# AN11242

## Frequency counter using the analog comparator of the LPC11Axx

Rev. 1 — 23 July 2012 Application note

**Document information**

| Info | Content |
|---|---|
| **Keywords** | LPC11Axx, LPC11A14, Analog Comparator, Frequency Counter |
| **Abstract** | This application note shows how to use the analog comparator in the LPC11Axx parts for a frequency counter/zero crossing detection application. |

### Revision history

| Rev | Date | Description |
|---|---|---|
| 1 | 20120723 | Initial version. |

# Contact information

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

# 1. Introduction

This application note describes how the analog comparator of the LPC11Axx can be used to perform frequency and duty-cycle measurements of a signal up to 10 MHz without the usage of any additional circuitry. This introduction chapter introduces the new LPC11Axx series and the frequency counter application briefly. In the following chapters, the analog comparator and how it is used in this application are described in more detail.

## 1.1 LPC11Axx

The LPC11Axx is a series of microcontrollers from NXP featuring the ARM Cortex M0 core. The LPC11Axx differs from the standard LPC11xx by:

- Highly flexible analog comparator with a programmable voltage reference.
- UnderVoltage LockOut (UVLO) protection against power-supply drop below 2.4 V.
- Integrated temperature sensor.
- 10-bit DAC with flexible conversion triggering.
- Internal voltage reference.

For this application the LPC11Axx has been selected because of its analog comparator, which makes it ideal to do zero-crossing detection and frequency measurement. The comparator is used in three different ways:

- Frequency measurement.
- Duty-cycle measurement.
- Auto-calibration of the comparator.

## 1.2 Running the application

This application has been designed to work on the IAR LPC11A14-SK evaluation board. It measures both the frequency and the duty cycle of a single input signal. The result of the measurement is displayed on the 1x8 character LCD on the board and is transmitted over the UART. Three IDEs are supported:

- LPCXpresso.
- Keil µVision.
- IAR Embedded Workbench.

Before the application can be run, the project needs to be built in any of the three mentioned IDEs and programmed through the SWD (Fig 1).

The power to the LPC11A14-SK board is derived from the USB cable. In order to use the application, the following connections need to be made:

- Input signal to pin P0.16/ACMP_I3. More details on how to properly connect the input signal is given in Chapter 1.2.1.
- RS232 cable to the RS232 connector.
- A terminal program (e.g. PuTTY) should connect to this board at 115200 baud, 8 databits, 1 stop bit, no parity.
- USB cable to the USB connector to supply power.

Fig 1 shows the board connected and running.

