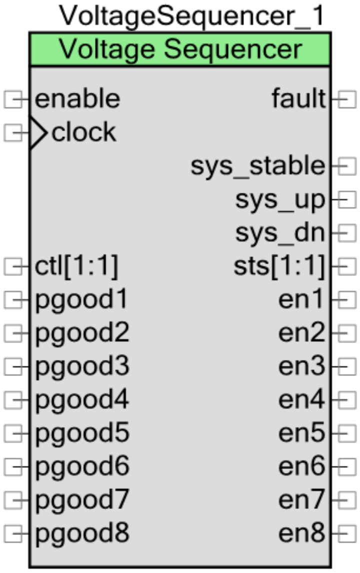
Voltage Sequencer

2.0

Features

- Supports sequencing and monitoring of up to 32 power converter rails
- Supports power converter circuits with logic-level enable inputs and logic-level power good (pgood) status outputs
- Autonomous (standalone) or host driven operation
- Sequence order, timing and inter-rail dependencies can be configured through an intuitive, easy-to-use graphical configuration GUI



General Description

The Voltage Sequencer component provides a simple way to define power-up and power-down sequencing of up to 32 power converters to meet user-defined system requirements. Once the sequencing requirements have been entered into the easy-to-use graphical configuration GUI, the component will automatically take care of the sequencing implementation without requiring any firmware development by the user.

When to Use a Voltage Sequencer

The Voltage Sequencer component should be used in any application that requires sequencing of multiple DC-DC power converters.

For sequencing-only applications, the component can be directly connected to the enable (en) and power good (pg) pins of the DC-DC power converter circuits.

For more comprehensive power supervisor applications, the component can be connected to the Power Monitor or Voltage Fault Detector components in PSoC Creator^{TM} design schematics. The APIs for these components have also been designed to simplify the firmware interaction between them. The Power Monitor and Voltage Fault Detector components are available in the Power Supervision category of the Cypress component catalog.

Input/Output Connections

This section describes the various input and output connections for the Voltage Sequencer component. An asterisk (*) in the list of I/Os indicates that the I/O may be hidden on the symbol under the conditions listed in the description of that I/O.

Enable – Input

Global enable pin that can optionally be used to initiate a power up sequence or a power down sequence**.**

Clock – Input

Timing source used by the component.

System Stable – Output

Active high signal is asserted when all power converters have powered up successfully (all sequencer state machines are in the ON state) and have been running normally for a user-defined amount of time.

System Up – Output

Active high signal is asserted when all power converters have powered up successfully (all sequencer state machines are in the ON state).

System Down – Output

Active high signal is asserted when all power converters have powered down successfully (all sequencer state machines are in the OFF state).

Warning – Output *

Active high signal is asserted when one or more power converters did not shut down within the user-specified time period. This terminal is visible when you de-select the checkbox labeled **Disable TOFF_MAX warnings** on the Power Down tab of the Configure dialog.

Fault – Output

Active high signal is asserted when a fault condition has been detected on one or more power converters. Avoid connecting this terminal to an interrupt component since this component has a buried interrupt service routine that needs to respond to faults as soon as possible. The intended usage model for this terminal is driving other logic or pins.

Sequencer Control Inputs – Input *

General purpose inputs with user defined polarity that may be used to gate power-up sequencing state changes, to force partial or complete power-down sequencing or both. These terminals are visible when a non-zero value is entered into the **Number of control inputs** parameter on the General tab of the Configure dialog.

Sequencer Status Outputs – Output *

General purpose outputs with user defined polarity that can be asserted and de-asserted at any point throughout the sequencing process to indicate the sequencer's progress. These terminals are visible when a non-zero value is entered into the **Number of status outputs** parameter on the General tab of the Configure dialog.

Power Converter Enables – Output

Power converter enable outputs. When asserted, these outputs enable the selected power converter so that it will begin regulating power to its output.

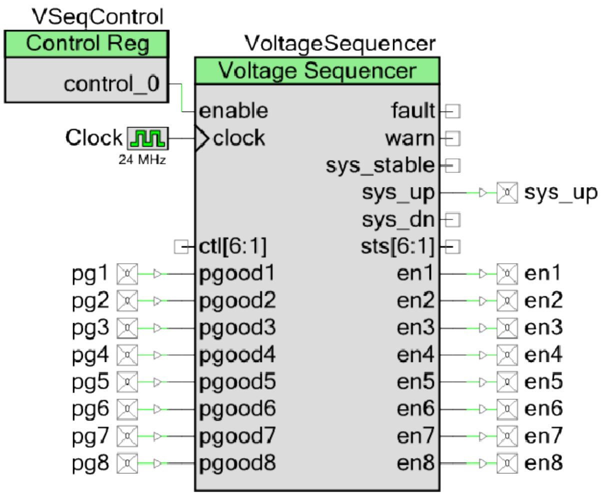
Power Converter Power Goods – Input

Power converter power good status inputs. These signals may come directly from the power converter status output pins or be derived inside PSoC from ADC monitoring of power converter voltage outputs (using the **PowerMonitor** component, for example) or over-voltage/under-voltage window comparator threshold detection (using the **VoltageFaultDetector** component, for example).

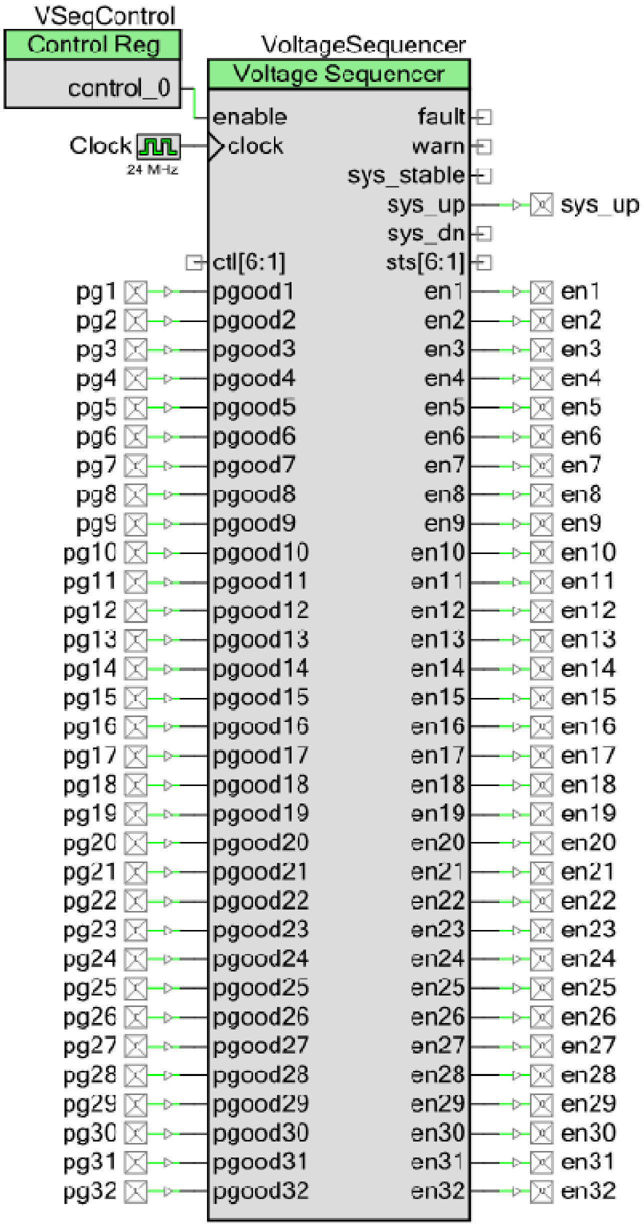
Schematic Macro Information

By default, the PSoC Creator Component Catalog contains two Schematic Macro implementations for the Voltage Sequencer component. These macros contain the Voltage Sequencer component already connected to digital pin components. The Schematic Macros use the Voltage Sequencer component configured for 8 and 32 power converters, as shown in the following diagrams.

8 Rail Voltage Sequencer



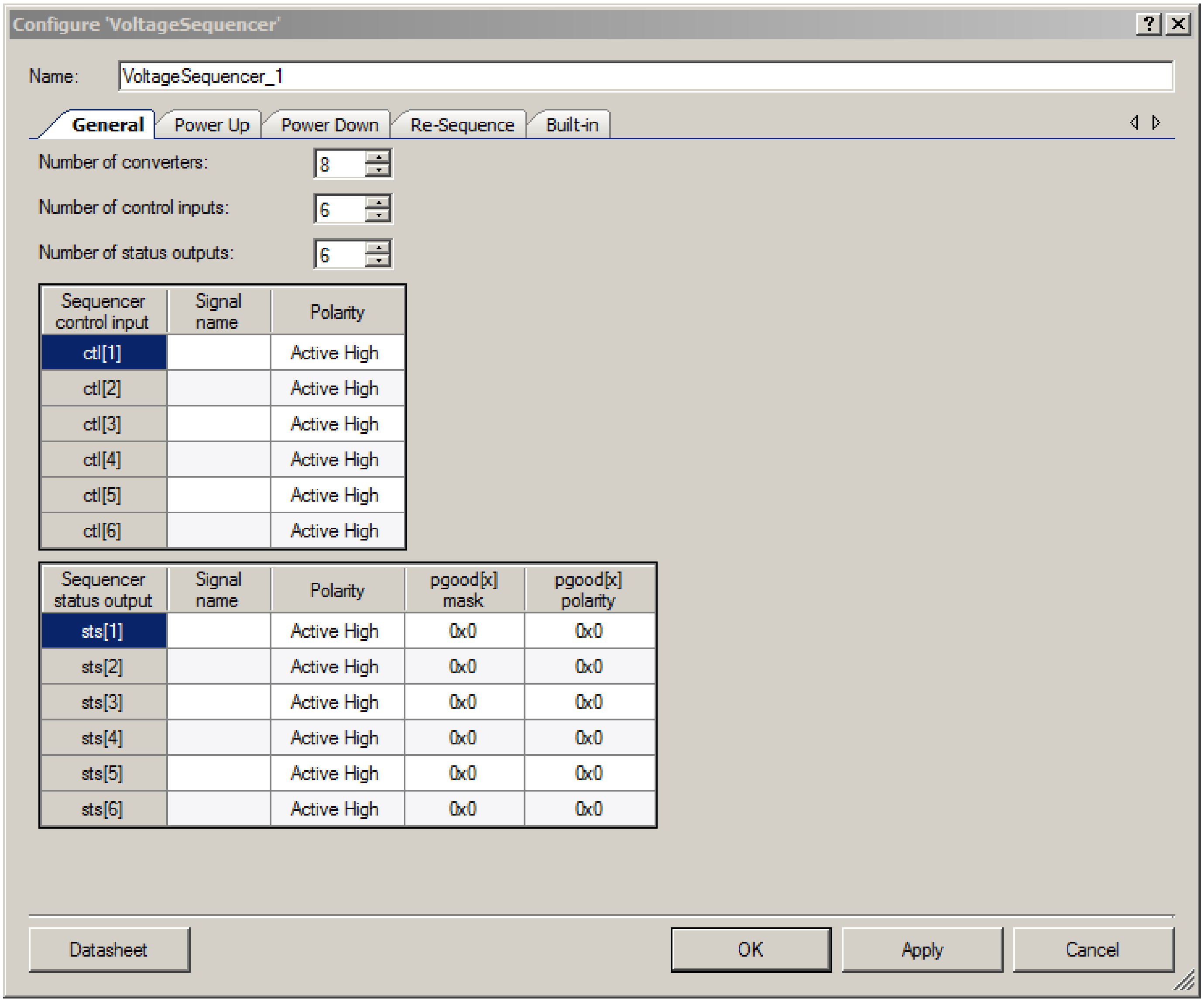
32 Rail Voltage Sequencer



Component Parameters

Drag a Voltage Sequencer component onto your design and double click it to open the Configure dialog. This dialog has three tabs to guide you through the process of setting up the Voltage Sequencer component.

