Application Note, V1.1, Mar. 2007

AP32112

TriCore

Incremental Encoder Interface using the General Purpose Timer Unit (GPTU)

Microcontrollers

Never stop thinking

Edition 2007-03

Published by
Infineon Technologies AG
81726 München, Germany

LEGAL DISCLAIMER

THE INFORMATION GIVEN IN THIS APPLICATION NOTE IS GIVEN AS A HINT FOR THE IMPLEMENTATION OF THE INFINEON TECHNOLOGIES COMPONENT ONLY AND SHALL NOT BE REGARDED AS ANY DESCRIPTION OR WARRANTY OF A CERTAIN FUNCTIONALITY, CONDITION OR QUALITY OF THE INFINEON TECHNOLOGIES COMPONENT. THE RECIPIENT OF THIS APPLICATION NOTE MUST VERIFY ANY FUNCTION DESCRIBED HEREIN IN THE REAL APPLICATION. INFINEON TECHNOLOGIES HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND LIABILITIES OF ANY KIND (INCLUDING WITHOUT LIMITATION WARRANTIES OF NON-INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS OF ANY THIRD PARTY) WITH RESPECT TO ANY AND ALL INFORMATION GIVEN IN THIS APPLICATION NOTE.

Information

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office (**www.infineon.com**).

Warnings

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

TriCore

Revision History: V1.1, 2007-03

Previous Version(s):
none

Page	Subjects (major changes since last revision)

We Listen to Your Comments
Any information within this document that you feel is wrong, unclear or missing at all? Your feedback will help us to continuously improve the quality of this document. Please send your proposal (including a reference to this document) to:
mcdocu.comments@infineon.com

1 Introduction

A single track of slots patterns the periphery of an incremental encoder disk, as shown in **Figure 1**. These slots create an alternating pattern of dark and light lines. The disk count is defined as the number of dark/light line pairs that occur per revolution (lines per revolution). As a rule, a second track is added to generate a signal that occurs once per revolution (index signal), which can be used to indicate an absolute position.

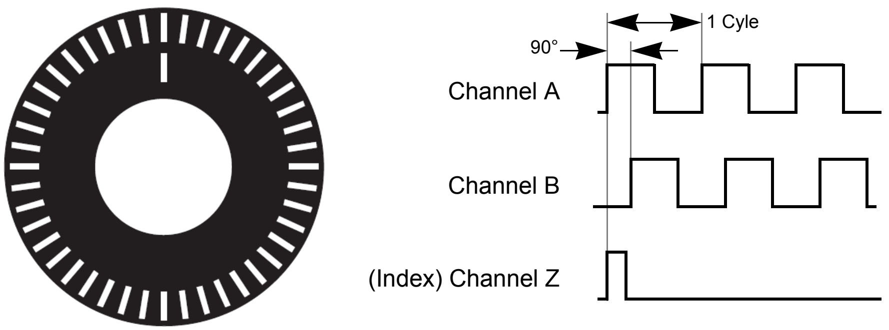

Figure 1 Optical Encoder disk

To derive direction information, the lines on the disk are read out by two different photo-elements that "look" at the disk pattern with a mechanical shift of 1/4 the pitch of a line pair between them. This shift is realized with a reticle or mask that restricts the view of the photo-element to the desired part of the disk lines. As the disk rotates, the two photo-elements generate signals that are shifted typically 90° out of phase from each other. These are commonly called the quadrature channel A and channel B signals. The clockwise direction for most encoders is defined as the channel A going positive before the channel B channel.

The encoder wheel typically makes one revolution for every revolution of the motor or the wheel may be at a geared rotation ratio with respect to the motor. Therefore, the frequency of the digital signal coming from the channel A and channel B outputs varies proportionally with the velocity of the motor. For example, a 1024-line encoder directly coupled to a motor running at 6000 revolutions per minute (rpm) results in a frequency of 102.4 KHz, so by measuring the frequency of either the channel A or channel B output, the processor can determine the speed of the motor.

