

# Application Note

## SLG46880/1 F(1) Block

### AN-CM-237

#### Abstract

*This application note how to use the F(1) computation macrocell in the SLG46880/1.  
This application note comes complete with design files which can be found in the References section.*

---

---

**SLG46880/1 F(1) Block**

## Contents

<b>Abstract</b> .....	<b>1</b>
<b>Contents</b> .....	<b>2</b>
<b>Figures</b> .....	<b>2</b>
<b>Tables</b> .....	<b>2</b>
<b>1 Terms and Definitions</b> .....	<b>3</b>
<b>2 References</b> .....	<b>3</b>
<b>3 Introduction</b> .....	<b>4</b>
<b>4 Creating a New F(1) Configuration</b> .....	<b>5</b>
<b>5 F(1) Commands</b> .....	<b>5</b>
<b>6 F(1) Example Commands</b> .....	<b>6</b>
<b>7 Sensor Application</b> .....	<b>7</b>
<b>8 Other F(1) Applications</b> .....	<b>11</b>
<b>9 Conclusion</b> .....	<b>13</b>
<b>Revision History</b> .....	<b>14</b>

## Figures

Figure 1: F(1) Block and Connections .....	4
Figure 2: LOAD Configuration .....	6
Figure 3: DELAY Configuration .....	6
Figure 4: LOOP with DELAY Configuration.....	7
Figure 5: Example #1 from Datasheet.....	7
Figure 6: Example #1 Schematic, State 0 .....	8
Figure 7: Example #1 Schematic, State 1 .....	8
Figure 8: ASM Diagram .....	9
Figure 9: ASM Output Matrix .....	9
Figure 10: Example #1 Application Circuit .....	9
Figure 11: Room Temperature (25 °C); Potentiometer at 50 kΩ .....	10
Figure 12: Over Temperature (120 °C); Potentiometer at 1.5 kΩ .....	11
Figure 13: Example #2 from Datasheet.....	11
Figure 14: Example #2 Schematic .....	12
Figure 15: Example #3 from Datasheet.....	12
Figure 16: Example #3 Schematic, State 0 .....	13
Figure 17: Example #3 Schematic, State 1 .....	13

## Tables

Table 1: E(1) Commands .....	5
------------------------------	---

---

**SLG46880/1 F(1) Block**

## 1 Terms and Definitions

ASM                      Asynchronous state machine

## 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-237 SLG46880/1 F\(1\) Block.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] 50K6A1 SERIES 1 Thermistor Datasheet [Online]. Available:  
[https://www.mouser.com/datasheet/2/418/NG\\_DS\\_Series\\_I\\_Thermistor\\_50K6A1\\_A-779622.pdf](https://www.mouser.com/datasheet/2/418/NG_DS_Series_I_Thermistor_50K6A1_A-779622.pdf). [Accessed 11 01 2018]

## SLG46880/1 F(1) Block

### 3 Introduction

The F(1) Computation Macrocell, also referred to as the F(1) block, is a specialized block within the SLG46880/1's Asynchronous State Machine (ASM) which allows the designer to trigger sequences of commands upon entering a new state of the ASM.

Whenever the ASM enters a new state, the F(1) block can execute a sequence of up to 12 commands for loading and storing single-bit data in a stack up to 16 bits deep, as well as perform logical operations on bits at the top of the stack, such as AND, OR, XOR, and INV.

Once the F(1) completes its sequence, control is relinquished back to the ASM. While the F(1) block is active, there is no ASM activity (i.e. the ASM cannot change states).

The F(1) block has two digital inputs: ASM\_nRESET and f1\_Interrupt. An active HIGH signal on either of those inputs will immediately halt any command execution for the F(1) block, and will immediately relinquish control back to the ASM.

The user can define up to 4 different F(1) configurations, which can each hold a set of up to 12 commands. Only one configuration may be executed per ASM state. The user can also decide to not use the F(1) block for a particular state.

The F(1) block also has 4 analog inputs coming from various pins that can be muxed into the positive input for the F(1) block's dedicated Analog Comparator.

The F(1) stack can serve as a persistent data source or data destination for commands running in the F(1) block. LOADx commands will push data down into the stack. OUTx commands will pop data off the stack, and send to the contents to one of three outputs to the Connection Matrix. The contents of this memory are not changed during state transitions, and are only changed by the commands running inside the F(1) Computation Macrocell itself.

