

# RedLab 4301

9513-based Counter/Timer

Five independent 16-bit counters

## User's Guide



# **RedLab 4301**

## **9513-based Counter/Timer**

### **User's Guide**



Document Revision 1.1 E, January, 2008  
© Copyright 2008, Meilhaus Electronic

# Imprint

## **User's Guide RedLab® Series**

Document Revision 1.1 E  
Revision Date: January 2008

Meilhaus Electronic GmbH

Fischerstraße 2  
D-82178 Puchheim near Munich, Germany  
<http://www.meilhaus.de>

© Copyright 2008 Meilhaus Electronic GmbH

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form by any means, electronic, mechanical, by photocopying, recording, or otherwise without the prior written permission of Meilhaus Electronic GmbH.

## **Important note:**

All the information included in this user's guide were put together with utmost care and to best knowledge. However, mistakes may not have been erased completely.

For this reason, the firm Meilhaus Electronic GmbH feels obliged to point out that they cannot be take on neither any warranty (apart from the claims for warranty as agreed) nor legal responsibility or liability for consequences caused by incorrect instructions.

We would appreciate it if you inform us about any possible mistakes.

The trademark Personal Measurement Device, TracerDAQ, Universal Library, InstaCal, Harsh Environment Warranty, Measurement Computing Corporation, and the Measurement Computing logo are either trademarks or registered trademarks of Measurement Computing Corporation.

Windows, Microsoft, and Visual Studio are either trademarks or registered trademarks of Microsoft Corporation.

LabVIEW is a trademark of National Instruments.

CompactFlash is a registered trademark of SanDisk Corporation.

XBee is a trademark of MaxStream, Inc.

All other trademarks are the property of their respective owners.

---

# Table of Contents

<b>Preface</b>	
<b>About this User's Guide .....</b>	<b>6</b>
What you will learn from this user's guide .....	6
Conventions in this user's guide .....	6
Where to find more information .....	6
<b>Chapter 1</b>	
<b>Introducing the RedLab 4301 .....</b>	<b>7</b>
Software features .....	7
Connecting a RedLab 4301 to your computer is easy .....	7
<b>Chapter 2</b>	
<b>Installing the RedLab 4301 .....</b>	<b>8</b>
What comes with your RedLab 4301 shipment? .....	8
Hardware .....	8
Additional documentation .....	8
Unpacking the RedLab 4301 .....	9
Installing the software .....	9
Installing the RedLab 4301 .....	9
<b>Chapter 3</b>	
<b>Functional Details .....</b>	<b>10</b>
RedLab 4301 block diagram .....	10
External components .....	11
USB connector .....	11
Status LED .....	11
Power LED .....	11
Screw terminal banks .....	11
Counter terminals (INP1 to INP5, GAT1 to GAT5, and OUT1 to OUT5) .....	13
Digital input (DI0 to DI7) and output (DO0 to DO7) terminals .....	13
Digital input control terminal (DI CTL) for pull-up/down configuration .....	14
Interrupt input terminal (INT) .....	14
Oscillator output terminal (OSC) .....	14
Ground terminals (GND) .....	14
Oscillator output (OSC) .....	14
Internal sources .....	14
External sources .....	15
Divider (OSC divider) .....	15
Counter operation .....	15
Count source .....	16
Gate control .....	16
Counting modes .....	16
Reload .....	16
Recycle .....	17
BCD/Binary mode .....	17
The "Special" gate .....	17
Output control .....	18
Example 1 .....	18
Example 2 .....	18
Interrupt input pin (INT) .....	20
<b>Chapter 4</b>	
<b>Specifications .....</b>	<b>21</b>
Counter .....	21
Digital input / output .....	22

Interrupt Input.....	22
Memory .....	23
Microcontroller.....	23
USB +5V voltage .....	23
LEDs.....	23
Power.....	23
USB specifications .....	24
Environmental .....	24
Mechanical .....	24
Screw terminal connector type .....	24
Screw terminal pin out.....	25

---

# About this User's Guide

## What you will learn from this user's guide

This user's guide explains how to install, configure, and use the RedLab 4301 so that you get the most out of its counter features. This user's guide also refers you to related documents available on our web site, and to technical support resources.

## Conventions in this user's guide

### For more information on ...

Text presented in a box signifies additional information and helpful hints related to the subject matter you are reading.

**Caution!** Shaded caution statements present information to help you avoid injuring yourself and others, damaging your hardware, or losing your data.

<#:#> Angle brackets that enclose numbers separated by a colon signify a range of numbers, such as those assigned to registers, bit settings, etc.

**bold text** **Bold** text is used for the names of objects on the screen, such as buttons, text boxes, and check boxes. For example:  
1. Insert the disk or CD and click the **OK** button.

*italic text* *Italic* text is used for the names of manuals and help topic titles, and to emphasize a word or phrase. For example:  
The *InstaCal* installation procedure is explained in the *Quick Start Guide*.  
*Never* touch the exposed pins or circuit connections on the board.

## Where to find more information

The following electronic documents provide helpful information relevant to the operation of the RedLab 4301.

