UG464
User Guide
PI Controller v4.1

Microsemi
Power Matters.™

Power Matters.™

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
E-mail: sales.support@microsemi.com
www.microsemi.com

© 2016 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.

Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.

About Microsemi

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 4,800 employees globally. Learn more at www.microsemi.com.

50200464.2.0 / 10.16

Contents

Figures

Tables

1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 2.0

In the revision 2.0 of the document, configurable bit-width for input and output ports are removed.

1.2 Revision 1.0

Revision 1.0 was the first publication of the document.

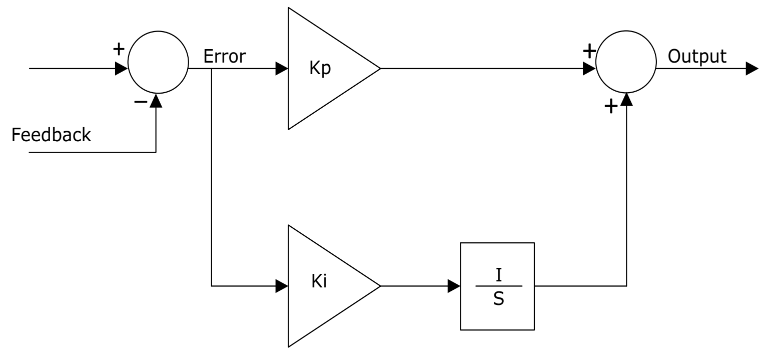
2 PI Controller Theory

2.1 Introduction

PI controller is the widely used closed loop controller for controlling a first order system. The basic functionality of a PI controller is to make the feedback measurement track the reference input, which is done by controlling its output till the error between reference and feedback signals becomes zero.

There are two components that contribute to the output, the proportional term and the integral term as shown in the following figure. The proportional term is only dependent on the instantaneous value of the error signal whereas the integral term is dependent on the present and previous values of error.

Figure 1 • **PI Controller in Continuous Domain**



Pi controller in continuous time domain can be expressed as:

$$y(t) = k_p \times e(t) + k_i \times \int_0^t e(\tau)d\tau$$

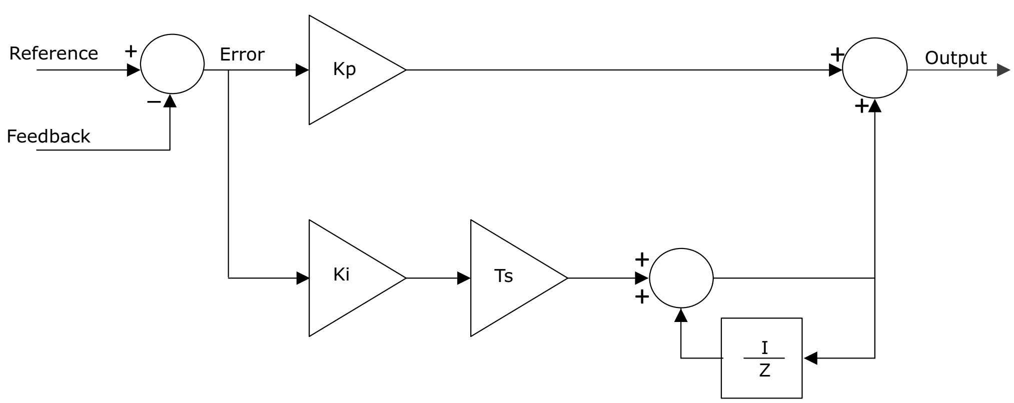
where,

e(t) = reference(t) - feedback(t) is the error between Reference and feedback

y(t) = PI controller output

To implement the PI controller in digital domain, it has to be discretized. The discretized form of PI controller based on zero order hold method is shown in the following figure.

Figure 2 • **The Discretized Form of PI Controller**



$$P(n) = K_p \times e(n)$$

$$I(n) = Ki \times Ts \times e(n) + I(n-1)$$

$$Y(n) = P(n) + I(n)$$

where,

P(n) = Proportional term output

I(n) = Integral term output

I(n-1)=Previous (buffered) value of Integral output

Ts = Sampling time in discrete domain

2.2 Anti Windup and Initialization

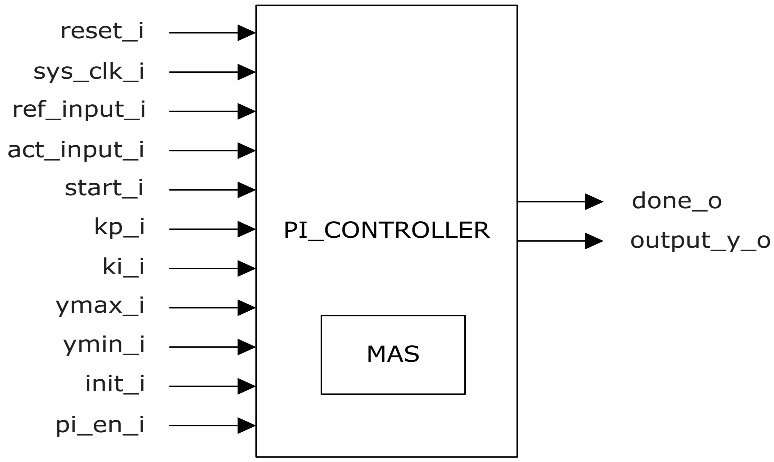
The PI controller has maximum and minimum limits for its output, so as to keep it within practical values. If a non-zero error signal persists for a long time, the Integral component of the controller keeps increasing and may reach a value limited by its bit width. This phenomenon is called integrator windup and has to be avoided to have proper dynamic response. The PI controller IP has an automatic anti-windup function which limits integrator as soon as PI controller reaches saturation.

