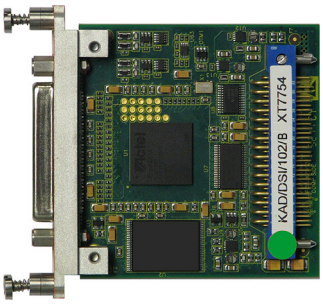
KAD/DSI/102

Discrete input (programmable counters, time tagging) - 24ch

CURTISSWRIGHTDS.com



Key Features

- 24 differential ended, discrete, bi-level input channels
- 24 independent, 32-bit counters with ten different operating modes
- 40ns internal resolution
- FIFO based time tagging
- Resettable counters
- Detection of 0.5µs wide input signal pulses

Applications

- Engine speed measurement
- Data capture from a parallel bus
- Timestamping of events

Overview

The KAD/DSI/102 monitors the status (high/low) of up to 24 differential ended discrete input channels. Each of these has an assigned programmable 32-bit counter. Additionally each of the inputs can be used to trigger time tagged events.

Each counter can be programmed to operate in one of the following modes: Period, Pulse Width, Duty Cycle, Frequency, Events Since Sample, Events Since Power Up, Events Since Reset, Samples Since Power Up, Samples Since Reset, or Time Since Event. The range of each counter is programmable, as is the threshold (within ±28V), hysteresis (0.8 to 20V) and sensitivity to the rising/falling edge.

All the channels control the time tagging to the 1K word deep FIFO (80 bits wide each word). For each input, time tagging can be triggered by a rising edge, falling edge, both edges, or neither (when the channel is disabled). Every time a trigger occurs, an 80-bit word is written to the FIFO consisting of the 24 inputs (configurable to be either input state after the change or value representing the bits which triggered the event) and the 48-bit binary coded decimal IRIG time that the event happened. There are also three FIFO flags which indicate that the FIFO is empty, that a message has been skipped, or that a message is stale.

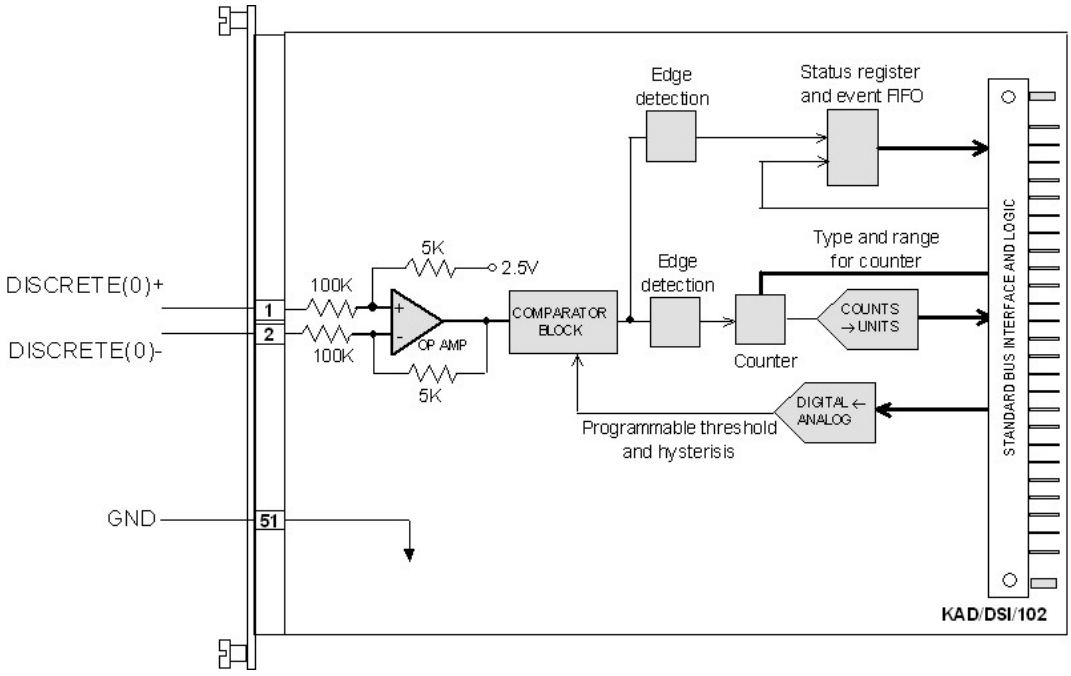


Figure 1: First of 24 discrete inputs; first counter and event FIFO

Specifications

All values provided in the following specification tables are valid within the operating temperature range specified under "Environmental ratings" in the "General specifications" table.

TABLE 1	General specifications				
PARAMETER	MIN.	TYP.	MAX.	UNITS	CONDITION/DETAILS
Slots	–	–	1	–	Can be placed in any user-slot in any combination.
Mass					
	–	80	–	g	
	–	2.82	–	oz	Design metric is grams.
Height above chassis					For recommended clearance requirements see the *CON/KAD/002/CP* data sheet.
bare connector	–	–	11	mm	
bare connector	–	–	0.43	in.	Design metric is millimeters.
Access rate	–	–	2	Msps	Maximum combined access rate for read and write.
Power consumption					
+5V	80	–	120	mA	
±7V	0	–	0	mA	
+12V	0	–	10	mA	
-12V	0	–	0	mA	
total power	0.4	–	0.72	W	Particular combinations of chassis and Acra KAM-500 modules may have power or current limitations. For details, see *TEC/NOT/016 - Power dissipation, TEC/NOT/049 - Power estimation*, and the relevant chassis data sheet.
Environmental ratings					See *Environmental Qualification Handbook*.
operating temperature	-40	–	85	°C	Chassis base/side plate temperature.
storage temperature	-55	–	105	°C	

TABLE 2	Differential ended digital inputs				
PARAMETER	MIN.	TYP.	MAX.	UNITS	CONDITION/DETAILS
Inputs	–	–	24	–	
Input voltage					
operating range	-30	–	30	V	
overvoltage protection	-50	–	50	V	Voltages outside of this range can damage input.
Input signal					
frequency range	0	–	1	MHz	High and low pulses of input signal detected as a result of threshold settings; each pulse must be at least 0.5µs wide (the shortest supported pulse width to be measured). All time and duty cycle measurements are taken based on a 40ns internal resolution and then scaled to output range.
threshold	-28	–	28	V	
threshold step	–	25	–	mV	
hysteresis	0.8	–	20	V	We recommend that hysteresis is set as wide as possible, usually in the order of 2 to 10 times lower than the step between logic levels.
hysteresis step	–	50	–	mV	