General Tab



Name: VoltageSequencer_1

Number of converters: 8

Number of control inputs: 6

Number of status outputs: 6

Sequencer control input	Signal name	Polarity
ctl[1]		Active High
ctl[2]		Active High
ctl[3]		Active High
ctl[4]		Active High
ctl[5]		Active High
ctl[6]		Active High

Sequencer status output	Signal name	Polarity	pgood[x] mask	pgood[x] polarity
sts[1]		Active High	0x0	0x0
sts[2]		Active High	0x0	0x0
sts[3]		Active High	0x0	0x0
sts[4]		Active High	0x0	0x0
sts[5]		Active High	0x0	0x0
sts[6]		Active High	0x0	0x0

Number of converters

Number of power converters to sequence. Range=1-32. (Default=8).

Number of control inputs

Number of general purpose control inputs. Range=0-6. (Default=1).

Number of status outputs

Number of general purpose status outputs. Range=0-6. (Default=1).

ctl[x] Signal name

Text field, 16 characters, for annotation purposes only. Use it to enter a descriptive name of the control input signal. By default this field is empty and no value is required. It will be visible only when the **Number of control inputs** parameter is non-zero.

ctl[x] Polarity

Options=Active High or Active Low. It will be visible only when the **Number of control inputs** parameter is non-zero. (Default = Active High).

sts[x] Signal name

Text field, 16 characters, for annotation purposes only. Use it to enter a descriptive name of the status output signal. By default this field is empty and no value is required. It will be visible only when the **Number of status outputs** parameter is non-zero.

sts[x] Polarity

Options=Active High or Active Low. It will be visible only when the **Number of status outputs** parameter is non-zero. (Default = Active High).

pgood[x] mask

Hexadecimal encoding of which pgood[x] signals participate in the logic equation for the sts[x] output where bit 0 corresponds to pgood[1] and bit 31 corresponds to pgood[32]. The encoding value will display 2, 4, 6 or 8 hex digits depending on the **Number of converters** parameter. The encoding for each bit is as follows:

1=pgood[x] participates

0=pgood[x] does not participate

The hexadecimal encoding can be entered manually, or the helper form can be used to select the participating pgood[x] signals from the array, automatically generating the hexadecimal encoding for you.

It will be visible only when the **Number of status outputs** parameter is non-zero.

(Default = 0)

pgood[x] polarity

Hexadecimal encoding of the polarity of the pgood[x] signal that will be used in the logic equation for the sts[x] output where bit 0 corresponds to pgood[1] and bit 31 corresponds to pgood[32]. The encoding value will display 2, 4, 6 or 8 hex digits depending on the Number of converters parameter. The encoding for each bit is as follows:

> 1=use the true pgood[x] in the logic equation
>
> 0=use the inverted pgood[x] in the logic equation

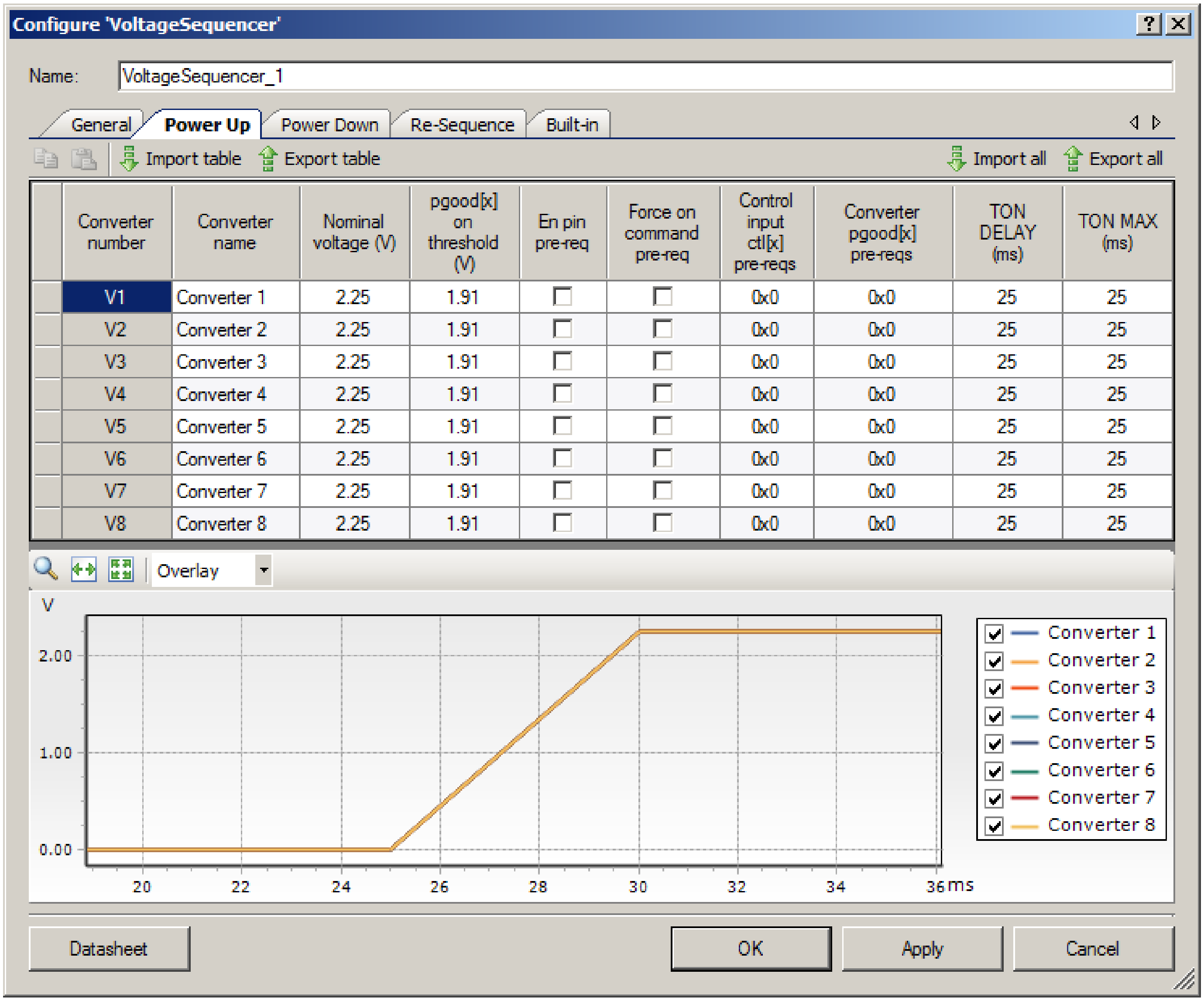
The associated sts[x] is the logical AND of the pgood[x] signals of the selected power converters.

The hexadecimal encoding can be entered manually, or the helper form can be used to select the participating pgood[x] signals from the array, automatically generating the hexadecimal encoding for you.

It will be visible only when the **Number of status outputs** parameter is non-zero.

(Default = 0)

Power Up Tab



Configure 'VoltageSequencer'

Name: VoltageSequencer_1

	Converter number	Converter name	Nominal voltage (V)	pgood[x] on threshold (V)	En pin pre-req	Force on command pre-req	Control input ctl[x] pre-reqs	Converter pgood[x] pre-reqs	TON DELAY (ms)	TON MAX (ms)
	V1	Converter 1	2.25	1.91	☐	☐	0x0	0x0	25	25
	V2	Converter 2	2.25	1.91	☐	☐	0x0	0x0	25	25
	V3	Converter 3	2.25	1.91	☐	☐	0x0	0x0	25	25
	V4	Converter 4	2.25	1.91	☐	☐	0x0	0x0	25	25
	V5	Converter 5	2.25	1.91	☐	☐	0x0	0x0	25	25
	V6	Converter 6	2.25	1.91	☐	☐	0x0	0x0	25	25
	V7	Converter 7	2.25	1.91	☐	☐	0x0	0x0	25	25
	V8	Converter 8	2.25	1.91	☐	☐	0x0	0x0	25	25

Overlay

Converter name

Text field, 16 characters, for annotation purposes only. Use it to enter a descriptive name of the power converter. By default this field is empty and no value is required.

Nominal Voltage (V)

Nominal converter output voltage. Annotation purposes only. Range=0.01–65.54.

pgood[x] on threshold

Minimum converter output voltage required to be considered good during power up sequencing. Range=0.01–65.54. Must be <= **Nominal Voltage** parameter for that converter.

En pin pre-req

When true, the associated power converter will not begin power-up sequencing until the enable pin toggles from low to high. (Default = False).

Force on command pre-req

When true, the associated power converter will not begin power-up sequencing until instructed to do so by calling the ForceOn() or ForceAllOn() APIs. (Default = False).

Control input ctl[x] pre-reqs

Hexadecimal encoding of which ctl[x] signals are pre-requisite inputs required for the associated power converter to power-up. The encoding for each bit is as follows:

1=ctl[x] is a pre-requisite for power-up sequencing

0= ctl[x] is not a pre-requisite for power-up sequencing

The hexadecimal encoding can be entered manually, or the helper form can be used to select the participating ctl[x] signals from the array, automatically generating the hexadecimal encoding for you.

