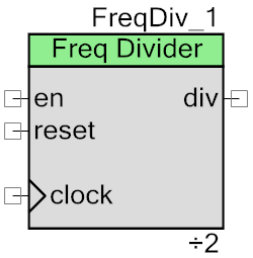
Frequency Divider

1.0



Features

- Divides a clock or arbitrary signal by a specified value.
- Enable and Reset inputs to control and align divided output.

General Description

The Frequency Divider component produces an output that is the clock input divided by the specified value.

When to Use a Frequency Divider

Use the Frequency Divider as a simple clock divider for UDB components, or to divide the frequency of another signal.

The global clock dividers provided by the Clock component are more efficient and have more features, but the Frequency Divider is especially useful when all global clock dividers are used.

Input/Output Connections

This section describes the various input and output connections for the Edge Detector.

en – Input

The en input enables or disables the internal counter. When this signal is low, rising edges on the **clock** input will not be counted, and the **div** output will not change.

reset – Input

The reset input resets the internal counter on the next rising edge of the **clock** input. This causes the **div** output to go low until reset goes low again, restarting the counter.

clock – Input

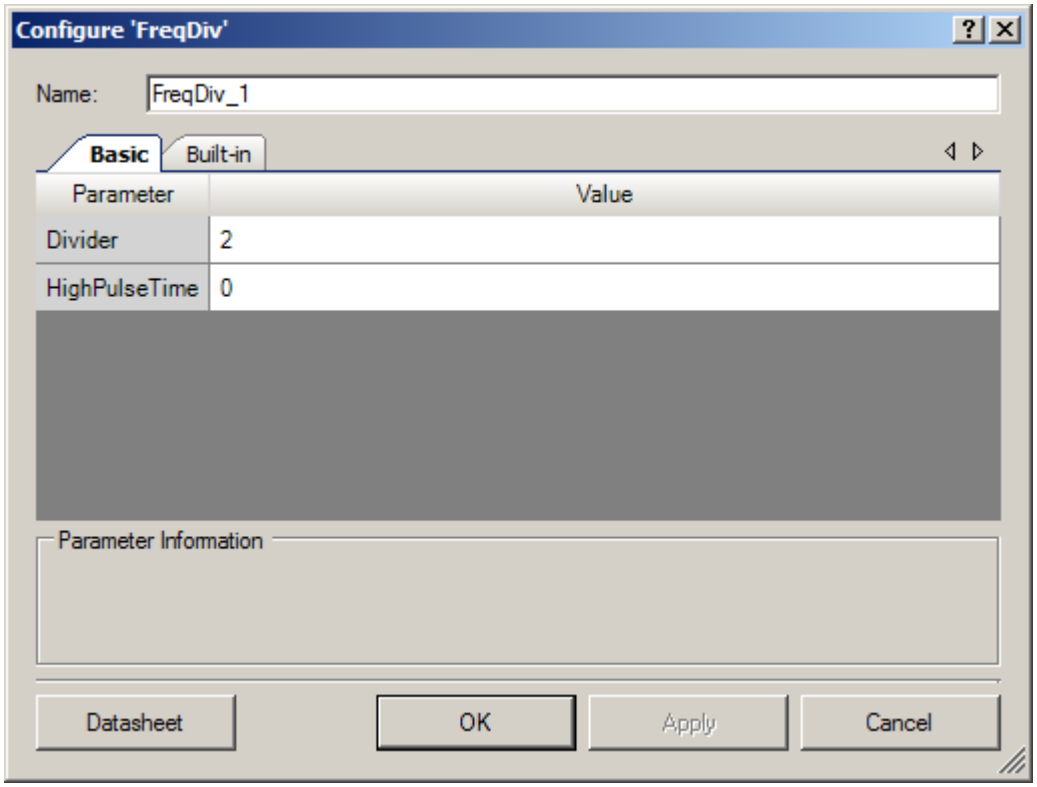
The clock input determines the signal to divide.

div – Output

The div outputs the **clock** signal divided by the specified value.

Component Parameters

Drag a Frequency Divider onto your design and double-click it to open the **Configure** dialog.



The Frequency Divider provides the following parameters.

Divider

The value by which to divide the **clock** input. The value must be between 2 and 4294967295. The default is **2**.

HighPulseTime

This parameter controls the duty cycle of the **div** output. Its value specifies the number of rising edges of the **clock** input for which the **div** output will pulse high for each period. HighPulseTime set to **0** is a special case, and indicates a 50% duty cycle (or slightly greater than 50% for odd values of Divider). The value of HighPulseTime must be less than Divider. The default is **0**.

Functional Description

The Frequency Divider internally uses an N-bit up-counter synthesized in PLDs, where N is the smallest integer greater than or equal to $\log_2$(Divider). This counter value is compared to the Divider parameter and the HighPulseTime parameter to produce the div output.

Figure 1. Divide-by-2 Example

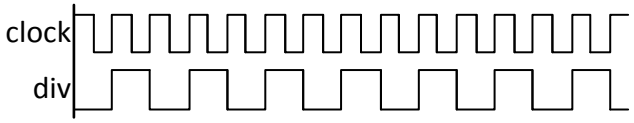


Figure 1 shows the waveform of a Frequency Divider with the Divider parameter set to 2, and the HighPulseTime parameter set to 0, indicating a 50% duty cycle.