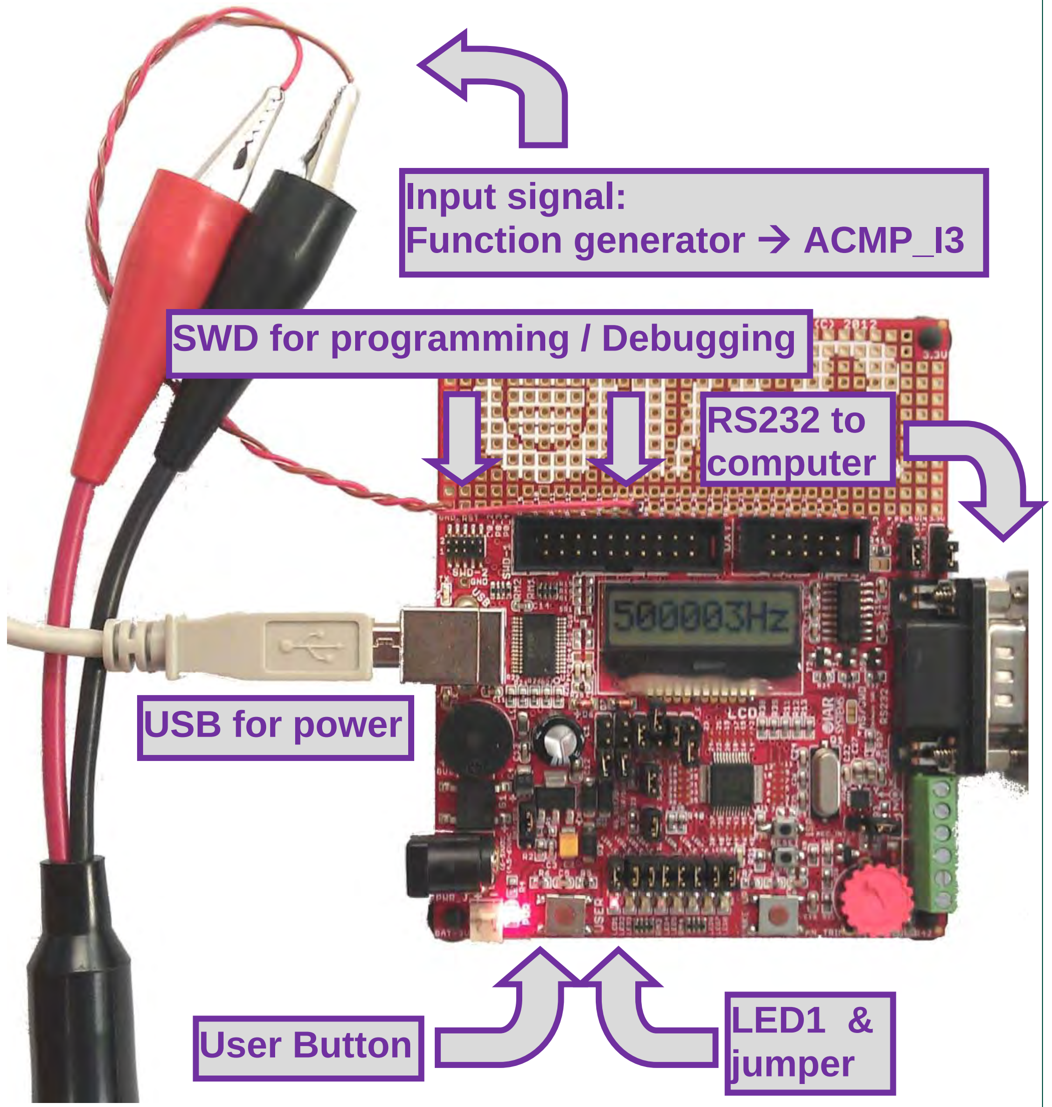


**Fig 1.** IAR LPC11A14-SK evaluation board as a frequency counter. Input signal is a 500 kHz sine wave. Note: LED1 is also connected to ACMP_I3. Therefore, it is also possible to apply the signal to the jumper associated with LED1 instead of soldering wires as shown in the picture.

### 1.2.1 Supplying an input signal

The input signal to be measured needs to be connected to pin P0.16/ACMP_I3. To avoid damaging the input pin from overvoltage, protection circuitry should be employed.

The simplest solution is to add a 10 kΩ resistor in series with the signal to be measured.

A safer and more advanced solution for input protection is shown in Fig 2. This circuit enables the frequency counter to operate with an input signal up to 50 $V_{pp}$.

 

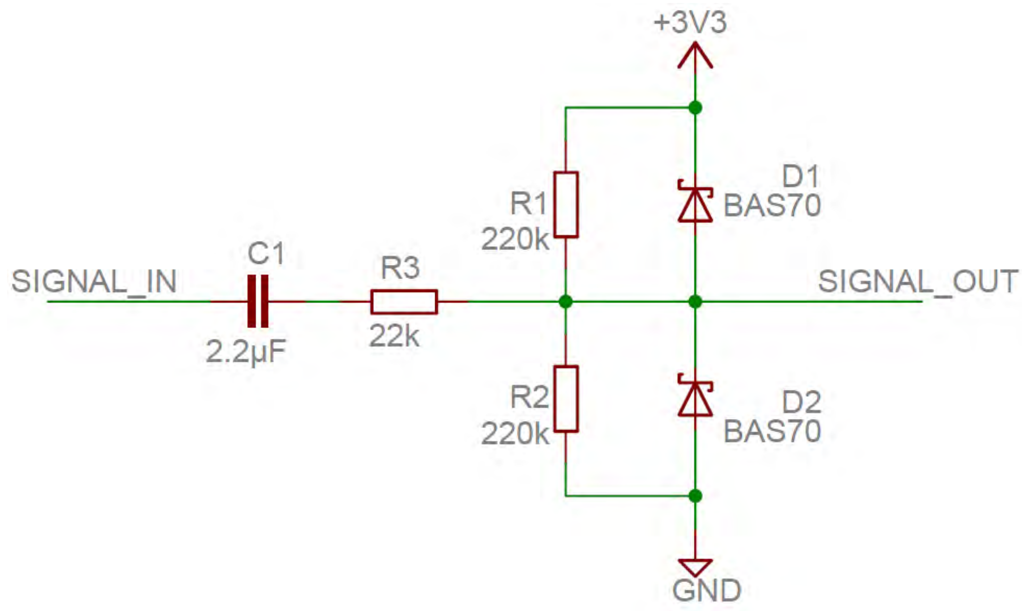


**Fig 2. Proposed circuit to protect the input of the MCU against overvoltage. Circuit protects up to 50$V_{pp}$.**

### 1.2.2 Using the application

The application can be used after programming the board and making the required connections (Fig 1). After startup, the system auto-calibrates the comparator for 1 second for best performance. The frequency and duty cycle of the input signal is then measured and displayed periodically. The result can be seen in Fig 1 and Fig 3 where a 500 kHz sine wave was measured. The auto-calibration can be repeated, if required, by pressing the user button (Fig 1).

