

Application Note

Servo Overload Protection

AN-CM-247

Abstract

Servos are one of the most used actuators in robotics. Some servos, especially unprogrammable servos, do not have overload protection. Consequently, a user will only find that the servo was overloaded after it has become noticeably burnt. This application note provides a design to try to protect an unprogrammable servo from overload risk caused by mechanical problems. This design uses an SLG46140V GreenPAK. There are two designs, the first one uses an absolute value for a failsafe PWM that sets when overload is detected. The second uses a delay block to increase or decrease the PWM duty cycle relative to the input when overload is detected.

This application note comes complete with design files which are listed in the References section.

 Servo Overload Protection

Contents

Abstract	1
Contents	2
Figures	2
1 Terms and Definitions	3
2 References	3
3 Introduction	4
4 Designed Solution	5
4.1 Feedback Input.....	6
4.2 PWM-out for Failsafe Mode	7
4.3 GreenPAK Design Component.....	10
5 Results	11
6 Conclusion	13
Revision History	14

Figures

Figure 1: Flow or State Diagram of System.....	5
Figure 2: Simple Voltage Divider with Adjustable Resistor.....	6
Figure 3: Simple Low Pass Filter.....	6
Figure 4: Feedback Voltage Signal.....	7
Figure 5: Circuit Design of Design A.....	7
Figure 6: GreenPAK Design of Design A.....	8
Figure 7: Circuit Design of Design B.....	9
Figure 8: GreenPAK Design of Design B.....	9
Figure 9: Counter Setting for Waiting Time.....	10
Figure 10: Delay Settings.....	11
Figure 11: Measurement Result.....	12
Figure 12: Critical Mode of Design A	13

Servo Overload Protection

1 Terms and Definitions

ACMP	Analog comparator output
CNT	Counter
DLY	Delay
GND	Ground
IC	Integrated circuit
LED	Light-emitting diodes
LPF	Low pass filter
PWM	Pulse width modulation
PWM-in	Origin PWM signal from main servo controller
PWM-out	PWM output that will control servo directly
RC Oscillator	A circuit with both a resistor (R) and a capacitor (C)

2 References

For related documents and software, please visit:

<https://www.dialog-semiconductor.com/configurable-mixed-signal>.

Download our free **GreenPAK™** Designer software [1] to open the .gp files [2] and view the proposed circuit design. Use the **GreenPAK** Development Tools [3] to freeze the design into your own customized IC in a matter of minutes. Dialog Semiconductor provides a complete library of application notes [4] featuring design examples as well as explanations of features and blocks within the Dialog IC.

- [1] [GreenPAK Designer Software](#), Software Download and User Guide, Dialog Semiconductor
- [2] [AN-CM-247 Servo Overload Protection.gp](#), GreenPAK Design File, Dialog Semiconductor
- [3] [GreenPAK Development Tools](#), GreenPAK Development Tools Webpage, Dialog Semiconductor
- [4] [GreenPAK Application Notes](#), GreenPAK Application Notes Webpage, Dialog Semiconductor
- [5] [SLG46140V](#), Datasheet, Dialog Semiconductor
- [6] [AN-1057 Servo Motor Control](#), Application note, Dialog Semiconductor
- [7] [Attiny85](#), Datasheet, Microchip

Servo Overload Protection

3 Introduction

This application note provides a design for protecting unprogrammable servos from overload risk caused by mechanical problems or other failure. If an overload is neglected the servo may burn, become damaged, or malfunction. Therefore, it is necessary to detect the overload indicators and perform a preventative action before the servo becomes damaged. Throughout this document this action is referred to as the failsafe mode.

Dialog's [GreenPAK](#) is used within this system as a pulse width modulation (PWM) modifier to make sure the servo works properly, and it accomplishes this by measuring current through the servo motor's ground connection. By default, the system just passes through the PWM signal from input to output. When improper current flow from the servo is detected for a few seconds the system enters failsafe mode, whereupon the PWM output is changed.

Within this application note there are two options to generate a PWM signal output in failsafe mode:

1. Decrease or increase the width of the PWM input with a rising or falling delay. This stops the servo from undergoing a fault when a PWM signal is attempting to push a servo beyond its actual range of rotation, like when an automatic door does not have the full range of motion theoretically available to the servo motor.
2. Generate a predetermined PWM by using a PWM generator from an internal [GreenPAK](#) clock. This option moves the servo motor to a predefined position that is assumed to be safe, such as the midpoint of the servo's range of motion.

The system reads a feedback voltage from a simple low pass filter which is placed between the servo ground and system ground. If overload occurs, the voltage should increase significantly. Overload detection sensitivity is made configurable by adding a voltage reference input with a simple voltage divider.

