AN0031
Application Note

AT32 PWM input test

Introduction

This application describes how to implement PWM input test using AT32 timers.

Note: The code in this application note is based on Artery's V2.x.x BSP (board support package). When in use, attention should be paid to the differences regarding to the versions of BSP.

Applicable products:

MCU	AT32F series AT32L series

Contents

List of tables

List of figures

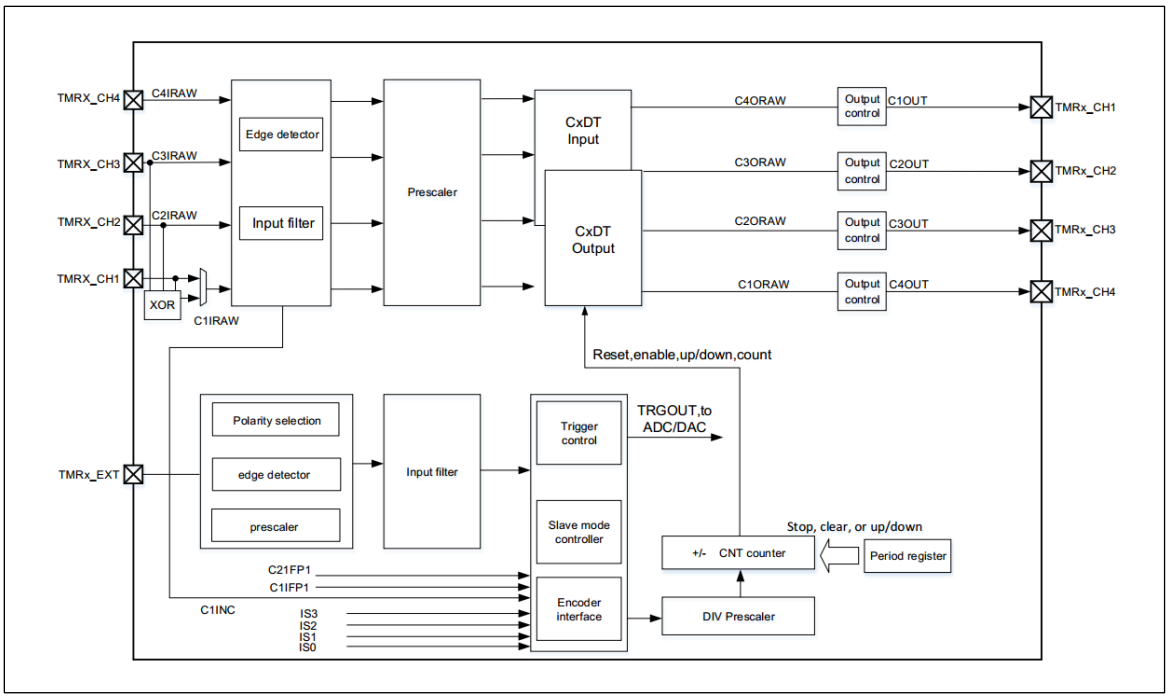
1 AT32 timers

1.1 Timer introduction

The timers consist of a 16-bit auto-reload counter (except for TM2 and TMR5 with 32-bit auto-reload counters) driven by a programmable prescaler. They can be used for various purposes, including measuring the pulse width of input signals (input capture) or generating output waveforms (output compare, PWM, dead-time complementary PWM)

Pulse width and waveform periods can be modulated from a few microseconds to several milliseconds using the timer prescaler and the RCC clock control prescaler.

Figure 1. Block diagram of general-purpose timers



The timers consist of four parts (Figure 1), including the Clock Unit that provides clock driver for timers, the Time-base Unit with counter function, the Input Capture that allows input signals to enter timer mode, and the Output Compare featuring PWM output of integrated timers.