2 Implementation

This TC1130 implements the General Purpose Timer Unit GPTU module. The GPTU consists of three 32-bit timers T0, T1 and T2 designed to solve such application tasks as event timing, event counting, and event recording (**Figure 2**). For a detailed description of the GPTU module see **[1]**. This quadrature encoder interface uses timer T2 to measure speed and position.

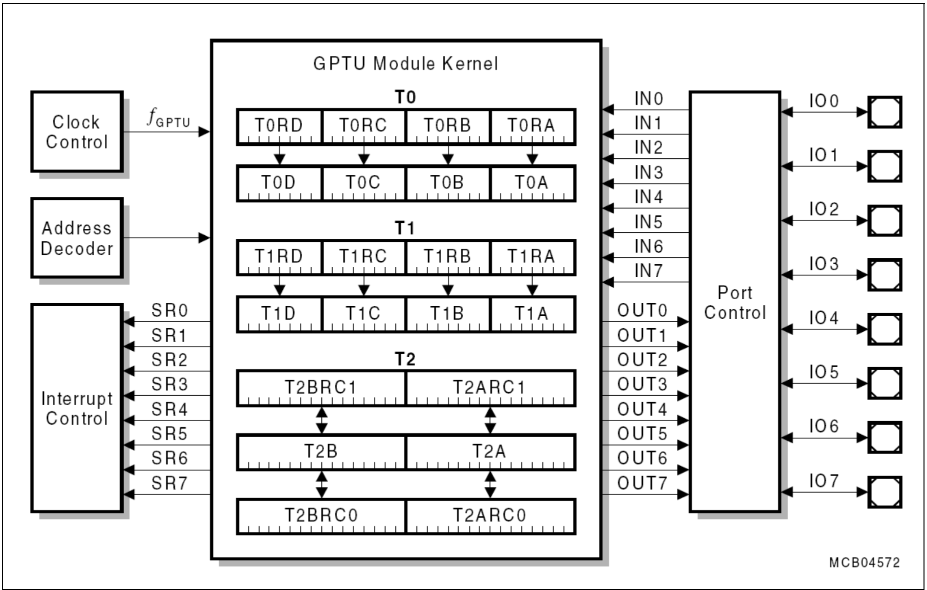

Figure 2 General Block Diagram of the GPTU Interface

Position tracking can be performed with timer T2 in quadrature counting mode, sometimes referred to as incremental or phase encoded interface. The standard way of tracking positions is to use two phase-shifted input signals. These provide the counting and direction information necessary for this task. As shown in **Figure 3**, the edges of the signals provide the count signal, while the phase relation between the two signals provides the direction information. To operate Timer T2 in this mode, the two signals are connected such that they trigger the Count_A/Count_B and the UpDown_A/UpDown_B inputs of the timer block.