The key advantages to using a [GreenPAK](#) are that they are small, low-cost, simple, and customizable. Another typical solution would be to use a microcontroller, which would be overkill for this system when only a small, low power, and configurable solution is required. A power-hungry device that is anywhere near the size of the servo motor is impractical and not industry-feasible. Furthermore, due to the variety of servos available on the market the solution must have some capacity to be tailored to most use-cases, which prevents a dedicated IC solution from being favorable.

Servo Overload Protection

4 Designed Solution

The designed solution can be shown as a block diagram, see [Figure 1](#). There are four states to describe how this system works:

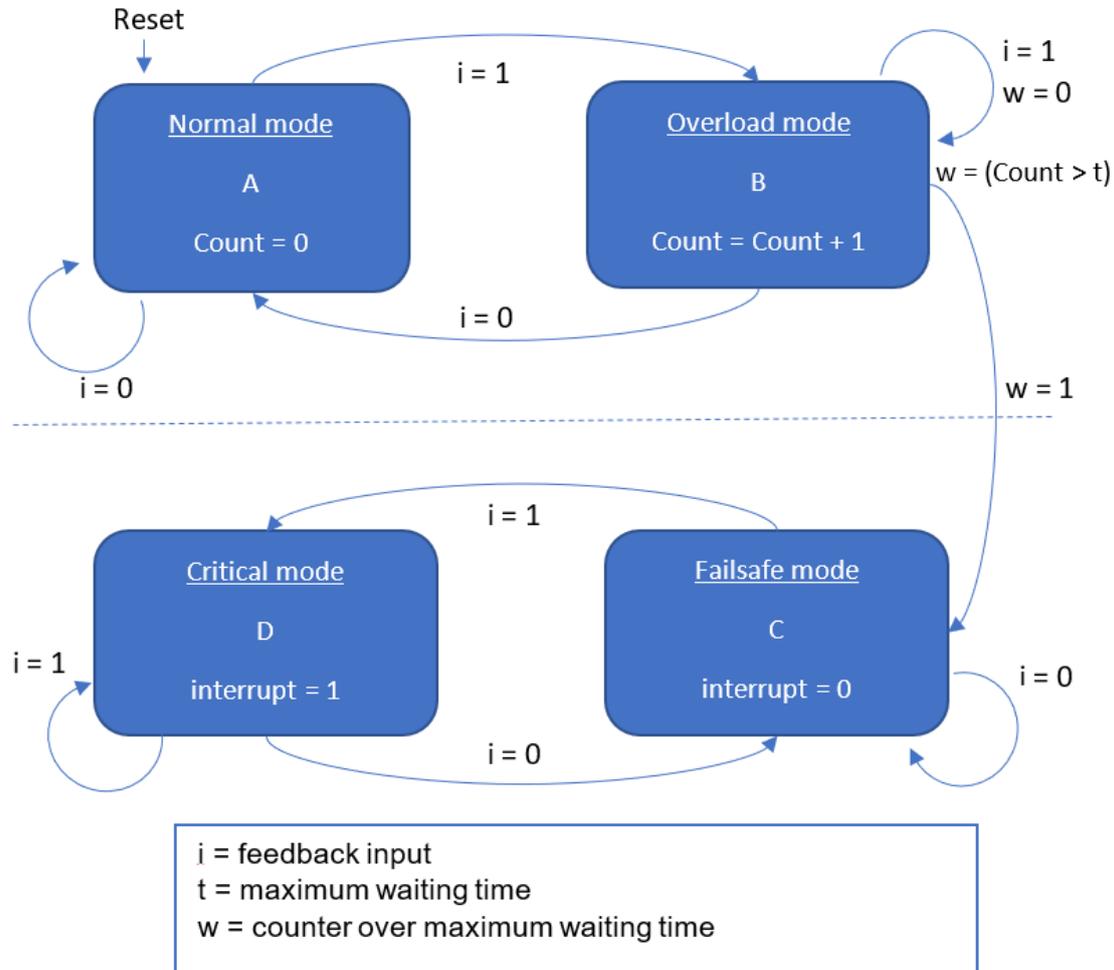


Figure 1: Flow or State Diagram of System

In state A the servo system runs normally. In this state, the PWM output matches the PWM input. When overcurrent is detected from the feedback pin's input ($i = 1$) the system transitions to state B and switches to overload mode.

In state B the PWM output is still the same as the PWM input. This state begins to increment a counter. If the overcurrent is resolved before the counter finishes the state transitions back to state A, otherwise the state transitions to C.