TABLE 2	Differential ended digital inputs (continued)				
PARAMETER	MIN.	TYP.	MAX.	UNITS	CONDITION/DETAILS
Threshold accuracy					The error can be determined separately for Threshold Voltage High and Threshold Voltage Low as a summation of the offset error and the gain error. Gain error should be calculated as a percentage of Threshold Voltage High or Low.
offset error	–	0.1	0.4	V	
gain error	–	0.1	0.4	%	
Time accuracy					
internal resolution	–	40	–	ns	Applies to Period, Pulse Width, Duty Cycle, Frequency, and Time Since Event counter operation modes and edge detection.
reference clock precision	–	1	5	ppm	Applies to Period, Pulse Width, Duty Cycle, Frequency and Time Since Event measurements. It is determined by the stability of the Temperature Compensated Crystal Oscillator (TCXO). Maximum value stated is inclusive of aging.
Delay of edge detection	300	500	700	ns	This includes the delay introduced by analog circuits (filter and comparators) and digital logic (noise filtering, reference clock, and system clock resolution). This value is to be understood as offset for Time Since Event measurements. For the Event FIFO time tag error it is below time tag of 1µs resolution, and can therefore be treated as negligible.
Analog filter					
poles	–	1	–	–	
filter cutoff	–	30	–	MHz	
Event FIFO					
FIFO depth	–	–	1,024	msg	msg = messages.
FIFO message width	–	–	80	bits	
time tag resolution	–	1	–	µs	
Input resistance					
between inputs	–	210	–	kΩ	Module powered off.
between inputs	–	210	–	kΩ	Module powered on.
each input to GND	–	105	–	kΩ	Module powered off.
each input to GND	–	105	–	kΩ	Module powered on.
Input impedance					Measured at 100kHz.
between inputs	–	100	–	kΩ	Module powered off.
between inputs	–	100	–	kΩ	Module powered on.
each input to GND	–	50	–	kΩ	Module powered off.
each input to GND	–	50	–	kΩ	Module powered on.

Setting up the KAD/DSI/102

All module setup can be defined in XML using XidML® schemas (see http://www.xidml.org).

Instrument settings

SETUP DATA	CHOICE	DEFAULT	NOTES
Manufacturer	-	-	-
Name	ACRA CONTROL	ACRA CONTROL	Name of manufacturer.
PartReference	KAD/DSI/102/B	KAD/DSI/102/B	The instrument part reference.
SerialNumber	AB1234	AB1234	Unique name for each module.
Settings	-	-	-
On FIFO Empty	Flag Only All High	Flag Only	Configures the Event FIFO operation regarding EventIrigTimeTag. When there are no new messages either the time tag is repeated (Flag Only) or all EventIrigTimeTag bits are set high (All High).
On Event Record	Status After Event Trigger Delta	Status After Event	Configures Event Status subparameter to be either status of discrete inputs after triggering (Status After Event) or to indicate which of discrete inputs have triggered message to be written into the Event FIFO (Trigger Delta).
On FIFO Full Overflow	No Yes	Yes	Configures Event FIFO to overflow (reset) when FIFO gets full and new event message can't be written.
Duty Cycle Mode Timeout	0.1 to 150	1.0	Defines timeout (in seconds) for counters operating in Duty Cycle mode.
Pulse Width Mode Timeout	0.1 to 150	1.0	Defines timeout (in seconds) for counters operating in Pulse Width mode. The value can't be lower than Range Maximum for such counters.
Frequency Mode Reference Period	0.1 to 150	1.0	Sets reference period (in seconds) for Frequency mode of counter in step size of 0.05.
Channels	-	-	-
Discrete(23:0) Discrete Input	-	-	Represents a discrete input channel on an instrument.
Settings	-	-	-
Trigger Edge	None Rising Falling Both	None	Used on discrete channels to specify which edge causes the writing of an event message to the Event FIFO.
Threshold Voltage Level	-28 to 28	2	Specifies the threshold voltage level of the input signal to be recognized as either logic 1 or logic 0 state.
Threshold Voltage Hysteresis	0.8 to 20	1.0	Specifies the hysteresis width. The hysteresis is centered around the Threshold Voltage Level.
Counter(23:0) Counter Input	-	-	Represents counter related to each discrete input.
Settings	-	-	-

SETUP DATA	CHOICE	DEFAULT	NOTES
Trigger Edge	None Rising Falling Both	Rising	Used on counter channels to specify which signal edge causes related counter incrementation or is used as reference edge.
Counter Type	Period Pulse Width Duty Cycle Frequency Events Since Sample Events Since Power Up Events Since Reset Time Since Event Samples Since Power Up Samples Since Event	Events Since Power Up	Specifies how the counter operates.

Parameter definitions

NAME/DESCRIPTION	BASE UNIT	DATA FORMAT	BITS	REGISTER DEFINITION
Global Parameters				
Discrete Current state of discrete inputs.	BitVector	BitVector	32	R[31:0] R[31:24] Reserved - Reserved for future use. R[23:0] Status - The value representing the current state of 24 discrete inputs.
DiscreteHigh Current state of discrete inputs 23 down to 16. Padded with 8 0s at MSB side.	BitVector 16	BitVector		R[15:8] Reserved - Reserved for future use. R[7:0] Status_23_16 - The value representing the current state of 8 discrete inputs, from 23 to 16.
DiscreteLow Current state of discrete inputs 15 down to 0.	BitVector	BitVector	16	R[15:0] Status_15_0 - The value representing the current state of 15 discrete inputs, from15 to 0.
Discrete_23_8 Current state of 16 discrete inputs, 23 to 8.	BitVector	BitVector	16	R[15:0] R[15:0] Status_23_8 - The value representing the current state of 16 discrete inputs, from 23 to 8.
Discrete_11_0 Current state of 12 discrete inputs, 11 to 0. Padded with 4 0s on LSB side.	BitVector	BitVector	16	R[15:0] R[15:4] Status_11_0 - The value representing the current state of 12 discrete inputs, from 11 to 0. R[3:0] Reserved - Reserved for future use.
Counter(23:0) Parameters				
Counter 32-bit register. Depending on the Counter Type setting, it expresses counts, ratio, frequency, or time.	Count Second Second Second Ratio Hertz Count	OffsetBinary OffsetBinary OffsetBinary OffsetBinary OffsetBinary OffsetBinary OffsetBinary	32	R[31:0] Default condition Counter Type = Period Counter Type = Pulse Width Counter Type = Time Since Event Counter Type = Duty Cycle Counter Type = Frequency Other
Parser parameters				

NAME/DESCRIPTION	BASE UNIT	DATA FORMAT	BITS	REGISTER DEFINITION
Event 80-bit wide event message.	BitVector	BitVector	80	R[79:0] R[79:32] EventIrigTimeTag - BCD IRIG time at the start of the event. When the FIFO data are stale (were already read and no new messages are in the FIFO since then), the last values are repeated (On FIFO Empty set to Flag Only). Otherwise all bits are 1 (On FIFO Empty set to All High). R(31) EventEmptyFlag - This bit is active when the Event FIFO is empty after power-up or reprogramming the system, and no message has been written since. R(30) EventStaleFlag - This bit is active when Event FIFO has been emptied and the same data is read more than once (no new messages). R(29) EventSkippedFlag - This bit is active when Event FIFO becomes full and one or more event messages could not be written, therefore were skipped. For On FIFO Full Overflow set to Yes, FIFO is reset additionally. R[28:24] Reserved - Reserved for future use. R[23:0] EventStatus - The value representing state of the 24 discrete inputs after an event or which input has caused triggering (On Event Record set to Status After Event or Trigger Delta respectively). When the FIFO was emptied or no new messages are in the FIFO, the last values are repeated.

