GigaDevice Semiconductor Inc.

Arm® Cortex®-M3/4/23/33 32-bit MCU

应用笔记

AN013

目录

图索引

表索引

1. 简介

GD32E501的可配置逻辑阵列（CLA）为外部引脚、ADC和定时器提供256个可编程数字逻辑操作，而无需CPU干预。GD32E501共有八个独立的CLA单元。每个CLA单元支持GPIO引脚的可配置异步/同步输出。本文提出了一种使用CLA搭建方波滤波器的设计方案。

2. CLA 方波滤波方案

2.1. CLA 方波滤波器原理

由 CLA 组建的方波滤波器架构如*图 2-1. CLA 方波滤波器架构*所示。CLA0~CLA4、CLA6 作为滤波单元，通过各 CLA 之间的逻辑组合，实现滤波功能；CLA5 作为滤波单元的触发信号转换器，CLA5 可将 TIMER2_TRGO 信号转换为 CLA5_ASYNC_OUT，CLA5_ASYNC_OUT 作为滤波单元的同步触发时钟源，滤波单元中只有 CLA3 使用 HCLK 作为触发时钟源。

用户可通过配置 TIMER2_TRGO 的时间间隔，来设置方波滤波器的够滤掉最大毛刺的宽度。

图 2-1. CLA 方波滤波器架构

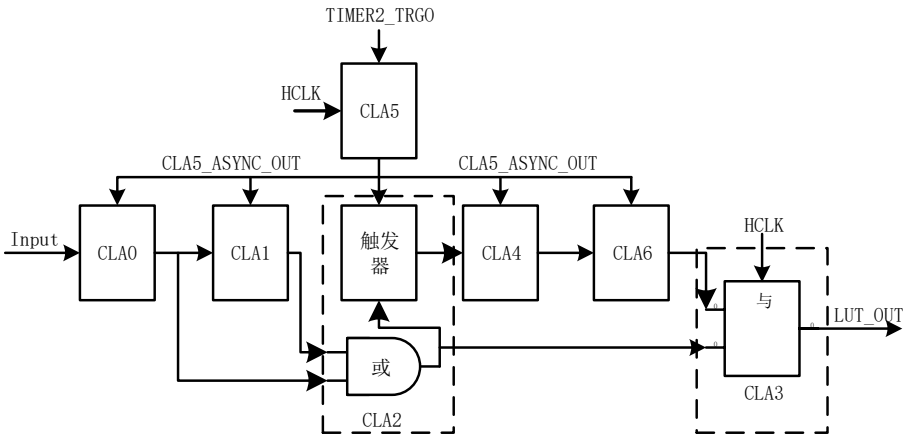
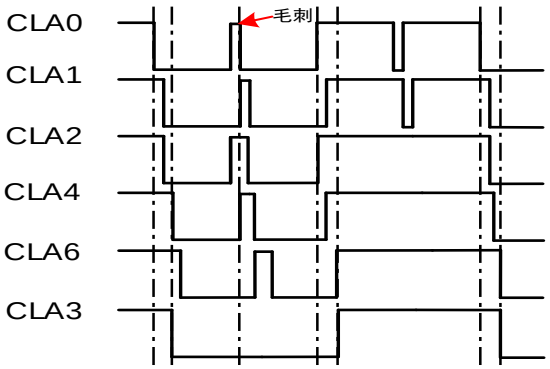


*图 2-2. CLA 方波滤波器的原理*展示了 CLA 方波滤波器对方波信号进行滤波的原理。

图 2-2. CLA 方波滤波器的原理



2.2. CLA 软件配置

方波滤波器中的每个 CLA 都被配置为特定的逻辑，再组合在一起，以达到滤波的目的。CLA0~CLA6 详细配置如*表 2-1. GD 工程下的 CLA 配置*:

表 2-1. GD 工程下的 CLA 配置

```
/*!
    \brief        configure the CLA interface
    \param[in]    none
    \param[out] none
    \retval       none
*/
void cla_config(void)
{
    /* configure the CLA0 */
    /* select CLA5_ASYNC_OUT as input of MUX0 */
    cla_multiplexer_input_config(CLA0,MUX0,CLA0MUX0_CLA5_ASYNC_OUT);
    /* select CLAIN11(PA8) as input of MUX1 */
    cla_multiplexer_input_config(CLA0,MUX1,CLA0MUX1_CLAIN11);
    /* IN1 as CLA0 output */
    cla_lut_control_config(CLA0,0xCC);
    /* select flip-flop result as CLA output */
    cla_output_config(CLA0,FLIP_FLOP_OUTPUT);
    cla_flip_flop_clocksource_config(CLA0,MUX0_OUTPUT);
    cla_flip_flop_clockpolarity_config(CLA0,CLA_CLOCKPOLARITY_POSEDGE);
    cla_flip_flop_output_reset(CLA0);

    /* configure the CLA1 */
    /* select CLA5_ASYNC_OUT as input of MUX0 */
    cla_multiplexer_input_config(CLA1,MUX0,CLA1MUX0_CLA5_ASYNC_OUT);
    /* select CLA0_ASYNC_OUT as input of MUX1 */
    cla_multiplexer_input_config(CLA1,MUX1,CLA1MUX1_CLA0_ASYNC_OUT);
    /* IN0 as CLA1 output */
    cla_lut_control_config(CLA1,0xCC);
    /* select flip-flop result as CLA output */
    cla_output_config(CLA1,FLIP_FLOP_OUTPUT);
    cla_flip_flop_clocksource_config(CLA1,MUX0_OUTPUT);
    cla_flip_flop_clockpolarity_config(CLA1,CLA_CLOCKPOLARITY_POSEDGE);
    cla_flip_flop_output_reset(CLA1);

    /* configure the CLA2 */
    /* select CLA0_ASYNC_OUT as input of MUX0 */
    cla_multiplexer_input_config(CLA2,MUX0,CLA2MUX0_CLA0_ASYNC_OUT);
    /* select CLA1_ASYNC_OUT as input of MUX0 */
    cla_multiplexer_input_config(CLA2,MUX1,CLA2MUX1_CLA1_ASYNC_OUT);
    /* IN0|IN1 */
    cla_lut_control_config(CLA2,0xFC);
    /* select flip-flop result as CLA output */
```

```
cla_output_config(CLA2,FLIP_FLOP_OUTPUT);
cla_flip_flop_clocksource_config(CLA2,TIMER_TRGO);
cla_flip_flop_clockpolarity_config(CLA2,CLA_CLOCKPOLARITY_POSEDGE);
cla_flip_flop_output_reset(CLA2);
cla_output_enable(CLA2);

/* configure the CLA3 */
/* select CLA5_ASYNC_OUT as input of MUX0 */
cla_multiplexer_input_config(CLA3,MUX0,CLA3MUX0_CLA5_ASYNC_OUT);
/* select CLA4_ASYNC_OUT as input of MUX1 */
cla_multiplexer_input_config(CLA3,MUX1,CLA3MUX1_CLA6_ASYNC_OUT);
/* IN0|IN3 */
cla_lut_control_config(CLA3,0x88);
/* select LUT result as CLA output */
cla_output_config(CLA3,LUT_RESULT);
cla_output_enable(CLA3);

/* configure the CLA4 */
/* select CLA5_ASYNC_OUT as input of MUX0 */
cla_multiplexer_input_config(CLA4,MUX0,CLA4MUX0_CLA5_ASYNC_OUT);
/* select CLA2_ASYNC_OUT as input of MUX1 */
cla_multiplexer_input_config(CLA4,MUX1,CLA4MUX1_CLA2_ASYNC_OUT);
/* IN0 */
cla_lut_control_config(CLA4,0xCC);
/* select flip-flop result as CLA output */
cla_output_config(CLA4,FLIP_FLOP_OUTPUT);
cla_flip_flop_clocksource_config(CLA4,MUX0_OUTPUT);
cla_flip_flop_clockpolarity_config(CLA4,CLA_CLOCKPOLARITY_POSEDGE);
cla_flip_flop_output_reset(CLA4);
cla_output_enable(CLA4);

/* configure the CLA5 */
/* select TIMER2_TRGO as input of MUX0 */
cla_multiplexer_input_config(CLA5,MUX0,CLA5MUX0_TIMER2_TRGO);
/* IN0 */
cla_lut_control_config(CLA5,0xF0);
/* select LUT result as CLA output */
cla_output_config(CLA5,LUT_RESULT);

/* configure the CLA6 */
/* select CLA5_ASYNC_OUT as input of MUX0 */
cla_multiplexer_input_config(CLA6,MUX0,CLA6MUX0_CLA5_ASYNC_OUT);
/* select CLA2_ASYNC_OUT as input of MUX1 */
```

```
    cla_multiplexer_input_config(CLA6,MUX1,CLA6MUX1_CLA4_ASYNC_OUT);
    /* IN0&IN1 */
    cla_lut_control_config(CLA6,0xCC);
    /* select flip-flop result as CLA output */
    cla_output_config(CLA6,FLIP_FLOP_OUTPUT);
    cla_flip_flop_clocksource_config(CLA6,MUX0_OUTPUT);
    cla_flip_flop_clockpolarity_config(CLA6,CLA_CLOCKPOLARITY_POSEDGE);
   cla_flip_flop_output_reset(CLA6);

    /* enable CLA0~CL6 */
    cla_enable(CLA5);
    cla_enable(CLA0);
    cla_enable(CLA1);
    cla_enable(CLA2);
    cla_enable(CLA4);
    cla_enable(CLA3);
    cla_enable(CLA6);
}
```