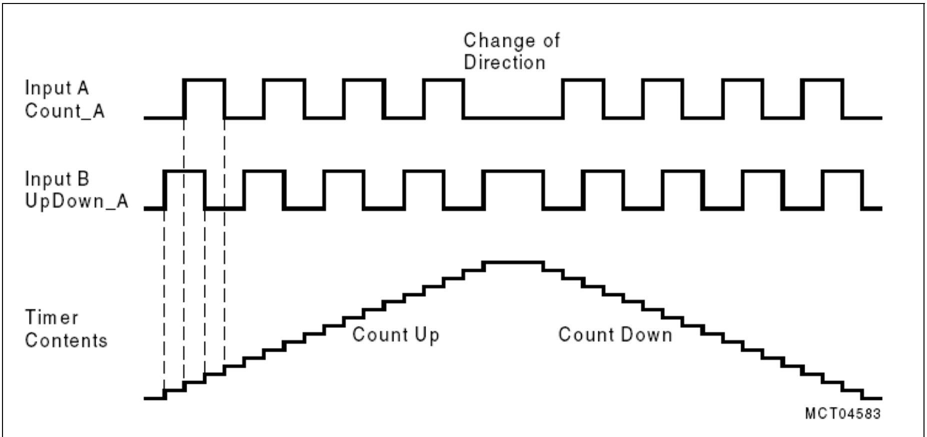

Figure 3 Quadrature Counting Operation

2.1 Position sensing

The position and speed sensing algorithm uses the GPTU module timer T2 in split mode (**Figure 4**). Timer T2A is used for position sensing. The encoder outputs channel A and channel B are connected to the T2A inputs Count_A and UpDown_A (**Figure 5**). Timer T2A is set to quadrature counting mode, so that the value of timer T2 corresponds to the rotor position. Most quadrature encoder have 2^n lines. T2A is set up to count both, negative and positive, edges of channel A and channel B so that one revolution gives 2^{n+2} edges. The lower (n+2) bits of T2A determines the rotor position, the upper (14-n) bits the number of revolutions, so that a multiturn interface can easily realized. If more revolutions are required an interrupt can be configured on the overflow of the timer T2A.

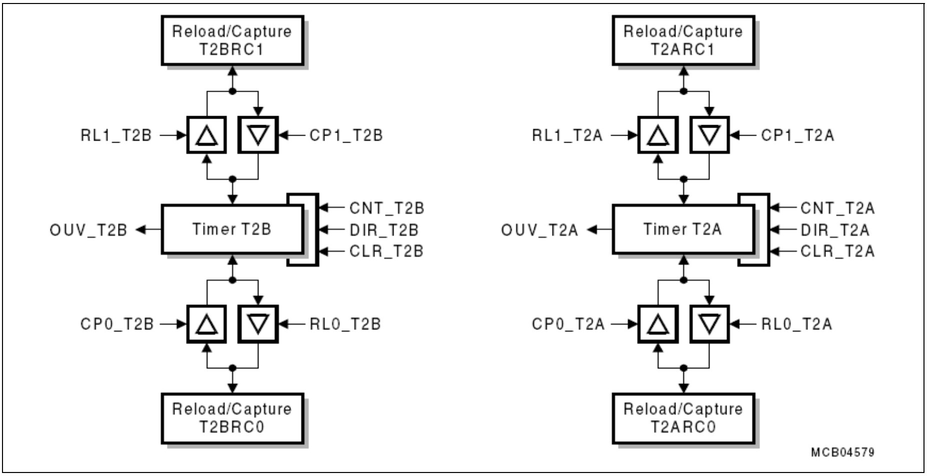

Figure 4 Block Diagram of Timer 2 in Split Mode

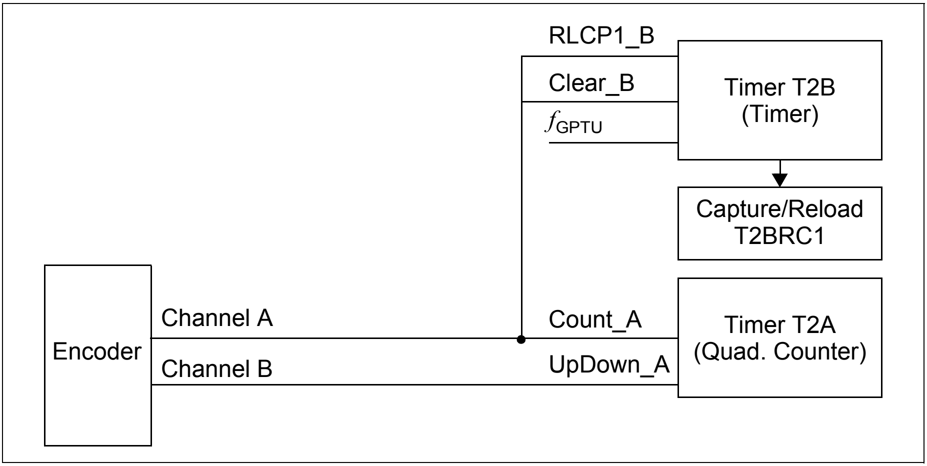

Figure 5 Encoder interface configuration for low to moderate speed (Method 1)