In state C, the system enters failsafe mode and turns on the failsafe signal. The failsafe signal manipulates the input PWM before sending it as an output, or it generates a predesigned PWM signal using an internal clock and PWM block. If, while still in failsafe mode, the feedback input still reads an overload condition, the system outputs an interrupt signal and triggers state D.

State D, also known as critical mode, is used to send an interrupt signal to cut off servo power, either by switching a relay or pinging the main servo controller to request a shutdown point. This state is necessary for situations where the servo is blocked from movement in both directions, such as during a mechanical failure or when the servo arm is locked in place by a hand, vise or similar object.

Servo Overload Protection

4.1 Feedback Input

The feedback input for this system comes from an analog comparator output (ACMP). ACMP has two sources; an overload reference voltage and a feedback voltage.

Rather than use one of the GreenPAK’s internal voltage references, this design uses an adjustable voltage for the overload reference voltage by using a simple voltage divider circuit. Different types of servos may have different overload levels. Therefore, this approach allows a sensitivity level for overload to be set as required, without the need for programming a different IC for each type of servo.

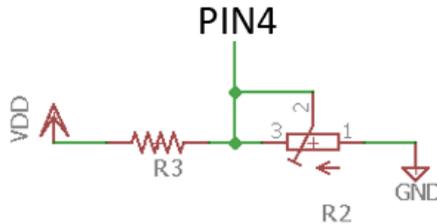


Figure 2: Simple Voltage Divider with Adjustable Resistor

Feedback voltage is obtained from a small current sense resistor that is inserted between the GND of the system and GND of the servo. However, this signal cannot be used directly because the current of the servo may exhibit some ripple. Firstly, the signal needs converting to a constant analog voltage by use of a low pass filter, see Figure 3.

The resistor and capacitor value can be determined depending upon the signal frequency. For this implementation, the feedback signal is set at approximately 50 Hz (almost like the PWM input frequency), $R8 = 1\text{ k}\Omega$ and $C1 = 100\ \mu\text{F}$.

This LPF design will have a frequency cutoff of about 1.5 Hz.

$$f_c = \frac{1}{2\pi RC}$$

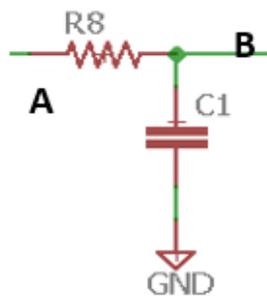


Figure 3: Simple Low Pass Filter

Feedback voltage signal: The 1st channel is filtered signal at point B and the 2nd is from a small resistor inserted between GND of IC and GND of servo.

Servo Overload Protection

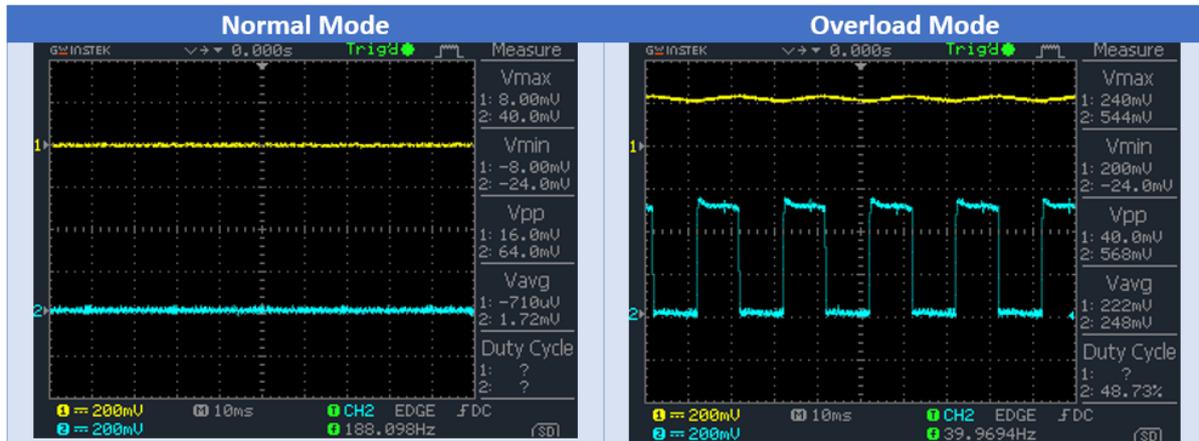


Figure 4: Feedback Voltage Signal

To generate the feedback logic the ACMP compares the overload reference voltage to the feedback voltage. If the feedback voltage is more than the overload reference voltage the ACMP sets the feedback input to HIGH (i = 1). The system then enters overload mode. The simulation is set to a constant voltage of 200 mV on pin 4.

4.2 PWM-out for Failsafe Mode

There are two designs that can be implemented to set the PWM for failsafe mode.