(Default = 0)

Converter pgood[x] pre-reqs

Hexadecimal encoding of which pgood[x] signals are pre-requisite inputs required for the associated power converter to power-up. The encoding for each bit is as follows:

1=pgood[x] is a pre-requisite for power-up sequencing

0= pgood[x] is not a pre-requisite for power-up sequencing

The hexadecimal encoding can be entered manually, or the helper form can be used to select the participating pgood[x] signals from the array, automatically generating the hexadecimal encoding for you.

(Default = 0)

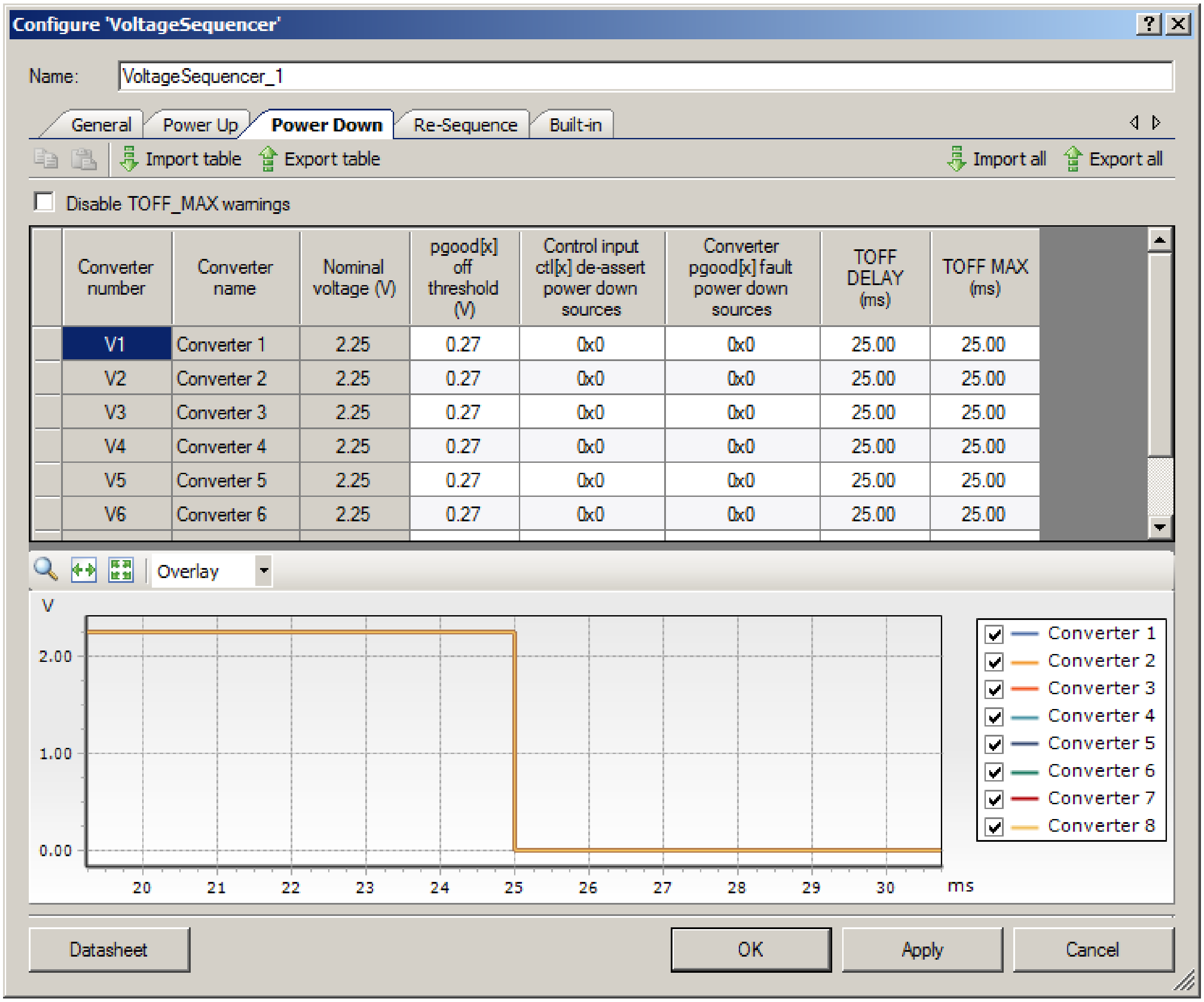
TON delay (ms)

Turn on delay. The time between all sequencing pre-requisites being met and enabling the power converter. Units are ms. Step size is 0.25 ms. Range=0–65535 (0-16.384 s).(Default = 25).

TON_MAX (ms)

Maximum turn on delay. The maximum time allowable between enabling the power converter and it asserting its **pg** status output. Units are ms. Step size is 0.25 ms. Range=0–65535 (0-16.384 s). (Default = 25).

Power Down Tab



Name: VoltageSequencer_1

□ Disable TOFF_MAX warnings

Converter number	Converter name	Nominal voltage (V)	pgood[x] off threshold (V)	Control input ctl[x] de-assert power down sources	Converter pgood[x] fault power down sources	TOFF DELAY (ms)	TOFF MAX (ms)
V1	Converter 1	2.25	0.27	0x0	0x0	25.00	25.00
V2	Converter 2	2.25	0.27	0x0	0x0	25.00	25.00
V3	Converter 3	2.25	0.27	0x0	0x0	25.00	25.00
V4	Converter 4	2.25	0.27	0x0	0x0	25.00	25.00
V5	Converter 5	2.25	0.27	0x0	0x0	25.00	25.00
V6	Converter 6	2.25	0.27	0x0	0x0	25.00	25.00

Disable TOFF_MAX warnings

Globally enable or disable warnings caused by TOFF_MAX_WARN_LIMIT timeouts

Options = Checked or un-checked. Disabling this option removes the **warn** terminal from the symbol (Default = Un-Checked).

Converter name

Text field, 16 characters, for annotation purposes only. This is a display (not editable) brought forward from the Power Up tab.

Nominal voltage (V)

Nominal converter output voltage for annotation purposes only. This is a display (not editable) brought forward from the Power Up tab.

pgood[x] off threshold (V)

The voltage level that the power converter output must drop to in order to be considered powered-off. Range=0.00–65.54. Must be <= Nominal Voltage

Control input ctl[x] de-assert power down sources

Hexadecimal encoding of which ctl[x] signals will force the associated power converter to power-down when they are de-asserted. The encoding for each bit is as follows:

> 1=ctl[x] will force a power-down sequence when de-asserted
>
> 0= ctl[x] will not force a power-down sequence when de-asserted

The hexadecimal encoding can be entered manually, or the helper form can be used to select the participating ctl[x] signals from the array, automatically generating the hexadecimal encoding for you.

(Default = 0)

Converter pgood[x] fault power down sources

Hexadecimal encoding of which pgood[x] signals will force the associated power converter to power-down when they are de-asserted. The encoding for each bit is as follows:

> 1=pgood[x] will force a power-down sequence when de-asserted
>
> 0= pgood[x] will not force a power-down sequence when de-asserted

The hexadecimal encoding can be entered manually, or the helper form can be used to select the participating pgood[x] signals from the array, automatically generating the hexadecimal encoding for you.

(Default = 0)

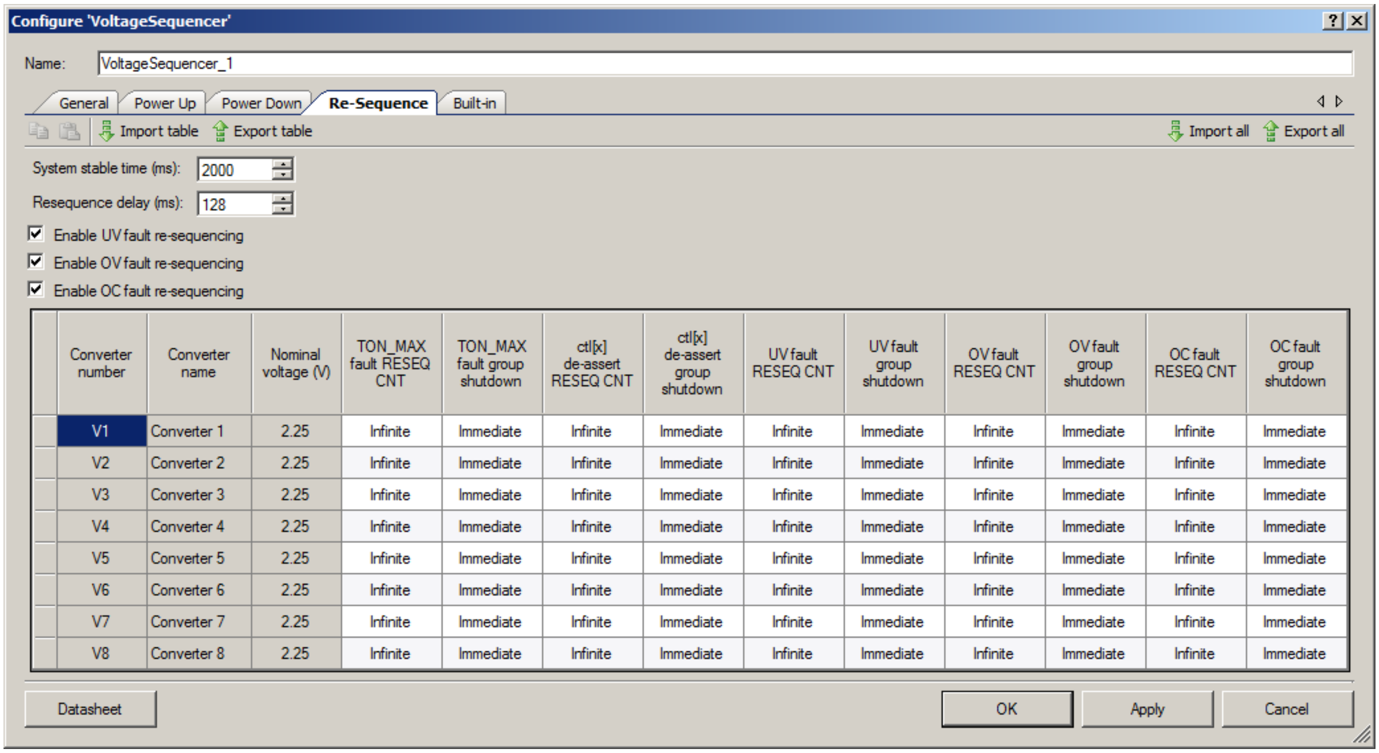
TOFF delay (ms)

Turn off delay. The time between initiating a power-down of the associated power converter and actually de-asserting the **en** output to the power converter. Units are ms. Step size is 0.25 ms. Range=0–65535 (0-16.384 s). Set to 0 for immediate shutdown. (Default = 25).