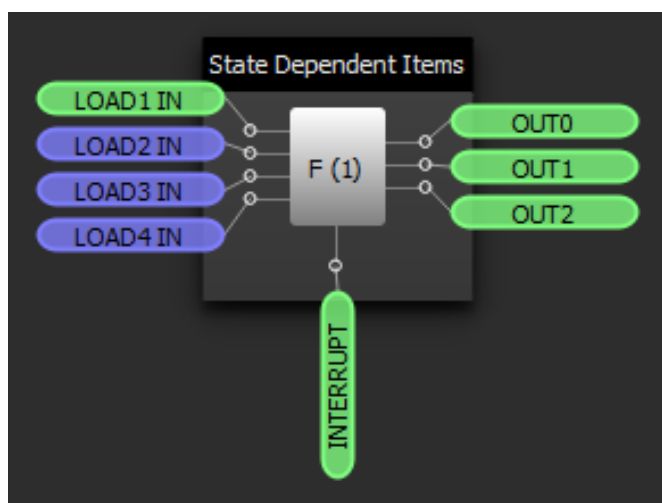


Figure 1: F(1) Block and Connections

## SLG46880/1 F(1) Block

### 4 Creating a New F(1) Configuration

To create a new F(1) configuration, display the F(1) block on your workspace by clicking on its checkbox in the Components List. Then double-click the F(1) block to bring up its Properties panel.

Create a new F(1) configuration by clicking on the green “+” button at the top right of the Properties panel. You can rename configurations to whatever you want, but for this app note I’ll leave the default configuration names in place.

For each of your F(1) configurations, you can choose whether or not you want an interrupt to reset the stack’s memory. You can also select the initial state for each of the F(1) block’s 3 outputs. The options for the OUTx initial states are: keep, 0, 1, and none.

### 5 F(1) Commands

Table 1 includes all of the commands available in the F(1) block.

**Table 1: E(1) Commands**

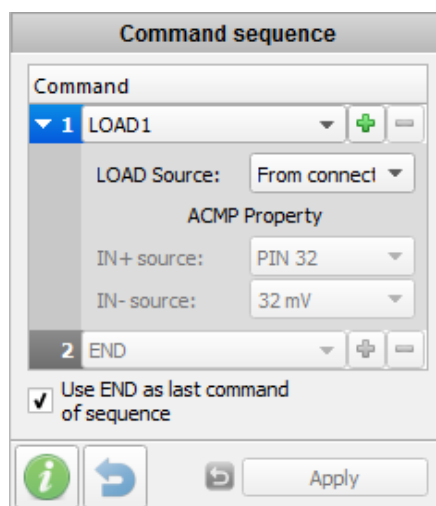
Command	Description
LOAD1	Push one bit. Source defined by LOAD1 configuration.
LOAD2	Push one bit. Source defined by LOAD2 configuration.
LOAD3	Push one bit. Source defined by LOAD3 configuration.
LOAD4	Push one bit. Source defined by LOAD3 configuration.
AND	Pop two bits. Perform Logical AND, then Push the output.
OR	Pop two bits. Perform Logical OR, then Push the output.
XOR	Pop two bits. Perform Logical XOR, then Push the output.
INV	Pop one bit. Perform Logical INV, then Push the output.
PUSH0	Push a 0 to the top of the stack.
POP	Pop one bit from the top of the stack.
DELAY	Delay according to F(1) delay configuration. Then continue.
LOOP w/ DLY	If the top bit is 0, execute delay and then jump to specified command. Otherwise, continue.
OUT0	Output the top value to OUT0
OUT1	Output the top value to OUT1
OUT2	Output the top value to OUT2
END	End F(1). Return control to ASM.

## SLG46880/1 F(1) Block

### 6 F(1) Example Commands

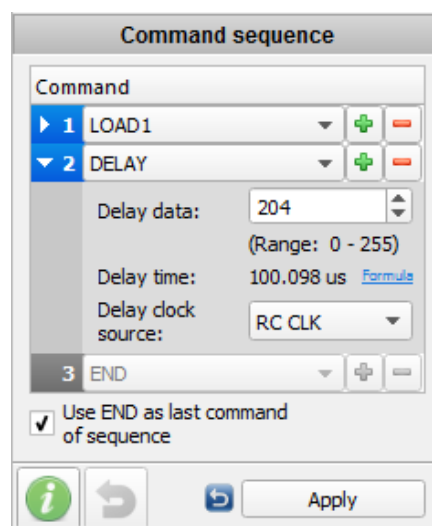
To create a LOAD command, select LOADx from the dropdown menu in the Command Sequence section. You can choose whether you want to load a bit from the LOADx IN matrix connector or from an ACMP comparison. If you choose to load from the ACMP, you can choose both the IN+ source and the IN- reference.