For design A, when in failsafe mode, the PWM output will come from an internal PWM generator. The signal width can be adjusted by the analog voltage reference.

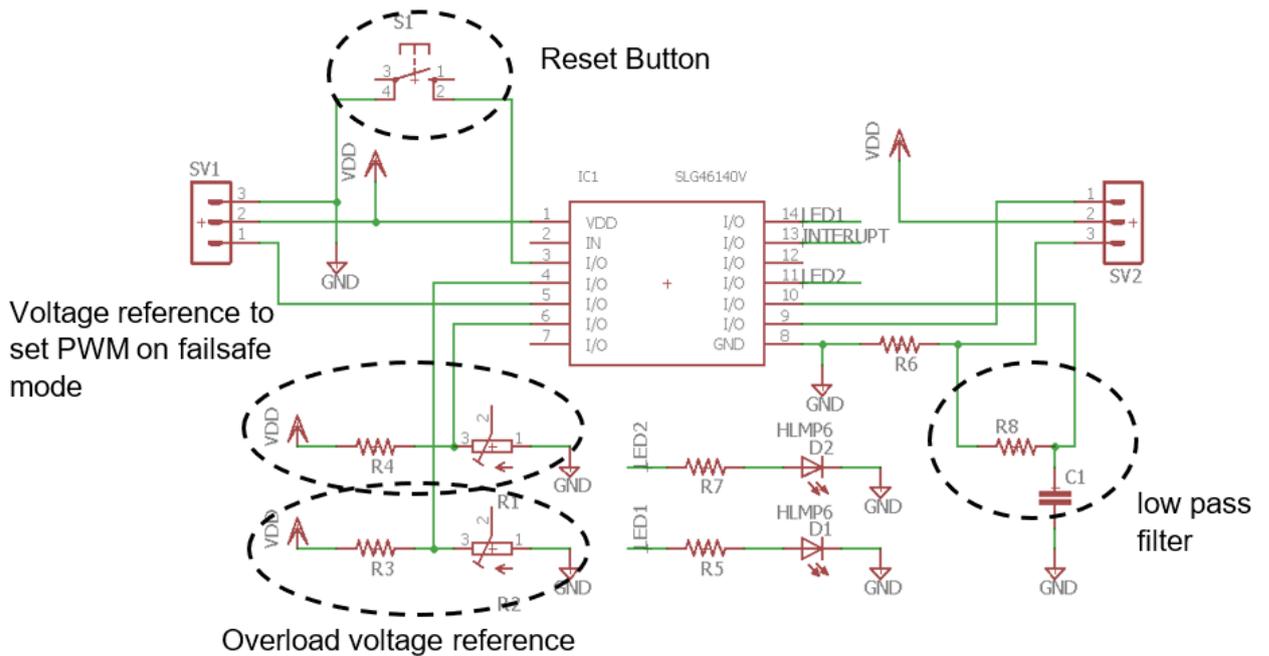


Figure 5: Circuit Design of Design A

Servo Overload Protection

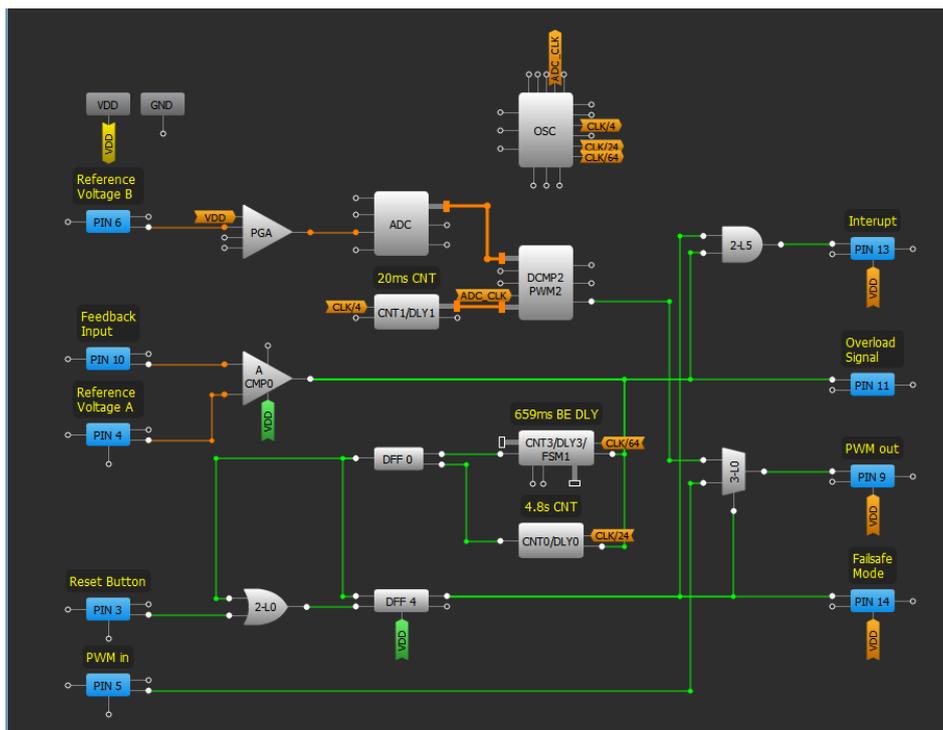


Figure 6: GreenPAK Design of Design A

Design A uses the analog to PWM module referenced in AN-1057 [6]. The PWM output from this module is used when the system is in failsafe mode. The PWM can be configured by changing the voltage reference originating from pin 6. However, before using this design, make sure that PWM-out is a safe value for servo operation. For this simulation, the benchmark of 254 mV is used to generate a pulse width of about 1500 μ s, (the midpoint of a typical servo). The frequency of this output can be configured by changing the counter data in CNT1. This example uses a typical frequency of 50 Hz, which is the standard period for most servo motors.

For design B, when in failsafe mode, PWM-out comes from PWM-in, modified by a defined delay. A delay block can be used to increase or decrease the width of PWM-in. Therefore, PWM-out will be relative to PWM-in. By setting *Falling* as edge select in the delay block, it increases the width of PWM-in. Otherwise, by setting *Rising* as edge select in the delay block, it decreases the width of PWM-in. This design is simpler than design A.

