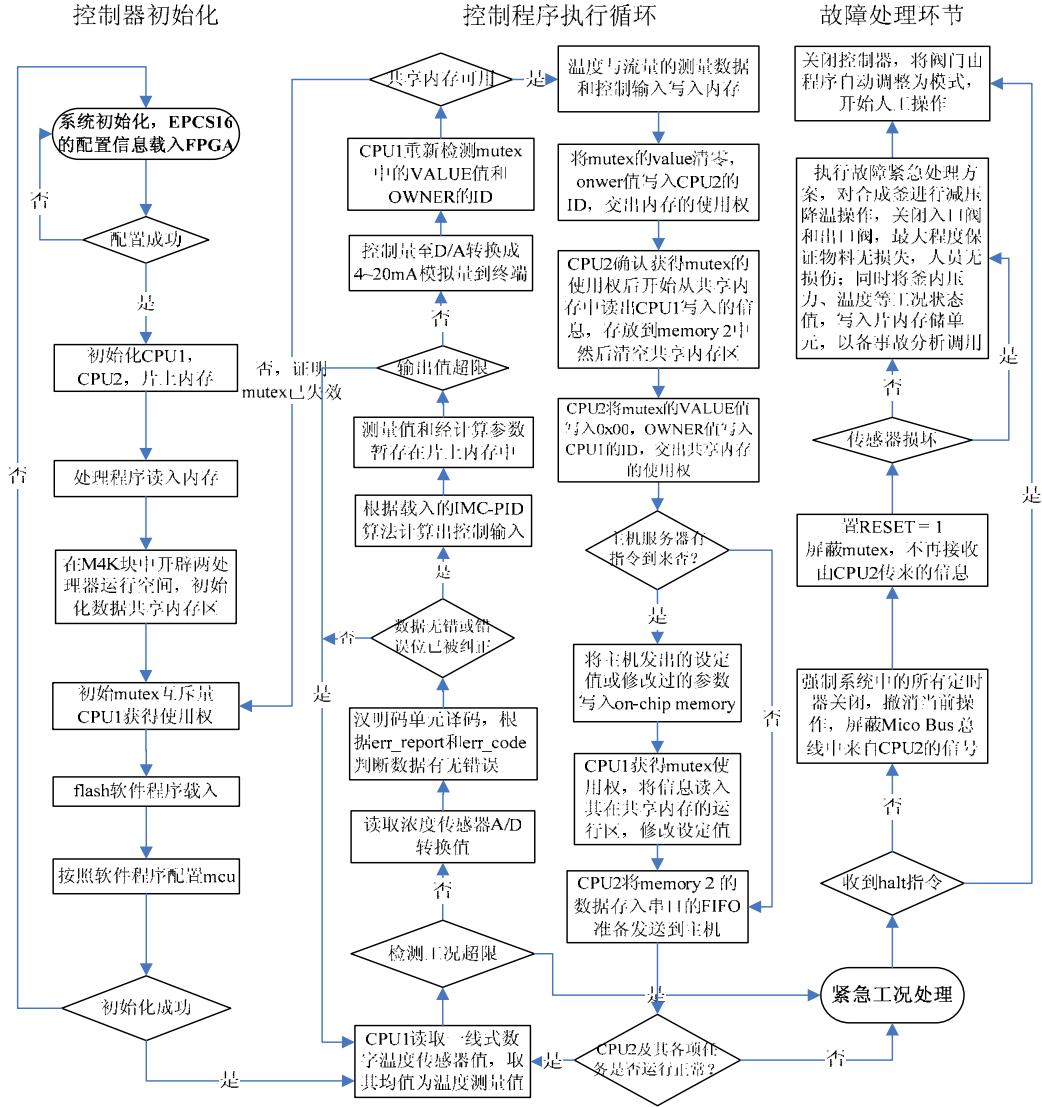


图 13 稀释合成釜控制系统软件程序流程图

6 IMC-PID 内模比例积分微分控制算法^[5]

内模控制 (Internal Model Control, 简称 IMC) 是一种基于过程数学模型进行控制器设计的新型控制策略。由于设计便捷、控制性能优异及其在系统分析方面的优越性，使得内模控制不仅是一种实用的先进控制算法，更是研究预测控制等基于模型控制策略的重要理论基础，以及提高常规控制系统设计水平的有力工具。本设计的重点是控制合成釜的稀释聚合，因此主要以稀释率的控制进行说明。该设备对控制器的系统辨识模型为：