- The *Quick Start Guide* is available on our RedLab CD in the root directory.
- The *Guide to Signal Connections* is available on our RedLab CD under „ICalUL\Documents“.
- The *Universal Library User's Guide* is available on our RedLab CD under „ICalUL\Documents“.
- The *Universal Library Function Reference* is available is on our RedLab CD under „ICalUL\Documents“.
- The *Universal Library for LabVIEW™ User's Guide* is available is on our RedLab CD under „ICalUL\Documents“.

---

# Introducing the RedLab 4301

The RedLab 4301 is a USB-based high-performance, low-cost counter/timer device.

The RedLab 4301 is designed with a 9513 counter/timer chip. The 9513 chip has five independent 16-bit counters (65,536 counts). Each counter has an input source, internal count register, load register, hold register, output, and gate.

The 9513 is software-programmable for event counting, pulse and frequency measurement, alarm comparisons, and other input functions. The 9513 can generate frequencies with either complex duty cycles, or with one-shot and continuous-output modes. You can chain up to five 9513 counters together using software to enable a 32-, 48-, 64-, or 80-bit counter that does not require hardware connections. The gate source and gating functions are software-programmable.

An eight-bit, high-current digital output port provides logic-level control, and can be used to switch solid state relays. An eight-bit digital input port can be used to sense contact closures and other TTL level signals.

For more information on the 9513 counter/timer, refer to the 9513 data sheet. This document is available on our RedLab CD under „ICalUL\Documents“.

## Software features

For information on the features of *InstaCal* and the other software included with your RedLab 4301, refer to the *Quick Start Guide* that shipped with your device. The *Quick Start Guide* is also available in PDF on our RedLab CD (root directory).

## Connecting a RedLab 4301 to your computer is easy

Installing a data acquisition device has never been easier.

- The RedLab 4301 relies upon the Microsoft Human Interface Device (HID) class drivers. The HID class drivers ship with every copy of Windows that is designed to work with USB ports. We use the Microsoft HID because it is a standard, and its performance delivers full control and maximizes data transfer rates for your RedLab 4301. No third-party device driver is required.
- The RedLab 4301 is plug-and-play. There are no jumpers to position, dual in-line package (DIP) switches to set, or interrupts to configure.
- You can connect the RedLab 4301 before or after you install the software, and without powering down your computer first.

When you connect an HID to your system, your computer automatically detects it and configures the necessary software. You can connect and power multiple HID peripherals to your system using a USB hub.

- You can connect your system to various devices using a standard four-wire cable. The USB connector improves upon serial and parallel port connectors with one standardized plug-and-port combination.
- Data can flow two ways between a computer and peripheral over USB connections.

---

# Installing the RedLab 4301

## What comes with your RedLab 4301 shipment?

As you unpack your board, make sure that the following components are included.

### Hardware

- RedLab 4301



- USB cable (2 meter length)



### Additional documentation

In addition to this hardware user's guide, you should also receive the *Quick Start Guide* (available on our RedLab CD (root directory)). This booklet supplies a brief description of the software you received with your RedLab 4301 and information regarding installation of that software. Please read this booklet completely before installing any software or hardware.

## Unpacking the RedLab 4301

As with any electronic device, you should take care while handling to avoid damage from static electricity. Before removing the RedLab 4301 from its packaging, ground yourself using a wrist strap or by simply touching the computer chassis or other grounded object to eliminate any stored static charge.

If your RedLab 4301 is damaged, notify Meilhaus Electronic immediately by phone, fax, or e-mail. For international customers, contact your local distributor where you purchased the RedLab 4301.

- Phone: +49 (0) 89/8901660
- Fax: +49 (0) 89/89016628
- E-Mail: support@meilhaus.com

## Installing the software

Refer to the *Quick Start Guide* for instructions on installing the software *Guide* (available on our RedLab CD (root directory)).

## Installing the RedLab 4301

To connect the RedLab 4301 to your system, turn your computer on, and connect the USB cable to a USB port on your computer or to an external USB hub that is connected to your computer. The USB cable provides power and communication to the RedLab 4301.

The RedLab 4301 installs as a composite device with separate devices attached. When you connect the RedLab 4301 for the first time, **Found New Hardware** popup balloons (Windows XP) or dialogs (Windows 2000) display as each RedLab 4301 interface is detected.

After the RedLab 4301 is installed, the **Power** LED remains lit to indicate that communication is established between the RedLab 4301 and your computer.

**Caution!** Do not disconnect **any** device from the USB bus while the computer is communicating with the RedLab 4301, or you may lose data and/or your ability to communicate with the RedLab 4301.

### If the Power LED turns off

If the Power LED is illuminated but then turns off, the computer has lost communication with the RedLab 4301. To restore communication, disconnect the USB cable from the computer, and then reconnect it. This should restore communication, and the Power LED should turn back *on*.

# Functional Details

## RedLab 4301 block diagram

RedLab 4301 functions are illustrated in the block diagram shown here.