Figure 2. Divide-by-3 Example

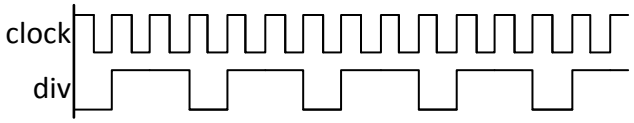


Figure 2 shows the waveform of a Frequency Divider with the Divider parameter set to 3, and the HighPulseTime parameter set to 0, indicating that the duty cycle should be just above 50%.

The duty cycle of the div output can be customized by modifying the HighPulseTime parameter to something other than 0.

Figure 3. Divide-by-4 Example

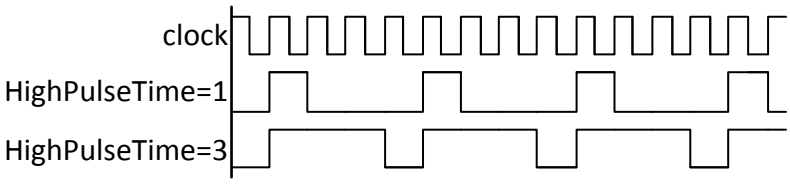


Figure 3 shows two waveforms produced by a Frequency Divider with the Divider parameter set to 4. One waveform has the HighPulseTime parameter set to 1, resulting in a 25% duty cycle. The other has the HighPulseTime parameter set to 3, resulting in a 75% duty cycle.

The Frequency Divider provides optional enable and reset inputs to allow further control of the div output.

The reset input provides control over when the first rising edge of the div output occurs. As long as the reset is high, the div output will remain low. As soon as the reset goes low, the div output will begin generating its specified waveform. Multiple Frequency Divider components can be phase-aligned by connecting them to a common reset signal.

Figure 4. Phase Alignment Using Reset Input

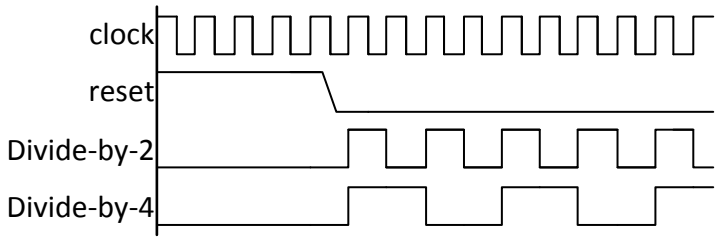


Figure 4 shows the waveforms produces when two Frequency Dividers are connected to the same reset signal in order to phase-align their outputs.

The en input provides a way to gate the clock input, effectively stalling the div output.

Figure 5. Clock Gating Using Enable Input

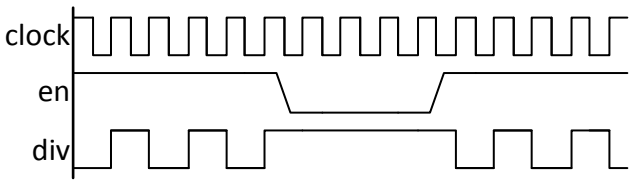


Figure 5 shows the resulting waveform when the enable input is held low for a period of time.

Resources

The Frequency Divider is synthesized to macrocells in the UDB array. Macrocell usage is dependent on optimizations performed during synthesis. Table 1 provides an estimate of the resource usage for different Divider values. For large divider values, it is usually more efficient to use the built-in clock divider in the Clock component.

Table 1. Resource Usage

Configuration	Resource Type					
	Datapath Cells	Macrocells	Status Cells	Control Cells	DMA Channels	Interrupts
Divider = 2	–	3	–	–	–	–
Divider = 10	–	6	–	–	–	–
Divider = 100	–	9	–	–	–	–
Divider = 1000	–	19	–	–	–	–
Divider = 10000	–	36	–	–	–	–
Divider = 100000	–	51	–	–	–	–
Divider = 1000000	–	63	–	–	–	–

MISRA Compliance

This section describes the MISRA-C:2004 compliance and deviations for the component. There are two types of deviations defined: project deviations – deviations that are applicable for all PSoC Creator components and specific deviations – deviations that are applicable only for this component. This section provides information on component specific deviations. The project deviations are described in the MISRA Compliance section of the *System Reference Guide* along with information on the MISRA compliance verification environment.

The Frequency Divider component does not have any C source code APIs.

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.

Table 2. AC Characteristics

Parameter	Description	Min	Typ	Max[1]	Units
f_{CLOCK}	Component clock frequency				
	Divider = 2	–	–	67	MHz
	Divider = 10	–	–	67	MHz
	Divider = 100	–	–	67	MHz
	Divider = 1000	–	–	43	MHz
	Divider = 10000	–	–	34	MHz
	Divider = 100000	–	–	28	MHz
	Divider = 1000000	–	–	21	MHz

Component Changes

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

Version	Description of Changes	Reason for Changes / Impact
1.0.b	Added note about built-in global clock dividers provided by Clock component.	Guide users to use the built-in clock dividers when it makes sense.
1.0.a	Corrected description of “reset” terminal	Previous version of datasheet incorrectly stated that “div” output is held high until reset goes low.

1 The values provide a maximum safe operating frequency of the component. The component may run at higher clock frequencies, at which point validation of the timing requirements with STA results is necessary.

Version	Description of Changes	Reason for Changes / Impact
1.0	First version of the component.	