2.2 Speed sensing

There are three common ways to measure speed. The first method measures the time between two following edges of the quadrature encoder, the second method measures the time between multiple edges and the third method measures a position difference per constant time period. The first method is used at low and moderate speed. When the measured period is too short that the speed calculation is not precise, the speed calculation algorithm switches to the second method.

2.2.1 Minimum and Maximum Speed Calculation

The speed calculation for method 1 is done by capturing the time Δt between two following edges of channel A. The angular frequency can be expressed as:

$$\omega = \frac{2\pi}{2N \cdot \Delta t} \quad (1)$$

where

- Δt : edge to edge time of channel A
- N : Number of lines per revolution
- ω : angular frequency

The accuracy of the speed measurement $\delta = \Delta\omega/\omega$ is limited due to the timer resolution of f_{GPTU} and results in:

$$\delta = \frac{\Delta\omega}{\omega} = 1 - \frac{k}{k+1} \quad (2)$$

where:

- k : number of f_{GPTU} timer ticks in Δt

The minimum and maximum speed measurement is limited on the hand by the accuracy δ and for low speed by the maximum value of the timer $FFFF_H$. The product $\Delta t \cdot f_{GPTU}$ has to be in the range of:

$$\frac{1-\delta}{\delta} < \Delta t \cdot f_{GPTU} < 65535 \quad (3)$$

The speed measurement is therefor limited to:

$$\omega_{min} = \frac{2\pi}{2N} \cdot \frac{1}{65535} \cdot f_{GPTU} \quad (4)$$

and

$$\omega_{max} = \frac{2\pi}{2N} \cdot \frac{\delta}{1-\delta} \cdot f_{GPTU} \quad (5)$$

Example: In an application with a 1024 lines encoder, the GPTU running at 75 MHz and a required speed accuracy of 0.3% the speed that can be measured with method 1 ranges from 33.5 rpm to 6612.3 rpm.

To measure even lower speed values, f_{GPTU} can be e.g. dynamically reduced. For higher speed values the active edges can be set from both edges to only one edge to double the ω_{max}. To further exceed the maximum speed and/or the accuracy the software can switch to method 2 to measure the timer between multiple edges. A configuration is shown in **Figure 6**. Timer T2B is not cleared on every edge. The time and position changes are measured in a control loop by the difference to the last values. The calculation can be done quite fast if the 16-bit values of T2A and T2B are read using the 32-bit register GPTU_T2RC1 and the values are subtracted using packed halfword substraction instruction **sub.h**.

Timer T0A with a reload value of FF_H is used for signal conditioning, because Timer T2A has dedicated edge selection for COUT_A and PLCP1_A. A direct connection of channel A to COUT_A and PLCP1_A would results in an undefined signal order.