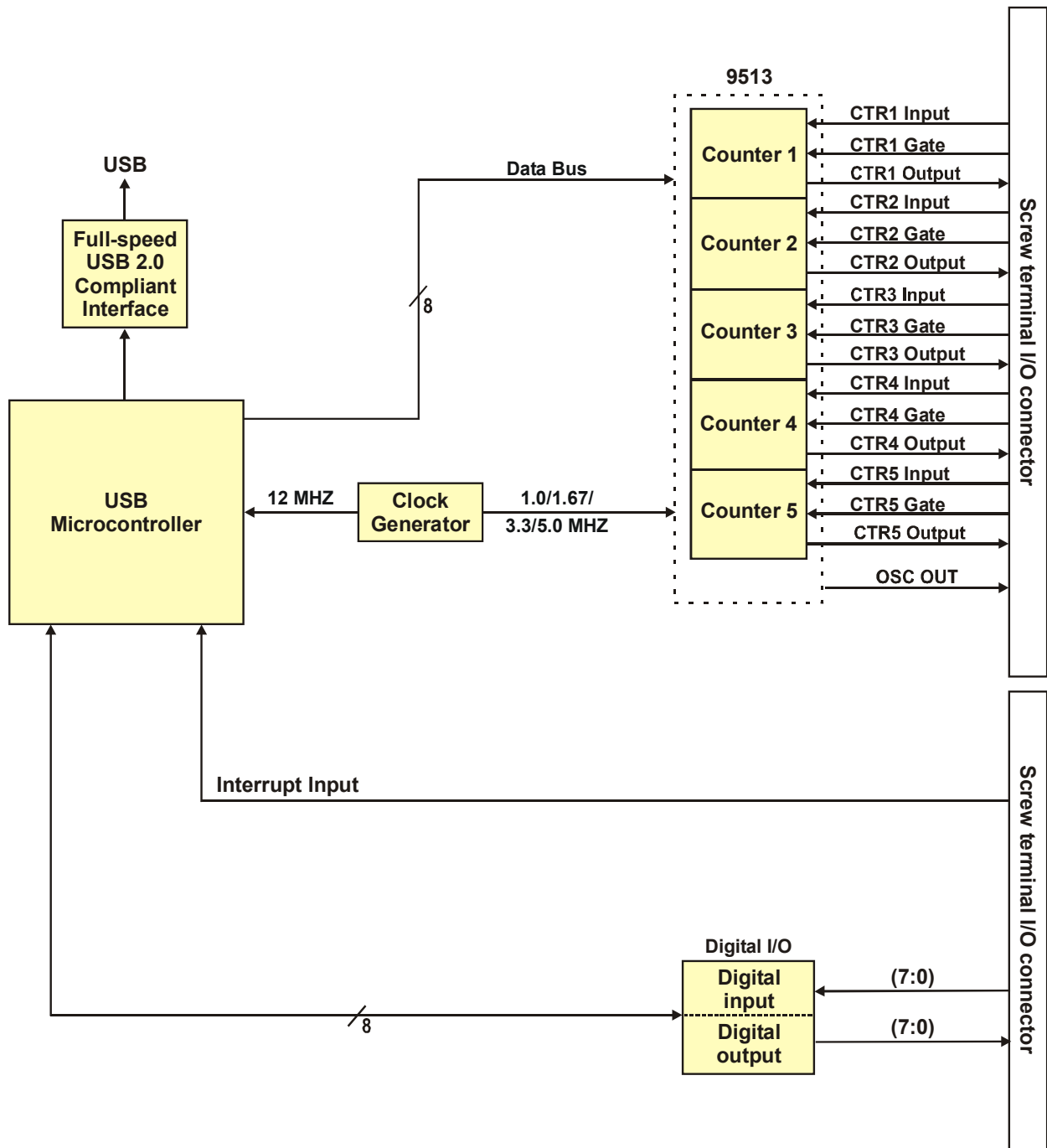


Figure 1. RedLab 4301 functional block diagram

## External components

The RedLab 4301 has the following external components, as shown in Figure 2.

- USB connector
- Status LED
- Power LED
- Screw terminal banks (2)

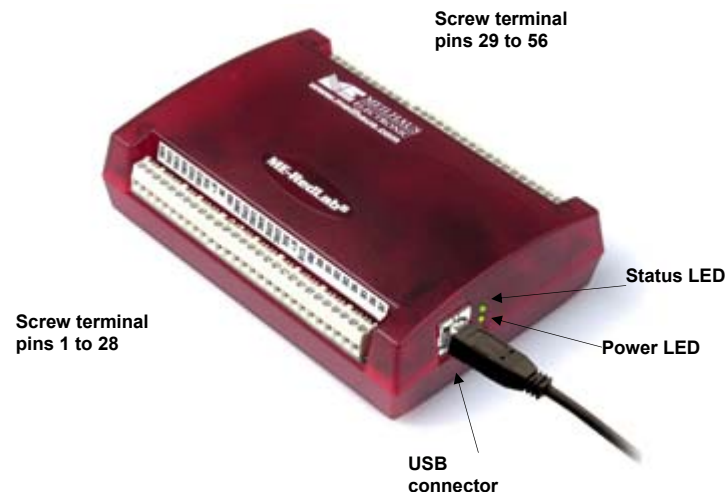


Figure 2. RedLab 4301 components

### USB connector

The USB connector provides power and communication. The voltage supplied through the USB connector is system-dependent, and may be less than 5 V. No external power supply is required.

### Status LED

The Status LED indicates the communication status of the RedLab 4301. It flashes when data is being transferred, and is off when the RedLab 4301 is not communicating. This LED uses up to 10 mA of current and cannot be disabled.