Servo Overload Protection

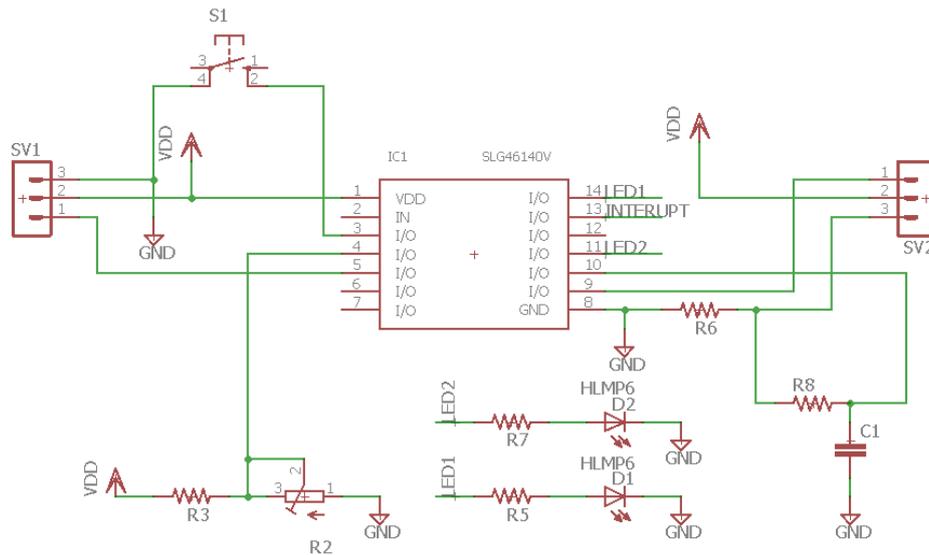


Figure 7: Circuit Design of Design B

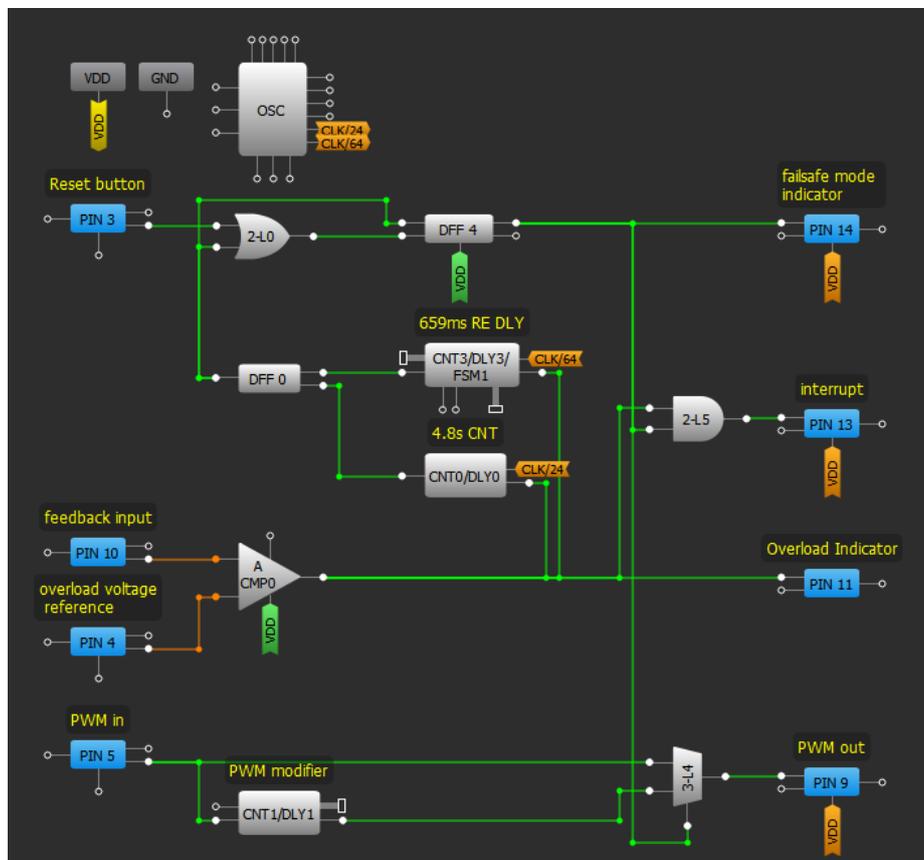


Figure 8: GreenPAK Design of Design B

The delay block is only used to increase or decrease the pulse by a set amount. PWM width can be increased by setting Edge Select in DLY 1 to *Falling* and Counter Data as the extra time desired to add to the pulse. Alternatively, PWM width can be decreased by setting Edge Select to *Rising*. Unfortunately, in this design, servos only have one direction protected via failsafe mode, either clockwise or counter clockwise. Therefore, it is necessary to decide the edge select needed for the application before programming this design.

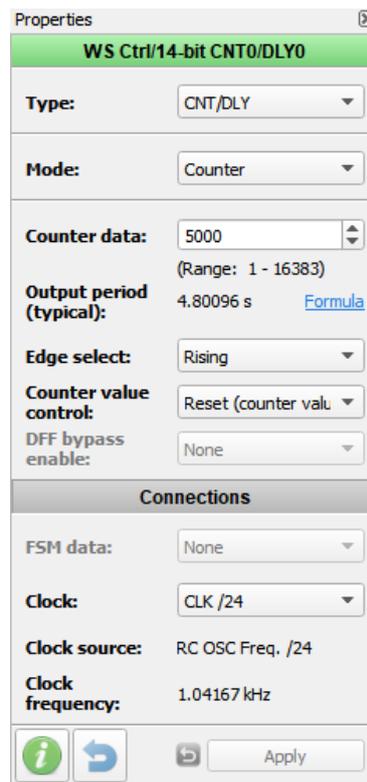
Servo Overload Protection

4.3 GreenPAK Design Component

As shown in [Figure 6](#), [Figure 7](#), and [Figure 8](#) the GreenPAK Design is very similar. There are some components, such as a digital multiplexer, a counter for the wait time, delay for PWM-out (design B only), analog comparator, PWM generator (design A only), and some logic blocks.

The digital multiplexer acts as a mode switcher. When the system is in the normal mode (state A and B) it selects PWM-in as PWM-out. When not in normal operation it selects the failsafe PWM as PWM-out.