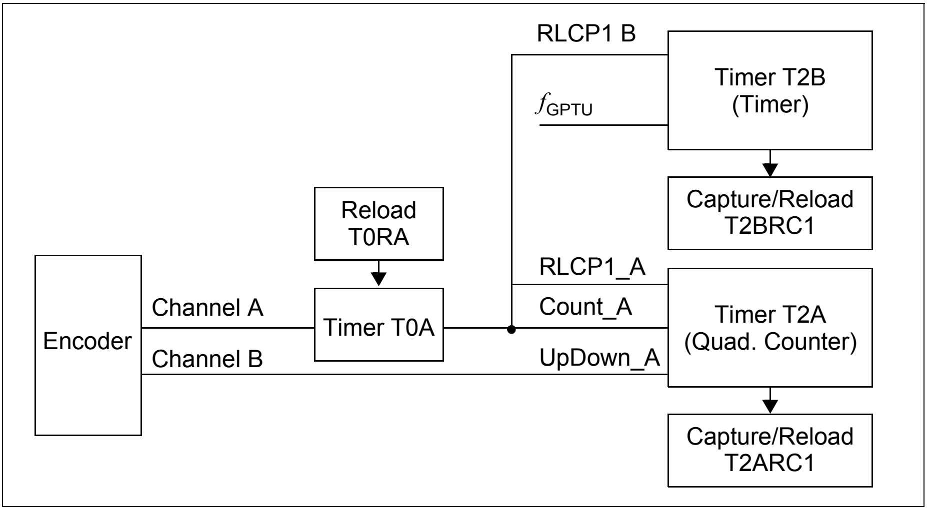

Figure 6 Encoder interface configuration for high speed (Method 2)

The maximum speed that can be measured can be expressed by $\omega_{max} = \frac{2\pi}{2N} \cdot f_{GPTU}$ which means that there must be at least a timer tick for every pulse.

A more detailed analysis that takes the accuracy δ and the control loop frequency f_C into account results in a transcendental equation:

$$\frac{1-\delta}{\delta} = \left\lfloor \frac{2N}{2\pi} \cdot \frac{\omega}{f_C} \right\rfloor \cdot \frac{2\pi}{2N} \cdot \frac{f_{GPTU}}{\omega} \tag{6}$$

where

f_C :Control loop frequency

$\lfloor x \rfloor$: floor(x) function

For typical values for high end drives **Figure 7** shows that the accuracy is decreasing with higher angular frequency and oscillating due to the floor function in equation **(6)**.

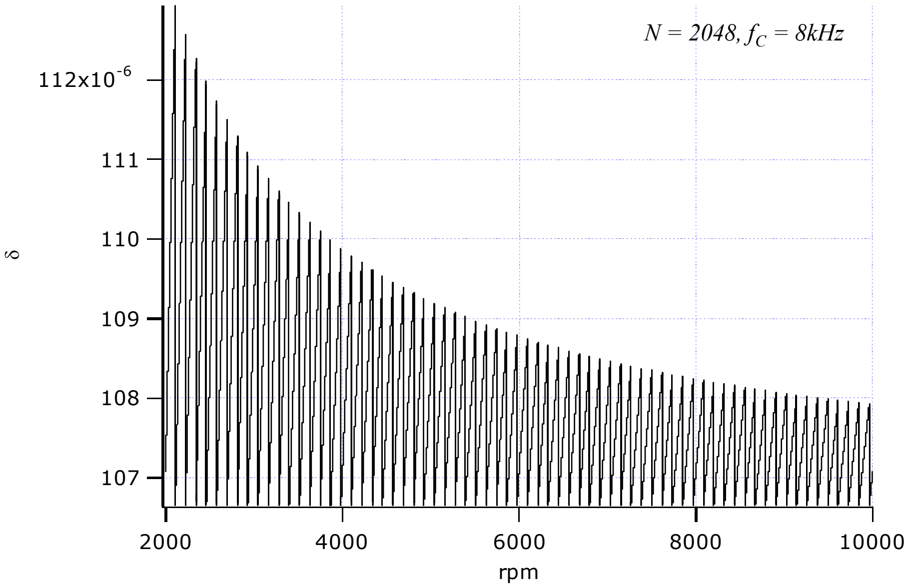

Figure 7 Accuracy versus rpm (Method 2).

Example: In an application with a 2048 lines encoder, the GPTU input clock at 75 MHz, a control loop frequency of 8kHz the accuracy value is decreasing from 0.0113% at 2000 rpm to 0.0108% at 10000 rpm.

3 References

[1] TC1130 32-Bit Single-Chip Microcontroller Volume 2 (of 2): Peripheral Units

w w w . i n f i n e o n . c o m

Published by Infineon Technologies AG