

云途芯片 SDK ETMR PWM 输出 0% 与 100% 占空比实现说明

版本

Config Tool Version: 3.0.0

SDK version: 1.4.1 & 1.4.0

1. 前言



为了说明云途芯片 SDK 中 ETMR PWM 在 0% 和 100% 占空比场景下的实现差异，本文基于 Config Tool 3.0.0 和 SDK 1.4.1 & SDK 1.4.0，对相关 SDK 接口、寄存器配置方式及各芯片系列支持情况进行整理。PWM 输出 0% 和 100% 属于边界占空比场景，不同芯片系列在 ETMR 模块设计和 SDK 封装上存在差异，可能导致同一接口在不同芯片上的支持能力不同。本文用于指导开发人员正确选择实现方式，避免在配置、调试和平台迁移过程中出现 PWM 输出异常。

2. 测试验证

2.1 测试条件

- SDK 版本: 1.4.1 & 1.4.0
- 对齐方式: 中心对齐
- 测试目的: PWM 0%/100% 占空比输出

2.2 测试步骤

1. 首先使用 `eTMR_DRV_UpdatePwmChannel` 设置 0% 和 100% 占空比。
2. 输出 0%，持续 1 秒。
3. 输出 100%，持续 1 秒。
4. 循环执行并观察波形。
5. 如果 SDK 接口无法正常实现 100%，则改用 `MOD+1` 实现 100%，0% 继续使用 SDK 接口。
6. 单独使用 CHMASK 强制输出高、低电平，并按相同时序循环。

2.3 测试结果

2.3.1 LE04/LE05:

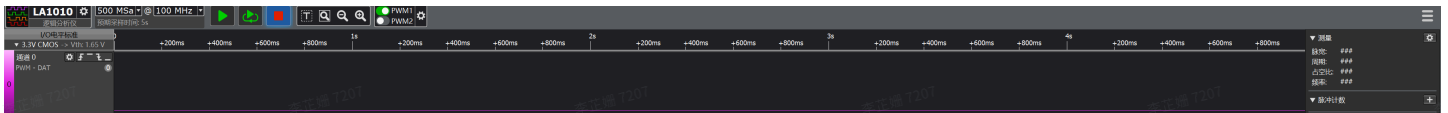
SDK1_4_0:

- eTMR_DRV_UpdatePwmChannel:

```
#if 1
    eTMR_DRV_ClearLdok(instance: PWM2_INST);
    eTMR_DRV_UpdatePwmChannel(instance: 2, channel: 0, dutyCycle: 0x0, offset: 0);
    eTMR_DRV_UpdatePwmChannel(instance: 2, channel: 1, dutyCycle: 0x0, offset: 0);
    eTMR_DRV_SetLdok(instance: PWM2_INST);
    OSIF_TimeDelay(delay: 1000);

    eTMR_DRV_ClearLdok(instance: PWM2_INST);
    eTMR_DRV_UpdatePwmChannel(instance: 2, channel: 0, dutyCycle: 0x8000, offset: 0);
    eTMR_DRV_UpdatePwmChannel(instance: 2, channel: 1, dutyCycle: 0x8000, offset: 0);
    eTMR_DRV_SetLdok(instance: PWM2_INST);
    OSIF_TimeDelay(delay: 1000);
#elif
```