### Power LED

The power LED lights up when the RedLab 4301 is connected to a USB port on your computer or to an external USB hub that is connected to your computer.

### Screw terminal banks

The RedLab 4301 has two rows of screw terminals—one row on the top edge of the housing, and one row on the bottom edge. Each row has 28 connections. Use 16 AWG to 30 AWG wire gauge when making screw terminal connections. Pin numbers are identified in Figure 3.

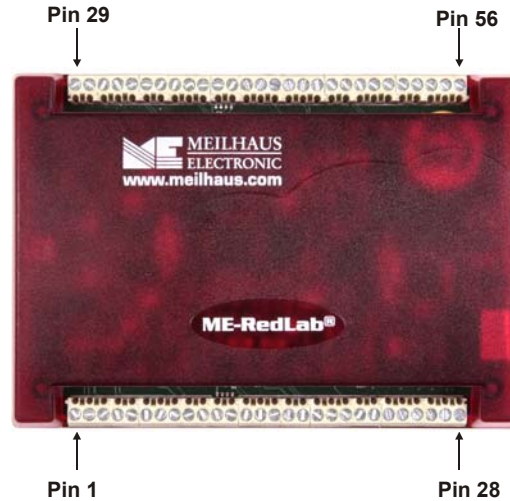


Figure 3. RedLab 4301 screw terminal pin numbering

### Screw terminal – pins 1-28

The screw terminals on the bottom edge of the RedLab 4301 (pins 1 to 28) provide the following connections:

- Two counter input connections (**INP1** and **INP2**)
- Two counter gate connections (**GAT1** and **GAT2**)
- Two counter output connections (**OUT1** and **OUT2**)
- Eight digital input connections (**DI0** to **DI7**)
- One pull-up/down control connection (**DI CTL**)
- One oscillator output connection (**OSC**)
- One interrupt input connection (**INT**)
- Two voltage output power connections (**+5V**)
- Two ground connections (**GND**)
- Seven pins are reserved (**RSVD**). Do not connect signals to these pins.

### Screw terminal – pins 29-56

The screw terminals on the top edge of the RedLab 4301 (pins 29 to 56) provide the following connections:

- Three counter input connections (**INP3 – INP5**)
- Three counter gate connections (**GAT3 – GAT5**)
- Three counter output connections (**OUT3 – OUT5**)
- Eight digital output connections (**DO0** to **DO7**)
- One pull-up/down control connection (**DI CTL**)
- Two ground connections (**GND**)
- Nine pins are reserved (**RSVD**). Do not connect signals to these pins.



## Digital input control terminal (DI CTL) for pull-up/down configuration

All digital input pins are floating by default. When inputs are floating, the state of unwired inputs is undefined—they may read high or low. You can use the **DI CTL** connection (pin 19) to configure the inputs to read a high or low value when they aren't wired.

- To pull up the digital pins to +5 V — inputs read high when unwired — wire the **DI CTL** terminal pin to a **+5V** terminal pin (pin 8 or 18).
- To pull down the digital pins to ground — inputs read low when unwired — wire the **DI CTL** terminal pin to a **GND** terminal pin (pin 10, 20, 38, or 48).

The **DI CTL** connection pulls the inputs to +5V or GND through a 47 k  $\Omega$  resistor.

## Interrupt input terminal (INT)

You can configure the interrupt input terminal (pin 9) with *InstaCal* to trigger off rising or falling edge inputs. You can program this pin to perform the following tasks:

- Send an event notification to the computer. The transfer rate is system-dependent.
- Latch in the digital inputs. With this option, the current value of the digital inputs (0 or 1) is read and stored. The stored value is updated when an active edge occurs on this pin.
- Latch out the digital outputs. With this option, digital outputs are not set to the value written until an active edge occurs on this pin.
- Save the current value of a counter. You can configure this option for each counter individually.

## Oscillator output terminal (OSC)

The oscillator output terminal (pin 7) outputs a configurable clock frequency. You can select the source of the oscillators and also select dividers for the oscillators programmatically to set the output rate.

## Ground terminals (GND)

The ground (**GND**) connections (pin 10, 20, 38, or 48) provide a common ground for the digital and voltage output connections.

## Oscillator output (OSC)

You can set the oscillator input programmatically to one of five internal frequency sources or one of 10 external input pins. The oscillator output can be divided by any number from 1 to 16 before being output to the **OSC** pin.

### Internal sources

The base frequency for the internal frequency sources can be set through *InstaCal* to one of four different frequencies — 1.0000 MHz, 1.6667 MHz, 3.3333 MHz, or 5.0000 MHz. This base frequency is used to generate the five internal frequency sources — **FREQ1** through **FREQ5**.

**FREQ1** is the same as the base frequency and each successive internal frequency following **FREQ1** divides the base frequency by another multiple of 10, for example **FREQ2** is the base frequency divided by 10, **FREQ3** is the base frequency divided by 100, and so on. This is illustrated in the following table.