CNT0 is the latency counter for the failsafe mode after overcurrent has been detected. It can be set as required for whichever servo is used. The failsafe point should be a point where the overcurrent is confirmed to be a consistent problem but not yet enough to cause burning. This example uses 4.8 s.



Properties	
WS Ctrl/14-bit CNT0/DLY0	
Type:	CNT/DLY
Mode:	Counter
Counter data:	5000 (Range: 1 - 16383)
Output period (typical):	4.80096 s Formula
Edge select:	Rising
Counter value control:	Reset (counter valu
DFF bypass enable:	None
Connections	
FSM data:	None
Clock:	CLK /24
Clock source:	RC OSC Freq. /24
Clock frequency:	1.04167 kHz
<input type="button" value="Apply"/>	

Figure 9: Counter Setting for Waiting Time

The delay for the PWM-out can be configured as shown in [Figure 10](#). Counter data can be set as needed. For this example, to increase PWM value by 500 μ s (left) and decrease PWM value by 500 μ s (right).

Servo Overload Protection



Figure 10: Delay Settings

5 Results

For simulation purposes, an RC oscillator was used to generate a 1100 μ s PWM signal. According to Design B, set pin 4 and pin 6 as a signal generator with constant voltage via the GreenPAK Designer Emulator Tool software. Feedback goes through the low pass filter circuit to pin 10. The results of the LPF can be seen in Figure 4. The overload reference voltage is set to 200 mV. With this configuration, the PWM is chosen to increase when in failsafe mode.