TIMER2_TRGO 信号作为 CLA5_ASYNC_OUT 的信号源，通过修改产生 TRGO 信号的时间间隔，来调整方波滤波器的性能。具体的 TIMER2 配置如*表 2-2. GD 工程下配置 TIMER2_TRGO 信号*。

表 2-2. GD 工程下配置 TIMER2_TRGO 信号

```
void timer_trgo_config(void)
{
    /* ----------------------------------------------------------------------
    TIMER Configuration:
    TIMERxCLK = SystemCoreClock/100 = 1MHz.
    TIMER configuration is timing mode, and the timing is 12 microsecond (12/1000000 = 12us).
    ---------------------------------------------------------------------- */
    timer_oc_parameter_struct timer_ocinitpara;
    timer_parameter_struct timer_initpara;
    /* enable the peripherals clock */
    rcu_periph_clock_enable(RCU_TIMER2);
    /* deinit a TIMER2 */
    timer_deinit(TIMER2);
    /* initialize TIMER init parameter struct */
    timer_struct_para_init(&timer_initpara);
    /* TIMER2 configuration */
    timer_initpara.prescaler         = 99;
    timer_initpara.alignedmode       = TIMER_COUNTER_EDGE;
    timer_initpara.counterdirection  = TIMER_COUNTER_UP;
    timer_initpara.period            = 12;
```

```
    timer_initpara.clockdivision        = TIMER_CKDIV_DIV1;
    timer_init(TIMER2, &timer_initpara);
    timer_master_output_trigger_source_select(TIMER2, TIMER_TRI_OUT_SRC_UPDATE);
    /* enable a TIMER */
    timer_enable(TIMER2);
}
```

3. 方波滤波器性能测试

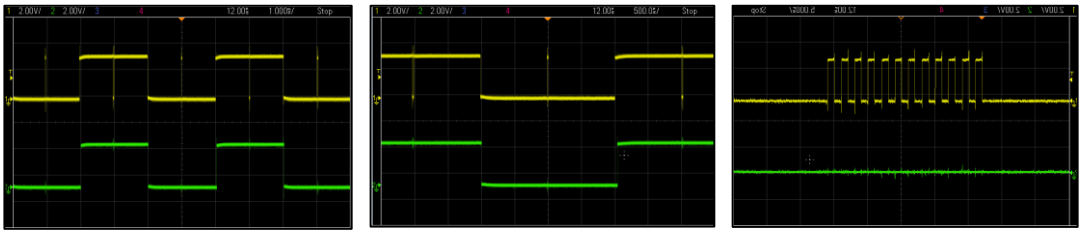
3.1. 测试条件

测试的输入信号是占空比为 50%，周期为 4ms 的方波，每半个周期有 12 个连续的宽度为 1us 的毛刺。通过配置 TIMER2，将 TIMER2_TRGO 信号的时间间隔设置为 12us，CLA 滤波器能使实现滤掉最大脉宽的毛刺为 12us。

3.2. 测试结果

方波滤波器的性能测试的结果如*图 3-1. CLA 方波滤波器性能测试*所示，图中黄色信号代表原始信号，绿色信号代表滤波后的信号。图片从左至右依次为放大的信号细节。测试表明，CLA 滤波器已将 12 个连续宽度为 1us 的毛刺滤掉。

图 3-1. CLA 方波滤波器性能测试



3.3. CLA 方波滤波器注意事项

在使用 CLA 方波滤波器时，需要注意以下几点。

1. CLA 方波滤波器的 TIMER2_TRGO 触发间隔与信号的毛刺宽度之间的关系，TIMER2_TRGO 信号确定了 CLA 滤波器可滤掉的单个最大毛刺宽度。若方波信号的毛刺是连续的，TIMER2_TRGO 时间间隔需要大于成连续出现的毛刺信号宽度的和。若不能将毛刺完全滤掉，可尝试将 TIMER2_TRGO 时间间隔加大。
2. CLA 方波滤波对毛刺信号不连续的方波信号，将 TIMER2_TRGO 配置为毛刺中最大的时间宽度。
3. 可在 CLA6 与 CLA3 之间增加 CLA7,功能配置与 CLA6 相同，可提升 CLA 方波滤波器在毛刺宽度较大时的滤波性能。

4. 版本历史

表 4-1. 版本历史

版本号.	说明	日期
1.0	首次发布	2021 年 12 月 13 日

Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the “Product”), is owned by the Company under the intellectual property laws and treaties of the People’s Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do release the Company as well as it’s suppliers and/or distributors from any claim, damage, or other liability arising from or related to all Unintended Uses of the Products. Customers shall indemnify and hold the Company as well as it’s suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Products.

Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and Products and services described herein at any time, without notice.