Internal frequency values for different base frequencies

	<b>1.0000 MHz</b>	<b>1.6667 MHz</b>	<b>3.3333 MHz</b>	<b>5.0000 MHz</b>
FREQ1	1.0000 MHz	1.6667 MHz	1.3333 MHz	5.0000 MHz
FREQ2	100.00 kHz	166.67 kHz	133.33 kHz	500.00 kHz
FREQ3	10.000 kHz	16.667 kHz	13.333 kHz	50.000 kHz
FREQ4	1.0000 kHz	1.6667 kHz	1.3333 kHz	5.0000 kHz
FREQ5	100.00 Hz	166.67 Hz	133.33 Hz	500.00 Hz

## External sources

In addition to the five internal sources, the frequency source for the oscillator can be an input signal connected to any of the five counter input pins (**INP1 – INP5**) or any of the five gate pins (**GAT1 – GAT5**). The input signal can have a maximum frequency of 20 MHz on the counter input pins, and a maximum frequency of 7 MHz on the gate pins.

The table below applies to all internal and external sources.

Oscillator sources

<b>Source</b>	<b>Description</b>
CTRINPUT1	Counter 1 input pin
CTRINPUT2	Counter 2 input pin
CTRINPUT3	Counter 3 input pin
CTRINPUT4	Counter 4 input pin
CTRINPUT5	Counter 5 input pin
GATE1	Counter 1 gate pin
GATE2	Counter 1 gate pin
GATE3	Counter 1 gate pin
GATE4	Counter 1 gate pin
GATE5	Counter 1 gate pin
FREQ1	Internal base frequency
FREQ2	Internal base frequency/10
FREQ3	Internal base frequency/100
FREQ4	Internal base frequency/1000
FREQ5	Internal base frequency/10,000

## Divider (OSC divider)

The signal output from the **OSC** pin is the same frequency as the frequency source divided by a value from 1 to 16. For example, if you select a base frequency of *5.0000 MHz* in *InstaCal*, and choose *FREQ3* as the internal frequency source (50.000 kHz from the *Internal frequency values* table above), and 8 as the oscillator divider, the frequency of the signal at the OSC pin is  $(50.000 \text{ kHz} / 8) = 6.2500 \text{ kHz}$ .

## Counter operation

Each of the five counters performs essentially the same function: their internal value either increments or decrements on each rising or falling edge of their source signal. Since they are all 16-bit counters, they can count up from 0 to  $2^{16}-1 = 65,535$ , or down from 65,535 to 0. When counting up, 65,535 is considered to be the **Terminal Count (TC)**; when counting down, 0 is the TC. There are a variety of configuration options that can be used to unlock the power of this basic functionality.

## Count source

Each counter on the device can count from any of the available sources. The sources for counting are the same five internal sources and ten external sources listed in the Oscillator sources table (on page 15) with one addition — each counter can also increment whenever the previous numbered counter reaches its TC (TC  $n-1$ ).

For example, if you select TC  $n-1$  as the counter source for Counter 2 Input (**INP2**), counter 2 increments or decrements when counter 1 reaches TC. The previous numbered counter for counter 1 is counter 5. Note that any of the 16 sources is available for any of the five counters. In other words, the signal source for counter 3 can be Counter 5 Input or Counter 2 Gate, or **FREQ4**, and so on.

## Gate control

The counter can be gated in several ways. Level gating typically allows the counter to count only while the gate is active. Edge gating, on the other hand, typically starts the counting when an active gate edge is received.

- Level gating configuration options are active high level Gate  $n$ , active low level Gate  $n$ , active high level Gate  $n-1$  and active high level Gate  $n+1$ , where Gate  $n$  is the Gate pin associated with the counter being configured.
- Edge gating configuration options are active high (rising edge) Gate  $n$  and active low (falling edge) Gate  $n$ . The counter can be gated by the TC of the previous numbered counter. You can also disable gating.

## Counting modes

Each of the five counters has two associated 16-bit registers which can be used to store count values — the Load register and the Hold register.

- The Load register is used to load the starting value for the counter.
- The Hold register can be used to store a second counter value, or to save the current counter value.

Most applications only use the Load register. To start counting, configure the counter for the desired mode, and then load the Load register with the starting value with which to start the counting operation.

Each counter can be configured to count up or to count down. The following sections explain the various configuration options.

## Reload

When the counter reaches TC it is always reloaded with a new value. This new value comes from one of two sources:

- Reload from the Load register. This mode can be used to generate constant frequency pulse trains.
- Alternately load from the Load register and then the Hold register. This mode can be used for applications that require delayed starts or to generate pulse trains with variable duty cycles.

## Recycle

Each counter can be configured to count to TC and then stop counting, or to continue counting after reaching TC. In either case, the counter is reloaded from the appropriate register when it reaches TC. However, if the Recycle mode is set to stop counting at TC, and the reload mode is set to load from the Load and Hold registers, it will count once to TC, reload from the Hold register, count again to TC, and then stop counting.

For example, if Counter 1 is configured as follows:

- The Load 1 register contains the value 10,000
- The Hold 1 register contains the value 5,000
- Count down
- Alternately load from the Load and Hold registers
- Not recycle

The counter will count from 10,000 to 0, and then start at 5,000 and count to 0, then stop. If you select an edge gating option, and Recycle mode is active, the counter only resumes counting after TC when a new active gate edge is received.

## BCD/Binary mode

Most applications use the binary mode of counting. Binary mode causes the counter to always increment or decrement by a value of 1. The counter can have a value from 0 to 65,535.