1.2 Timer configuration

1) **Clock enable**

crm_periph_clock_enable(CRM_TMR2_PERIPH_CLOCK, TRUE);;

2) **Initialize timer parameters, set auto-reload value, division factors and count mode**

In the library function, the timer parameters are initialized by the initialization function tmr_base_init () and tmr_cnt_dir_set():

void tmr_base_init(tmr_type* tmr_x, uint32_t tmr_pr, uint32_t tmr_div):

Where, the first parameter indicates the specific timer, the second parameter (tmr_pr) periodic counter value, and the third parameter (tmr_div) the frequency division factor.

void tmr_cnt_dir_set(tmr_type *tmr_x, tmr_count_mode_type tmr_cnt_dir):
Where, the first parameter indicates the specific timer, the second parameter (tmr_cnt_dir) the specific counting mode (such as, up, down, or up/down counting).

In particularly, the Plus mode is a unique feature of TMR2 and TMR5.

The tmr_32_bit_function_enable() function is used to enable the Plus mode.

When the Plus mode is enabled, the TMRx_CVAL,TMRx_PR and TMRx_CxDT are expanded from 16 bits to 32 bits.

void tmr_32_bit_function_enable(tmr_type *tmr_x, confirm_state new_state):

The tmr_clock_source_div_set() function is used to configure timer frequency division parameters (note that it is different from TMR_DIV, which is used to configure filtering and dead time);

The tmr_repetition_counter_set() function is used to configure repetition period register (for advanced timers TMR1/TMR8/TMR15 only). Such both functions are not used in this example, with only brief introduction about them here.

void tmr_clock_source_div_set(tmr_type *tmr_x, tmr_clock_division_type tmr_clock_div);

void tmr_repetition_counter_set(tmr_type *tmr_x, uint8_t tmr_rpr_value);

3) **Set TMRx_IDEN to enable interrupt update**

void tmr_interrupt_enable(tmr_type *tmr_x, uint32_t tmr_interrupt, confirm_state new_state);

The parameter tmr_interrupt indicates the type of interrupts that are enabled, including interrupt update, trigger interrupt and input capture interrupt.

4) **Set TMRx interrupt priority**

This is done by calling the nvic_irq_enable().

5) **Enable TMRx.**

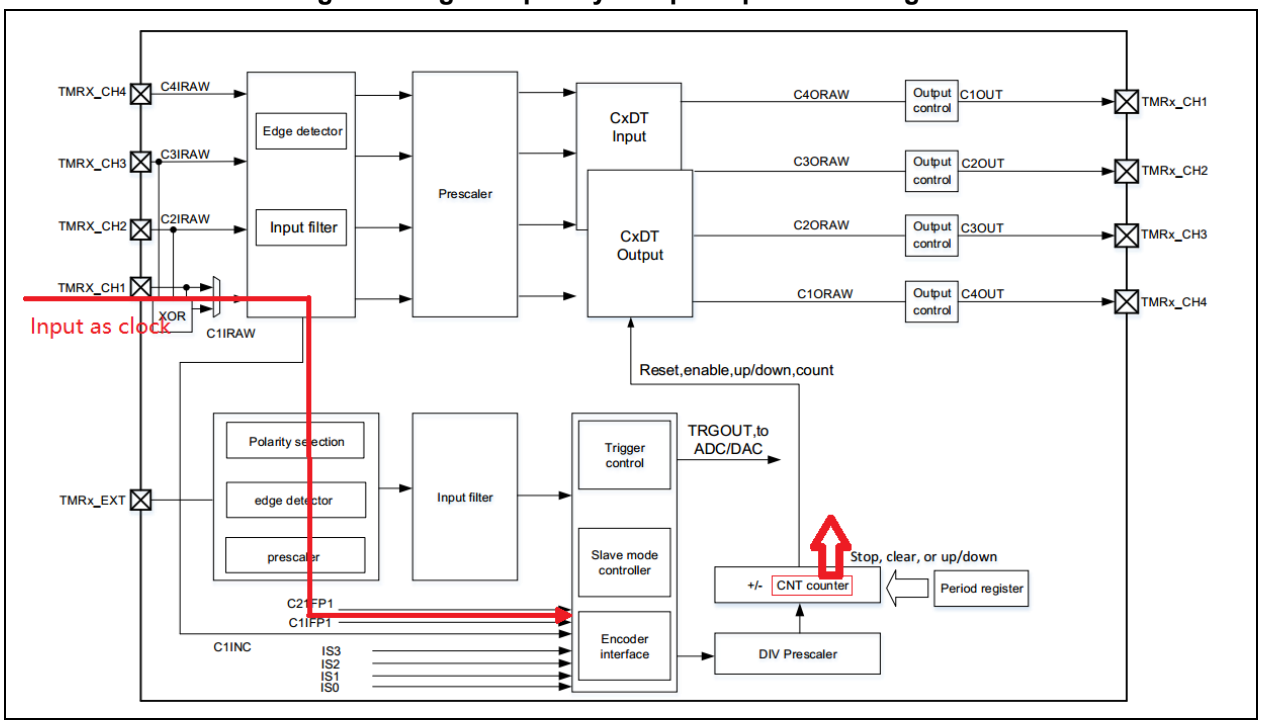
void tmr_counter_enable(tmr_type *tmr_x, confirm_state new_state);