现象：100%占空比无法正常输出

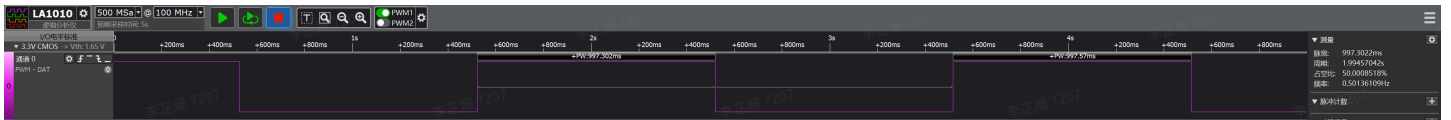


- 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口:

```
eTMR_DRV_ClearLdok(instance: PWM2_INST);
eTMR_DRV_UpdatePwmChannel(instance: 2, channel: 0, dutyCycle: 0x0, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 2, channel: 1, dutyCycle: 0x0, offset: 0);
eTMR_DRV_SetLdok(instance: PWM2_INST);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: PWM2_INST);
g_etmrBase[2]->CH[0].VAL0 = 0;
g_etmrBase[2]->CH[0].VAL1 = g_etmrBase[2]->MOD + 1;
g_etmrBase[2]->CH[1].VAL0 = 0;
g_etmrBase[2]->CH[1].VAL1 = g_etmrBase[2]->MOD + 1;
// eTMR_DRV_UpdatePwmChannel(2, 0, 0x8000, 0);
// eTMR_DRV_UpdatePwmChannel(2, 1, 0x8000, 0);
eTMR_DRV_SetLdok(instance: PWM2_INST);
OSIF_TimeDelay(delay: 1000);
```

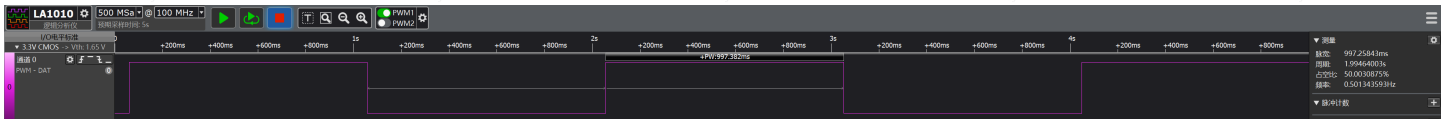
现象：输出符合预期，但是第一个周期无法100%，后续周期可以实现；不建议使用；



- 使用 CHMASK 强制输出高、低电平。

```
g_etmrBase[2]->CHMASK = (1 << eTMR_CHMASK_CH0M_SHIFT) | (1 << eTMR_CHMASK_CH1M_SHIFT) |  
                        (0 << eTMR_CHMASK_CH0MV_SHIFT) | (0 << eTMR_CHMASK_CH1MV_SHIFT);  
eTMR_DRV_SetLdok(instance: PWM2_INST);  
OSIF_TimeDelay(delay: 1000);  
  
eTMR_DRV_ClearLdok(instance: PWM2_INST);  
g_etmrBase[2]->CHMASK = (1 << eTMR_CHMASK_CH0M_SHIFT) | (1 << eTMR_CHMASK_CH1M_SHIFT) |  
                        (1 << eTMR_CHMASK_CH0MV_SHIFT) | (1 << eTMR_CHMASK_CH1MV_SHIFT);  
eTMR_DRV_SetLdok(instance: PWM2_INST);  
OSIF_TimeDelay(delay: 1000);
```

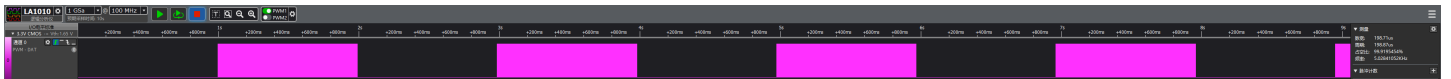
现象：输出符合预期，**建议使用**；



SDK1_4_1:

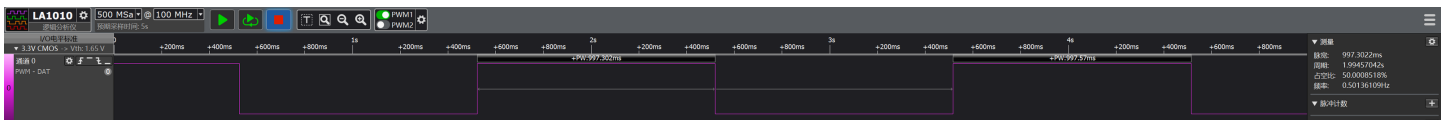
- eTMR_DRV_UpdatePwmChannel :

现象：占空比设置为 100% 时，实际输出约为 99.9%，无法达到 100%。



- 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口:

现象：输出符合预期



小结:

SDK1_4_0:

- SDK 接口可实现 0% 占空比，但无法正常实现 100% 占空比；
- 100% 可通过 MOD+1 实现，**但是第一个周期无法100%，后续周期可以实现；不建议使用**；
- CHMASK 也可稳定实现 0% 和 100%。

SDK1_4_1:

- SDK 接口可实现 0% 占空比，但无法正常实现 100% 占空比；
- 100% 可通过 MOD+1 实现，**但是第一个周期无法100%，后续周期可以实现；不建议使用**；
- CHMASK 也可稳定实现 0% 和 100%。

2.3.2 LE14/LE15:

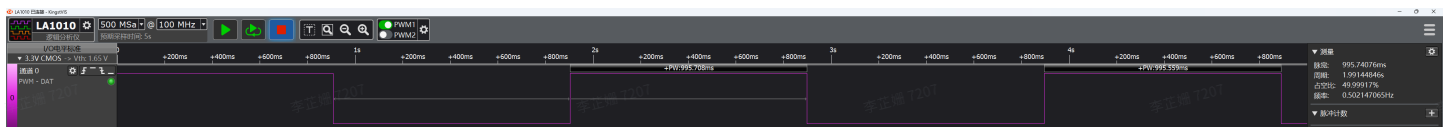
SDK1_4_0:

- eTMR_DRV_UpdatePwmChannel:

```
eTMR_DRV_ClearLdok(instance: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 0, dutyCycle: 0x0000, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 1, dutyCycle: 0x0000, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
// g_etmrBase[2]->CH[0].VAL0 = 0;
// g_etmrBase[2]->CH[0].VAL1 = g_etmrBase[2]->MOD + 1;
// g_etmrBase[2]->CH[1].VAL0 = 0;
// g_etmrBase[2]->CH[1].VAL1 = g_etmrBase[2]->MOD + 1;
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 0, dutyCycle: 0x8000, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 1, dutyCycle: 0x8000, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

现象：输出符合预期

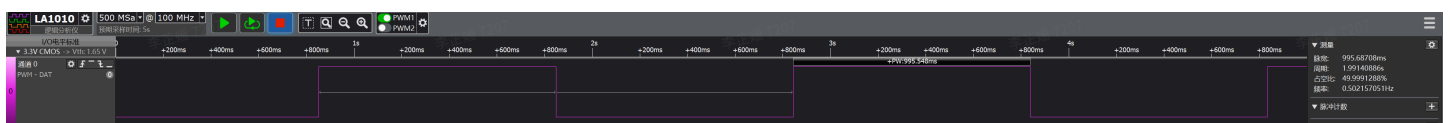


- 使用 CHMASK 强制输出高、低电平。

```
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH0M_SHIFT) | (1 << eTMR_CHMASK_CH1M_SHIFT) |
                        (0 << eTMR_CHMASK_CH0MV_SHIFT) | (0 << eTMR_CHMASK_CH1MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH0M_SHIFT) | (1 << eTMR_CHMASK_CH1M_SHIFT) |
                        (1 << eTMR_CHMASK_CH0MV_SHIFT) | (1 << eTMR_CHMASK_CH1MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

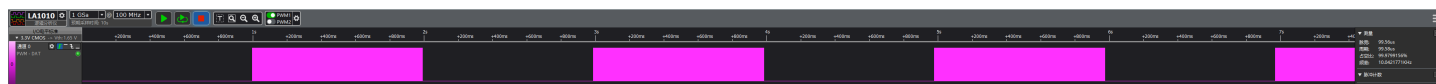
现象：输出符合预期



SDK1_4_1:

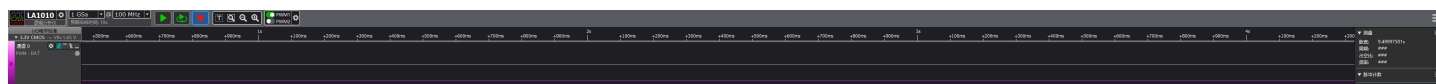
- `eTMR_DRV_UpdatePwmChannel` :

现象：占空比设置为 100% 时，实际输出约为 99.9%，无法达到 100%。



- 使用 `MOD+1` 实现 100%，0% 继续使用 SDK 接口:

现象：100%占空比无法正常输出



小结:

SDK1_4_0:

1. SDK 接口可实现0%占空和100%占空比;
2. `CHMASK` 可稳定实现 0% 和 100%。

SDK1_4_1:

1. SDK 接口可实现 0% 占空比，但无法正常实现 100% 占空比;
2. `MOD + 1` 无法输出 100%占空比
3. `CHMASK` 可稳定实现 0% 和 100%。

2.3.3 MC03:

SDK1_4_0:

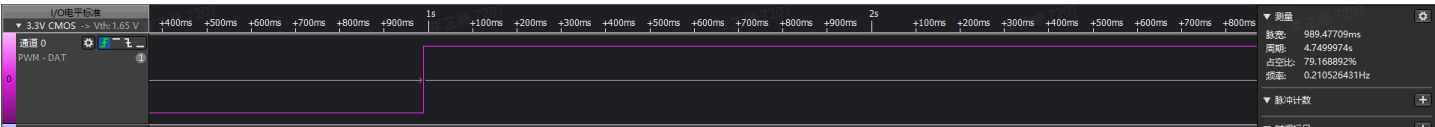
- `eTMR_DRV_UpdatePwmChannel` :

```
eTMR_DRV_ClearLdok(instance: 0);

eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0x0, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0x0, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
// g_etmrBase[2]->CH[0].VAL0 = 0;
// g_etmrBase[2]->CH[0].VAL1 = g_etmrBase[2]->MOD + 1;
// g_etmrBase[2]->CH[1].VAL0 = 0;
// g_etmrBase[2]->CH[1].VAL1 = g_etmrBase[2]->MOD + 1;
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0x8000, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0x8000, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

现象：0%占空比第一次输出正常，后续就无法正常输出



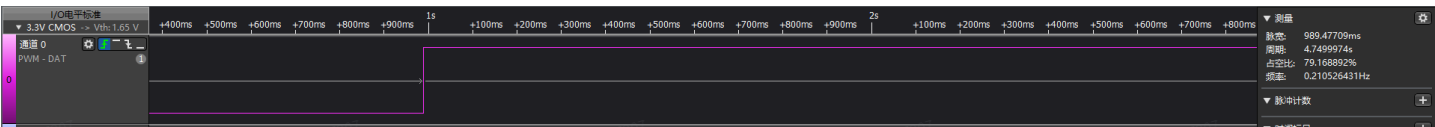
使用 MOD+1 实现 100%，0% 继续使用 SDK 接口：

```
eTMR_DRV_ClearLdok(instance: 0);

eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0x0, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0x0, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CH[2].VAL0 = 0;
g_etmrBase[0]->CH[2].VAL1 = g_etmrBase[0]->MOD + 1;
g_etmrBase[0]->CH[3].VAL0 = 0;
g_etmrBase[0]->CH[3].VAL1 = g_etmrBase[0]->MOD + 1;
// eTMR_DRV_UpdatePwmChannel(0, 2, 0x8000, 0);
// eTMR_DRV_UpdatePwmChannel(0, 3, 0x8000, 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