With binary mode, a 4-bit binary number is represented by the 16 decimal numbers 0 to 15: 0 = 0b0000, 1 = 0b0001, ..., 9 = 0b1001, 10 = 0b1010, ..., 15 = 0b1111. (0b is used to denote binary numbers). For the numbers 0-9, there are four binary bits to represent one decimal digit, and from 10-15, there are four binary bits to represent two decimal digits.

However, in some applications, it is useful to maintain the four-bit-to-one-digit correspondence, so the numbers 0-9 remain the same, but 10-15 and up to 99 are represented by eight bits—four for each digit. The numbers 100-999 are represented by 12 bits, and 1000-9999 are represented by 16 bits. This makes it easy to decode long binary numbers.

For example, 0b1001 0001 0000 1001 equals 9109 in BCD (Binary Coded Decimal) mode, and equals 37,129 in binary mode. In BCD mode, the counter can only contain values from 0 to 9999 instead of 0 to 65,535 as in binary counting.

## The "Special" gate

When you enable the Special gate option and specify gating, the gate can function as a hardware re-trigger of the counting operation. When an active gate edge is received, the counter is reloaded from the Load register and begins counting, even if the counter is already counting when the gate edge is received.

When you enable the Special gate option and disable gating, the Gate pin associated with the counter being configured can select the reload source for the counter. This requires the counter be in Load and Hold reload mode. If the gate pin is low, the counter is reloaded from the Load register. If the gate pin is high, the counter is reloaded from the Hold register.

## Output control

You can configure the output pin associated with a counter to behave in five ways:

- Inactive with high impedance to ground.
- Inactive with low impedance to ground.
- Generate an active high pulse when the associated counter reaches TC.
- Generate an active low pulse when the associated counter reaches TC.
- Toggle its value every time the counter reaches TC.

### Example 1

**Problem:** The application generates an active low pulse every time a button<sup>1</sup> is pressed. You need a simple event counter to count button presses.

**Solution:** Any of the counters can be configured to do this fairly easily. Do the following:

- Connect the input signal to the counter input pin, and select that input pin as the source for the counter.
- Program the counter to:
  - Count negative edge transitions
  - Count up
  - No recycle
  - Count in binary mode
  - Reload from the Load register
  - Disable special gate
  - No gate control
  - Make output control inactive
- Load the Load register with a value of 0 to start counting.

### Example 2

**Problem:** An external device generates an active high signal while a switch is depressed. The switch is depressed for at most one second. You need to measure the time the switch is depressed with microsecond accuracy.

**Solution:** Microsecond accuracy for one second requires counting to at most 1,000,000. However, each counter can count only to 65,535. You can use some counter features to *cascade* two of the counters in order to count higher. Configure Counter 1 and Counter 2 as follows:

Configure Counter 1:

1. Using *InstaCal*, configure the counter 1 source for 1 MHz. As shown in *the "Internal frequency values" table on page 15*, an internal base frequency of 1.0000 MHz corresponds to *FREQ1*.  
The internal base frequency of 1 MHz provides an ideal way to measure time to the microsecond, since 1 count at 1 MHz equals 1 microsecond.
2. The counter can count either up or down, so for this example set the counter to count up.
3. The counter should count only while the input signal from the switch is active. To do this, connect the input signal to the gate 1 pin (**GAT1**), and configure the counter gate control for active high level Gate *n*.

---

<sup>1</sup> For more information about debouncing mechanical switches, refer to the "*Low pass to de-bounce inputs*" section in document *Signals.pdf* available on our RedLab CD under „ICaUL\Documents“.

- Load the Counter 1 Load Register with a value of zero, since you want the counter to reach as high a count as possible before reaching TC.
- To cascade Counter 1 with Counter 2, turn Recycle mode *on*. This is done so Counter 1 continuously counts to TC, then wraps around and starts counting at 0 again.
- Set reload to always load from the Load register, since you want to restart at 0 after every TC.
- Enable binary mode.
- Set the Counter 1 output to inactive; Counter 1 requires no special gating.

Configure Counter 2:

1. Set the count source for Counter 2 to be the (TC n-1). This causes Counter 2 to increment when Counter 1 reaches TC.
2. Load the Counter 2 Load Register 2 with a value of zero, since you want Counter 2 to start at 0 and count up. Counter 2 requires no gating, no special gate, no reloading, and no recycling.
3. Enable binary mode.

When the button is depressed, Counter 1 will start counting from 0 to 65,535 over and over again. Every time Counter 1 reaches 65,535, Counter 2 increments by 1 (starting from 0). When the button is released, Counter 1 stops counting. At that point, the count values can both be read.

The total time the button is pressed in  $\mu\text{s}$  is  $(\text{Counter 2 counts} * 65,536) + \text{Counter 1 counts } \mu\text{s}$ . If Counter 2 is at 9 and Counter 1 is at 34,671, the total time is  $(9 * 65,536) + 34,671 = 624,495 \mu\text{s} = 0.624495 \text{ s}$ .

## Interrupt input pin (INT)

You can configure the interrupt input pin to perform the following tasks:

- Generate an event notification that will be sent to the computer
- Latch in the eight digital inputs.

When this option is active, the digital inputs always return the same value until a new active edge is received on the interrupt pin. When the active edge is received, the current value is latched at the input pins, and held until the next active edge.