When the LOAD command executes, it will PUSH the value from either the connection matrix or the ACMP into the stack.



**Figure 2: LOAD Configuration**

To add a DELAY command, click on the “+” button next to LOAD1 and select DELAY from the next dropdown menu. Now, you can select your delay clock source and data to configure the delay length.



**Figure 3: DELAY Configuration**

The LOOP with DELAY command essentially works as a GOTO command. It allows the designer to jump to an out-of-sequence command after a delay. First, select the command you wish to jump to in the “Delay loop to location” dropdown. Then configure the delay just as was shown in the DELAY command above.

SLG46880/1 F(1) Block

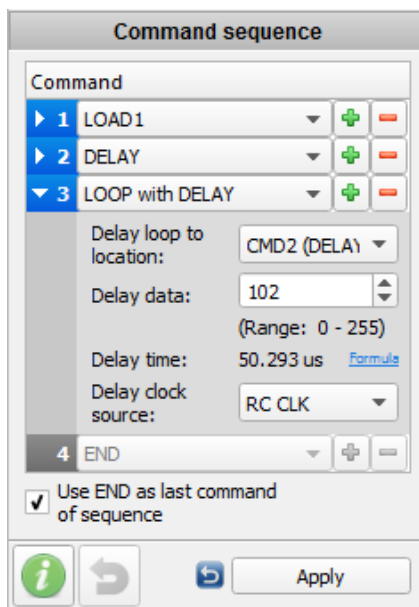


Figure 4: LOOP with DELAY Configuration

Use the OUTx commands to output the value at the top of the stack to the matrix connections. Finally, use the END command to return control back to the ASM.

## 7 Sensor Application

Example #1: Sensor Application showing the command sequence for a potential use case of the f(1) Macrocell.

f(1) Command Sequence	Command	Description
1	INV	Change Top Memory location from 0 → 1
2	OUT1	Output a 1 to a pin defined by OUT1 register. This can be used to turn on or bias an external sensor
3	DELAY	Wait for a time defined by DELAY register. This can be used to allow sensor to settle.
4	LOAD1	Capture the output of f(1) ACMP from an analog pin connected to the sensor as defined by the LOAD1 register.
5	LOAD2	Capture the output of a pin as defined by the LOAD2 register. This can be used to check a power good signal
6	AND	Logically AND the top two values of the 1x16 memory that were just loaded with LOAD1 and LOAD2
7	OUT2	Output the result of sensor output AND power good signal for a control decision for the ASM
8	END	ASM can now act on OUT2 signal

Figure 5: Example #1 from Datasheet

Example #1 of the datasheet provides an example using the F(1) block to detect the voltage of a sensor when the state changes in the ASM (Figure 5). OUT1 is connected to PIN #20 (SENSOR\_ON) and OUT2 is connected to PIN #18 (OVER\_TEMP). The delay on command 3 is defined as 2 μs. LOAD 1's ACMP IN+ is connected to PIN#1 (SENSOR\_IN) and has a threshold of 992 mV. LOAD 2 is connected to PIN #2 (PG). See Figure 6 and Figure 7 for a view of the schematic in GreenPAK Designer. See Figure 8 and Figure 9 for a view of the ASM settings.