TOFF_MAX (ms)

Maximum turn off max delay. The maximum time allowable between disabling the power converter and it de-asserting its **pg** status output. Units are ms. Step size is 0.25 ms. Range=0–65535 (0-16.384 s). (Default = 25).

Re-Sequence Tab



Name: VoltageSequencer_1

System stable time (ms): 2000

Resequence delay (ms): 128

☑ Enable UV fault re-sequencing

☑ Enable OV fault re-sequencing

☑ Enable OC fault re-sequencing

Converter number	Converter name	Nominal voltage (V)	TON_MAX fault RESEQ CNT	TON_MAX fault group shutdown	ctl[x] de-assert RESEQ CNT	ctl[x] de-assert group shutdown	UV fault RESEQ CNT	UV fault group shutdown	OV fault RESEQ CNT	OV fault group shutdown	OC fault RESEQ CNT	OC fault group shutdown
V1	Converter 1	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate
V2	Converter 2	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate
V3	Converter 3	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate
V4	Converter 4	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate
V5	Converter 5	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate
V6	Converter 6	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate
V7	Converter 7	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate
V8	Converter 8	2.25	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate	Infinite	Immediate

System stable time (ms)

Number of ms that all power converters must remain in the ON state before the system is considered “stable”. 16-bit value, 8 ms resolution, 0-524 sec range. (Default = 2000).

Resequence delay (ms)

Global re-sequence delay for all power converter state machines. Units are steps of 8 ms. Range=0-65535 (0-534.28 s). (Default = 128).

Enable UV fault re-sequencing

Checking this option gives you the ability to enter automatic re-sequencing parameters unique to under voltage fault conditions. This component cannot determine the specific source of power converter faults, so only enable this option if your design has the capability to do so.

Checking this option will disable (by hiding) the re-sequencing parameters for pgood[x] fault conditions. (Default = Unchecked).

Enable OV fault re-sequencing

Checking this option gives you the ability to enter automatic re-sequencing parameters unique to over voltage fault conditions. This component cannot determine the specific source of power converter faults, so only enable this option if your design has the capability to do so.

Checking this option will disable (by hiding) the re-sequencing parameters for pgood[x] fault conditions. (Default = Unchecked).

Enable OC fault re-sequencing

Checking this option gives you the ability to enter automatic re-sequencing parameters unique to over current fault conditions. This component cannot determine the specific source of power converter faults, so only enable this option if your design has the capability to do so.

Checking this option will disable (by hiding) the re-sequencing parameters for pgood[x] fault conditions. (Default = Unchecked).

Converter name

Text field, 16 characters, for annotation purposes only. This is a display (not editable) brought forward from the Power Up tab.

Nominal voltage (V)

Nominal converter output voltage for annotation purposes only. This is a display (not editable) brought forward from the Power Up tab.

TON MAX fault RESEQ CNT

TON_MAX fault re-sequence count for the associated power converter. Options=None, 1-30, Infinite. (Default = Infinite)

TON MAX fault group shutdown

TON_MAX fault group shutdown response pull-down box. Options=Soft or Immediate. When “Soft” is chosen, the power down delay time for each slave is determined by the **TOFF delay** parameter set for that slave in the Power Down tab. (Default = Immediate).

ctl[x] de-assert RESEQ CNT

Ctl[x] fault re-sequence count for the associated power converter. Options=None, 1-30, Infinite. (Default = Infinite).

ctl[x] de-assert group shutdown

Ctl[x] fault group shutdown response pull-down box. Options=Soft or Immediate. When “Soft” is chosen, the power down delay time for each slave is determined by the **TOFF delay** parameter set for that slave in the Power Down tab. (Default = Immediate).

pgood[x] de-assert RESEQ CNT

pgood[x] fault re-sequence count for the associated power converter. Options=None, 1-30, Infinite. (Default = Infinite).

pgood[x] de-assert group shutdown

pgood[x] fault group shutdown response pull-down box. Options=Soft or Immediate. When “Soft” is chosen, the power down delay time for each slave is determined by the **TOFF delay** parameter set for that slave in the Power Down tab. (Default = Immediate).

UV fault RESEQ CNT

UV fault re-sequence count for the associated power converter. Options=None, 1-30, Infinite. (Default = Infinite).

UV fault group shutdown

UV fault group shutdown response pull-down box. Options=Soft or Immediate. When “Soft” is chosen, the power down delay time for each slave is determined by the **TOFF delay** parameter set for that slave in the Power Down tab. (Default = Immediate).

OV fault RESEQ CNT

OV fault re-sequence count for the associated power converter. Options=None, 1-30, Infinite. (Default = Infinite).

OV fault group shutdown

OV fault group shutdown response pull-down box. Options=Soft or Immediate. When “Soft” is chosen, the power down delay time for each slave is determined by the **TOFF delay** parameter set for that slave in the Power Down tab. (Default = Immediate).

OC fault RESEQ CNT

OC fault re-sequence count for the associated power converter. Options=None, 1-30, Infinite. (Default = Infinite).

OC fault group shutdown

group shutdown response pull-down box. Options=Soft or Immediate. When “Soft” is chosen, the power down delay time for each slave is determined by the **TOFF delay** parameter set for that slave in the Power Down tab. (Default = Immediate).

Application Programming Interface

Application Programming Interface (API) routines allow you to configure the component using software. The following table lists and describes the interface to each function. The subsequent sections cover each function in more detail.

By default, PSoC Creator assigns the instance name “ VoltageSequencer_1” to the first instance of a component in a given design. You can rename the instance to any unique value that follows the syntactic rules for identifiers. The instance name becomes the prefix of every global function name, variable, and constant symbol. For readability, the instance name used in the following table is “Sequencer”.

Function	Description
Sequencer_Start()	Enables the component and places all power converter state machines into the appropriate state
Sequencer_Stop()	Disables the component
Sequencer_Init()	Initializes the component
Sequencer_Enable()	Enables the component
Sequencer_SetCtlPolarity()	Sets the polarity of the selected general purpose sequencer control input
Sequencer_GetCtlPolarity()	Returns the polarity of the selected general purpose sequencer control input
Sequencer_SetStsPgoodMask()	Specifies which pgood[x] signals participate in the generation of the specified general purpose sequencer control output pin
Sequencer_GetStsPgoodMask()	Returns which pgood[x] signals participate in the generation of the specified general purpose sequencer control output pin
Sequencer_SetStsPgoodPolarity()	Configures the logic conditions that will cause the selected general purpose sequencer control output pin to be asserted
Sequencer_GetStsPgoodPolarity()	Returns the polarity of the signals used in the AND expression for the selected general purpose sequencer control output pin
Sequencer_SetPgoodOnThreshold()	Sets the power good voltage threshold for power on detection
Sequencer_GetPgoodOnThreshold()	Returns the power good voltage threshold for power on detection
Sequencer_SetEnPinPrereq()	Determines which power converter state machines have the enable pin as a power up pre-requisite
Sequencer_GetEnPinPrereq()	Returns which power converter state machines have the enable pin as a power up pre-requisite
Sequencer_SetOnCmdPrereq()	Determines which power converter state machines have a call to ForceOn() or ForceAll On() as a power up prerequisite
Sequencer_GetOnCmdPrereq()	Determines which power converter state machines have a call to ForceOn() or ForceAll On() as a power up prerequisite

Function	Description
Sequencer_SetPgoodPrereq()	Determines which pgood[x] pins are power up prerequisites for the selected power converter state machine
Sequencer_GetPgoodPrereq()	Determines which pgood[x] pins are power up prerequisites for the selected power converter state machine
Sequencer_SetTonDelay()	Sets the TON delay parameter for the selected power converter
Sequencer_GetTonDelay()	Returns the TON delay parameter for the selected power converter
Sequencer_SetTonMax()	Sets the TON_MAX parameter for the selected power converter
Sequencer_GetTonMax()	Returns the TON_MAX parameter for the selected power converter
Sequencer_SetPgoodOffThreshold()	Sets the power good voltage threshold for power down detection
Sequencer_GetPgoodOffThreshold()	Returns the power good voltage threshold for power down detection
Sequencer_SetCtlShutdownMask()	Determines which ctl[x] pins will cause the selected power converter to shutdown when de-asserted
Sequencer_GetCtlShutdownMask()	Returns which ctl[x] pins will cause the selected power converter to shutdown when de-asserted
Sequencer_GetCtlStatus()	Returns which ctl[x] pins have caused one or more converters to shutdown
Sequencer_SetPgoodShutdownMask()	Determines which other pgood[x] pins will shutdown the selected power converter when de-asserted
Sequencer_GetPgoodShutdownMask()	Returns which other pgood[x] pins will shutdown the selected power converter when de-asserted
Sequencer_SetToffDelay()	Sets the TOFF delay parameter for the selected power converter
Sequencer_GetToffDelay()	Returns the TOFF delay parameter for the selected power converter
Sequencer_SetToffMax()	Sets the TOFF_MAX_DELAY parameter for the selected power converter
Sequencer_GetToffMax()	Returns the TOFF_MAX_DELAY parameter for the selected power converter
Sequencer_SetSysStableTime()	Sets the global System Stable parameter for all power converter state machines
Sequencer_GetSysStableTime()	Returns the global System Stable parameter for all power converter state machines
Sequencer_SetReseqDelay()	Sets the global Re-sequence Delay parameter for all power converter state machines
Sequencer_GetReseqDelay()	Returns the global Re-sequence Delay parameter for all power converter state machines
Sequencer_SetTonMaxReseqCnt()	Sets the re-sequence count for TON_MAX fault condition.
Sequencer_GetTonMaxReseqCnt()	Returns the re-sequence count for TON_MAX fault conditions