In certain applications like motor control, it is important to initialize the PI controller to a proper value before enabling it. Initializing the PI to a good value avoids jerky operation. The IP block has a enable input to enable or disable the PI controller. If disabled, the output will be equal to the init input and when enabled, the output is PI computed value.

3 PI Controller Hardware Implementation

The following figure shows the block diagram of an implemented PI controller.

Figure 3 • **Implementation of PI Controller**



The ref_input_i port connects to the reference input, while the act_input_i port connects to the actual input. The kp_i, ki_i inputs represent the proportional and integral gains. ymax_i and ymin_i represent the minimum and maximum outputs. The init_i input represents the initialization value (which is output if the PI controller is diabled), and the pi_en_i input is used to enable the PI controller operation. A pulse of one clock cycle width is used to start the computation. The done_o output signal represents valid data on the output_y_o port. A single clock pulse appears at the done_o signal to represent the end of computation.

The entire system is synchronized with the rising edge of clock and controlled by a finite state machine (FSM). The PI controller block uses a multiply-accumulate-subtract (MAS) block to perform operations like multiplication, addition, and subtraction.

3.1 Inputs and Outputs

The following table describes input and output ports of the PI controller.

Table 1 • **Input and Output Ports of the PI Controller**

Signal Name	Direction	Description
reset_i	Input	Asynchronous reset signal (default is active low)
sys_clk_i	Input	System clock
ref_input_i	Input	Reference input to be tracked
act_input_i	Input	Input from feedback measurement/quantity
start_i	Input	Start signal to trigger the FSM on the signal's rising edge; to initiate the PI Controller algorithm
kp_i	Input	Proportional gain value input
ki_i	Input	Integral gain value input
ymin_i	Input	Input representing the minimum allowable output value
ymax_i	Input	Input representing the maximum allowable output value
init_i	Input	The initial output value when the PI controller is not enabled
pi_en_i	Input	When ‘1’, the PI Controller functionality is enabled. When ‘0’, the PI Controller functionality is disabled, and the input value in the init port will be available at the output
done_o	Output	Done signal which is asserted for one clock cycle when computations are completed and output of PI controller is available
output_y_o	Output	Output of the PI controller

3.2 Configuration Parameters

The following table describes the configuration parameters used in the hardware implementation of the PI controller. These are generic parameters and can be varied as per the application requirements.

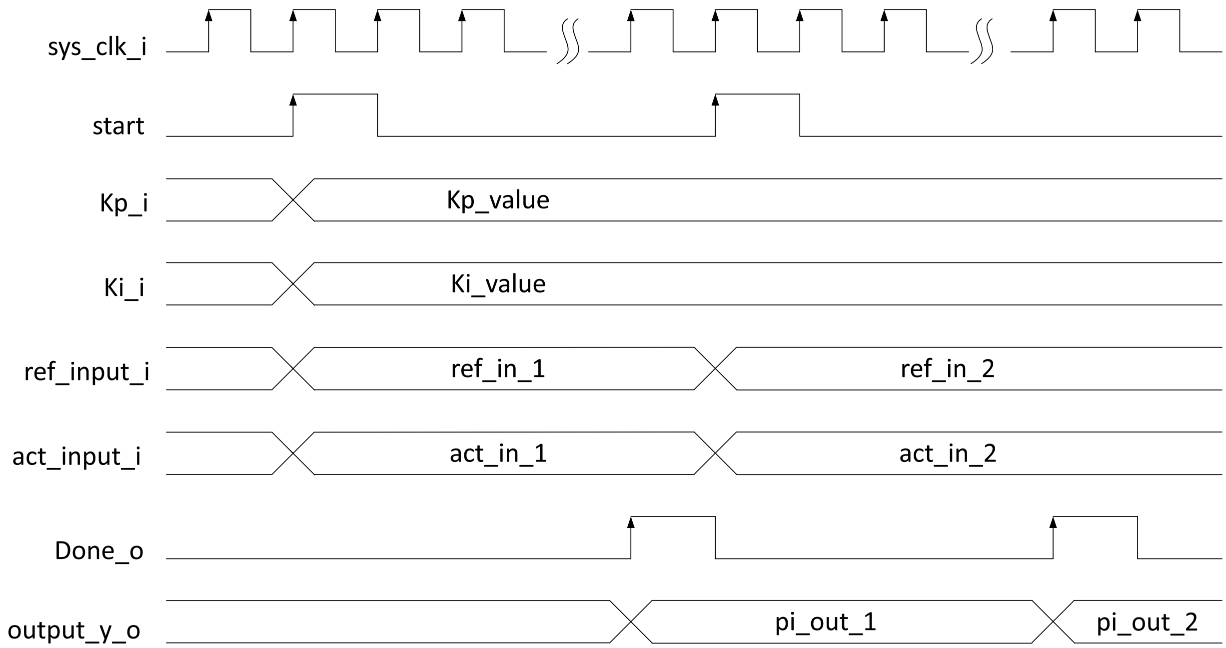
Table 2 • **Configuration Parameters Used in the PI Controller Hardware Implementation**

Name	Description
g_NO_MCYCLE_PATH	The number of clock delays required before the multiplication product ready signal is asserted

3.3 Timing Diagram of PI Controller

The timing diagram of the PI controller block is shown in the following figure.

Figure 4 • **PI Controller Timing Diagram**



3.4 Resource Utilization of Rate Limiter Block

The resource utilization of the rate limiter block, implemented on a SmartFusion®2 / IGLOO®2 device is as shown in the following table.

Table 3 • **Resource Utilization of Rate Limiter Block**

Resource	Usage
Sequential Elements	328
Combinational Logic	248
MACC	1
RAM1kx18	0
RAM64x18	0