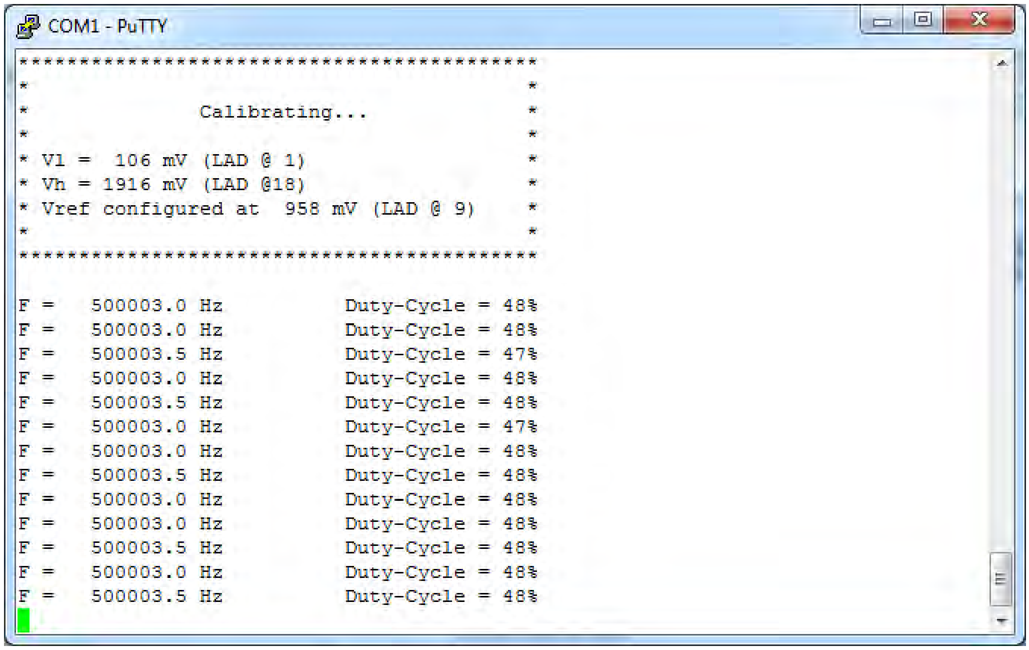


**Fig 3. Output in PuTTY session after calibration. A 500 kHz sine wave was applied to the input.**

# 2. Analog comparator peripheral

The analog comparator is a new peripheral introduced in the LPC11Axx series. This chapter gives a short overview of this peripheral and its features.

The analog comparator has the following features:

- Selectable external inputs can be used as either the positive or negative input of the comparator.
- Internal voltage reference and temperature sensor can be used as either the positive or negative input of the comparator.
- 32-stage voltage ladder can be used as either the positive or negative input of the comparator.
- Voltage ladder source selectable between the VDD/VDD(3V3) or VDDCMP pin.
- Voltage ladder can be powered down when not in use.
- Interrupt capability.
- Selectable 0 mV, 10 mV (±5 mV), and 20 mV (±10 mV), 40 mV (± 20 mV) input hysteresis.

Fig 4 shows the block diagram of the comparator peripheral.

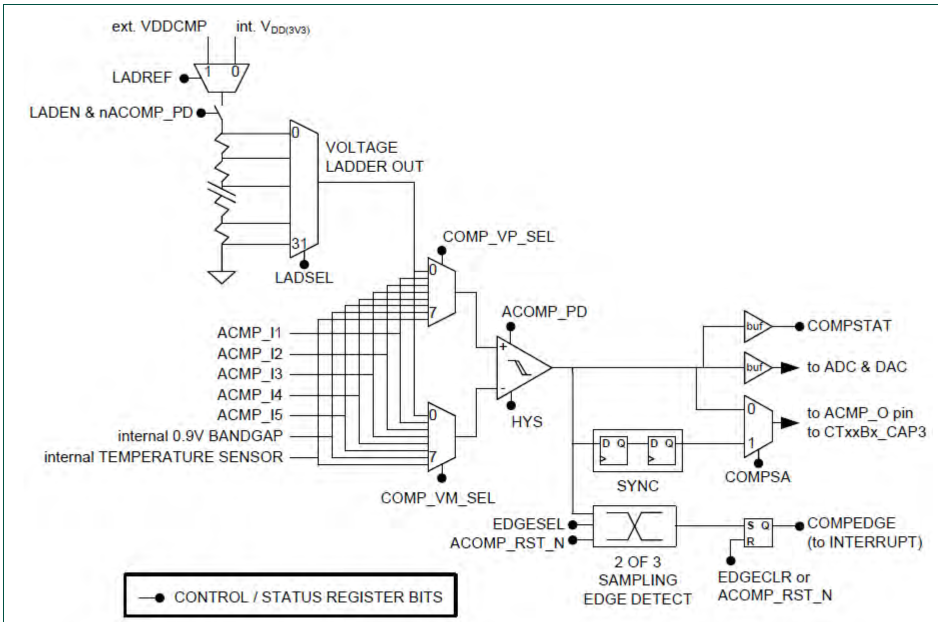


**Fig 4. Block diagram of the analog comparator peripheral.**

The output of the comparator can be connected to the ADC/DAC and the timer capture input 3 (CTxxBx_CAP3).

The connection to the ADC/DAC enables the comparator to trigger an Analog-to-digital or Digital-to-analog conversion.

The connection to the capture input of the timer modules enables the comparator to control the timer modules, e.g. increment the counter on an edge trigger or copy the value of the timer to the associated match register upon an edge trigger.

# 3. Frequency measurement using the analog comparator

This chapter explains how the analog comparator in the LPC11Axx can be used for frequency measurement.

First, the incoming analog signal is converted to a digital signal via the comparator. For this example, the inverting input (V-) of the comparator is connected to the voltage ladder and the non-inverting input (V+) is connected to the signal to be measured.

The voltage ladder should be set to half the peak-to-peak voltage ($V_{pp}$) of the non-inverting input (Chapter 5). To make the comparator more immune to noise, hysteresis is enabled on the comparator.

The connection for the analog comparator is shown in Fig 5. The output of this configuration is shown in Fig 6.