Configurable parameters

Counter23:0

SETUP DATA	CHOICE	DEFAULT	NOTES
Range Maximum	4294967295 0 to 150 0 to 150 0 to 1e6 1 0 to 1e6 4294967295	4294967295	The maximum range. Default condition Counter Type = Period Counter Type = Pulse Width Counter Type = Time Since Event Counter Type = Duty Cycle Counter Type = Frequency Other
Range Minimum	0	0	The minimum range.

NOTE: It is recommended that names are less than 20 characters, have no white space or contain any of the following five characters "/><\.

Getting the most from the KAD/DSI/102

Wiring the inputs

For single ended signals with 28V/0V or 28V/open modes (see the following figure), connect the signal to the positive input for each channel. The negative input can be left floating but we recommend that it is tied to ground (GND) as it is then less prone to picking up noise or cross-talk.

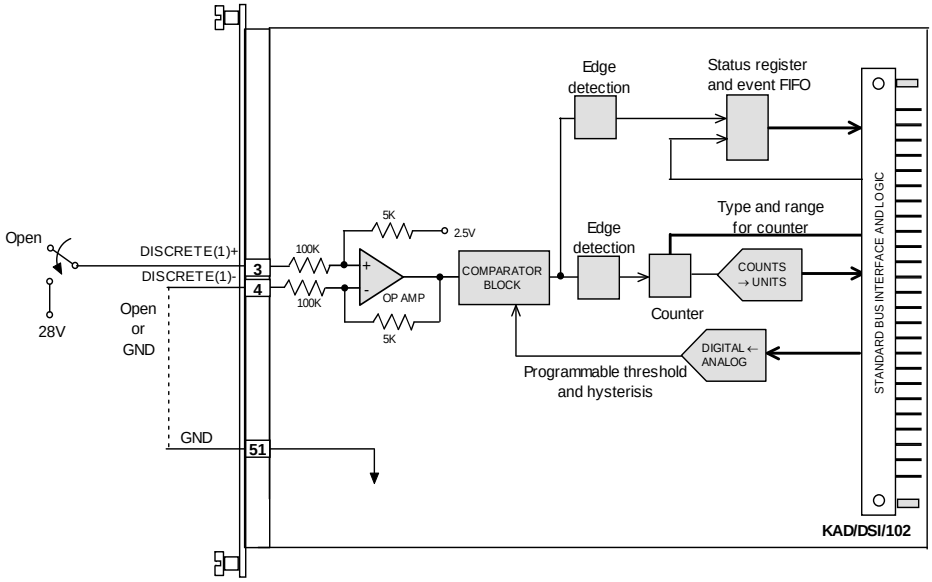


Figure 2: Second channel wired for operation in 28V/open mode

To configure the input to operate correctly with 0V/open mode, an external pull-up resistor is recommended to increase the voltage step seen by the KAD/DSI/102 module input. An example is shown in the following figure.

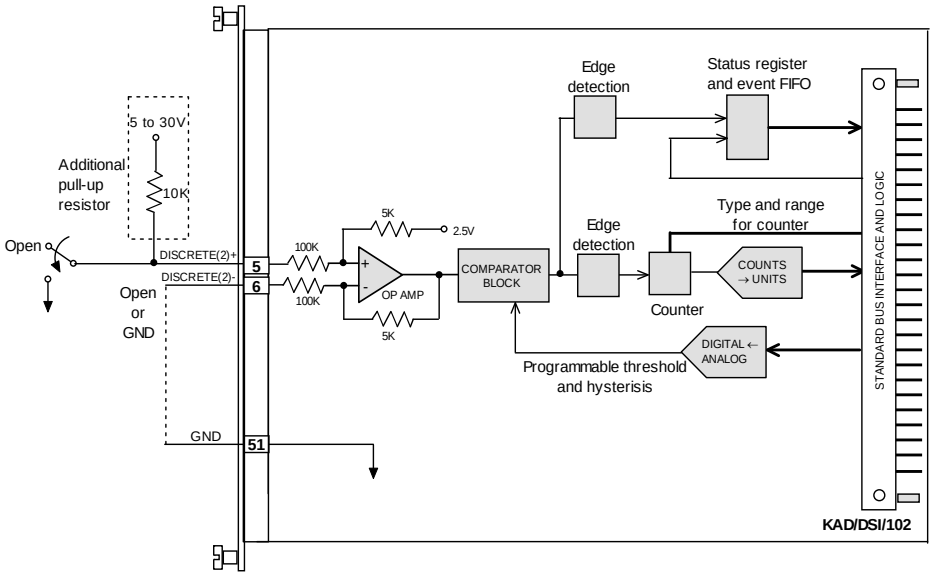


Figure 3: Third channel wired for operation in 0V/open mode using an external pull-up resistor

Setting threshold and hysteresis

The following figure shows an example of input signals, threshold levels, and hysteresis as well as points at which the KAD/DSI/102 detects valid edges.

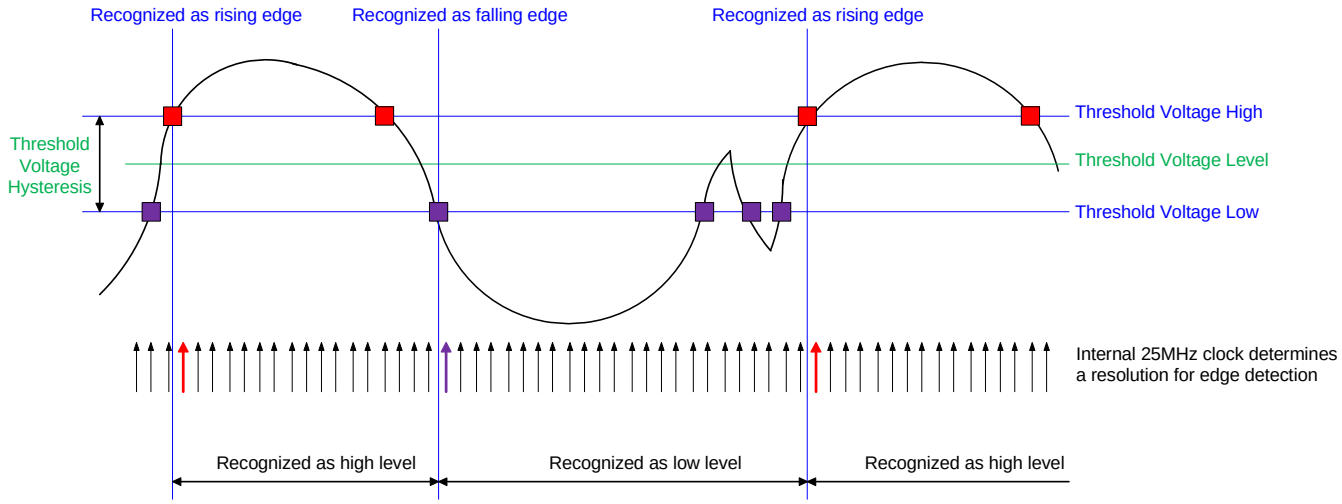


Figure 4: Edge and level detection

NOTE: Transition is considered a rising edge only after a full transition from the signal level below Threshold Voltage Low to above Threshold Voltage High takes place and vice versa for a falling edge. This has an impact for time when the module recognizes high and low pulses and also when Event messages are timestamped.