6) **Program interrupt service functions.**

2 Specification introduction

2.1 AT32 PWM high-frequency test

Figure 2. High-frequency test principle block diagram



When testing high-frequency signals, the high-frequency signal input is used as the clock source of timer TMR2 (shown in Figure 2) to enable TMR2 counter to count while using another timer as a clock benchmark , the change value of TMR2 counter is captured base on certain intervals, such as 1s, which is the frequency value of the high-frequency signals.

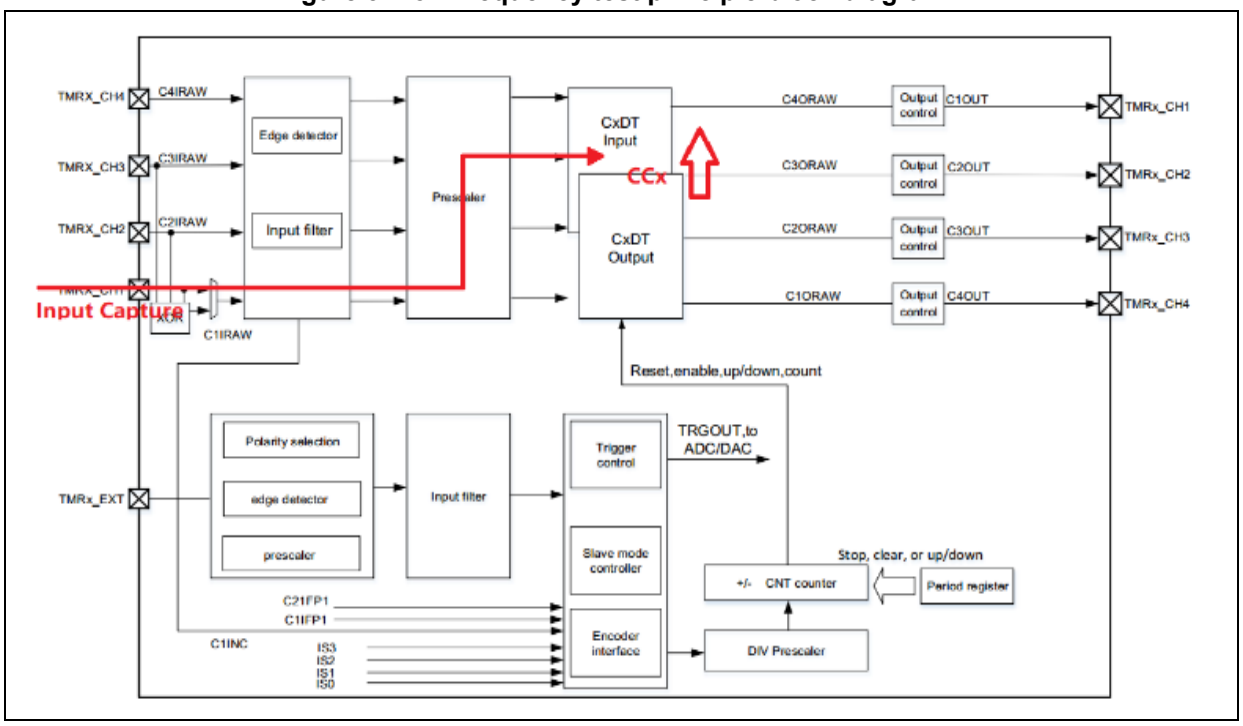
Two timers are used, one of which is TMR2 (because the PMEN bit in the TMR2_CTRL1 can be set to enable TMRx plus mode so that TMRx_CNT, TMRx_PR and TMRx_CxDT can be extended from 16 bit to 32 bit) that is helpful for the counter to count without overflow generation when testing high frequency signals.