现象：0%占空比第一次输出正常，后续就无法正常输出

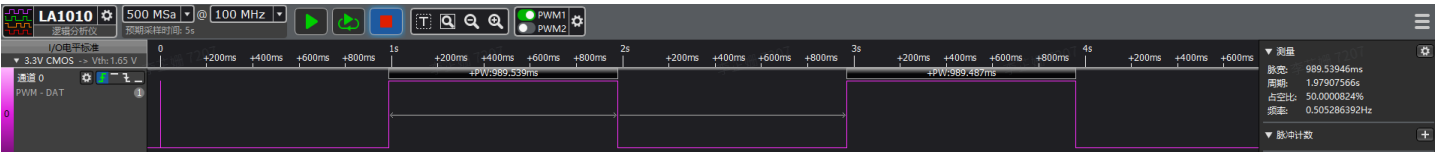


- 使用 CHMASK 强制输出高、低电平。

```
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
                        (0 << eTMR_CHMASK_CH2MV_SHIFT) | (0 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
                        (1 << eTMR_CHMASK_CH2MV_SHIFT) | (1 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

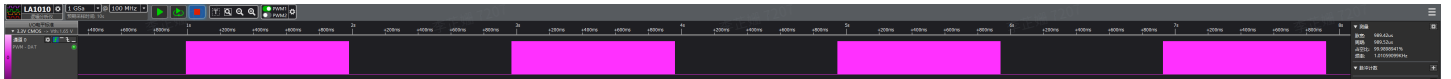
现象：输出符合预期



SDK1_4_1:

- eTMR_DRV_UpdatePwmChannel :

现象：占空比设置为 100% 时，实际输出约为 99.9%，无法达到 100%。



- 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口：
- 现象：输出符合预期



小结：

SDK1_4_0:

1. SDK 接口首次可实现 0% 占空比，但循环切换时无法输出 0% 占空比；
2. 采用 MOD+1 可实现 100% 占空比，但循环切换时无法输出 0% 占空比；
3. CHMASK 也可稳定实现 0% 和 100%。

SDK1_4_1:

1. SDK 接口可实现 0% 占空比，但无法正常实现 100% 占空比；
2. 100% 可通过 MOD+1 实现，但是第一个周期无法100%，后续周期可以实现；不建议使用；
3. CHMASK 也可稳定实现 0% 和 100%。

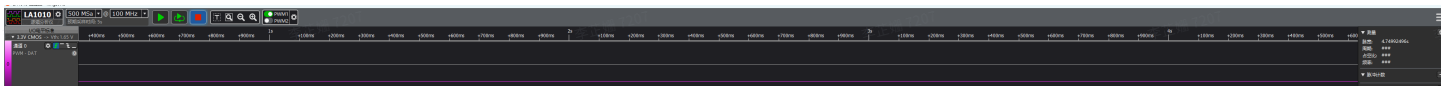
2.3.4 MD14:

SDK1_4_0:

- eTMR_DRV_UpdatePwmChannel

```
#if 1
    eTMR_DRV_ClearLdok(instance: 0);
    eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0000, offset: 0);
    eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0000, offset: 0);
    eTMR_DRV_SetLdok(instance: 0);
    OSIF_TimeDelay(delay: 1000);
    // g_etmrBase[0]->CH[2].VAL0 = 0;
    // g_etmrBase[0]->CH[2].VAL1 = g_etmrBase[0]->MOD + 1;
    // g_etmrBase[0]->CH[3].VAL0 = 0;
    // g_etmrBase[0]->CH[3].VAL1 = g_etmrBase[0]->MOD + 1;
    eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0x8000, offset: 0);
    eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0x8000, offset: 0);
    eTMR_DRV_SetLdok(instance: 0);
    OSIF_TimeDelay(delay: 1000);
#else
```

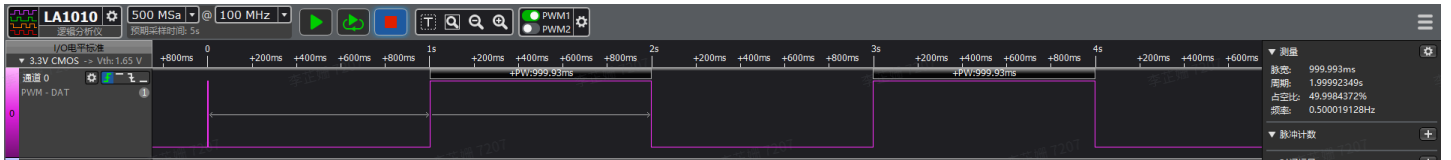
现象：100%占空比无法正常输出



- 使用 `MOD+1` 实现 100%，0% 继续使用 SDK 接口

```
eTMR_DRV_ClearLdok(instance: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0000, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0000, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
g_etmrBase[0]->CH[2].VAL0 = 0;
g_etmrBase[0]->CH[2].VAL1 = g_etmrBase[0]->MOD + 1;
g_etmrBase[0]->CH[3].VAL0 = 0;
g_etmrBase[0]->CH[3].VAL1 = g_etmrBase[0]->MOD + 1;
//eTMR_DRV_UpdatePwmChannel(0, 2, 0x8000, 0);
//eTMR_DRV_UpdatePwmChannel(0, 3, 0x8000, 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