Normally the Threshold Voltage Level is set to a voltage midway between input signal logic states, while the Threshold Voltage Hysteresis is set to a value around 2 to 10 times lower than the distance between input signal logic states. A large hysteresis value is required if the incoming signal contains a lot of noise (including common mode voltage noise) or its edge is not sharp (such as with a slowly changing signal). Setting a wider hysteresis moves the edge detection voltage levels by (±0.5 × Threshold Voltage Hysteresis) from the Threshold Voltage Level. Therefore the value must not be too high as this may cause the effective edge detection voltage level to move outside the input signal range. Also, setting the Threshold Voltage Hysteresis value high may have an impact on the exact time the edge is detected, causing the EventIrigTimeTag to be further delayed by the time difference between the input signal crossing Threshold Voltage Level and the effective edge detection point caused by Threshold Voltage Hysteresis. The appropriate values must be considered for a specific application, signal change rate, and noise expected on signal lines.

For 28V/0V and 28V/open modes, when the signal is connected to the DISCRETE(x)+ input, and the DISCRETE(x)- input is left floating (or connected to GND), the recommended value of the Threshold Voltage Level is around 14V with the recommended Threshold Voltage Hysteresis being between 3V to 15V, depending on noise, input signal change rate, and allowed time tag error.

For 0V/open mode, we recommend using an external pull-up resistor and connecting DISCRETE(x)- to GND. This increases the voltage step and reduces the influence of unwanted noise in external cabling (open/floating inputs have high impedance so noise is easily picked up). Threshold Voltage Level and Threshold Voltage Hysteresis depend on the pull-up resistor value, the pull-up voltage, and whether DISCRETE(x)- is connected to GND, or left floating. If an external pull-up resistor is not used, ensure the noise level in the external cables do not cause unwanted edges being detected by the module. For 0V/open mode, when the signal is connected to the DISCRETE(x)+ input, and the DISCRETE(x)- input is left floating, the Threshold Voltage Level should be around -1.2V and Threshold Voltage Hysteresis around 1V. For a similar case, but with DISCRETE(x)- input connected to GND, values are 1.25V and 1V respectively.

Event FIFO

When time tagging events, make sure unused inputs or inputs that are not of interest are masked such that transitions on these inputs are ignored (Trigger Edge for Discrete[23:0] set to None). This helps ensure that the FIFO does not fill up due to too many events.

EventIrigTimeTag is 48 bits describing TimeHi, TimeLo, and TimeMicro.

TimeHi:

R[15:13] Reserved - Reserved for future use

R[12:7] Hours - BCD Hours 0 to 23

R[6:0] Minutes - BCD Minutes 0 to 59

TimeLo:

R(15) Reserved - Reserved for future use

R[14:8] Seconds - BCD Seconds 0 to 59

R[7:0] Centiseconds - BCD Centiseconds 0 to 99

TimeMicro:

R[15:0] Microseconds - BCD Microseconds 0 to 9999

To guarantee that the event FIFO is never completely filled, ensure that events are read more quickly than they occur. The event messages (80-bit wide word Event[79:0]) are written into FIFO, which can store up to 1,024 events.

For example, a square-wave input with a frequency of 500Hz with both edges set to trigger a write to FIFO, generates events with a frequency of 1,000Hz. The minimum frequency of reading events to avoid filling up FIFO and skipping some events must be greater than 1,000Hz.

If a burst of 1,000 events occurs every 250 seconds, then a minimum reading frequency of 4Hz must be used to safeguard that the FIFO is empty before the next burst arrives.

By default Event[23:0] bits contain the status of inputs after the event (On Event Record set to Status After Event), but can be configured to indicate which input(s) trigger a write to FIFO (Trigger Delta).

Event(31) indicates that the FIFO is empty and that nothing was written since last power-up or programming. Event(30) indicates that the message in the FIFO was read before (stale). By default (when On FIFO Empty is set to Flag Only), when the FIFO is empty, or when a message is stale, the other Event[79:0] bits keep their previous value. An alternative operation can be set to indicate an empty or stale FIFO by setting all bits of time tag (Event[79:32]) to 1 (All High setting), which is an invalid value for a time tag, that is, it’s not a BCD number.

Event(29) indicates that a message (or messages) was skipped. This happens if the FIFO is full and a new triggered message can't be written into the FIFO and is therefore skipped (when the On Event Record setting is Flag Only). In such a case, if the On FIFO Full Overflow setting is set to Yes, the FIFO is also reset (cleared).

An example of how the FIFO acts if the On Event Record is set to Status After Event can be seen in Figure 5 on page 10.

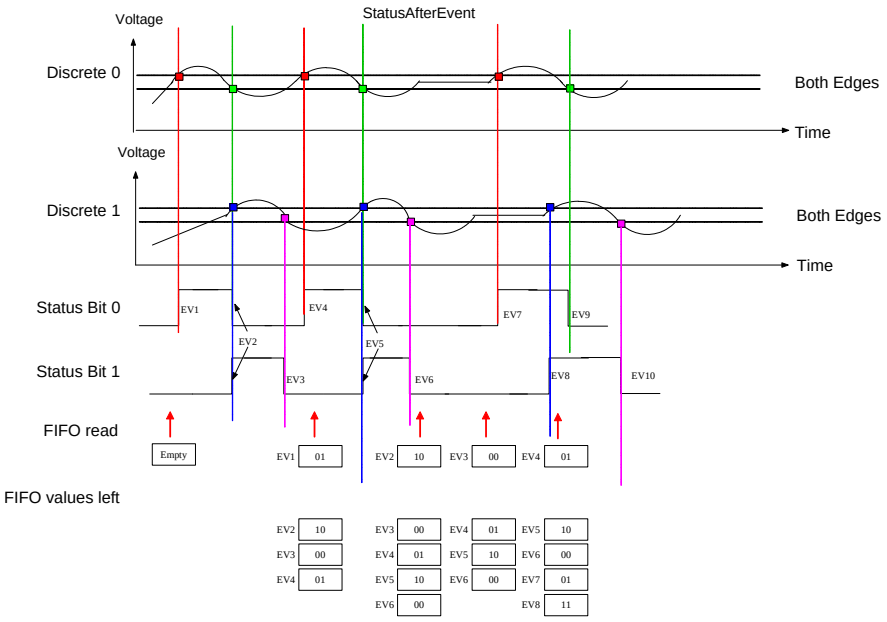


Figure 5: Event FIFO example with On Event Record set to Status After Event

Counter modes

Each counter can be set to any bit length (4 to 32 bits), counting values from 0 to ($2^{\text{Size In Bits}}$ - 1). Each counter event is configured independently from the Event FIFO triggering edge (threshold and hysteresis is common for counter and FIFO within one input channel). For some modes (Counter Type selection), previous counter input events can reset the next counter. This feature can be used when counting external or internal events and enables resetting based on another event. Note that the next counter/input for channel 23 (last channel) is channel 0 counter/input (first channel), and that the previous counter/input for channel 0 is channel 23 counter/input. The details of counter operation depends on the mode. There are 10 possible modes of operation and they are described below.