Function	Description
Sequencer_SetTonMaxFaultResp()	Sets the shutdown mode for a fault group when a TON_MAX fault condition occurs on the selected master converter
Sequencer_GetTonMaxFaultResp()	Returns the shutdown mode for a fault group when a TON_MAX fault condition occurs on the selected master converter
Sequencer_SetCtlReseqCnt()	Sets the re-sequence count for fault conditions due to de-asserted ctl[x] inputs
Sequencer_GetCtlReseqCnt()	Returns the re-sequence count for fault conditions due to de-asserted ctl[x] inputs
Sequencer_SetCtlFaultResp()	Sets the shutdown mode for a fault group in response to fault conditions due to de-asserted ctl[x] inputs
Sequencer_GetCtlFaultResp()	Returns the shutdown mode for a fault group in response to fault conditions due to de-asserted ctl[x] inputs
Sequencer_SetFaultReseqSrc()	Sets the power converter fault re-sequence sources
Sequencer_GetFaultReseqSrc()	Returns the power converter fault re-sequence sources
Sequencer_SetPgoodReseqCnt()	Sets the re-sequence count for fault conditions due to de-asserted pgood[x] inputs
Sequencer_GetPgoodReseqCnt()	Returns the re-sequence count for fault conditions due to de-asserted pgood[x] inputs
Sequencer_SetPgoodFaultResp()	Sets the shutdown mode for a fault group due to de-asserted pgood[x] inputs
Sequencer_GetPgoodFaultResp()	Returns the shutdown mode a fault group due to de-asserted pgood[x] inputs
Sequencer_SetOvReseqCnt()	Sets the re-sequence count for over-voltage (OV) fault conditions
Sequencer_GetOvReseqCnt()	Returns the re-sequence count for over-voltage (OV) fault conditions
Sequencer_SetOvFaultResp()	Sets the shutdown mode for a fault group due to overvoltage (OV) fault conditions
Sequencer_GetOvFaultResp()	Returns the shutdown mode for a fault group due to overvoltage (OV) fault conditions
Sequencer_SetUvReseqCnt()	Sets the re-sequence count for under-voltage (UV) fault conditions
Sequencer_GetUvReseqCnt()	Returns the re-sequence count for under-voltage (UV) fault conditions
Sequencer_SetUvFaultResp()	Sets the shutdown mode for a fault group due to undervoltage (UV) fault conditions
Sequencer_GetUvFaultResp()	Returns the shutdown mode for a fault group due to undervoltage (UV) fault conditions
Sequencer_SetOcReseqCnt()	Sets the re-sequence count for over-current (OC) fault conditions
Sequencer_GetOcReseqCnt()	Returns the re-sequence count for over-current (OC) fault conditions
Sequencer_SetOcFaultResp()	Sets the shutdown mode for a fault group due to overcurrent (OC) fault

Function	Description
	conditions
Sequencer_GetOcFaultResp()	Returns the shutdown mode for a fault group due to overcurrent (OC) fault conditions
Sequencer_EnFaults()	Enables/disables assertion of the fault output signal
Sequencer_SetFaultMask()	Sets which power converters have fault detection enabled
Sequencer_GetFaultMask()	Returns which power converters have fault detection enabled
Sequencer_GetFaultStatus()	Returns a bit mask containing the pgood fault status for all power converters
Sequencer_EnWarnings()	Enables/disables assertion of the warn output signal
Sequencer_SetWarnMask()	Sets which power converters have warnings enabled
Sequencer_GetWarnMask()	Returns which power converters have warnings enabled
Sequencer_GetWarnStatus()	Returns a bit mask containing TOFF_MAX_WARN warning status for all power converters
Sequencer_GetState()	Returns the current state machine state for the selected power converter
Sequencer_ForceOff()	Forces the selected power converter to power down either immediately or after the TOFF delay
Sequencer_ForceAllOff()	Forces all power converters to power down either immediately or after their TOFF delays
Sequencer_ForceOn()	Forces the selected power converter to power up
Sequencer_ForceAllOn()	Forces all power converters to power up

Global Variables

Variable	Description
Sequencer_initVar	Indicates whether the Voltage Sequencer has been initialized
Sequencer_ctlPolarity	Polarity of the general purpose control inputs
Sequencer_ctlShutdownMaskList[]	Defines which ctl[x] pins will cause shutdown for each converter
Sequencer_stsPgoodMaskList[]	Defines which pgood[x] pins are used to generate each sts[x] output
Sequencer_stsPgoodPolarityList[]	Defines the logic conditions for generation of each sts[x] output
Sequencer_pgoodOnThresholdList[]	Defines power good voltage threshold for power on detection
Sequencer_enPinPrereqMask	Defines which converter have the enable pin as a power up pre-requisite
Sequencer_onCmdPrereqMask	Defines which converter have a call to ForceOn() or ForceAllOn() as a power up pre-requisite.
Sequencer_ctlPrereqList[]	Defines which ctl[x] pins are power up pre-requisites for each converter

Variable	Description
Sequencer_pgoodPrereqList[]	Defines which pgood[x] pins are power up pre-requisites for each converter
Sequencer_tonDelayList[]	Defines TON_DELAY parameter for each power converter
Sequencer_tonMaxDelayList[]	Defines TON_MAX_DELAY parameter for each power converter
Sequencer_pgoodOffThresholdList[]	Defines power good voltage threshold for power off detection
Sequencer_pgoodShutdownMaskList[]	Defines which pgood[x] pins will cause shutdown for each converter
Sequencer_toffDelayList[]	Defines TOFF_DELAY parameter for each power converter
Sequencer_toffMaxDelayList[]	Defines TOFF_MAX_DELAY parameter for each power converter
Sequencer_sysStableTime	System Stable Time parameter
Sequencer_globalReseqDelay	Global TRESEQ_DELAY parameter
Sequencer_tonMaxFaultReseqCfg[]	Defines the re-sequence configuration for TON_MAX fault conditions
Sequencer_ctlFaultReseqCfg []	Defines the re-sequence configuration for CTL fault conditions
Sequencer_faultReseqSrcList[]	Defines the power converter fault re-sequence sources
Sequencer_pgoodFaultReseqCfg[]	Defines the re-sequence configuration for pgood fault conditions
Sequencer_ovFaultReseqCfg[]	Defines the re-sequence configuration for OV fault conditions
Sequencer_uvFaultReseqCfg[]	Defines the re-sequence configuration for UV fault conditions
Sequencer_ocFaultReseqCfg[]	Defines the re-sequence configuration for OC fault conditions
Sequencer_faultEnable	Enable/disable assertion of the fault output signal
Sequencer_faultMask	Defines which power converters have fault detection enabled
Sequencer_faultStatus	Bit mask containing the pgood fault status for all power converters
Sequencer_warnEnable	Enable/disable assertion of the warn output signal
Sequencer_warnStatus	Bit mask containing TOFF_MAX_WARN warning status for all power converters
Sequencer_warnMask	Defines which power converters have warnings enabled
Sequencer_ctlStatus	Bit mask containing which ctl[x] pins have caused a shutdown

void Sequencer_Start(void)

Description: Enables the component and places all power converter state machines into the appropriate state (OFF or PEND_ON). Calls the Init() API if the component has not been initialized before. Calls the Enable() API.

Parameters: None

Return Value: None

Side Effects: None

void Sequencer_Stop (void)

Description: Disables the component

Parameters: None

Return Value: None

Side Effects: All output terminals are de-asserted

void Sequencer_Init(void)

Description: Initializes the component. Parameter settings are initialized based on parameters entered into the customizer.

Parameters: None

Return Value: None

Side Effects: None

void Sequencer_Enable(void)

Description: Enables the component.

Parameters: None

Return Value: None

Side Effects: None

void Sequencer_SetCtlPolarity(uint8 ctlNum, uint8 ctlPolarity)

Description: Sets the polarity of the selected general purpose sequencer control input (ctl[x])

Parameters: uint8 ctlNum
Specifies the control pin number
Valid range: 1-6

uint8 ctlPolarity
Specifies the polarity of the control pin
Options: 1=active high, 0=active low

Return Value: None

Side Effects: None

uint8 Sequencer_GetCtlPolarity(uint8 ctlNum)

Description: Returns the polarity of the selected general purpose sequencer control input (ctl[x])

Parameters: uint8 ctlNum
Specifies the control pin number
Valid range: 1-6

Return Value: uint8 ctlPolarity
Specifies the polarity of the control pin
Options: 1=active high, 0=active low

Side Effects: None

void Sequencer_SetStsPgoodMask(uint8 stsNum, uint32 stsPgoodMask)

Description: Specifies which pgood[x] signals participate in the generation of the specified general purpose sequencer control output pins (sts[x])

Parameters: uint8 stsNum
Specifies the status pin number
Valid range: 1-6

uint32 stsPgoodMask

Bit Field	Status Pgood Mask
0	1=Sts output depends on pgood[1]
1	1=Sts output depends on pgood[2]

…	…
31	1=Sts output depends on pgood[32]

Return Value: None

Side Effects: None

uint32 Sequencer_GetStsPgoodMask(uint8 stsNum)

Description: Returns which pgood[x] signals participate in the generation of the specified general purpose sequencer control output pins (sts[x])

Parameters: uint8 stsNum
Specifies the status pin number
Valid range: 1-6

Return Value: uint32 stsPgoodMask

Bit Field	Status Pgood Mask
0	1=Sts output depends on pgood[1]
1	1=Sts output depends on pgood[2]
…	…
31	1=Sts output depends on pgood[32]

Side Effects: None

void Sequencer_SetStsPgoodPolarity(uint8 stsNum, uint32 pgoodPolarity)

Description: Configures the logic conditions that will cause the selected general purpose sequencer control output pins (sts[x]) to be asserted

Parameters: uint8 stsNum
Specifies the status pin number
Valid range: 1-6

uint32 stsPgoodPolarity
Specifies the polarity of the pgood[x] signal required to assert the specified sts[x] signal

Bit Field	Status Polarity
0	0=pgood[1] must be low, 1=pgood[1] must be high
1	0=pgood[2] must be low, 1=pgood[2] must be high
…	…

31	0=pgood[32] must be low, 1=pgood[32] must be high

Return Value: None

Side Effects: None

uint32 Sequencer_GetStsPgoodPolarity(uint8 stsNum)

Description: Returns the polarity of the **pgood[x]** signals used in the AND expression for the selected general purpose sequencer control output (sts[x]).