现象：输出符合预期

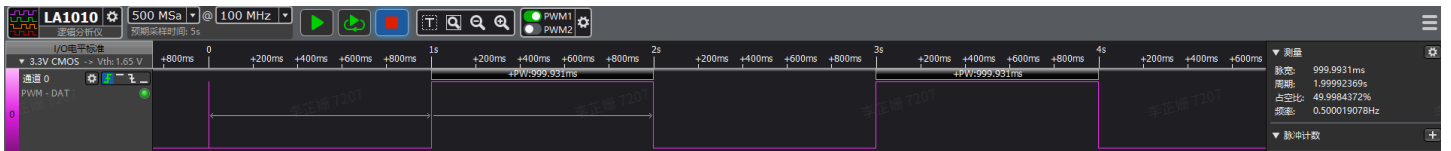


- 使用 CHMASK 强制输出高、低电平

```
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
                        (0 << eTMR_CHMASK_CH2MV_SHIFT) | (0 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
                        (1 << eTMR_CHMASK_CH2MV_SHIFT) | (1 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

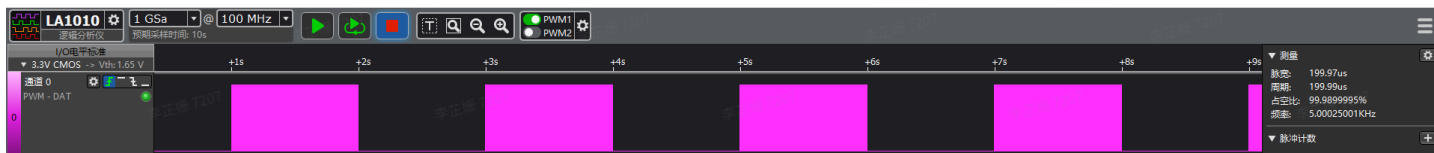
现象：输出符合预期



SDK1_4_1_HF001:

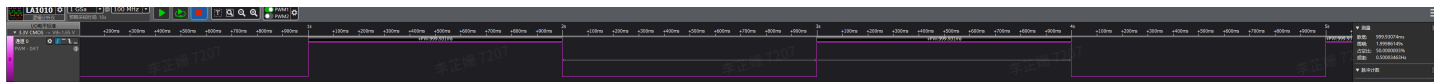
- `eTMR_DRV_UpdatePwmChannel`

现象：占空比设置为 100% 时，实际输出约为 99.9%，无法达到 100%。



- 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口

现象：输出符合预期



小结:

SDK1_4_0:

1. SDK 接口无法正常输出100%占空比
2. 100% 可通过 MOD+1 实现，但是第一个周期无法100%，后续周期可以实现；不建议使用；
3. CHMASK 也可稳定实现 0% 和 100%。

SDK1_4_1_HF001:

1. SDK 接口可实现 0% 占空比，但无法正常实现 100% 占空比；
2. 100% 可通过 MOD+1 实现，但是第一个周期无法100%，后续周期可以实现；不建议使用；
3. CHMASK 也可稳定实现 0% 和 100%。

2.3.5 MD24:

SDK1_4_0:

- eTMR_DRV_UpdatePwmChannel :

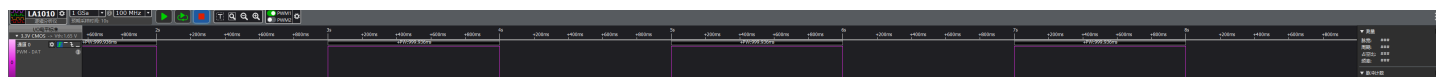
```

eTMR_DRV_ClearLdok(instance: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0x0, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0x0, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0x8000, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0x8000, offset: 0);
// g_etmrBase[0]->CH[2].VAL0 = 0;
// g_etmrBase[0]->CH[2].VAL1 = g_etmrBase[PWM_INST]->MOD + 1;
// g_etmrBase[0]->CH[3].VAL0 = 0;
// g_etmrBase[0]->CH[3].VAL1 = g_etmrBase[PWM_INST]->MOD + 1;
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

```

现象：输出符合预期



- 单独使用 CHMASK 强制输出高、低电平。

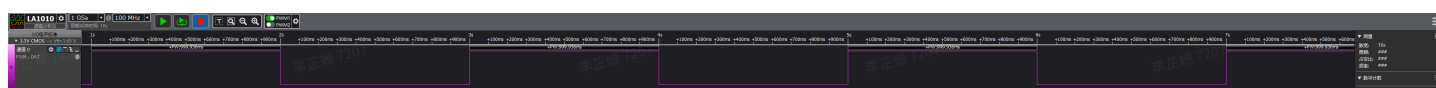
```