The advantage of this operation is that it not only can test the high-frequency signals up to 50 MHz, without frequent interrupt generation, but also has redundant codes to handle customer tasks. The frequencies tested in this method range from 500 MHz to 1 Hz (TMR2 operating frequency is 240 MHz).

Note: Plus mode is a unique feature of TMR2 and TMR5. The use of other TMR without plus mode, or AT32 MCU without plus mode TMR will cause the testing frequency to be limited.

2.2 AT32 PWM low-frequency test

Figure 3. Low-frequency test principle block diagram



When testing low-frequency signals, the low-frequency signal input is used as the input capture of TMR2 (shown in Figure3) to trigger the input capture interrupt of TMR2. The counter change value between two input captures divided by the working clock of TMR2 is equal to the frequency value of the low-frequency signals.

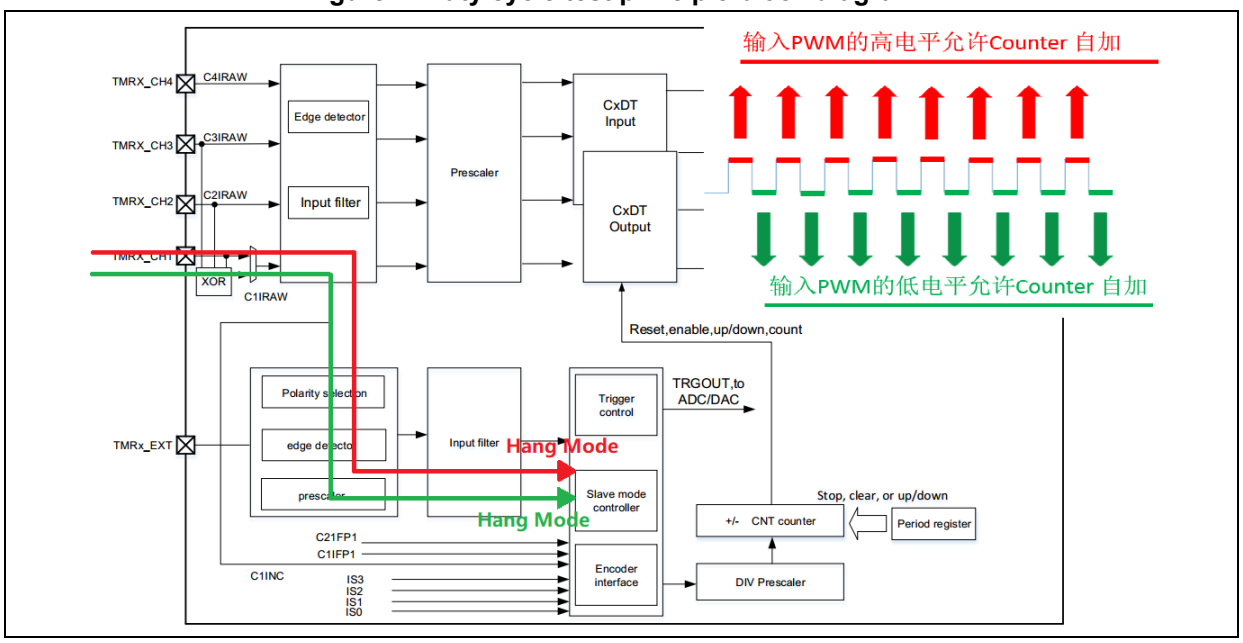
The reason of selection of TMR2 is that it can enable TMRx plus mode by setting the PMEN bit in the TMR2_CTRL1 to extend TMRx_CNT, TMRx_AR and TMRx_CxDT from 16 bits to 32 bits, making it convenient to conduct low frequency testing.