Parameters: uint8 stsNum
Specifies the status pin number
Valid range: 1-6

Return Value: uint32 stsPgoodPolarity
Specifies the polarity of the pgood[x] signal required to assert the specified sts[x] signal

Bit Field	Status Polarity
0	0=pgood[1] must be low, 1=pgood[1] must be high
1	0=pgood[2] must be low, 1=pgood[2] must be high
…	…
31	0=pgood[32] must be low, 1=pgood[32] must be high

Side Effects: None

void Sequencer_SetPgoodOnThreshold(uint8 converterNum, uint16 onThreshold)

Description: Sets the power good voltage threshold for power on detection

Parameters: uint8 ctlNum
Specifies the converter number
Valid range: 1-32

uint16 onThreshold
Specifies the power good power on threshold in mV
Valid range: 0-65535

Return Value: None

Side Effects: None

uint16 Sequencer_GetPgoodOnThreshold(uint8 converterNum)

Description: Returns the power good voltage threshold for power on detection

Parameters: uint8 ctlNum
Specifies the converter number
Valid range: 1-32

Return Value: uint16 onThreshold
Specifies the power good power on threshold in mV
Valid range: 0-65535

Side Effects: None

void Sequencer_SetEnPinPrereq(uint32 converterMask)

Description: Determines which power converter state machines have the enable pin as a power up pre-requisite

Parameters: uint32 converterMask

Bit Field	Converter Mask
0	1=power converter 1 has the enable signal as a sequencing pre-requisite
1	1=power converter 2 has the enable signal as a sequencing pre-requisite
…	…
31	1=power converter 32 has the enable signal as a sequencing pre-requisite

Return Value: None

Side Effects: None

uint32 Sequencer_GetEnPinPrereq(void)

Description: Returns which power converter state machines have the enable pin as a power up pre-requisite

Parameters: None

Return Value: uint32 converterMask

Bit Field	Converter Mask
0	1=power converter 1 has the enable signal as a sequencing pre-requisite

1	1=power converter 2 has the enable signal as a sequencing pre-requisite
…	…
31	1=power converter 32 has the enable signal as a sequencing pre-requisite

Side Effects: None

void Sequencer_SetOnCmdPrereq(uint32 converterMask)

Description: Determines which power converter state machines have a call to ForceOn() or ForceAllOn() as a power up pre-requisite

Parameters: uint32 converterMask

Bit Field	Converter Mask
0	1=power converter 1 has a call to ForceOn() or ForceAllOn() a sequencing pre-requisite
1	1=power converter 2 has a call to ForceOn() or ForceAllOn() a sequencing pre-requisite
…	…
31	1=power converter 32 has a call to ForceOn() or ForceAllOn() a sequencing pre-requisite

Return Value: None

Side Effects: None

uint32 Sequencer_GetOnCmdPrereq(void)

Description: Returns which power converter state machines have a a call to ForceOn() or ForceAllOn() as a power up pre-requisite

Parameters: None

Return Value: uint32 converterMask

Bit Field	Converter Mask
0	1=power converter 1 has a call to ForceOn() or ForceAllOn() a sequencing pre-requisite
1	1=power converter 2 has a call to ForceOn() or ForceAllOn() a sequencing pre-requisite
…	…
31	1=power converter 32 has a call to ForceOn() or ForceAllOn() a sequencing pre-requisite

Side Effects: None

void Sequencer_SetPgoodPrereq(uint8 converterNum, uint32 pgoodMask)

Description: Determines which pgood[x] pins are power up pre-requisites for the selected power converter state machine

Parameters: uint8 converterNum
Specifies the power converter state machine number
Valid range: 1-32

uint32 pgoodMask
Specifies which pgood[x] pins are power up pre-requisites for the selected power converter

Bit Field	Power Good Power Up Pre-Requisite Mask
0	1=pgood[1] must be asserted
1	1=pgood[2] must be asserted
…	…
31	1=pgood[32] must be asserted

Return Value: None

Side Effects: None

uint32 Sequencer_GetPgoodPrereq(uint8 converterNum)

Description: Determines which pgood[x] pins are power up pre-requisites for the selected power converter state machine

Parameters: uint8 converterNum
Specifies the power converter state machine number
Valid range: 1-32

Return Value: uint32 pgoodMask
Specifies which pgood[x] pins are power up pre-requisites for the selected power converter

Bit Field	Power Good Power Up Pre-Requisite Mask
0	1=pgood[1] must be asserted
1	1=pgood[2] must be asserted
…	…
31	1=pgood[32] must be asserted

Side Effects: None

void Sequencer_SetTonDelay(uint8 converterNum, uint16 tonDelay)

Description: Sets the **TON** delay parameter for the selected power converter. Defined as the time between a state machine's pre-requisites all becoming satisfied and the en[x] being asserted

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint16 tonDelay
units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Return Value: None

Side Effects: None

uint16 Sequencer_GetTonDelay(uint8 converterNum)

Description: Returns the **TON** delay parameter for the selected power converter. Defined as the time between a state machine's pre-requisites all becoming satisfied and the en[x] being asserted

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint16 tonDelay
units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Side Effects: None

void Sequencer_SetTonMax(uint8 converterNum, uint16 tonMax)

Description: Sets the **TON_MAX** parameter for the selected power converter. Defined as the maximum time allowable between a power converter's en[x] being asserted and pgood[x] being asserted. Failure to do so generates a fault condition

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint16 tonMax
units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Return Value: None

Side Effects: None

uint16 Sequencer_GetTonMax(uint8 converterNum)

Description: Returns the **TON_MAX** parameter for the selected power converter. Defined as the maximum time allowable between a power converter's en[x] being asserted and pgood[x] being asserted. Failure to do so generates a fault condition

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint16 tonMax
units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Side Effects: None

void Sequencer_SetPgoodOffThreshold(uint8 converterNum, uint16 onThreshold)

Description: Sets the power good voltage threshold for power off detection

Parameters: uint8 ctlNum
Specifies the converter number
Valid range: 1-32

uint16 offThreshold
Specifies the power good power off threshold in mV
Valid range: 0-65535

Return Value: None

Side Effects: None

uint16 Sequencer_GetPgoodOffThreshold(uint8 converterNum)

Description: Returns the power good voltage threshold for power off detection

Parameters: uint8 ctlNum
Specifies the converter number
Valid range: 1-32

Return Value: uint16 offThreshold

Specifies the power good power off threshold in mV
Valid range: 0-65535

Side Effects: None

void Sequencer_SetCtlShutdownMask(uint8 converterNum, uint8 ctlPinMask)

Description: Determines which ctl[x] pins will cause the selected power converter to shutdown when de-asserted

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 ctlPinMask
Specifies which ctl[x] pins can cause a shutdown

Bit Field	Control Pin Shutdown Mask
0	1=ctl[1] de-assertion will shutdown the converter
1	1=ctl[2] de-assertion will shutdown the converter
…	…
5	1=ctl[6] de-assertion will shutdown the converter
7..6	Reserved. Set to zeroes

Return Value: None

Side Effects: None

uint8 Sequencer_GetCtlShutdownMask(uint8 converterNum)

Description: Returns which ctl[x] pins will cause the selected power converter to shutdown when de-asserted

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 ctlPinMask
Specifies which ctl[x] pins can generate fault conditions

Bit Field	Control Pin Shutdown Mask
0	1=ctl[1] de-assertion will shutdown the converter

1	1=ctl[2] de-assertion will shutdown the converter
…	…
5	1=ctl[6] de-assertion will shutdown the converter
7..6	Reserved. Set to zeroes

Side Effects: None

uint8 Sequencer_GetCtlStatus(void)

Description: Returns which ctl[x] pins have caused one or more converters to shutdown

Parameters: None

Return Value: uint8 ctlStatus
Specifies which ctl[x] pins have caused a shutdown

Bit Field	Control Pin Shutdown Mask
0	1=ctl[1] de-assertion caused a shutdown
1	1=ctl[2] de-assertion caused a shutdown
…	…
5	1=ctl[6] de-assertion caused a shutdown
7..6	Reserved. Set to zeroes

Side Effects: None

void Sequencer_SetPgoodShutdownMask(uint8 converterNum, uint32 pgoodMask)

Description: Determines which other pgood[x] pins will shutdown the selected power converter when de-asserted.

Note that the pgood[converterNum] pin is automatically a fault source for the selected power converter whether or not the corresponding bit in the pgoodMask is set or not.

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint32 pgoodMask
Specifies which pgood[x] pins can cause a shutdown

Bit Field	Power Good Mask
0	1=pgood[1] de-assertion will shutdown the converter

1	1=pgood[2] de-assertion will shutdown the converter
…	…
31	1=pgood[32] de-assertion will shutdown the converter

Return Value: None

Side Effects: None

uint32 Sequencer_GetPgoodShutdownMask (uint8 converterNum)

Description: Returns which other pgood[x] pins will shutdown the selected power converter when de-asserted

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint32 pgoodMask
Specifies which pgood[x] pins can cause a shutdown

Bit Field	Power Good Mask
0	1=pgood[1] de-assertion will shutdown the converter
1	1=pgood[2] de-assertion will shutdown the converter
…	…
31	1=pgood[32] de-assertion will shutdown the converter

Side Effects: None

void Sequencer_SetToffDelay(uint8 converterNum, uint16 toffDelay)

Description: Sets the TOFF delay parameter for the selected power converter. Defined as the time between making the decision to turn a power converter of and to actually de-asserting the en[x] pin

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint16 toffDelay
units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Return Value: None

Side Effects: None

uint16 Sequencer_GetToffDelay(uint8 converterNum)

Description: Returns the TOFF delay parameter for the selected power converter. Defined as the time between making the decision to turn a power converter of and to actually de-asserting the en[x] pin

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint16 toffDelay
units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Side Effects: None

void Sequencer_SetToffMax(uint8 converterNum, uint16 toffMax)

Description: Sets the TOFF_MAX_DELAY parameter for the selected power converter. Defined as the maximum time allowable between a power converter's en[x] being de-asserted and power converter actually turning off. Failure to do so generates a warning condition

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint16 toffMax
units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Return Value: None

Side Effects: None

uint16 Sequencer_GetToffMax(uint8 converterNum)

Description: Returns the TOFF_MAX_DELAY parameter for the selected power converter. Defined as the maximum time allowable between a power converter's en[x] being de-asserted and power converter actually turning off. Failure to do so generates a warning condition

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint16 toffMax

units = 0.25 ms per LSB
Valid Range=0-65535 (0-16.384 s)

Side Effects: None

void Sequencer_SetSysStableTime(uint16 stableTime)

Description: Sets the global TRESEQ_DELAY parameter for all power converter state machines. Defined as the time between making the decision to re-sequence and beginning a new power up sequence

Parameters: uint16 stableTime
units = 8 ms per LSB
Valid Range=0-65535 (0-534.28 s)

Return Value: None

Side Effects: None

uint16 Sequencer_GetSysStableTime(void)