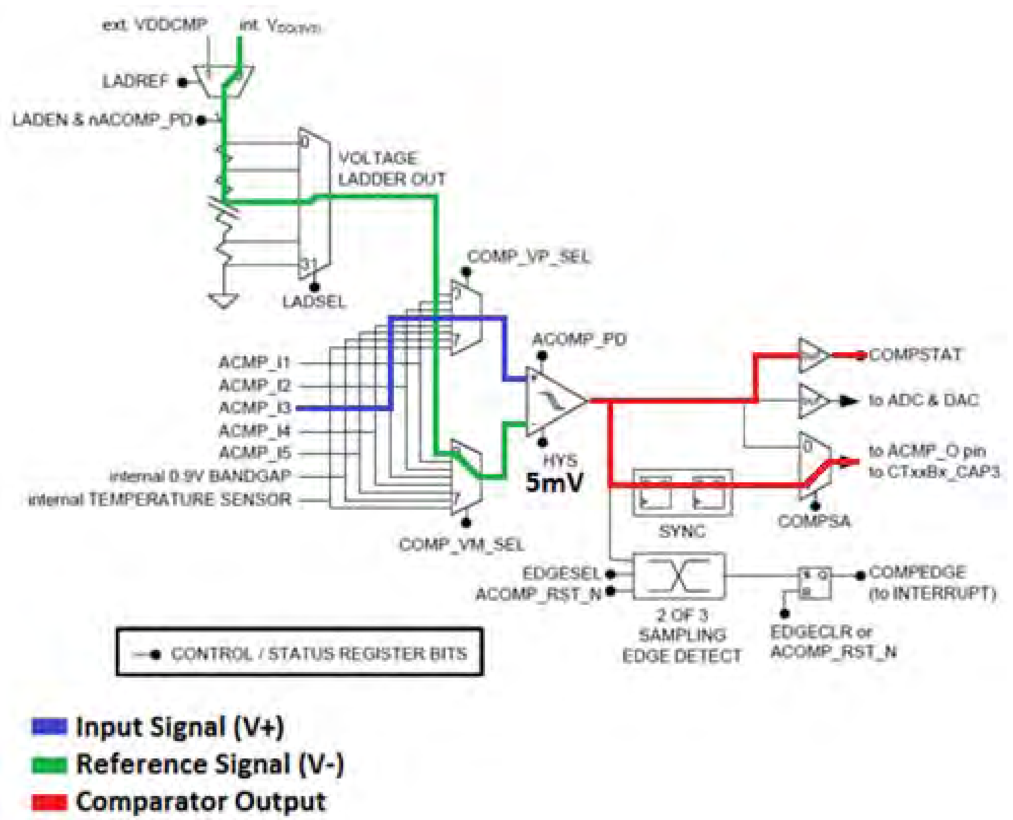


**Fig 5. Analog comparator peripheral as used in this application.**

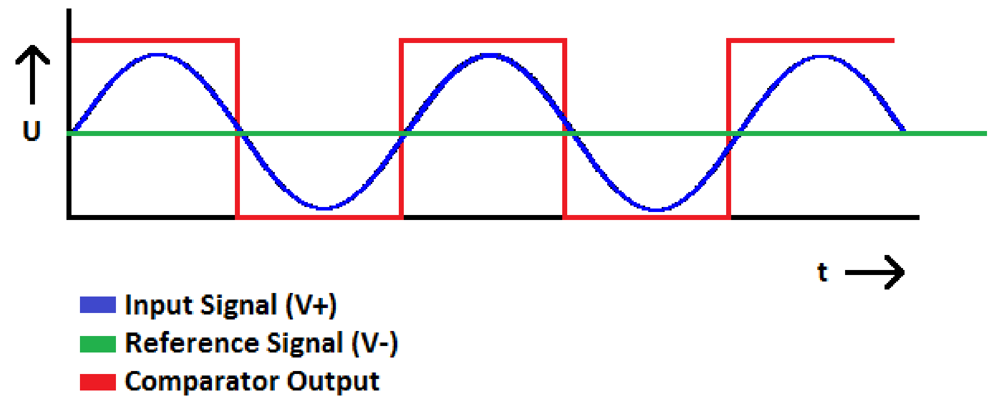


**Fig 6. Output of the analog comparator when configured as in Fig 5.**

The next step is to determine the frequency of the input signal. The two most straight-forward methods for frequency measurement are:

- Measuring the period of the signal.
- Count the number of edges in a certain timeframe.

Each of the two methods can be easily implemented using the LPC11Axx's analog comparator. Both of them have their own pros and cons. The $2^{nd}$ method is implemented for this application note. The two methods are discussed below.

## 3.1 Method 1: Measuring the period of the signal

The frequency of a signal can be easily measured by measuring the duration of one period, e.g. the time between two rising edges or between two falling edges (Fig 7).

With the LPC11Axx, this can easily be done by feeding the output of the comparator to both the timer value reset and the capture input of a timer module:

- The timer module is configured to increment as fast as possible. As the maximum frequency specified for the LPC11Axx is 50 MHz, the timer is set to increment at a 50 MHz rate. At this rate, the timer is incremented every 20 ns.
- The timer module is configured to capture its timer value on a rising edge to the capture register.
- The timer module is configured to clear its timer value after a rising edge.
- The captured timer value is the period of the signal in terms of $1/F_{timer}$. So, with a 50 MHz timer clock, each count in the capture register represents 20 ns.

Fig 7 graphically shows the above described measurement.