g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
(0 << eTMR_CHMASK_CH2MV_SHIFT) | (0 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
(1 << eTMR_CHMASK_CH2MV_SHIFT) | (1 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

```

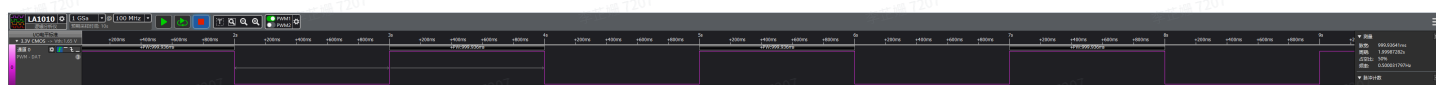
现象：输出符合预期



SDK1_4_1:

- eTMR_DRV_UpdatePwmChannel:

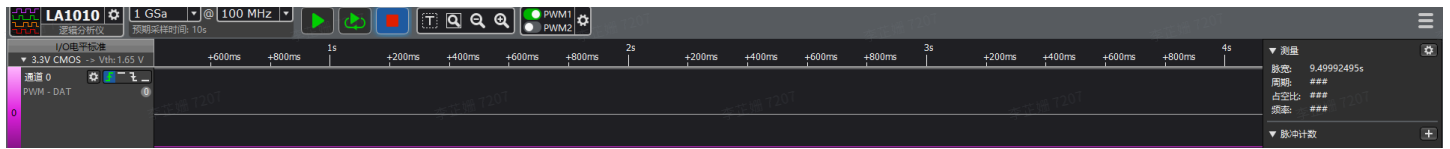
现象：输出符合预期



- 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口:

现象：100%占空比无法正常输出





小结:

SDK1_4_0:

1. SDK 接口可直接实现 0% 和 100% 占空比。
2. CHMASK 也可稳定实现 0% 和 100%。

SDK1_4_1:

1. SDK 接口可直接实现 0% 和 100% 占空比。
2. MOD + 1 无法输出 100%占空比
3. CHMASK 也可稳定实现 0% 和 100%。

2.3.6 ME05:

SDK1_4_0:

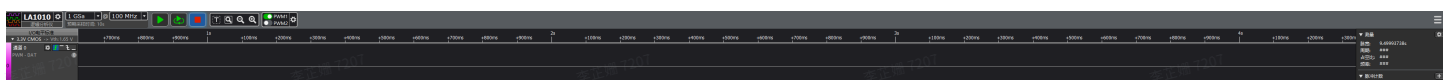
- eTMR_DRV_UpdatePwmChannel :

```
eTMR_DRV_ClearLdok(instance: 0);

eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 0, dutyCycle: 0x0, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 1, dutyCycle: 0x0, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
// g_etmrBase[0]->CH[0].VAL0 = 0;
// g_etmrBase[0]->CH[0].VAL1 = g_etmrBase[0]->MOD + 1;
// g_etmrBase[0]->CH[1].VAL0 = 0;
// g_etmrBase[0]->CH[1].VAL1 = g_etmrBase[0]->MOD + 1;
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 0, dutyCycle: 0x8000, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 1, dutyCycle: 0x8000, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

现象: 100%占空比无法正常输出



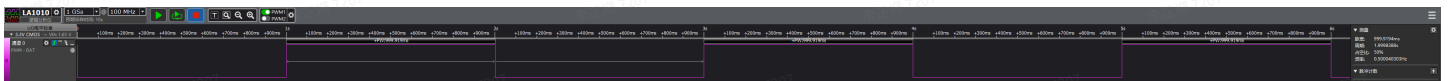
- 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口：

```
eTMR_DRV_ClearLdok(instance: 0);

eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 0, dutyCycle: 0x0, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 1, dutyCycle: 0x0, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CH[0].VAL0 = 0;
g_etmrBase[0]->CH[0].VAL1 = g_etmrBase[0]->MOD + 1;
g_etmrBase[0]->CH[1].VAL0 = 0;
g_etmrBase[0]->CH[1].VAL1 = g_etmrBase[0]->MOD + 1;
//eTMR_DRV_UpdatePwmChannel(0, 0, 0x8000, 0);
//eTMR_DRV_UpdatePwmChannel(0, 1, 0x8000, 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

现象：输出符合预期

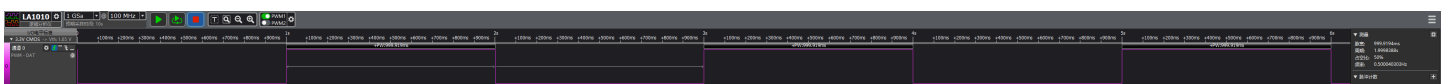


- 使用 CHMASK 强制输出高、低电平。