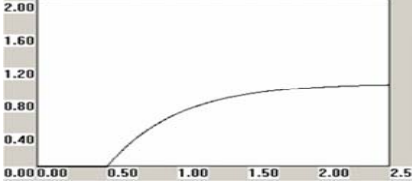
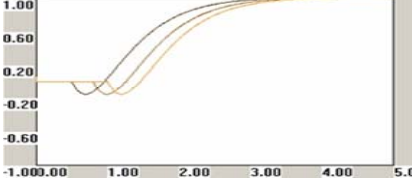
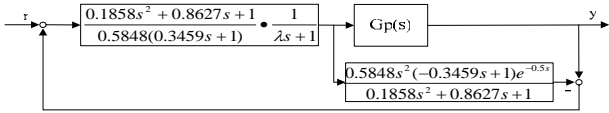
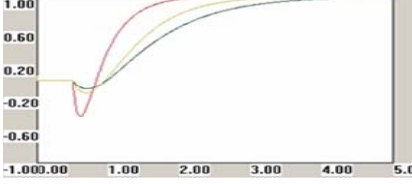
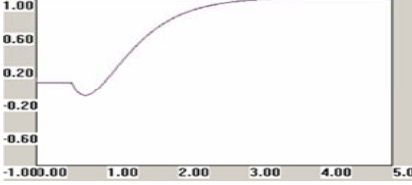
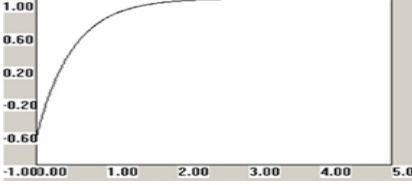
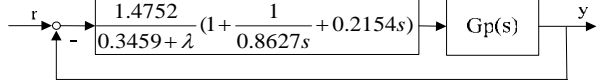
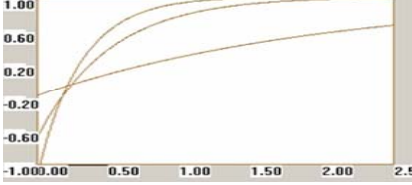
$$\tilde{G}_p(s) = \frac{0.5848(-0.3459s + 1)e^{-0.5s}}{0.1858s^2 + 0.8627s + 1} \quad (2)$$

按照内模控制原理设计带有滤波器的内模控制器。在反馈通道中插入滤波器，可起到抑制干扰的作用。反馈滤波器结构和参数的选取，既要保证模型失配时的闭环稳定性和零稳态偏差特性，还要考虑系统的抗干扰性和鲁棒性。滤波器传递函数如下，仿真结果见表 5：

$$G_{IMC}(z^{-1}) + G_f(z^{-1})[G_p(z^{-1}) - \hat{G}_p(z^{-1})] = 0 \quad g(z^{-1})[1 - z^{-1} + (1 - a_f)z^{-2}] = 0 \quad (3)$$