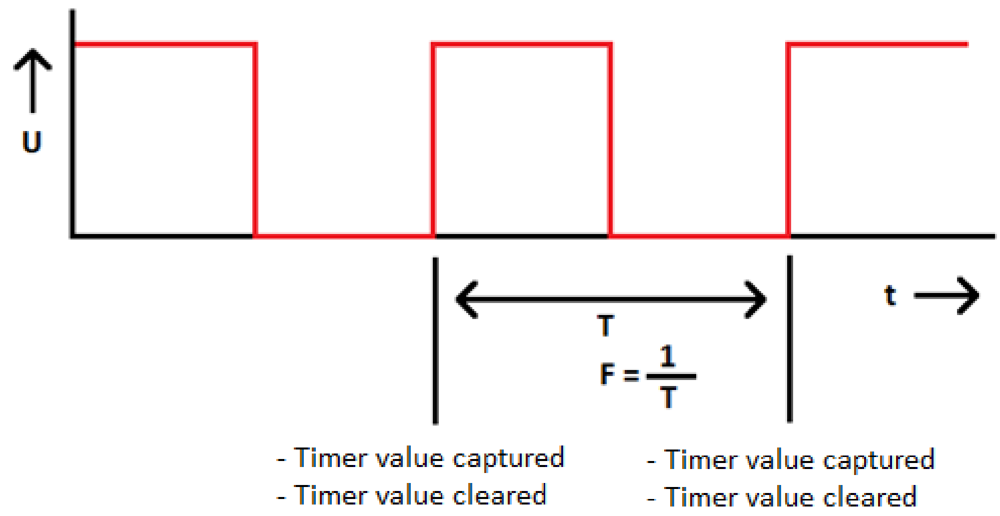


**Fig 7. The frequency of a signal can be calculated by measuring the duration of a single period. This can easily be done by feeding the comparator output to both the timer reset and timer capture input.**

This method has a number of pros and cons.

Pros:

- Fast for high frequency. Just a single period of the input signal is enough to calculate the frequency.
- High resolution for low frequencies. At low frequencies, the resolution is very good. In the above described example ($F_{timer}$ = 50 MHz), the resolution of a 10 Hz input signal is 2 µHz and the resolution of a 1 Hz input signal is 20 nHz (Fig 8).
- Easy to implement.

Cons:

- Inaccurate for high frequencies. Frequencies near $F_{timer}$ have a very poor resolution. With $F_{timer}$ at 50 MHz, the resolution of a 1 MHz signal is 20 kHz and a 10 MHz signal has a resolution of 2 MHz (Fig 8).

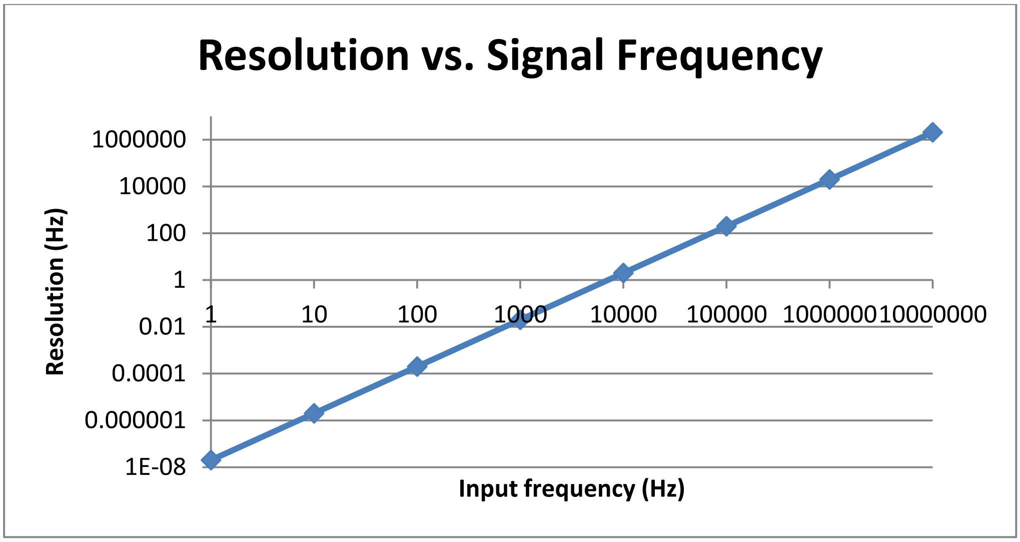


**Fig 8. Graph showing the relation between the resolution of the measurement and the frequency of the input signal when using the described Method 1. Assumed $F_{timer}$ of 50 MHz.**

## 3.2 Method 2: Count number of edges in a certain timeframe

The second method to measure the frequency is to count the number of edges in a certain timeframe (Fig 9).

With the LPC11Axx this can be easily done by feeding the output of the comparator to the capture input of a timer module:

- The timer module is configured as a counter. At each edge, rising and falling, the counter is incremented.
- After a defined period of time, e.g. 1s, the value of the counter is read and subtracted from the previous value. The result is the number of falling and rising edges within that period of time. The frequency can be calculated by dividing the number of edges by the time between the two read-outs.  The result is then divided by 2 to compensate for double-edged triggering:

$$F = \frac{\left(\frac{Count_2 - Count_1}{t}\right)}{2}$$

Fig 9 graphically shows the above described measurement.