The minimum frequency tested in this method is 56 MHz (TMR2 operating frequency is 240 MHz).

Note: Plus mode is a unique feature of TMR2 and TMR5. The use of other TMR without plus mode, or AT32 MCU without plus mode TMR will cause the testing frequency to be limited.

2.3 AT32 PWM duty cycle test principle

Figure 4. Duty cycle test principle block diagram



PWM duty cycle test is measured in gated mode (shown in Figure 4). The input signals are used as the input signals of two timers simultaneously to control the counter of timers. While one counts on the high level stage of the input signals, the other counts on the low level stage, with the third timer as a time benchmark, capture the counter values of the two timers during certain interrupt, for example, 1s, and the ratio of the two values refers to the latest PWM duty cycle.

The reason of selection of TMR2 and TMR5 is that they can enable TMRx plus mode by setting the PMEN bit in the TMRx_CTRL1 to extend TMRx_CNT, TMRx_AR and TMRx_CxDT from 16 bits to 32 bits to facilitate test.

The advantage of this method is being that they can measure the duty cycle of higher frequencies, such as, up to 10 MHz, within 1% error. Without frequent generation of interrupts, it also holds abundant codes to handle customer tasks.

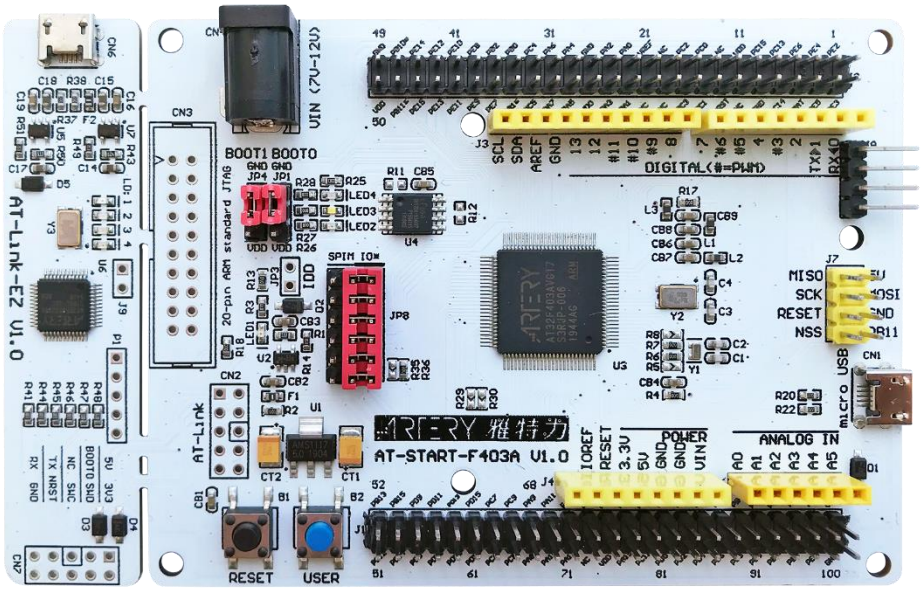
Note: Plus mode is a unique feature of TMR2 and TMR5. The use of other TMR without plus mode, or AT32 MCU without plus mode TMR will cause the testing frequency to be limited.

3 PWM test quick start

3.1 Hardware resources

1) AT-START-F403A evaluation board

Figure 5. AT-START-F403 V1.1 board



Note: This demo is based on the hardware conditions of AT32F403. If the user wants to apply it to other AT32 parts, please make the corresponding modifications.