Period mode

The Period mode configures the counter to measure the time between events. Internally the counter increments every 40ns. The raw counter value is scaled to user-programmed Range Maximum and Size In Bits for a specific counter, railed, and then output from the module. A simplified example of counter operation in Period mode (without scaling to reach Range Maximum) is presented in the following figure.

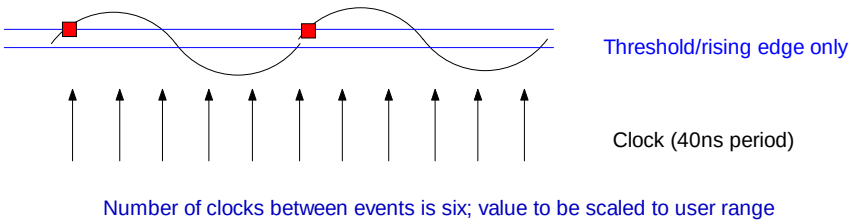


Figure 6: Example of counter operation in Period mode

When setting the counter Trigger Edge to Rising, a period is measured between consecutive rising edges. When setting it to Falling, a period is measured between falling edges.

Consider the following example where the counter Range Maximum is set to 1s and the counter Size In Bits is set to 24. For an input signal period of 0.25 seconds, the reading from the counter register is 0x0040 0000. Any signal with a period more than or equal to the Range Maximum rails at the maximum value defined by Size In Bits, that is, 0x00FF FFFF.

The equation for the Period mode is:

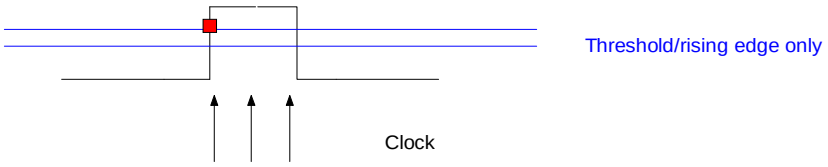
$$Period[counts] = Input\ Signal\ Period[s] \times \frac{2^{Size\ In\ Bits}}{Range\ Maximum[s]}$$

The error on this mode is determined by a 40ns internal time measurement resolution, 5ppm (1ppm typical) reference clock precision, and one-LSB resolution of the counter.

$$Error[s] = \pm\left(40 \times 10^{-9}[s] + 5 \times 10^{-6} \times Period[s] + \frac{Range\ Maximum[s]}{2^{Size\ In\ Bits}}\right)$$

Pulse Width mode

The Pulse Width mode configures the counter to measure the time of a high or low pulse. Internally, the counter increments every 40ns. The raw counter value is scaled to user-programmed Range Maximum and Size In Bits for a specific counter, railed, and then output from the module. It is possible to set the counter to measure a low or high pulse width by changing the Trigger Edge setting; a Rising setting means that the width of a high pulse is measured (see the following figure).



In this example, the number of clocks within high pulse is three

Figure 7: Example of counter operation in Pulse Width mode

The equation for the Pulse Width mode is:

$$Pulse\ Width[counts] = Input\ Signal\ Pulse\ Width[s] \times \frac{2^{Size\ In\ Bits}}{Range\ Maximum[s]}$$

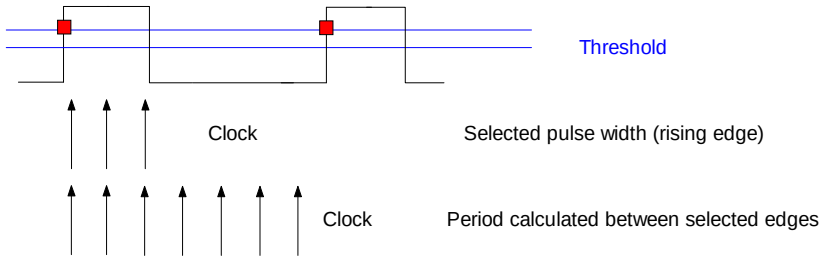
The error on this mode is determined by a 40ns internal time measurement resolution, 5ppm (1ppm typical) reference clock precision, and one-LSB resolution of the counter.

$$Error[s] = \pm\left(40 \times 10^{-9}[s] + 5 \times 10^{-6} \times Pulse\ Width[s] + \frac{Range\ Maximum[s]}{2^{Size\ In\ Bits}}\right)$$

NOTE: In Pulse Width mode, the counter Trigger Edge set to Both, results in the measurement of either high or low pulse width. The value read from the output buffer is the previous value which was propagated through scaling, railing blocks before a new sample window started. Such a setup can be useful where a signal source indicates some event by toggling its current output level, rather than generating a defined edge or pulse.

Duty Cycle mode

The Duty Cycle mode calculates the ratio between the selected pulse width and the period of input signal. It then represents it over an offset binary range of 0 to 1 across the selected counter Size In Bits. The counter's Trigger Edge defines the edge at which the ratio value is recalculated, and which pulse width is used for the ratio calculation. A simple example of counter operation in Duty Cycle mode with Trigger Edge set to Rising is presented in Figure 8 on page 12.



Three clocks for pulse width and eight clocks for period gives a ratio of 0.375

Figure 8: Example of counter operation in Duty Cycle mode

Setting the counter Trigger Edge to Rising means that the ratio between the high pulse and the period is calculated. Setting it to Falling means that the ratio between the low pulse and the period is calculated.

Consider the following example where an input signal of 12.5% duty cycle is connected to the input, the counter Trigger Edge is set to Rising, and Size in Bits is set to 16. In this scenario, the reading from the counter register is be 0x0000 2000 (meaning 0.125 in 16-bit size offset binary value representation).

NOTE: We recommend not using the Both setting in this mode.

The equation for Duty Cycle mode is:

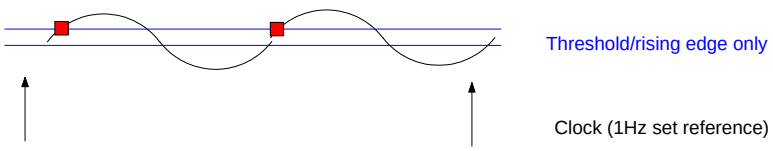
$$Duty\ Cycle[counts] = Input\ Signal\ Duty\ Cycle[unitless] \times 2^{Size\ In\ Bits}$$

The error on this mode is determined by a 40ns internal time measurement resolution for both pulse width and period measurements, and one-LSB resolution of the counter. A worst case scenario (input signal duty cycle being close to one) would be covered by the following formula:

$$Error[unitless] = \pm\left(\frac{2 \times 40 \times 10^{-9}[s]}{Period[s]} + \frac{1}{2^{Size\ In\ Bits}}\right)$$

Frequency mode

The Frequency mode configures the counter to count events over a fixed time, which is defined by the Frequency Mode Reference Period setting, so the roles of incrementing and windowing signals versus Period mode are opposite. A simple example of counter operation in Frequency mode is presented in the following figure.



Number of events between clocks is two; value to be scaled to a user-defined range

Figure 9: Example of counter operation in Frequency mode

NOTE: The internal counter value shown in the figure above is only for the purpose of illustrating the basics of operation when in Frequency mode; this value is scaled by user-defined values for Range Maximum and Size in Bits.