SLG46880/1 F(1) Block

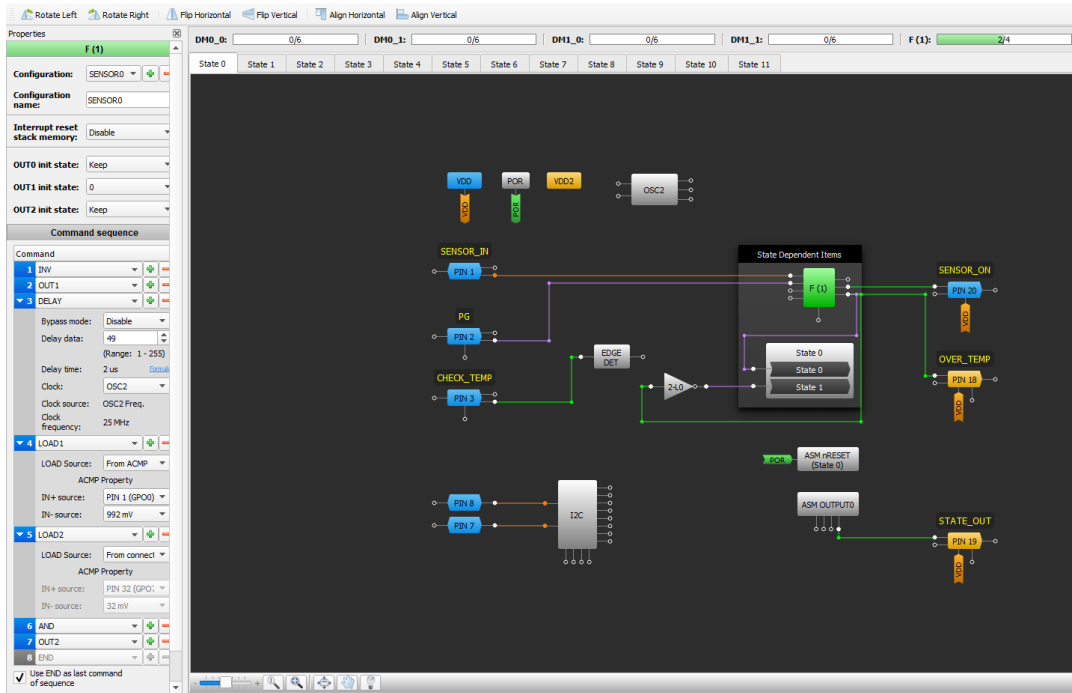


Figure 6: Example #1 Schematic, State 0

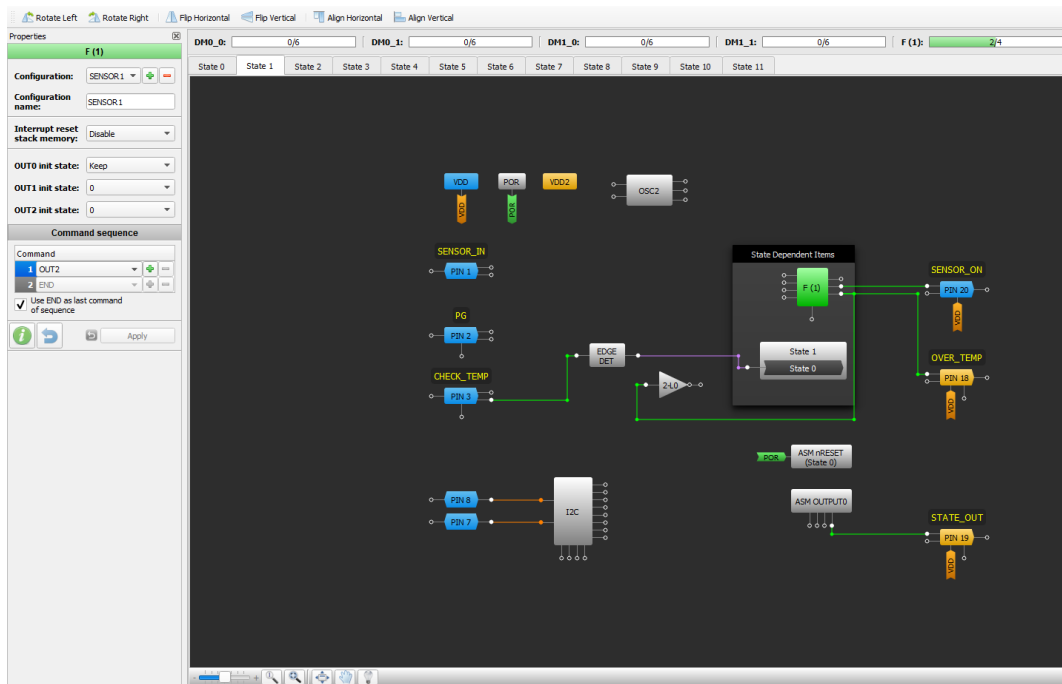


Figure 7: Example #1 Schematic, State 1



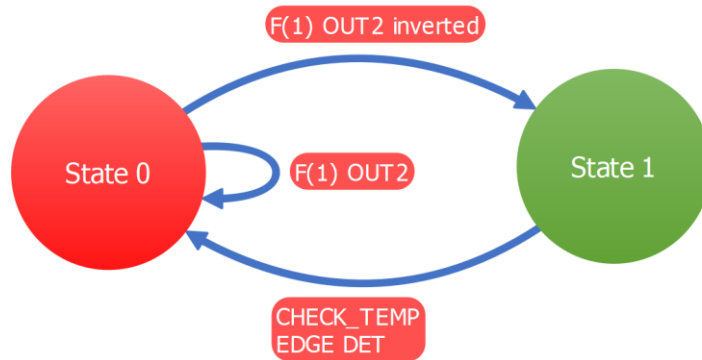


Figure 8: ASM Diagram

RAM				
Matrix		Pins		
Connection Matrix Output RAM				
State name	OUT3	OUT2	OUT1	OUT0
State 0	0	0	0	1
State 1	0	0	0	0

Figure 9: ASM Output Matrix

For a more specific use of this F(1) sequence, it can be used to flag an overtemperature condition while the state machine is transitioning to a new state and the device that the GreenPAK is sensing already has a power good condition. For this application, the SENSOR\_ON output is connected to one node of an NTC thermistor. Another resistor is connected to the other node of the thermistor and to ground to make a voltage divider for the SENSOR\_IN input (Figure 10).

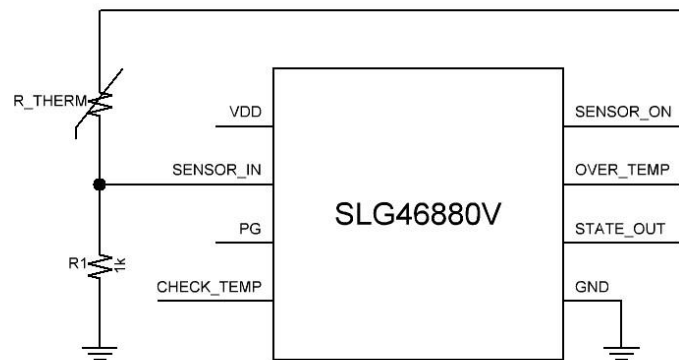


Figure 10: Example #1 Application Circuit

## SLG46880/1 F(1) Block

When the thermistor senses room temperature, the voltage at SENSOR\_IN is about 65 mV. This is well below the 992 mV threshold of the ACMP and the OVER\_TEMP output remains low (Figure 11). As the thermistor senses a higher temperature, it will decrease in resistance and increase the voltage seen on SENSOR\_IN. Once the thermistor senses a temperature above 110 °C, the voltage at SENSOR\_IN will be about 1 V. This crosses the 992 mV threshold set in the ACMP and the OVER\_TEMP output will go high while in State 0 if the PG input is also high (Figure 12).

At room temperature (25 °C) the thermistor was measured about 50 k $\Omega$  and according to a datasheet of a similarly rated NTC thermistor [5] it would be 1.5 k $\Omega$  at 120 °C. To test the design, a potentiometer was used to match what the thermistor's resistance would be at these given temperatures.

Channel 1 - PIN #3 (CHECK\_TEMP)

Channel 2 - PIN #19 (STATE\_OUT)

Channel 3 - PIN #1 (SENSOR\_ON)

Channel 4 - PIN #18 (OVER\_TEMP)