3.2 PWM input test demo

- Open *PWM Input Test project* source code, there are three macros in the at32f403a_407_clock.h:

```
#define high_frequency_test
#define low_frequency_test
#define duty_ration_test
```

They are used to test high-frequency, low-frequency signals and PWM duty cycle, respectively. Open the macro to be tested (Notice: only open one macro at a time)

Open the *PWM Output* source code, there are three macros in the at32f403a_407_clock.h:

```
#define Output_High_Frequency
#define Output_Low_Frequency
#define Output_PWM_Duty_Ration_10
```

They are used to test high-frequency, low-frequency signals and PWM duty cycle respectively.

The AT-LINK-EZ onboard AT-START has serial interface output feature so that it can output the PA9 of the USART1_TX port to PC. Besides, it is possible to use other serial interface tool for test result output.

- **For high-frequency signal test:**

1. Open *PWM Output* source code macro definition: #define Output_High_Frequency.

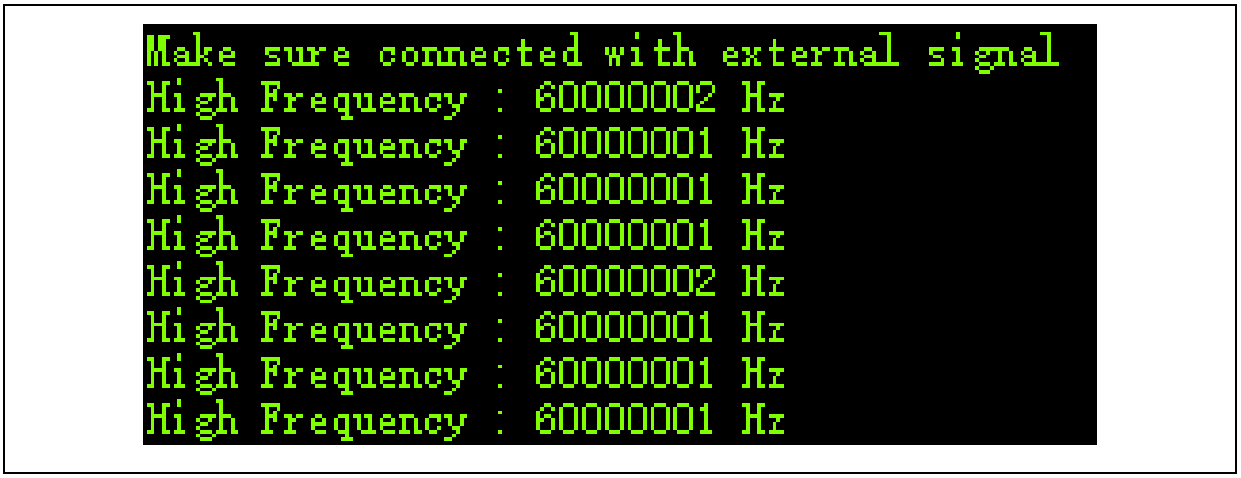
PA8 generates 30 MHz PWM (I/O is operating at high frequency, so the max frequency can be reduced accordingly). Compile and download to the test board 1.

2. Open *PWM Input Test* source code macro definition: #define High_Frequency_Test, compile and download to the test board 2.

3. Connect the PA8 of test board 1 to the PA0 of test board 2, and USART1 outputs the current PWM frequency information via PA9.

Serial port print information by is as follows:

Figure 6. Serial port print information for high-frequency signal test



- **For low-frequency signal test:**

1. Open *PWM Output* source code macro definition: #define Output_Low_Frequency, PA8 generates 500 MHz PWM, compile and download to the test board 1.

2. Open *PWM Input Test* source code macro definition: #define Low_Frequency_Test, compile and download to the test board 2.

3. Connect PA8 of the test board 1 to the PA0 of test board 2, and USART1 outputs the current PWM frequency information via PA9.