Description: Sets the global TRESEQ_DELAY parameter for all power converter state machines. Defined as the time between making the decision to re-sequence and beginning a new power up sequence

Parameters: None

Return Value: uint16 stableTime
units = 8 ms per LSB
Valid Range=0-65535 (0-534.28 s)

Side Effects: None

void Sequencer_SetReseqDelay(uint16 reseqDelay)

Description: Sets the global TRESEQ_DELAY parameter for all power converter state machines. Defined as the time between making the decision to re-sequence and beginning a new power up sequence

Parameters: uint16 reseqDelay
units = 8 ms per LSB
Valid Range=0-65535 (0-534.28 s)

Return Value: None

Side Effects: None

uint16 Sequencer_GetReseqDelay(void)

Description: Returns the global TRESEQ_DELAY parameter for all power converter state machines. Defined as the time between making the decision to re-sequence and beginning a new power up sequence

Parameters: None

Return Value: uint16 reseqDelay
units = 8 ms per LSB
Valid Range=0-65535 (0-534.28 s)

Side Effects: None

void Sequencer_SetTonMaxReseqCnt(uint8 converterNum, uint8 ReseqCnt)

Description: Sets the re-sequence count for TON_MAX fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 reseqCnt
5 bit number
Options: 0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing counts

Return Value: None

Side Effects: None

uint8 Sequencer_GetTonMaxReseqCnt(uint8 converterNum)

Description: Returns the re-sequence count for TON_MAX fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 reseqCnt
5 bit number
Options: 0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing counts

Side Effects: None

void Sequencer_SetTonMaxFaultResp(uint8 converterNum, uint8 faultResponse)

Description: Sets the shutdown mode for the fault group when a TON_MAX fault condition occurs on the selected master converter

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

uint8 Sequencer_GetTonMaxFaultResp(uint8 converterNum)

Description: Returns the shutdown mode for the fault group when a TON_MAX fault condition occurs on the selected master converter

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

Return Value: uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Side Effects: None

void Sequencer_SetCtlReseqCnt(uint8 converterNum, uint8 reseqCnt)

Description: Sets the re-sequence count for fault conditions due to de-asserted ctl[x] inputs

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing,
1-30=valid re-sequencing counts

Return Value: None

Side Effects: None

uint8 Sequencer_GetCtlReseqCnt(uint8 converterNum)

Description: Returns the re-sequence count for fault conditions due to de-asserted ctl[x] inputs

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing,
1-30=valid re-sequencing counts

Side Effects: None

void Sequencer_SetCtlFaultResp(uint8 converterNum, uint8 faultResponse)

Description: Sets the shutdown mode for the fault group in response to fault conditions due to de-asserted ctl[x] inputs

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

uint8 faultResponse
Specifies the shutdown mode for a fault group
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

uint8 Sequencer_GetCtlFaultResp(uint8 converterNum)

Description: Returns the shutdown mode for the fault group in response to fault conditions due to de-asserted ctl[x] inputs

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

Return Value: uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Side Effects: None

void Sequencer_SetFaultReseqSrc(uint8 converterNum, uint8 reseqSrc)

Description: Sets the power converter fault re-sequence sources

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 reseqSrc

Bit Field	Re-Sequence Source
0	1=OV fault source enabled
1	1=UV fault source enabled
2	1=OC fault source enabled
7:3	Reserved

Return Value: None

Side Effects: When reseqSrc is zero, power good (pgood) inputs become the fault re-sequence source.

uint8 Sequencer_GetFaultReseqSrc(uint8 converterNum)

Description: Returns the power converter fault re-sequence source

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 reseqSrc

Bit Field	Re-Sequence Source
0	1=OV fault source enabled
1	1=UV fault source enabled
2	1=OC fault source enabled
7:3	Reserved

Side Effects: None

void Sequencer_SetPgoodReseqCnt(uint8 converterNum, uint8 reseqCnt)

Description: Sets the re-sequence count for fault conditions due to de-asserted pgood[x] inputs

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing,
1-30=valid re-sequencing counts

Return Value: None

Side Effects: None

uint8 Sequencer_GetPgoodReseqCnt(uint8 converterNum)

Description: Returns the re-sequence count for fault conditions due to de-asserted pgood[x] inputs

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing,
1-30=valid re-sequencing counts

Side Effects: None

void Sequencer_SetPgoodFaultResp(uint8 converterNum, uint8 faultResponse)

Description: Sets the shutdown mode for the fault group for fault conditions due to de-asserted pgood[x] inputs

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

uint8 Sequencer_GetPgoodFaultResp(uint8 converterNum)

Description: Sets the shutdown mode for the fault group for fault conditions due to de-asserted pgood[x] inputs

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

Return Value: uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Side Effects: None

void Sequencer_SetOvReseqCnt(uint8 converterNum, uint8 reseqCnt)

Description: Sets the re-sequence count for over-voltage (OV) fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing counts

Return Value: None

Side Effects: None

uint8 Sequencer_GetOvReseqCnt(uint8 converterNum)

Description: Sets the re-sequence count for over-voltage (OV) fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing

counts

Side Effects: None

void Sequencer_SetOvFaultResp(uint8 converterNum, uint8 faultResponse)

Description: Sets the shutdown mode for the fault group due to over-voltage (OV) fault conditions

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

uint8 Sequencer_GetOvFaultResp(uint8 converterNum)

Description: Returns the shutdown mode for the fault group due to over-voltage (OV) fault conditions

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

Return Value: uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Side Effects: None

void Sequencer_SetUvReseqCnt(uint8 converterNum, uint8 reseqCnt)

Description: Sets the re-sequence count for under-voltage (UV) fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing counts

Return Value: None

Side Effects: None

uint8 Sequencer_GetUvReseqCnt(uint8 converterNum)

Description: Returns the re-sequence count for under-voltage (UV) fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing counts

Side Effects: None

void Sequencer_SetUvFaultResp(uint8 converterNum, uint8 faultResponse)

Description: Sets the shutdown mode for the fault group due to under-voltage (UV) fault conditions

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

uint8 Sequencer_GetUvFaultResp(uint8 converterNum)

Description: Returns the shutdown mode for the fault group due to under-voltage (UV) fault conditions

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

Return Value: uint8 faultResponse

Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Side Effects: None

void Sequencer_SetOcReseqCnt(uint8 converterNum, uint8 reseqCnt)

Description: Sets the re-sequence count for over-current (OC) fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing counts

Return Value: None

Side Effects: None

uint8 Sequencer_GetOcReseqCnt(uint8 converterNum)

Description: Returns the re-sequence count for over-current (OC) fault conditions

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 reseqCnt
5 bit number
0=no re-sequencing, 31=infinite re-sequencing, 1-30=valid re-sequencing counts

Side Effects: None

void Sequencer_SetOcFaultResp(uint8 converterNum, uint8 faultResponse)

Description: Sets the shutdown mode for the fault group due to over-current (OC) fault conditions

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

uint8 faultResponse

Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

uint8 Sequencer_GetOcFaultResp(uint8 converterNum)

Description: Returns the shutdown mode for the fault group due to over-current (OC) fault conditions

Parameters: uint8 converterNum
Specifies the master power converter number
Valid range: 1-32

Return Value: uint8 faultResponse
Specifies the shutdown mode for the fault group
Options: 0=immediate, 1=soft

Side Effects: None

void Sequencer_EnFaults(uint8 faultEnable)

Description: Enables/disables assertion of the fault output signal. Faults are still processed by the state machine and fault status is still available through the GetFaultStatus() API.

Parameters: uint8 faultEnable
Options: 0=disabled, 1=enabled
Enabled when the component is started

Return Value: None

Side Effects: None

void Sequencer_SetFaultMask(uint32 faultMask)

Description: Sets which power converters have fault detection enabled

Parameters: uint32 faultMask
All bits are set when the component is started

Bit Field	Fault Mask
0	1=enable fault detection for power converter 1
1	1=enable fault detection for power converter 2

…	…
31	1=enable fault detection for power converter 32

Return Value: None

Side Effects: None

void Sequencer_GetFaultMask(uint32 faultMask)

Description: Returns which power converters have fault detection enabled

Parameters: None

Return Value: uint32 faultMask
Fault mask of power converters

Bit Field	Fault Mask
0	1=fault detection for power converter 1 is enabled
1	1=fault detection for power converter 2 is enabled
…	…
31	1=fault detection for power converter 32 is enabled

Side Effects: None

uint32 Sequencer_GetFaultStatus(void)

Description: Returns a bit mask containing the pgood fault status for all power converters. Bits are sticky until cleared by calling this API.

Parameters: None

Return Value: uint32 faultStatus
Fault status of power converters

Bit Field	Fault Status
0	1=power converter 1 has/had a pgood fault
1	1=power converter 2 has/had a pgood fault
…	…
31	1=power converter 32 has/had a pgood fault

Side Effects: Calling this API de-asserts the fault output pin

void Sequencer_EnWarnings(uint8 warnEnable)

Description: Enables/disables assertion of the warn output signal. Warning status is still available through the GetWarningStatus() API.

Parameters: uint8 warnEnable
Options: 0=disabled, 1=enabled

Enabled when the component is started

Return Value: None

Side Effects: None

void Sequencer_SetWarnMask(uint32 warnMask)

Description: Sets which power converters have warnings enabled

Parameters: uint32 warnMask
All bits are cleared when the component is started

Bit Field	Warning Mask
0	1=enable warnings for power converter 1
1	1= enable warnings for power converter 2
…	…
31	1= enable warnings for power converter 32

Return Value: None

Side Effects: None

void Sequencer_GetWarnMask(uint32 warnMask)

Description: Returns which power converters have warnings enabled

Parameters: None

Return Value: uint32 warnMask
Warn mask of power converters

Bit Field	Warning Mask
0	1=warnings for power converter 1 are enabled
1	1=warnings for power converter 2 are enabled
…	…
31	1=warnings for power converter 32 are enabled

Side Effects: None

uint32 Sequencer_GetWarnStatus(void)

Description: Returns a bit mask containing TOFF_MAX_WARN warning status for all power converters. Bits are sticky until cleared by calling this API.

Parameters: None

Return Value: uint32 warnStatus
Warning status of power converters

Bit Field	Warning Status
0	1=power converter 1 has/had a warning
1	1=power converter 2 has/had a warning
…	…
31	1=power converter 32 has/had a warning

Side Effects: Calling this API de-asserts the warn output pin

uint8 Sequencer_GetState(uint8 converterNum)

Description: Returns the current state machine state for the selected power converter.