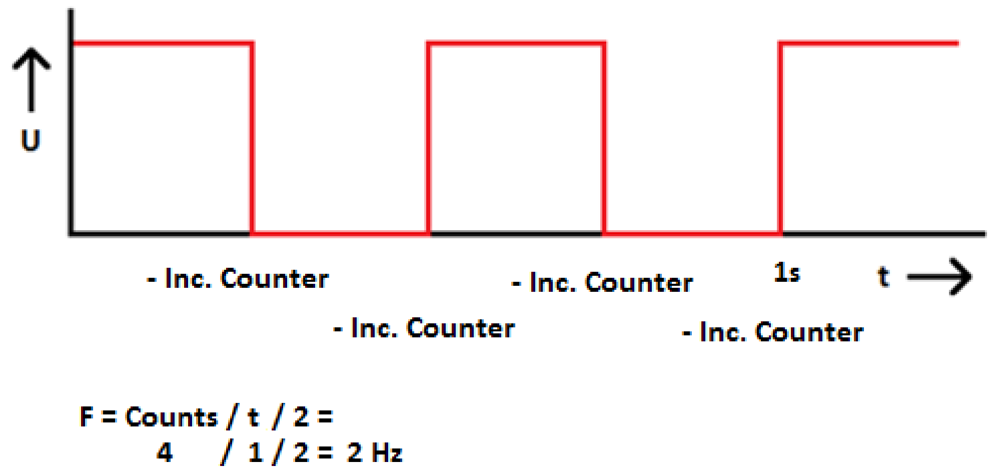


**Fig 9. The frequency of a signal can be calculated by measuring the number of edges in a defined period of time. This can easily be done by feeding the comparator output to a timer capture input and incrementing the counter on each edge on the capture input.**

This method has a number of pros and cons.

Pros:

- Good resolution for both low and high frequencies. When the timeframe is set to 1s, the resolution is 0.5 Hz, independent of the frequency of the signal.
- Easy to implement.

Cons:

- Resolution for low frequencies (0.5 Hz when using a 1s timeframe) not as good as with Method 1.

This method is chosen to be implemented for the frequency counter application because of the better resolution at high frequencies when compared to Method 1. While the resolution at low frequencies is not as good when compared to Method 1, it is good enough for a frequency counter. The implementation of this method can be found in Chapter 5.

 

# 4. Duty-cycle measurement using the analog comparator

The duty cycle of a square wave signal is defined as the ratio between the duration of its high level vs. the duration of its low level (Fig 10). With the analog comparator of the LPC11Axx, the duty cycle of a signal can easily be measured.


**Fig 10. The duty cycle of a square wave signal is defined as the ratio between the duration of its high level vs. the duration of its low level.**

For this application the method used to calculate the duty cycle is as follows:

- The frequency of the signal is known (Chapter 3).
- The high time of the signal is calculated using a second timer module using a similar method as described in Chapter 3.1. The timer value is reset on a rising edge and captured on a falling edge, effectively measuring the time that the signal is high.
- The duty cycle is calculated by dividing the high time (multiply by 100 to get the result as a percentage) by the time of a complete period (1/Frequency).

# 5. Software implementation

This chapter provides more information on how the frequency and duty-cycle measurements, as described in Chapter 3 and Chapter 4, and the auto-calibrate function are implemented in software.

The high-level block diagram is shown in Fig 11. The application uses three timers and one interrupt:

- CT32B0 is used for the frequency counter and is triggered by the analog comparator (Chapter 3.2).
- CT32B1 is used for the duty-cycle measurement (Chapter 4).
- Systick timer is used to generate an interrupt every 1/128$^{th}$ second.

At startup, the calibration mode is initialized. The calibration mode is executed in the main loop. The duration of the calibration, one second, is handled by the systick timer.

When the calibration is done, the mode is changed to NormalOperation. In NormalOperation, frequency and duty-cycle measurement are performed. The NormalOperation is fully controlled by the CT32 timers and the systick timer interrupt.

Upon completion of a frequency and duty-cycle measurement, the state of the application is changed to Display.  In this state, measured data is outputted to the LCD and the UART.

The status of the push button, USER, is polled every 1/128$^{th}$ of a second (in the systick timer interrupt). If the button is pressed, the state will change to the Calibration mode.

 