### Interrupt latency when latching inputs

There is a latency period between when an active interrupt edge occurs on the INT pin and when the action triggered by that interrupt occurs. This latency can be as long as 100  $\mu$ s, but typically varies from about 9  $\mu$ s to about 40  $\mu$ s between interrupts.

- Latch out a digital output value.

When this option is active, the digital outputs do not change state until a new active edge is received on the interrupt pin. New values sent to the digital outputs are stored. The most recently received value is latched out and held at the next active edge.

- Save the current value of all five counters to their respective Hold registers. You can read the value of the counter when the interrupt was received from the Hold registers.

For more information on the configuration options for your RedLab 4301, refer to the "USB-4300 Series" section of the "Counter Boards" chapter in the *Universal Library User's Guide*. This document (UL-User-Guide.pdf) is available on our RedLab CD under „ICaUL\Documents\UserGuides\Software“.

## Specifications

Typical for 25 °C unless otherwise specified.

Specifications in *italic text* are guaranteed by design.

### Counter

Refer to the CTS9513-2 data sheet for complete 9513 specifications and operating modes. The CTS9513-2 data sheet (9513A.pdf) is available on our RedLab CD under „ICaUL\Documents“.

Table 1. Counter specifications

Parameter	Conditions
Counter type	9513
Configuration	One 9513 device. Five up/down counters, 16-bits each.
Compatibility	5V/TTL
The 9513 device is programmable for:	
Clock source	Software selectable: External: 1. Counter 1-5 clock inputs 2. Counter 1-5 gate inputs Internal: 3. Terminal count of previous counter 4. Internal clock frequency scaler (default; divided by 1)
Gate	Software selectable source: External: 5. Active high or low level or edge, counter 1 – 5 gate input 6. Active high level previous gate or next gate 7. All external gate signals (GATx) individually pulled up through 47 K resistors to +5 V. Internal: 8. Active high previous counter terminal count 9. No gating (default)
Output	Software selectable: 10. Always low (default) 11. High pulse on terminal count 12. Low pulse on terminal count 13. Toggle on terminal count 14. Inactive, high impedance at user connector counter # output.
Osc Out	Software selectable source: 15. Counter # input 16. Gate # input 17. Prescaled internal clock (default) Software selectable divider: 18. Division by 1-16 (default = 16)
Clock input frequency	20 MHz max (50 nS min period)
Internal clock frequencies (Generated from 12 MHz crystal oscillator.)	Software selectable: 19. 5.0000 MHz (default) 20. 3.3333 MHz 21. 1.6667 MHz 22. 1.0000 MHz
Internal clock frequency prescaler	BCD scaling (Internal clock divided by 1, 10, 100, 1000 or 10000) or Binary scaling (Internal clock divided by 1, 16, 256, 4096 or 65536)
<i>Internal clock generator accuracy</i>	$\pm 2$ ppm

Parameter	Conditions
12 MHz crystal oscillator accuracy	±50 ppm
High pulse width (clock input)	25 ns min
Low pulse width (clock input)	25 ns min
Gate width	70 ns min
Input low voltage	-0.5 V min, 0.8V max
Input high voltage	2.0 V min, USB +5V power max
Output low voltage @ I <sub>OL</sub> =4 mA	0.4 V max
Output high voltage @ I <sub>OH</sub> = 4 mA	2.4 V min

## Digital input / output

Table 2. Digital I/O specifications

Digital type	Discrete, 5V/TTL compatible
	Output: 74ACT373
	Input: 74ACT373
Number of I/O	8 input, 8 output
Configuration	1 bank of 8 as output, 1 bank of 8 as input
Input high voltage	2.0 V min, 5.5 V absolute max
Input low voltage	0.8 V max, -0.5 V absolute min
Output high voltage	3.3 volts min @ -24 mA (V <sub>CC</sub> = 4.5 V)
Output low voltage	0.8 volts max @ 10 mA
Data transfer	Programmed I/O
Power-up / reset state	Digital outputs reset to TTL low
Digital I/O transfer rate (system paced)	System dependent, 33 to 1000 port reads/writes or single bit reads/writes per second.
Pull-up/pull-down configuration	User configurable for pull-up/-down through 47 kΩ resistor (Note 1). All pins floating (default)

**Note 1:** Pull-up and pull-down configurations are available using the DI CTL terminal block pin 19. The pull-down configuration requires the DI CTL pin (pin 19) to be connected to a GND pin (pin 10, 20, 38, or 48). For a pull-up configuration, the DI CTL pin should be connected to a +5V terminal pin (pin 8 or 18).

## Interrupt Input

Table 3. Interrupt specifications

Implementation	Interrupts the microcontroller operation on the device to execute one or more of several firmware routines.
Interrupt characteristics	Rising edge (default) or falling edge triggered, user selectable
Firmware routines	Any or all of the following can be activated by the user: <b>23.</b> Generate USB event notification <b>24.</b> Latch digital inputs (Reading digital inputs returns most recently latched value.) <b>25.</b> Latch digital outputs (Most recently written digital output value is latched.) <b>26.</b> Save counts on any/all of counters 1-5.
Event latency to PC	1-33 ms (4 ms typical)
Maximum event notification rate	33-1000 Hz (system dependent) (Note 2)
Interrupt latency for latch operations	100 μs maximum (80 μs typical)

**Note 2:** The interrupt rate, when transferring information to the PC (event notification), is limited by the USB to a theoretical limit of 1kHz. Some systems may not be able to achieve this maximum rate due to differences in USB controller implementation, traffic on the USB, or operating system activity.