In Frequency mode, the counter Trigger Edge set to Both, results in the values counted being doubled. The Both setting may be desirable when increasing the resolution of the frequency measurement by a factor of two (which is possible if the duty cycle is around 50%).

Setting the counter Trigger Edge to Rising means that the internal counter increments on every rising edge of the input signal. Setting it to Falling means that it increments on every falling edge.

Consider the following example where the counter Range Maximum is set to 1,000,000 (Hz), the counter Size In Bits is set to 32 and the Frequency Mode Reference Period is set to any value. For a 500,000Hz input signal, the reading from the counter register is 0x8000 0000. Any signal with a frequency more than or equal to the Range Maximum rails at the maximum value defined by Size in Bits, that is, 0xFFFF FFFF.

Low input signal frequency example

Applying a 2.9Hz signal to an input configured as Frequency counter set with Rising Edge, Size In Bits = 16 bits, Range Maximum = 10Hz, and the Frequency Mode Reference Period is set to 1 second, outputs nine samples of 3Hz value 0x4CCC (19660 decimal) and one sample of 2Hz 0x3333 (13107 decimal). The average of (9 × 3Hz + 1 × 2Hz) / 10 = 2.9Hz gives the result expected. However a single measurement is determined by counting an integer number of edges over 1 second (Frequency Mode Reference Period), as this is the method used by the module for Frequency mode of counter operation. This is clearly visible (biggest relative error of measurement) if the frequency applied to the input is low (for example, below 100Hz could be considered low). Accuracy can be improved by increasing Frequency Mode Reference Period, though the effect cannot be completely eliminated, and it is at a cost of the refresh rate of the Frequency measurement.

NOTE: The number of counted edges over ones is scaled (multiplied) to represent it as a user-defined Range Maximum, and this causes up to one LSB of the counter rounding error. For that reason the exact readings in the example above are not exactly 2Hz and 3Hz, but 1.999969482 and 2.99987793 respectively.

The equation for Frequency mode is:

$$Frequency[counts] = Input\ Signal\ Frequency[Hz] \times \frac{2^{Size\ In\ Bits}}{Range\ Maximum[Hz]}$$

The error on this mode is determined by one counted edge of input signal resolution, 5ppm (1ppm typical) reference clock precision, and one-LSB resolution of counter.

$$Error[Hz] = \pm\left(\frac{1}{Frequency\ Mode\ Reference\ Period\ [s]} + 5 \times 10^{-6} \times Frequency[Hz] + \frac{Range\ Maximum[Hz]}{2^{Size\ In\ Bits}}\right)$$

NOTE: For measuring low frequency signals, where a resolution of better than 1Hz is required, the more accurate method would be to use Period mode instead, and post-processing the reading to frequency by inverting it.

Events Since Sample mode[1]

In the Events Since Sample mode, the counter increments on events from related input (independently configured from Event FIFO triggering), and resets at the start of every sample window. The sample windows are synchronized to the start of the acquisition cycle, and are repeated every sample period. The Output buffer value is updated at the end of each sampling window value. A simple example of counter operation in Events Since Sample mode is presented in the following figure.

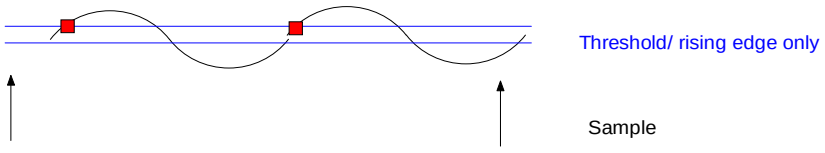


Figure 10: Example of counter operation in Events Since Sample mode

1. In previous Curtiss-Wright products, this mode was called EVENTS_SINCE.

NOTE: In Events Since Sample mode, the counter Trigger Edge set to Both, results in the values counted being doubled. The Both setting may be desirable in scenarios where the signal source toggles to indicate an event.

Events Since Power Up mode[2]

The Events Since Power Up mode configures the counter to count events since system power-up or programming. A simple example of counter operation in Events Since Power Up mode is presented in the following figure.

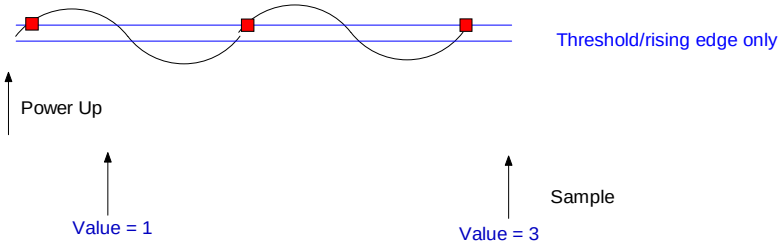


Figure 11: Example of counter operation in Events Since Power Up mode

NOTE: In Events Since Power Up mode, the counter Trigger Edge set to Both, results in the values counted being doubled. The Both setting may be desirable in scenarios where the signal source toggles to indicate an event.

Events Since Reset mode[3]

The Events Since Reset mode uses two adjacent inputs. This counter works in a mode similar to Events Since Sample mode, but the difference is that an event on the previous counter causes the counter to reset. The previous counter can be configured to any mode at the same time. This mode is useful if an external reset of counted events is required. A simple example of counter operation in Events Since Reset mode is presented in the following figure.

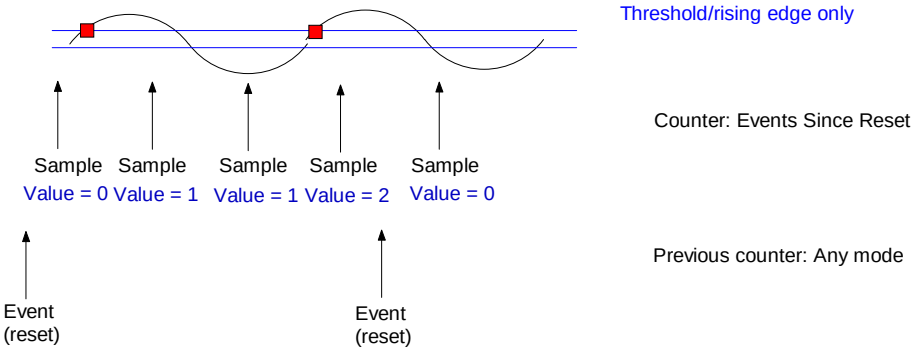


Figure 12: Example of counter operation in Events Since Reset mode

NOTE: In Events Since Reset mode, the counter Trigger Edge set to Both, results in the values counted being double in size. The Both setting may be desirable in scenarios where the signal source toggles to indicate an event.

Samples Since Power Up mode[4]

The Samples Since Power Up mode configures the counter to increment every time it is read. For example, if the counter Length is configured to 32 bits and it is read once per 1ms major frame, then it can act as a major frame counter that reaches its

2. In previous Curtiss-Wright products, this mode was called EVENTS.
3. In previous Curtiss-Wright products, this mode was called RESET.
4. In previous Curtiss-Wright products, this mode was called READ.

maximum value after approximately 49.7 days. This is the only mode where the counter returns to 0 after having reached the maximum value. After power-up or programming, the counter starts from 0. A simple example of counter operation in Samples Since Power Up mode is presented in Figure 13 on page 15.