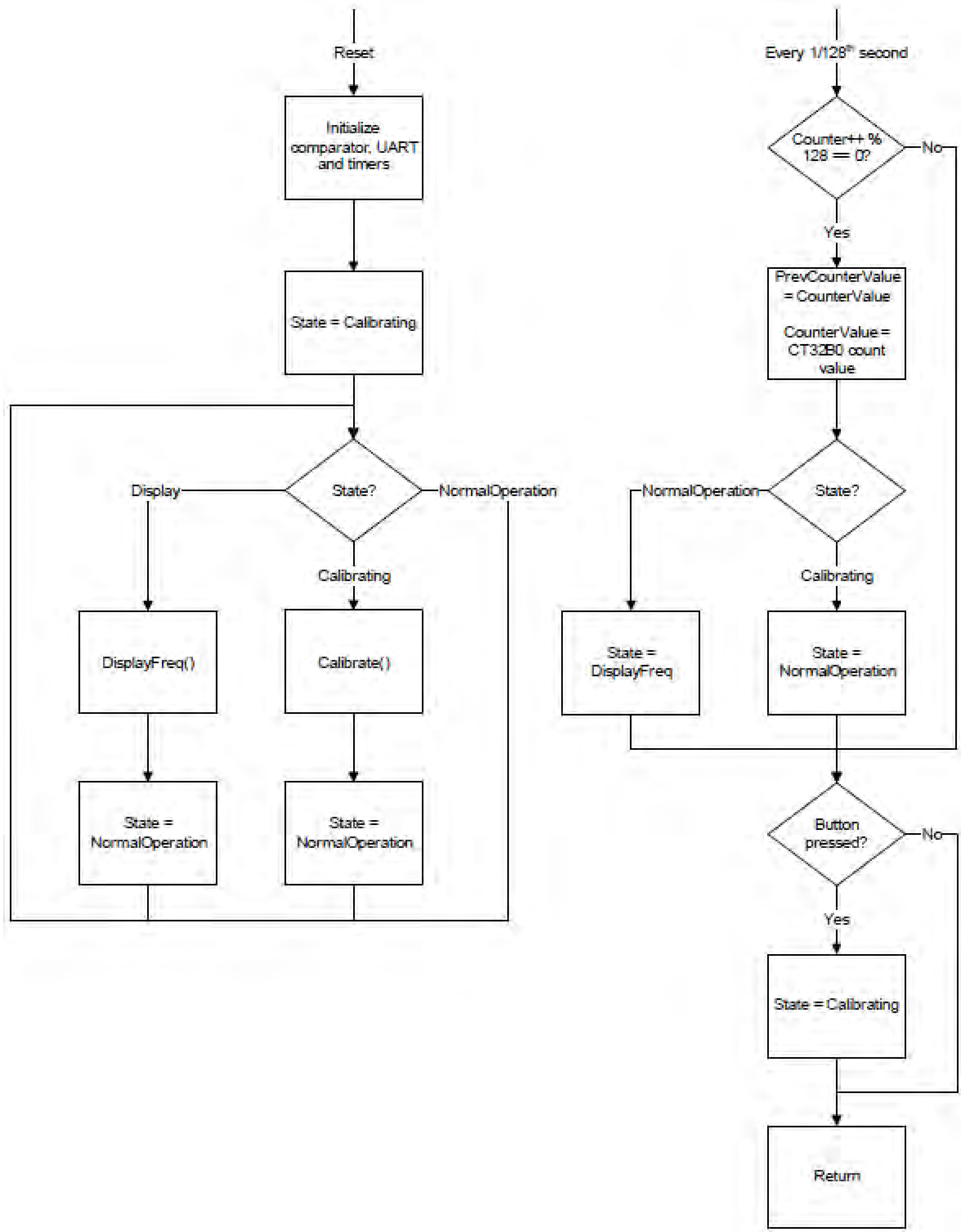


**Fig 11. High-level software flow chart**

## 5.1 Auto-calibration

In order for the frequency counter to work correctly on non full-scale signals, the reference of the comparator (V-, Fig 4) must be set to the mid-level of the input signal. An auto-calibration function is implemented for this.

As can be seen in Fig 4, the reference signal of the comparator is derived from the voltage ladder. The auto-calibration mode sweeps the voltage ladder as quickly as possible over its 32 positions while keeping track of the output of the comparator. This way the maximum and minimum voltage of the signal are determined and the mid level can be calculated (the average of the minimum and maximum voltage). This flow of the algorithm is outlined in Fig 12. The provided source code is listed below:

```
/* initialize detected minimum / maximum values */
LowestVal = 31;
HighestVal = 0;
while(State == Calibrating)
{
   for(LAD=0; LAD<32; LAD++)
   {
      /* Cycle through all 32 voltage ladder positions for one second */
      LPC_CMP->LAD = (1<<0)|(LAD<<1);
      /* short delay to get a stable output from the comparator */
      for(i=0;i<64;i++)
         __NOP();

      /* Peak detector for highest voltage ladder position */
      if((LPC_CMP->CTL & (1<<21)) && (LAD > HighestVal))
         HighestVal = LAD;
      /* Peak detector for lowest voltage ladder position */
      else if(!(LPC_CMP->CTL & (1<<21)) && (LAD < LowestVal))
         LowestVal = LAD;
   }
}
```

## 5.2 Frequency measurement

The frequency measurement is performed by counting the number of rising and falling edges.  This method is described in detail in Chapter 3.2.

The frequency measurement relies on the timer module CT32B0. CT32B0 is configured as a counter set to triggered on a rising and falling edge of the comparator output.  The previous number of edges and the current number of edges is saved to memory every second. When the state is changed to DisplayFreq, the frequency is calculated and outputted to the LCD and the UART.

The code below is used to configure the analog comparator:

```
/* Configure analog comparator */
LPC_CMP->CTL = (1<<6)|(3<<8)|(1<<25); /* Use synchronization logic, positive
                                         voltage input = ACMP_I3,
                                         negative voltage input = voltage ladder
                                         output, enable 5mV hysteresis */
LPC_CMP->LAD = (1<<0);                /* Enable voltage ladder, ref = VDD */
```

The code below is used to configure timer module CT32B0 to act as a counter and to increment on every rising and falling edge of the analog comparator:

```
/* Configure CT32B0 for frequency measurement */
LPC_CT32B0->CTCR = (3<<0)|(3<<2); /* Counter mode, triggered by rising
                                     and falling edge of analog comparator */
LPC_CT32B0->TCR = (1<<1);         /* Reset counter and prescaler */
LPC_CT32B0->TCR = (1<<0);         /* Enable counter and prescaler */
```

## 5.3 Duty-cycle measurement

The duty-cycle measurement is performed by calculating the ratio of the duration of the high level time to the duration of a full period. This method is described in detail in Chapter 4.

The duty-cycle measurement relies on the previous calculated frequency and timer module CT32B1. CT32B1 is configured to act as a timer which reset on every rising edge of the analog comparator's output. The timer value is captured to a capture register on every falling edge of the analog comparator's output (Chapter 4).

The analog comparator is already configured by the code used for the frequency measurement (Chapter 5.1).