指令集仿真（表 3）

表5 测试结果与说明

控制类型	控制曲线	测试结论
含RHP零点在内的IMC		$G_{IMC} = \frac{0.1858 s^2 + 0.8627 s + 1}{K(-0.3459 s + 1)} \cdot \frac{1}{\lambda s + 1}$ <p>当无模型偏差时控制因子 G_{IMC} 如上式，取 $\lambda = 0.5$，系统的输出为 $y(t) = 1(t - 0.5) - e^{-(t-0.5)/\lambda}$。系统的输出是有界的(曲线<1>)。</p>
不含RHP零点的IMC（调整死区）		 <p>当给定值发生单位阶跃变化后，此时系统的输出：</p> $y(t) = 1(t - 0.5) + \frac{0.3459 + \lambda}{0.3459 - \lambda} e^{-\frac{t-0.5}{\lambda}} - \frac{0.6918}{0.3459 - \lambda} e^{-2.891(t-0.5)}$ <p>控制曲线<2>的图形死区时间为：0.5, 0.8, 1。由此可以看出死区时间对系统输出的影响。</p>
不含RHP零点IMC（调整滤波器）		<p>曲线<3>是保持死区时间不变，滤波器常数分别为:0.1, 0.5 和 0.8 时，系统的输出。可以看出：滤波器常数变小，系统响应速度提升，同时震动加剧。</p>
不含RHP零点的IMC（调整增益）		<p>曲线<4>是当增益为：0.5848, 0.8, 2 时的输出。由图可知：三个图形重合在一起，所以理想状态下，系统增益对系统的输出没有影响。</p>
IMC-PID（模型无偏差）		 <p>在此IMC-PID控制器作用下，系统的输出为：</p> $y(t) = 1 - \frac{0.3459 + \lambda}{\lambda} e^{-t/\lambda}$ <p>非最小相位系统如曲线<5>，我们可以看到：系统的输出从负值开始，慢慢趋于给定值。</p>
IMC-PID（调整滤波器）		<p>滤波器常数从上到下依次为：0.3, 0.5, 2。从曲线<6>可以看出，滤波器常数越小，系统输出响应越快，震动加剧。因此，适当地选取死区和滤波器常数，IMC-PID 算法可以有效地将合成釜系统控制在最优化状态。</p>

结束语

经过仿真验证,应用双NIOSII组成的控制单元,以及片上高速硬件算法,借助Altera新型可编程逻辑器件Cyclone II的强大性能,实现了预期的控制效果,并使得将复杂的控制、运算逻辑集成在单一低成本芯片上的现代化设计成为了可能。

Altera FPGA 专用高速EPCS配置单元的特点,在电路板绘制时会节约大量的板面空间,同时减少了布线扰动可能性。其低成本Cyclone II器件对于运行环境非常苛刻的工业级应用来说,是非常适合的,也是同类竞争产品难以比拟的。我们相信,凭借Altera雄厚的技术实力,会逐步增强NIOSII的相关功能,并提供更高级的内管理单元,在SOPC方面继续担当领军角色。

参考文献

- [1] NIOSII Processor Reference Handbook. Altera Corporation, June 2006.
- [2] JEAN. J. LABROSSE. MicroC/OS-II The Real-Time Kernel[M].第2版. 北京:北京航空航天大学出版社, 2003, 15-138.
- [3] SAMIR PALNITKAR. Verilog HDL 数字设计与综合[M]. 北京:电子工业出版社, 2006年2月, 133-135.
- [4] MARY MCCARTHY. Peak-to-Peak Resolution Versus Effective Resolution[J]. Application Note AN-615. Analog Device Inc.2003.
- [5] RICHALET J. Model predictive control: application to industrial process[J]. *Automatica*, 1978, 14(4): 400-435.

作者简介:

牛思文(1982-),男,北京市,硕士,北京化工大学信息学院,研究生,研究方向为过程仿真软硬件技术。
牛爱杰(1953-),男,北京市,工程师,北京化工大学材料科学学院。导师。

原创性声明:

本设计是实验室开发人员在导师的指导下自主研发的项目成果,文章图表、文字等内容均没有仿照或抄袭任何第三方已有方案,特此声明。

联系方式:

姓 名: 牛爱杰
电 话: 13661042850/01064436589
地 址: 北京市朝阳区樱花东街9号楼6门
单 位: 北京化工大学
邮 编: 1 0 0 0 2 9