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: uint8 state
Power converter state machine state

Encoding	State
0	OFF
1	PEND_ON
2	TON_DELAY
3	TON_MAX
4	ON
5	TOFF_DELAY
6	TOFF_MAX
7	PEND_RESEQ
8	TRESEQ_DELAY
9..255	Undefined

Side Effects: None

void Sequencer_ForceOff(uint8 converterNum, uint8 powerOffMode)

Description: Forces the selected power converter to power down either immediately or after the TOFF delay

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

uint8 powerOffMode
Specifies the shutdown mode
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

void Sequencer_ForceAllOff(uint8 powerOffMode)

Description: Forces all power converters to power down either immediately or after their TOFF delays

Parameters: uint8 powerOffMode
Specifies the shutdown mode
Options: 0=immediate, 1=soft

Return Value: None

Side Effects: None

void Sequencer_ForceOn(uint8 converterNum)

Description: Forces the selected power converter to power up

Parameters: uint8 converterNum
Specifies the power converter number
Valid range: 1-32

Return Value: None

Side Effects: If the selected power converter state machine was in the OFF state, this API call will cause the state machine to transition into the PEND_ON state

void Sequencer_ForceAllOn(void)

Description: Forces all power converter to power up

Parameters: None

Return Value: None

Side Effects: If any power converter state machines were in the OFF state, this API call will cause them to transition into the PEND_ON state

API Constants

Name	Description
NUMBER_OF_CONVERTERS	Number of converters to sequence
NUMBER_OF_CTL_INPUTS	Number of sequencer control inputs
NUMBER_OF_STS_OUTPUTS	Number of sequencer status outputs
INFINITE_RESEQUENCING	Fixed value = 31 (from PMBus specification)

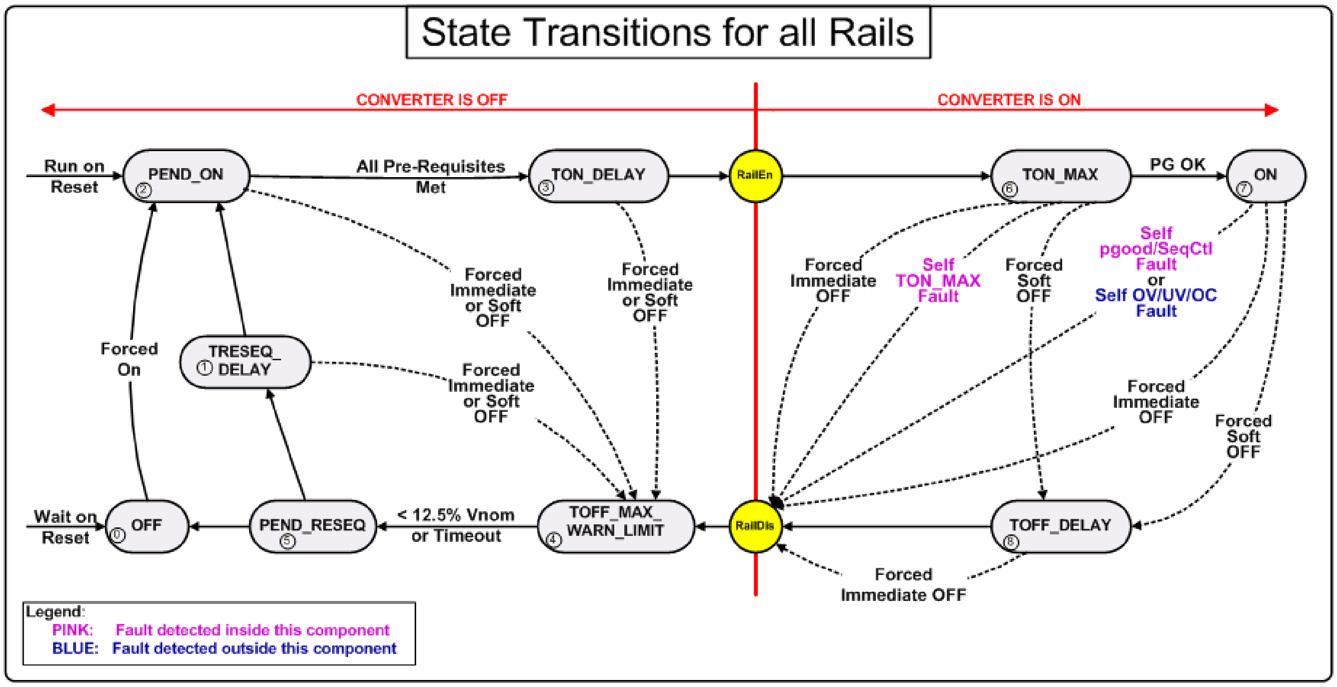
Sample Firmware Source Code

PSoC Creator provides numerous example projects that include schematics and example code in the Find Example Project dialog. For component-specific examples, open the dialog from the Component Catalog or an instance of the component in a schematic. For general examples, open the dialog from the Start Page or **File** menu. As needed, use the **Filter Options** in the dialog to narrow the list of projects available to select.

Refer to the “Find Example Project” topic in the PSoC Creator Help for more information.

Functional Description

To support complex event-based sequencing, management of each power converter is done through an independent firmware state machine that drives the enable output (en[x]) for the associated power converter. Each power converter has its own state machine. The state transition flow is shown in the diagram below.



At the start of time (after a power on reset for example), all state machines for all of the power converters begin in either the OFF state or the PEND_ON state under user control. The state machine for each power converter then transitions to a new state depending on how the user defines the sequencing conditions. Power converter fault conditions also drive the associated state machine to a new state as defined by the user. In the diagram above, the two identified fault response transitions (highlighted in pink and blue color) refer to faults that have occurred on this power converter. At any given point in time, any of the state machines can be in any one of the defined states.

State machine transitions for every power converter are always handled in the Sequencer State Machine ISR that gets invoked every 250 µs or 500 µs depending on the number of converters to sequence. The component has a built-in tick timer clock source, which is automatically configured to produce the appropriate time reference for this ISR. When a power converter’s state machine is in the ON state and a fault occurs, the Fault Handler ISR will be invoked. The Fault Handler ISR is responsible for time critical activities such as disabling the faulted power converter immediately. It also sets a fault flag that will be recognized the next time the Sequencer State Machine ISR is invoked. The Sequencer State Machine ISR will then take care of non-time critical fault handling activities such as state machine transitions.

In most real-world applications, power converters have a relationship to each other – they are not truly independent. This may occur when multiple power converters supply power to a single chip or a group of chips. In that case, when one power converter fails, the other power converters must be shutdown also. Another example is that there may be a hardware enforced relationship between two or more power converters. For example, the output of one power converter may be the power supply input of another power converter. In that case, when the primary power converter faults and will be shutdown, it is required to shut down the secondary power converter also because it will lose power anyway.

To support these use cases, fault conditions on one power converter state machine must be able to influence state transitions of the state machines for other power converters. To address this requirement, the concept of a Fault Group is introduced. If the user specifies that a fault on one power converter must force a shutdown on one or more operational power converters, then the operational power converters are referred to as the faulty power converter's Fault Group.

The Fault Group can be configured to shut down immediately or go through a soft shutdown process with user-configurable delays. When there is a hardware enforced relationship between power converters, the Fault Group that draws power from the faulty power converter must be set for immediate shutdown to ensure fault conditions are not generated by the Fault Group that is powering down.

Registers

The Voltage Sequencer component has several control and status registers that are used by the firmware APIs to control operation and monitor status. None of these registers are accessible directly by user firmware.

Resources

The Voltage Sequencer component is almost entirely firmware based. The component utilizes the following resources.

Configuration	Resource Type					
	Datapath Cells	Macrocells	Status Cells	Control Cells	DMA Channels	Interrupts
8 Converters	–	21	3	5	–	3
16 Converters	–	39	4	7	–	3
24 Converters	–	57	5	9	–	3
32 Converters	–	75	6	11	–	3

API Memory Usage

The component memory usage varies significantly, depending on the compiler, device, number of APIs used and component configuration. The following table provides the memory usage for all APIs available in the given component configuration.

The measurements have been done with associated compiler configured in Release mode with optimization set for Size. For a specific design the map file generated by the compiler can be analyzed to determine the memory usage.

Configuration	PSoC 3 (Keil_PK51)		PSoC 5 (GCC)		PSoC 5LP (GCC)	
	Flash Bytes	SRAM Bytes	Flash Bytes	SRAM Bytes	Flash Bytes	SRAM Bytes
8 Converters	7877	349	5236	397	5240	397
16 Converters	8199	627	5840	705	5836	705
24 Converters	8514	905	6140	1013	6144	1013
32 Converters	8819	1183	6404	1321	6408	1321

DC and AC Electrical Characteristics

Specifications are valid for –40 °C ≤ T_A ≤ 85 °C and T_J ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted.

DC and AC Characteristics

Parameter	Description	Min	Typ	Max	Unit
f_{CLOCK}	Component Clock frequency	–	–	66	MHz
f_{BUS_CLK}	Min Bus Clock Frequency				
	8 Converters	20	–	–	MHz
	16 Converters	30	–	–	MHz
	17-32 Converters	40	–	–	MHz
$t_{TRANSITION}$	Sequencer state transition time				
	Up to 16 Converters	–	250	275	µs
	17-32 Converters	–	500	550	µs
t_{FAULT_RESP}	Fault response time	$1/f_{CLOCK}$	–	100	ns
t_{ON_DELAY}	Programmable power-on delay	0	–	16.384	s
t_{OFF_DELAY}	Programmable power-off delay	0	–	16.384	s

Component Changes

This section lists the major changes in the component from the previous version.

Version	Description of Changes	Reason for Changes / Impact
2.0	Complete redesign. Voltage Sequencer v2.0 is NOT AT ALL backwards compatible with previous versions.	Redesigned sequencing engine, to support time-based sequencing as well as more convoluted and complicated event-based sequencing.

Version	Description of Changes	Reason for Changes / Impact
Version	Description of Changes	
1.50	Updated for compatibility with PSoC Creator v2.0 Re-classified as “Concept” component Corrected intermittent sequencing time delay error	
1.40	Status register clocking scheme changed in Verilog file to improve timing performance Corrected error detecting pgood failure of previously enabled rails during up sequencing (error was introduced in v1.30)	
1.30	Symbol colors and size updated, resource utilization updated, references to power management APIs removed	
1.20	Updated to support PSoC 5	
1.10	Updated for PSoC Creator 1.0 Beta 5 compatibility	
1.0	First release	

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.