The code used to configure timer module CT32B1 to act as a timer is shown below:

```
/* Configure CT32B1 for duty-cycle measurement */
LPC_CT32B1->CTCR = (1<<4)|(6<<5);  /* Timer mode, reset timer on rising edge of
                                       the comparator */
LPC_CT32B1->CCR = (1<<10);         /* Load CR3 on falling edge of comparator */
LPC_CT32B1->TCR = (1<<1);          /* Reset counter and prescaler */
LPC_CT32B1->TCR = (1<<0);          /* Enable counter and prescaler */
```

 

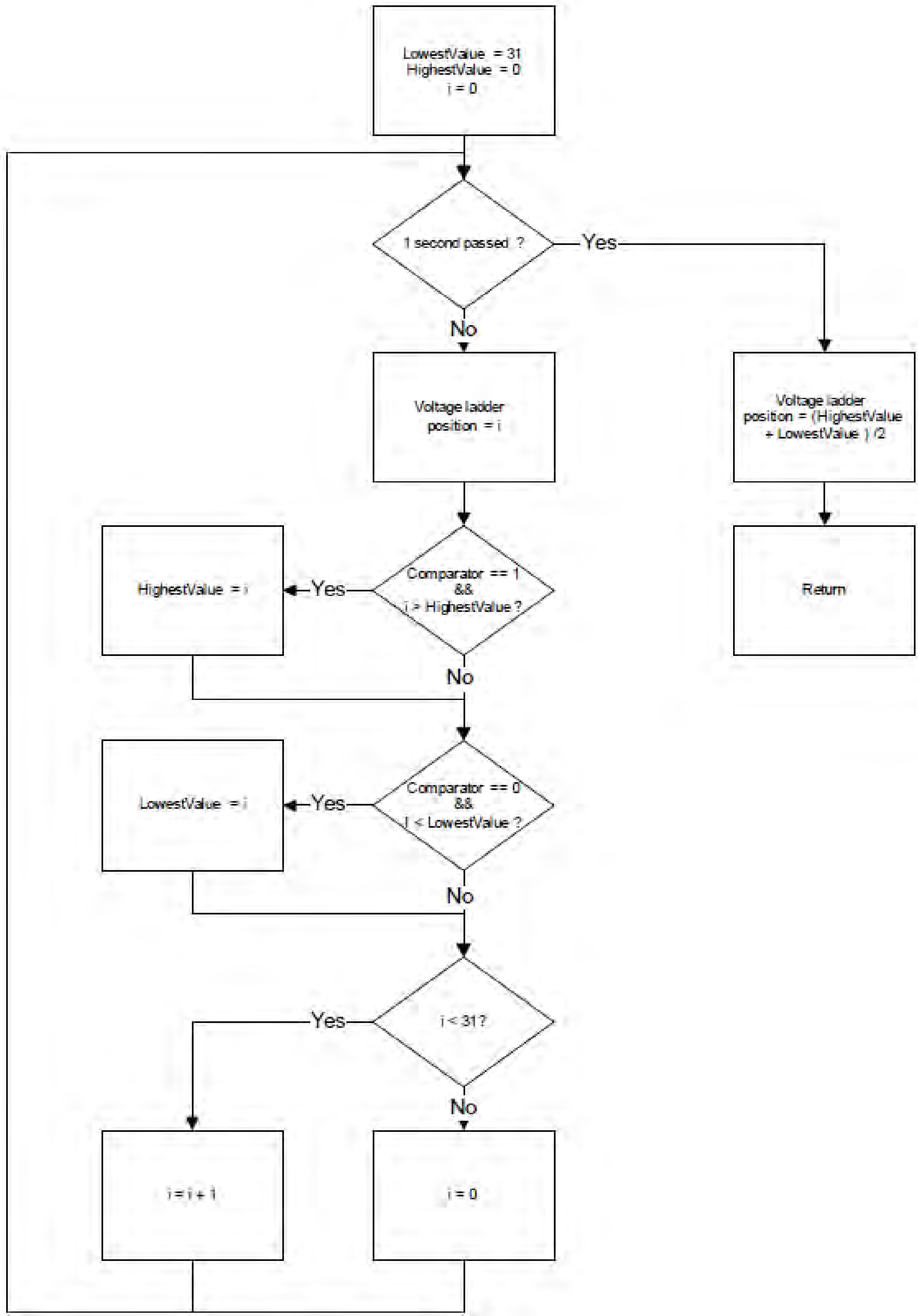


**Fig 12. Flowchart of auto-calibrate algorithm**

# 6. Conclusion

The LPC11Axx series contains an ARM Cortex M0 core with integrated analog peripherals, such as an analog comparator, 10-bit ADC, 10-bit DAC, UVLO and temperature sensor.

The analog comparator in the LPC11Axx series is ideal for frequency measurement and zero-crossing detection because of its good interconnectivity to the other peripherals of the LPC11Axx.

The analog comparator can be used to measure signals with low frequencies (e.g. 1 Hz) and high frequencies (e.g. 10 MHz).

Sample code and sample projects for LPCXpresso, IAR Embedded Workbench and Keil µVision are provided by NXP.

## 7. References

[1] LPC11Axx datasheet, NXP Semiconductors, 4 July 2012, Rev. 2.1

[2] LPC11Axx user manual, NXP Semiconductors, 7 June 2012, Rev 2.0

# 8. Legal information

## 8.1 Definitions

**Draft —** The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

## 8.2 Disclaimers

**Limited warranty and liability —** Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

**Right to make changes —** NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

**Suitability for use —** NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

**Applications —** Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

**Export control —** This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

**Evaluation products —** This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

## 8.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.

# 9. Contents





**Date of release: 23 July 2012**
**Document identifier: AN11242**