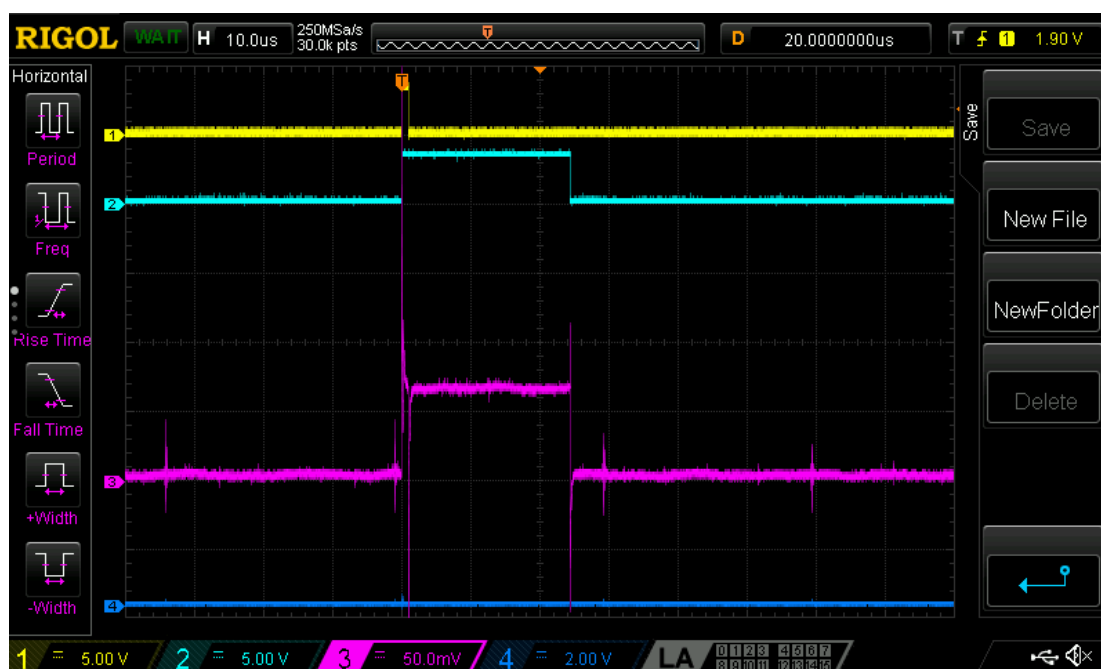


Figure 11: Room Temperature (25 °C); Potentiometer at 50 k $\Omega$

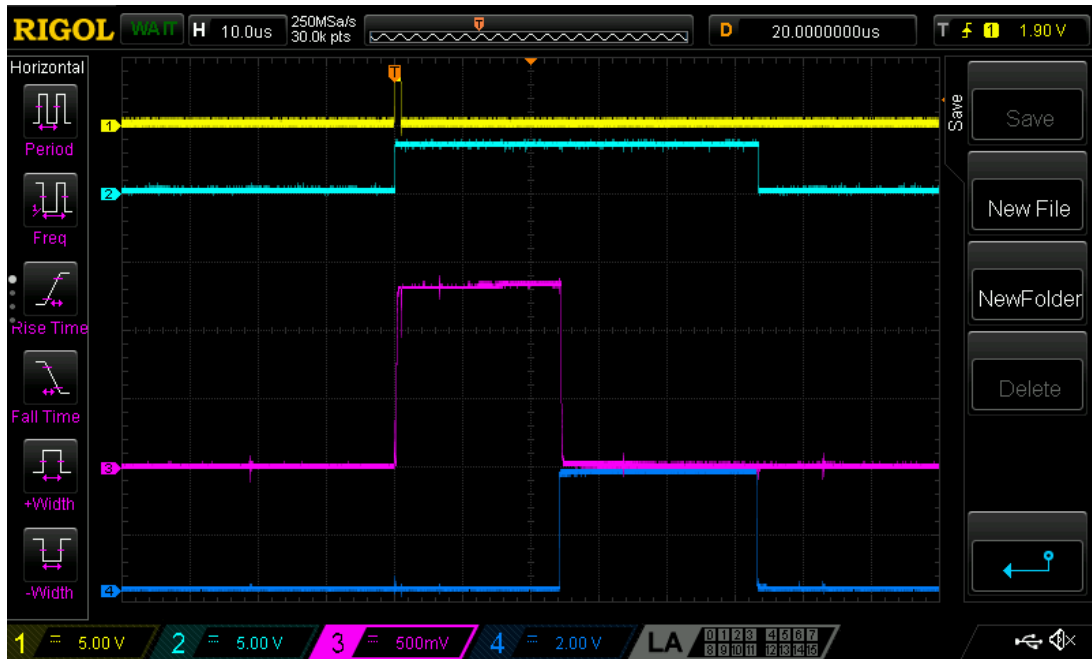


Figure 12: Over Temperature (120 °C); Potentiometer at 1.5 kΩ

## 8 Other F(1) Applications

Example #2: Power Good Application showing the command sequence for a potential use case of the f(1) Macrocell.