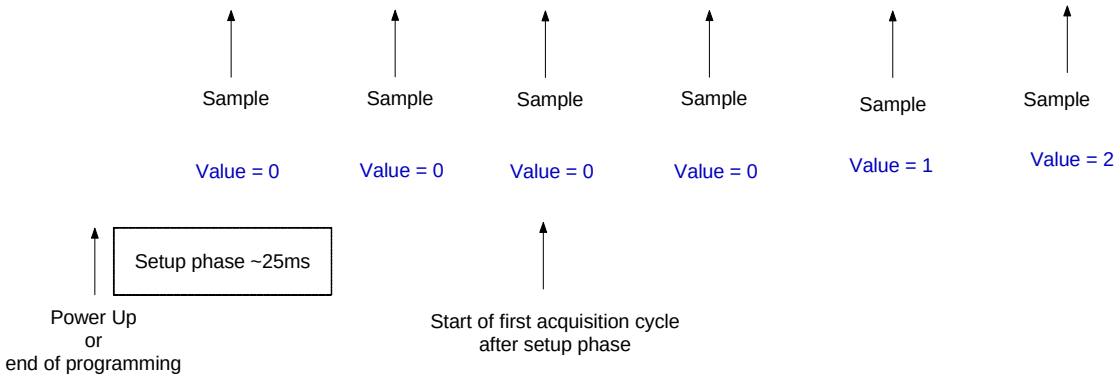


Figure 13: Example of counter operation in Samples Since Power Up mode

Samples Since Event mode[5]

The Samples Since Event mode configures the counter to work in a mode similar to the Samples Since Power Up mode, but the difference is that an event on the input causes the counter to reset to a value of 0, as power-up does. A simple example of counter operation in Samples Since Event mode is presented in the following figure.

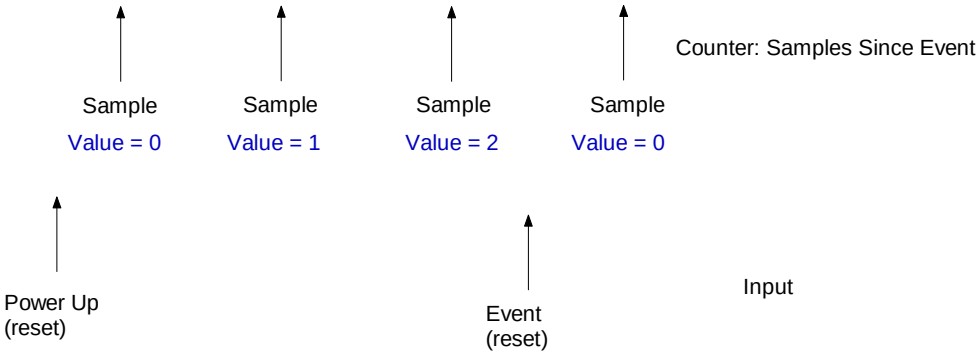


Figure 14: Example of counter operation in Samples Since Event mode

Time Since Event mode[6]

The Time Since Event mode configures the counter to reset on, and count time since, the last event. Output value is an offset binary representation of a user-defined Range Maximum over Size In Bits of counter. A simple example of counter operation in Time Since Event mode is presented in Figure 15 on page 16.

5. In previous Curtiss-Wright products, this mode was called EVENT.
6. In KAD/DSI/002, this mode was called TIMER and in KAD/DSI/003 it was called ELAPSED.

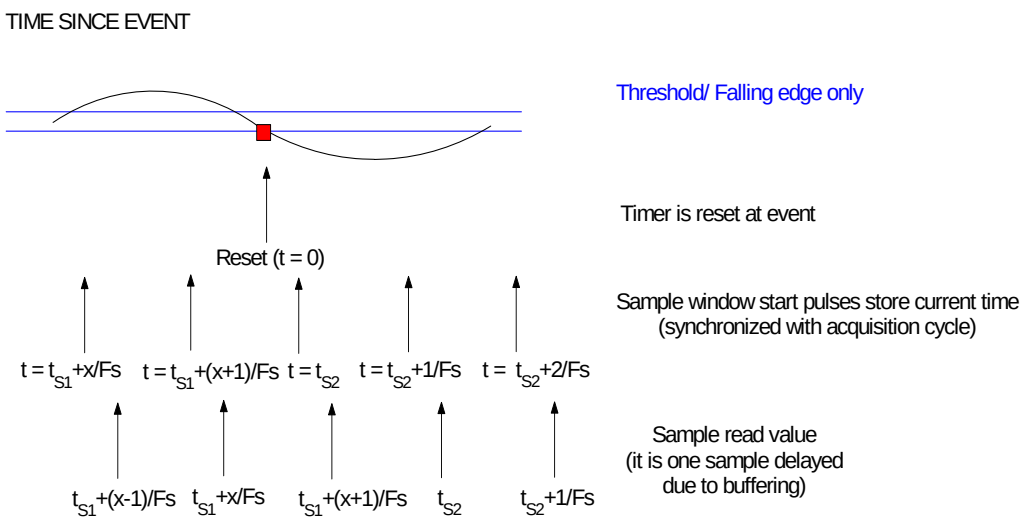


Figure 15: Example of counter operation in Time Since Event mode

The equation for the Time Since Event mode is:

$$Time\ Since\ Event[s] = Input\ Signal\ Time\ Since\ Event[s] \times \frac{2^{Size\ In\ Bits}}{Range\ Maximum[s]}$$

NOTE: Time value is sampled at the start of the sample window. The sample window is aligned with start of acquisition cycle and evenly spaced thereafter (with 1/Fs distance). The value read is one sample delayed due to the processing required (scaling to user-range), and so the buffered value is released for the next sample window. In this mode, the counter is also reset at power-up, and starts counting once the module setup phase is finished.

The error on this mode is determined by 40ns internal time measurement resolution, 5ppm (1ppm typical) reference clock precision, one-LSB resolution of the counter, and the delay of the analog input circuit and the digital edge detection block (which is 500ns ± 200ns).

$$Error[Hz] = \pm\left(40 \times 10^{-9}[s] + 5 \times 10^{-6} \times Time\ Since\ Event[s] + \frac{Range\ Maximum[s]}{2^{Size\ In\ Bits}}\right) + (500 \pm 200) \times 10^{-9}[s]$$

Timeout

If there are periods of measured signals which are longer than expected, it is desirable for the module to timeout and output a rail value. For example, with the KAD/DSI/102 the timeout feature applies to Period, Pulse Width and Duty Cycle modes. Such operation protects against a possible situation where the signal source gets disconnected (connection broken) and avoids displaying the last properly measured value of period.

In Period mode, the timeout is set automatically by software, based on the RangeMaximum set for a specific channel. If the RangeMaximum is exceeded, then a timeout occurs, and the counter value is propagated towards output buffer block (as the value must exceed RangeMaximum, then output value, after scaling and railing must be at the top rail).