Servo Overload Protection

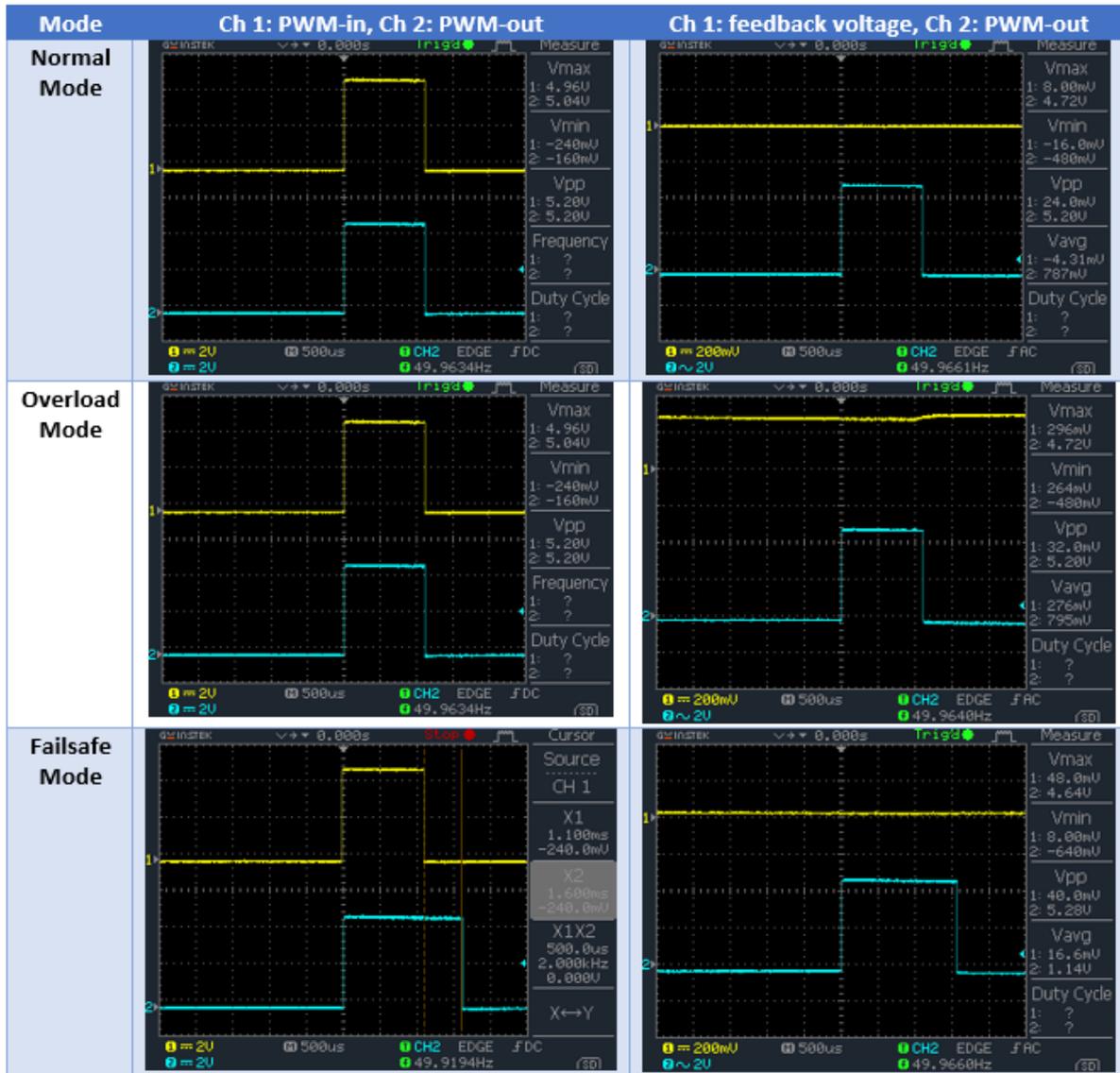


Figure 11: Measurement Result

In normal mode there is no overload, so the PWM output is the test input of 1100 μs. When feedback voltage is greater than the overload reference voltage it switches to overload mode. The output does not match the PWM input during the failsafe mode and instead is set at the PWM “safe spot” at 1.5 ms.

After the system has remained in overload mode for approximately 4.8 s, the system automatically switches to failsafe mode. Figure 11 shows, in this mode, there is a difference of pulse width between PWM-in and PWM-out by 500 μs. If overload still occurs in failsafe mode, the system enters critical mode and emits an interrupt signal through pin 13. For this simulation the LED on the development board is activated.