Serial port print information: (disregard the first data)

Figure 7. Serial port print information of low-frequency signal test

- **For PWM duty cycle test:**

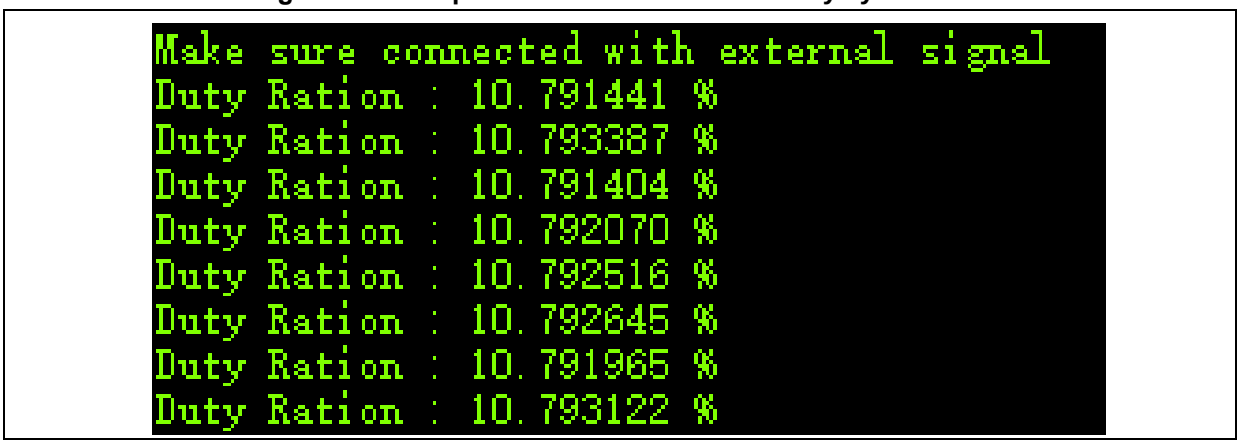
  1. Open *PWM Output* source code macro definition: #define Output_PWM_Duty_Ration_10, PA8 generates 6 MHz PWM, the duty cycle is 10%, compile and download to the test board 1.

  2. Open *PWM Input Test* source code macro definition: #define Duty_Ration_Test. Compile and download to the test board 2.

  3. Connect the PA8 in test board 1 to the PA0 of test board 2, and USART1 outputs the current PWM duty cycle information through PA9.

  Serial port information as follows:

Figure 8. Serial port information for PWM duty cycle test



4 Revision history

Table 1. Document revision history

Date	Revision	Changes
2021.12.30	2.0.0	Initial release

IMPORTANT NOTICE – PLEASE READ CAREFULLY

Purchasers are solely responsible for the selection and use of ARTERY's products and services; ARTERY assumes no liability for purchasers' selection or use of the products and the relevant services.

No license, express or implied, to any intellectual property right is granted by ARTERY herein regardless of the existence of any previous representation in any forms. If any part of this document involves third party's products or services, it does NOT imply that ARTERY authorizes the use of the third party's products or services, or permits any of the intellectual property, or guarantees any uses of the third party's products or services or intellectual property in any way.

Except as provided in ARTERY's terms and conditions of sale for such products, ARTERY disclaims any express or implied warranty, relating to use and/or sale of the products, including but not restricted to liability or warranties relating to merchantability, fitness for a particular purpose (based on the corresponding legal situation in any unjudicial districts), or infringement of any patent, copyright, or other intellectual property right.

ARTERY's products are not designed for the following purposes, and thus not intended for the following uses: (A) Applications that have specific requirements on safety, for example: life-support applications, active implant devices, or systems that have specific requirements on product function safety; (B) Aviation applications; (C) Aerospace applications or environment; (D) Weapons, and/or (E) Other applications that may cause injuries, deaths or property damages. Since ARTERY products are not intended for the above-mentioned purposes, if purchasers apply ARTERY products to these purposes, purchasers are solely responsible for any consequences or risks caused, even if any written notice is sent to ARTERY by purchasers; in addition, purchasers are solely responsible for the compliance with all statutory and regulatory requirements regarding these uses.

Any inconsistency of the sold ARTERY products with the statement and/or technical features specification described in this document will immediately cause the invalidity of any warranty granted by ARTERY products or services stated in this document by ARTERY, and ARTERY disclaims any responsibility in any form.