## Memory

Table 4. Memory specifications

EEPROM	256 bytes EEPROM memory available for external use.
--------	---

## Microcontroller

Table 5. Microcontroller specifications

Type	High performance 8-bit RISC microcontroller
------	---

## USB +5V voltage

Table 6. USB +5V voltage specifications

Parameter	Conditions	Specification
USB +5V (VBUS) input voltage range		4.75 V min. to 5.25 V max.

## LEDs

Table 7. USB +5V voltage specifications

Power LED	Indicates that the device's microcontroller has power and is running
Status LED	Indicates that the USB is configured; blinks to indicate USB traffic.

## Power

Table 8. Power specifications

Parameter	Conditions	Specification
Supply current (Note 3)	USB enumeration	100 mA max
Supply current	Maximum load	302 mA max.
User +5V output voltage range (terminal block pin 8 and 18)	Connected to self-powered hub. (Note 4)	4.75 V min. to 5.25 V max.
User +5V output current (terminal block pin 8 and pin 18)	Bus-powered and connected to a self-powered hub. (Note 4)	10 mA max.

**Note 3:** This is the total current requirement for the RedLab 4301 which includes up to 14 mA for the Power and Status LEDs, but does not include current sourced from the User +5V output or from the digital output pins.

**Note 4:** Self-Powered Hub refers to a USB hub with an external power supply. Self-powered hubs allow a connected USB device to draw up to 500 mA.

Root Port Hubs reside in the PC's USB Host Controller. The USB port(s) on your PC are root port hubs. All externally powered root port hubs (desktop PC's) provide up to 500 mA of current for a USB device. Battery-powered root port hubs provide 100 mA or 500 mA, depending upon the manufacturer. A laptop PC that is not connected to an external power adapter is an example of a battery-powered root port hub.

## USB specifications

Table 9. USB specifications

USB device type	USB 2.0 (full-speed)
Device compatibility	USB 1.1, USB 2.0
Power requirements	Self-powered, 500 mA consumption max
USB cable type	<i>A-B cable, UL type AWM 2725 or equivalent. (min 24 AWG VBUS/GND, min 28 AWG D+/D-)</i>
USB cable length	3 meters max.

## Environmental

Table 10. Environmental specifications

Operating temperature range	0 to 60 °C
Storage temperature range	-40 to 85 °C
Humidity	0 to 90% non-condensing

## Mechanical

Table 11. Mechanical specifications

Dimensions	127 mm (L) x 88.9 mm (W) x 35.56 (H)
User connection length	3 meters max.

## Screw terminal connector type

Table 12. Screw terminal connector specifications

Connector type	Screw terminal
Wire gauge range	16 AWG to 30 AWG

## Screw terminal pin out

Table 13. Screw terminal pin out

Pin	Signal Name	Pin Description	Pin	Signal Name	Pin Description
1	INP1	Counter 1 input	29	INP3	Counter 3 input
2	GAT1	Counter 1 gate	30	GAT3	Counter 3 gate
3	OUT1	Counter 1 output	31	OUT3	Counter 3 output
4	INP2	Counter 2 input	32	INP4	Counter 4 input
5	GAT2	Counter 2 gate	33	GAT4	Counter 4 gate
6	OUT2	Counter 2 output	34	OUT4	Counter 4 output
7	OSC	Oscillator output	35	INP5	Counter 5 input
8	+5V	+5V output	36	GAT5	Counter 5 gate
9	INT	Interrupt input	37	OUT5	Counter 5 output
10	GND	Ground	38	GND	Ground
11	RSVD	Do not connect	39	RSVD	Do not connect
12	RSVD	Do not connect	40	RSVD	Do not connect
13	RSVD	Do not connect	41	RSVD	Do not connect
14	RSVD	Do not connect	42	RSVD	Do not connect
15	RSVD	Do not connect	43	RSVD	Do not connect
16	RSVD	Do not connect	44	RSVD	Do not connect
17	RSVD	Do not connect	45	RSVD	Do not connect
18	+5V	+5V output	46	RSVD	Do not connect
19	DI CTL	Pull-up/down connection	47	RSVD	Do not connect
20	GND	Ground	48	GND	Ground
21	DI0	Digital input	49	DO0	Digital output
22	DI1	Digital input	50	DO1	Digital output
23	DI2	Digital input	51	DO2	Digital output
24	DI3	Digital input	52	DO3	Digital output
25	DI4	Digital input	53	DO4	Digital output
26	DI5	Digital input	54	DO5	Digital output
27	DI6	Digital input	55	DO6	Digital output
28	DI7	Digital input	56	DO7	Digital output

**Meilhaus Electronic GmbH**  
**Fischerstrasse 2**  
**D-82178 Puchheim, Germany**  
**Phone: +49 (0)89 89 01 66-0**  
**Fax: +49 (0)89 89 01 66-77**  
**E-Mail: [sales@meilhaus.com](mailto:sales@meilhaus.com)**  
**<http://www.meilhaus.com>**