f(1) Command Sequence	Command	Description
1	LOAD1	Capture the output of f(1) ACMP from an analog pin connected to power rail 1 as defined by the LOAD1 register. A "1" means that power is good on power rail 1.
2	LOAD2	Capture the output of f(1) ACMP from an analog pin connected to power rail 2 as defined by the LOAD2 register. A "1" means that power is good on power rail 2.
3	LOAD3	Capture the output of f(1) ACMP from an analog pin connected to power rail 3 as defined by the LOAD3 register. A "1" means that power is good on power rail 3.
4	LOAD4	Capture the output of f(1) ACMP from an analog pin connected to power rail 4 as defined by the LOAD4 register. A "1" means that power is good on power rail 4.
5	AND	LOAD4 result ANDs with LOAD3 result (LOAD4 & LOAD3). This combines two of the Power Good signals.
6	AND	(LOAD4 & LOAD3) & LOAD2. This combines three of the Power Good signals.
7	AND	((LOAD4 & LOAD3) & LOAD2) & LOAD1. This combines four of the Power Good signals
8	OUT1	Output the Master Power Good signal to a location defined by the OUT1 register

Figure 13: Example #2 from Datasheet

SLG46880/1 F(1) Block

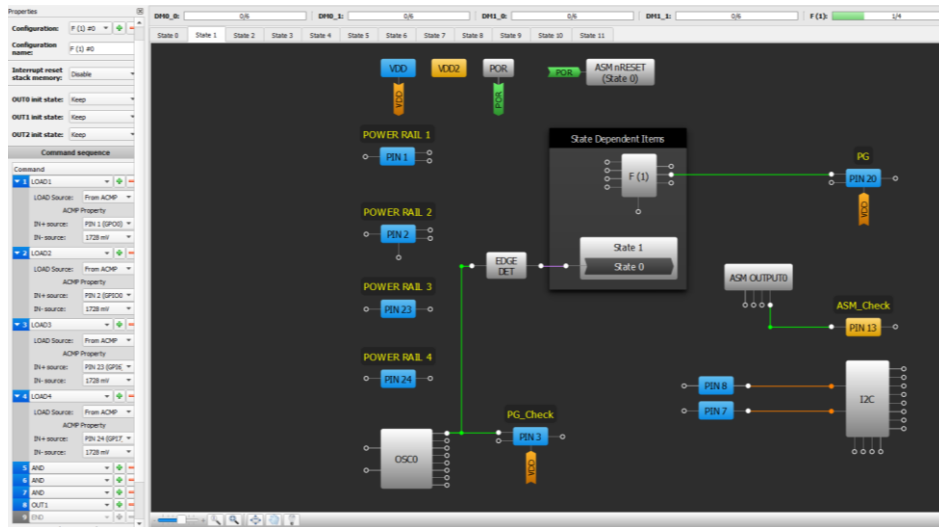


Figure 14: Example #2 Schematic

Example #3: Rising Edge Deglitch Application showing the command sequence for a potential use case of the f(1) Macrocell.

f(1) Command Sequence	Command	Description
1	LOAD1	Capture the output of a pin or any macrocell output defined by the LOAD1 register.
2	DELAY	Delay for the time defined by configuration register.
3	LOAD1	Capture the output of a pin or any macrocell output defined by the LOAD1 register.
4	DELAY	Delay for time defined by configuration register.
5	LOAD1	Capture the output of a pin or any macrocell output defined by the LOAD1 register.
6	AND	Logically AND the top two values of the 1x16 memory stack, that was loaded before. (LOAD1 & LOAD1 after Delay).
7	AND	(LOAD1 & LOAD1 after Delay) & LOAD1 after double Delay.
8	LOOP WITH DELAY	If the calculated value is 0, then start to execute first command once again, which is defined by the configuration bits. Otherwise f(1) continues to execute next command (command 9).
9	OUT1	Output the result value, which is a high level.
10	END	ASM can act on any other signals.

Figure 15: Example #3 from Datasheet

The rising edge deglitch will load the value from PIN #11. After 5 μs it will load the value from PIN #11 again and repeat this a second time after another 5 μs. If PIN #11 was held high for these three successive LOAD1 commands it will output to OUT1 of the F(1) block. If PIN #11 was not held high for these three successive LOAD1 commands, it will loop back to the first command (LOAD1) and restart the process.