In Pulse Width mode, there is a Pulse Width Mode Timeout setting available which defines the timeout value for all counters. In this mode, the output value could be 0, in the scenario where there is no new pulse ending edge since the last measurement taken (or since power-up/programming). Alternatively, if a timeout occurred after a new pulse trigger edge (beginning of pulse width measurement), then the output value is railed at the high rail of the range, for example 0xFFFFFFFF for 32-bit size counter.

In Duty Cycle mode, there is a Duty Cycle Mode Timeout setting available, which defines the maximum period of input signal on which the duty cycle ratio can be calculated. If timeout happens in this mode, the output value depends on the phase of the input signal during which timeout occurred. In the scenario that the last edge is the trigger edge, the output value is high rail (for example, 0x00FF FFFF where the Size in Bits for a specific channel is set to 24), otherwise the value is 0 (low rail). Low rail is also

the default output value after power-up.

The KAD/DSI/102 generates an internal timeout reference signal pulse every 50ms. This pulse is divided (individually per channel) by individual integer dividers (however, Pulse Width Mode Timeout and Duty Cycle Mode Timeout are set up for all channels by a single setting). As the reference signal is asynchronous versus input signals, KSM-500 and DAS Studio 3 software always rounds up the required timeout value divider and adds it to 1. This is to avoid the situation where timeout interrupts the measurement of a signal for which period is close (<50ms) to the required timeout. For example, if Pulse Width Timeout is set to 1s, then the value programmed is equivalent to 1.05 seconds.

Power-up

The 0 value from each counter, regardless of mode, is read until all blocks and channels are set up as per EEPROM configuration.

NOTE: In line with our policy for reading EEPROM setup, the EEPROM configuration is read and programmed in a loop.

After power-up or programming, at least one full setup pass (through all channels) needs to finish, before counter increments are allowed. This ensures all channels are already fully set up. This phase takes approximately 25ms, but for inter-channel synchronization purposes, finishes at the first acquisition cycle after all channels have been set up. Until this phase finishes, all samples read from counter registers are 0.

Connector pinout of the KAD/DSI/102

PIN	NAME	SEE SPECIFICATIONS TABLE	COMMENT
1	DISCRETE(0)+	Differential ended digital inputs	Discrete input
2	DISCRETE(0)-	Differential ended digital inputs	Discrete input
3	DISCRETE(1)+	Differential ended digital inputs	Discrete input
4	DISCRETE(1)-	Differential ended digital inputs	Discrete input
5	DISCRETE(2)+	Differential ended digital inputs	Discrete input
6	DISCRETE(2)-	Differential ended digital inputs	Discrete input
7	DISCRETE(3)+	Differential ended digital inputs	Discrete input
8	DISCRETE(3)-	Differential ended digital inputs	Discrete input
9	DISCRETE(4)+	Differential ended digital inputs	Discrete input
10	DISCRETE(4)-	Differential ended digital inputs	Discrete input
11	DISCRETE(5)+	Differential ended digital inputs	Discrete input
12	DISCRETE(5)-	Differential ended digital inputs	Discrete input
13	DISCRETE(6)+	Differential ended digital inputs	Discrete input
14	DISCRETE(6)-	Differential ended digital inputs	Discrete input
15	DISCRETE(7)+	Differential ended digital inputs	Discrete input
16	DISCRETE(7)-	Differential ended digital inputs	Discrete input
17	DISCRETE(8)+	Differential ended digital inputs	Discrete input
18	DISCRETE(8)-	Differential ended digital inputs	Discrete input
19	DISCRETE(9)+	Differential ended digital inputs	Discrete input
20	DISCRETE(9)-	Differential ended digital inputs	Discrete input
21	DISCRETE(10)+	Differential ended digital inputs	Discrete input
22	DISCRETE(10)-	Differential ended digital inputs	Discrete input
23	DISCRETE(11)+	Differential ended digital inputs	Discrete input
24	DISCRETE(11)-	Differential ended digital inputs	Discrete input
25	DISCRETE(12)+	Differential ended digital inputs	Discrete input
26	DISCRETE(12)-	Differential ended digital inputs	Discrete input
27	DISCRETE(13)+	Differential ended digital inputs	Discrete input
28	DISCRETE(13)-	Differential ended digital inputs	Discrete input
29	DISCRETE(14)+	Differential ended digital inputs	Discrete input
30	DISCRETE(14)-	Differential ended digital inputs	Discrete input
31	DISCRETE(15)+	Differential ended digital inputs	Discrete input
32	DISCRETE(15)-	Differential ended digital inputs	Discrete input
33	DISCRETE(16)+	Differential ended digital inputs	Discrete input
34	DISCRETE(16)-	Differential ended digital inputs	Discrete input
35	DISCRETE(17)+	Differential ended digital inputs	Discrete input
36	DISCRETE(17)-	Differential ended digital inputs	Discrete input
37	DISCRETE(18)+	Differential ended digital inputs	Discrete input
38	DISCRETE(18)-	Differential ended digital inputs	Discrete input
39	DISCRETE(19)+	Differential ended digital inputs	Discrete input
40	DISCRETE(19)-	Differential ended digital inputs	Discrete input
41	DISCRETE(20)+	Differential ended digital inputs	Discrete input
42	DISCRETE(20)-	Differential ended digital inputs	Discrete input
43	DISCRETE(21)+	Differential ended digital inputs	Discrete input
44	DISCRETE(21)-	Differential ended digital inputs	Discrete input
45	DISCRETE(22)+	Differential ended digital inputs	Discrete input
46	DISCRETE(22)-	Differential ended digital inputs	Discrete input
47	DISCRETE(23)+	Differential ended digital inputs	Discrete input
48	DISCRETE(23)-	Differential ended digital inputs	Discrete input
49	DNC		Do not connect
50	GND	Internal ground	
51	GND	Internal ground	
52	CHASSIS	Chassis	Double-density connector only

Ordering information

PART NUMBER	DESCRIPTION
KAD/DSI/102/B	Discrete input (programmable counters, time tagging) - 24ch (with 52-way double-density connector)
KAM/DSI/102/B	Discrete input (programmable counters, time tagging) - 24ch (with 51-way micro-miniature connector)

By default, the standard mating connector (CON/KAD/002/CP for KAD modules; or ACC/CON/008/04 for KAM modules), is included with each module in the shipment. Its part number will be added to the Confirmation of Order unless an alternative option is specified (see the *Cables* data sheet). In this data sheet, KAD/DSI/102 refers to both the KAD and KAM version of the module.

Revision history

REVISION	DIFFERENCES	STATUS
KAD/DSI/102/B	Added Time Since Event mode; improved threshold voltage hysteresis on channels (12) and (19)	Recommended for new programs
KAD/DSI/102	First release	Not recommended for new programs

Supporting software

MODULE	DETAILS
DAS Studio 3	User interface for setup and management of data acquisition, network switches, recorders and ground stations in an integrated environment
KSM-500	This module is supported by the KSM-500 suite of software tools

Related documentation

DOCUMENT	DETAILS
DOC/DBK/001	Acra KAM-500 Databook
DOC/HBK/002	Environmental Qualification Handbook
DOC/MAN/018	KSM-500 Databook
DOC/MAN/030	DAS Studio 3 User Manual
TEC/NOT/016	Power dissipation
TEC/NOT/049	Power estimation

This page is intentionally blank