There is no difference between design A and design B when in normal and overload mode. However, in failsafe mode the system generates a PWM by using an internal clock and PWM generator. For this simulation, to set failsafe PWM to 1500 μs, the voltage reference of the PWM needs to be set as 254.7 mV.

Servo Overload Protection

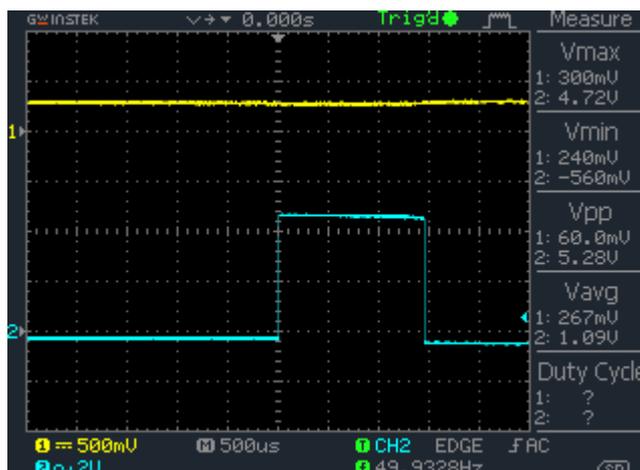


Figure 12: Critical Mode of Design A

Based on the results shown in [Figure 12](#), Dialog's design can protect the servo motor from overloading.

6 Conclusion

Servo overload protection using the Dialog [GreenPAK](#) SLG46140 is an effective and compact way to prevent servos from overloading. The [GreenPAK](#) Design modifies the PWM used to control the servo based upon feedback voltage. If overload occurs, the system can set the servo to a safe position. If the servo is found to not be able to move to a safe position an interrupt is sent to disable the servo and prevent damage. Other more complicated and application-specific schemes can be created using this application as a starting point. Beyond functionality, using this Dialog product has several advantages compared to competing solutions, such as its small size, lower power consumption, simplicity, and lower price.

Servo Overload Protection

Revision History

Revision	Date	Description
1.0	18-Jul-2018	Initial Version

Servo Overload Protection

Status Definitions

Status	Definition
DRAFT	The content of this document is under review and subject to formal approval, which may result in modifications or additions.
APPROVED or unmarked	The content of this document has been approved for publication.

Disclaimer

Information in this document is believed to be accurate and reliable. However, Dialog Semiconductor does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information. Dialog Semiconductor furthermore takes no responsibility whatsoever for the content in this document if provided by any information source outside of Dialog Semiconductor.

Dialog Semiconductor reserves the right to change without notice the information published in this document, including without limitation the specification and the design of the related semiconductor products, software and applications.

Applications, software, and semiconductor products described in this document are for illustrative purposes only. Dialog Semiconductor makes no representation or warranty that such applications, software and semiconductor products will be suitable for the specified use without further testing or modification. Unless otherwise agreed in writing, such testing or modification is the sole responsibility of the customer and Dialog Semiconductor excludes all liability in this respect.

Customer notes that nothing in this document may be construed as a license for customer to use the Dialog Semiconductor products, software and applications referred to in this document. Such license must be separately sought by customer with Dialog Semiconductor.

All use of Dialog Semiconductor products, software and applications referred to in this document are subject to Dialog Semiconductor's [Standard Terms and Conditions of Sale](http://www.dialog-semiconductor.com), available on the company website (www.dialog-semiconductor.com) unless otherwise stated.

Dialog and the Dialog logo are trademarks of Dialog Semiconductor plc or its subsidiaries. All other product or service names are the property of their respective owners.

© 2018 Dialog Semiconductor. All rights reserved.

Contacting Dialog Semiconductor

United Kingdom (Headquarters)

Dialog Semiconductor (UK) LTD
Phone: +44 1793 757700

Germany

Dialog Semiconductor GmbH
Phone: +49 7021 805-0

The Netherlands

Dialog Semiconductor B.V.
Phone: +31 73 640 8822

Email:

enquiry@diasemi.com

North America

Dialog Semiconductor Inc.
Phone: +1 408 845 8500

Japan

Dialog Semiconductor K. K.
Phone: +81 3 5769 5100

Taiwan

Dialog Semiconductor Taiwan
Phone: +886 281 786 222

Web site:

www.dialog-semiconductor.com

Hong Kong

Dialog Semiconductor Hong Kong
Phone: +852 2607 4271

Korea

Dialog Semiconductor Korea
Phone: +82 2 3469 8200

China (Shenzhen)

Dialog Semiconductor China
Phone: +86 755 2981 3669

China (Shanghai)

Dialog Semiconductor China
Phone: +86 21 5424 9058