SLG46880/1 F(1) Block

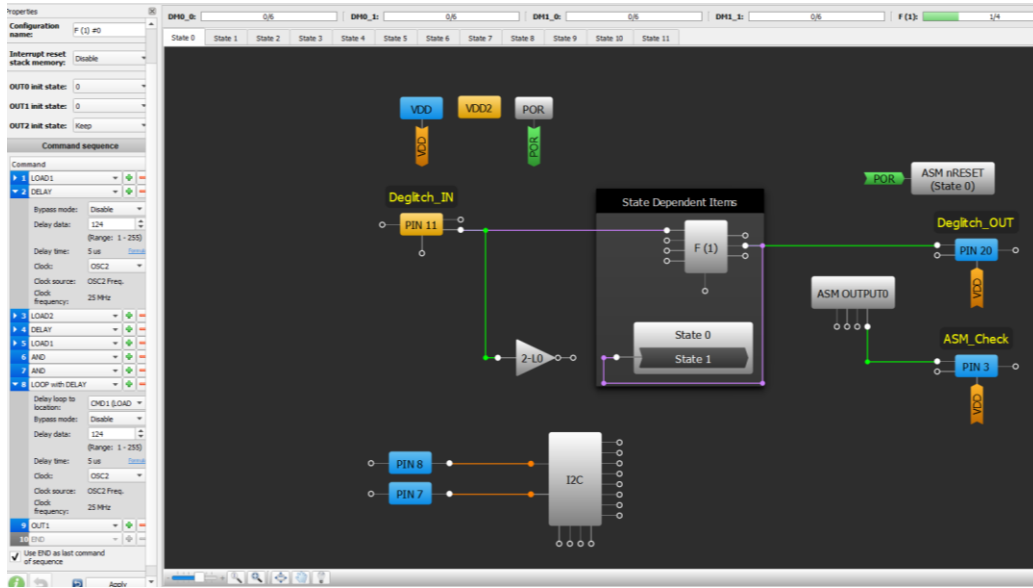


Figure 16: Example #3 Schematic, State 0

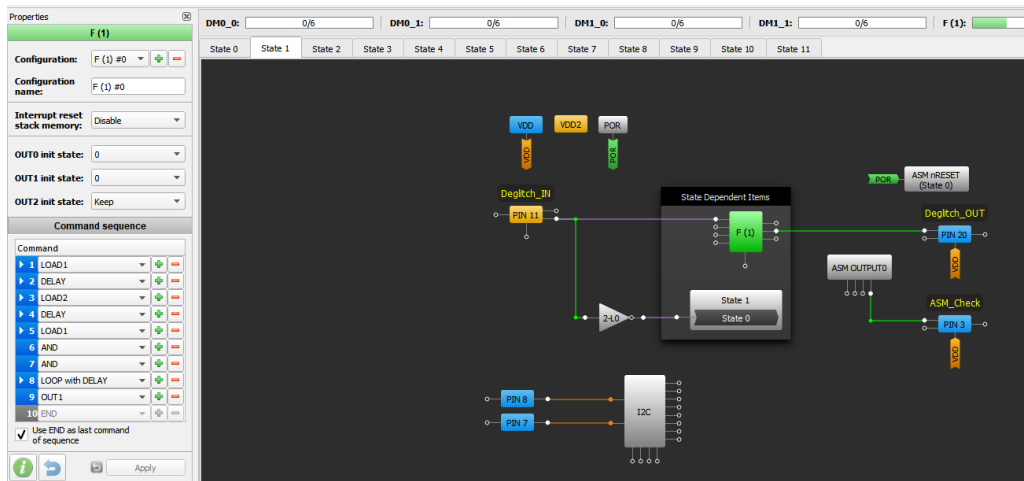


Figure 17: Example #3 Schematic, State 1

## 9 Conclusion

In this app note we discussed how to use the F(1) computation macrocell in the SLG46880/1. The F(1) block allows the designer to assign up to 12 commands to manipulate a 16-bit deep stack of bits. Since the F(1) block's has up to 4 configurations, it can be reused in different ASM states for different purposes. Thanks to its multi-purpose functionality, the F(1) block enhances the flexibility of the [GreenPAK](#) platform.

## Revision History

Revision	Date	Description
1.0	19-Mar-2018	Initial Version

## SLG46880/1 F(1) Block

### 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](http://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](mailto: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](http://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