```
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH0M_SHIFT) | (1 << eTMR_CHMASK_CH1M_SHIFT) |
(0 << eTMR_CHMASK_CH0MV_SHIFT) | (0 << eTMR_CHMASK_CH1MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH0M_SHIFT) | (1 << eTMR_CHMASK_CH1M_SHIFT) |
(1 << eTMR_CHMASK_CH0MV_SHIFT) | (1 << eTMR_CHMASK_CH1MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

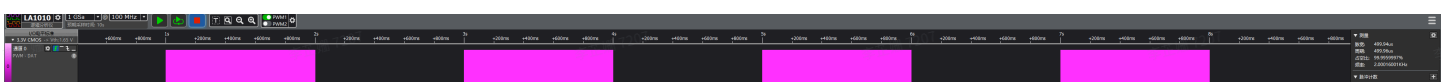
现象：输出符合预期



SDK1_4_1_HF001:

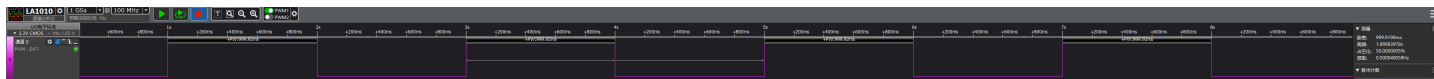
- eTMR_DRV_UpdatePwmChannel:

现象：占空比设置为 100% 时，实际输出约为 99.9%，无法达到 100%。



- 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口：

现象：输出符合预期



SDK1_4_0:

- SDK 接口无法正常输出100%占空比；
- 100% 可通过 MOD+1 实现，但是第一个周期无法100%，后续周期可以实现；不建议使用；
- CHMASK 也可稳定实现 0% 和 100%。

SDK1_4_1_HF001:

- SDK 接口无法正常输出100%占空比；
- 100% 可通过 MOD+1 实现，但是第一个周期无法100%，后续周期可以实现；不建议使用；
- CHMASK 也可稳定实现 0% 和 100%。

2.3.7 HA01:

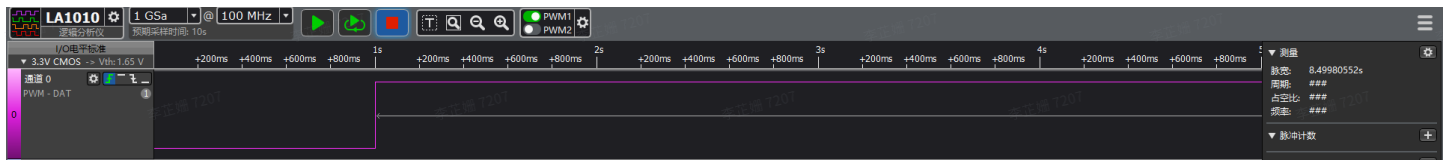
SDK1_4_0:

- eTMR_DRV_UpdatePwmChannel :

```
eTMR_DRV_ClearLdok(0);
eTMR_DRV_UpdatePwmChannel(0, 2, 0x0, 0);
eTMR_DRV_UpdatePwmChannel(0, 3, 0x0, 0);
eTMR_DRV_SetLdok(0);
OSIF_TimeDelay(1000);

eTMR_DRV_ClearLdok(0);
// g_etmrBase[0]->CH[2].VAL0 = 0;
// g_etmrBase[0]->CH[2].VAL1 = g_etmrBase[0]->MOD + 1;
// g_etmrBase[0]->CH[3].VAL0 = 0;
// g_etmrBase[0]->CH[3].VAL1 = g_etmrBase[0]->MOD + 1;
eTMR_DRV_UpdatePwmChannel(0, 2, 0x8000, 0);
eTMR_DRV_UpdatePwmChannel(0, 3, 0x8000, 0);
eTMR_DRV_SetLdok(0);
OSIF_TimeDelay(1000);
```

现象：0%占空比第一次输出正常，后续就无法正常输出

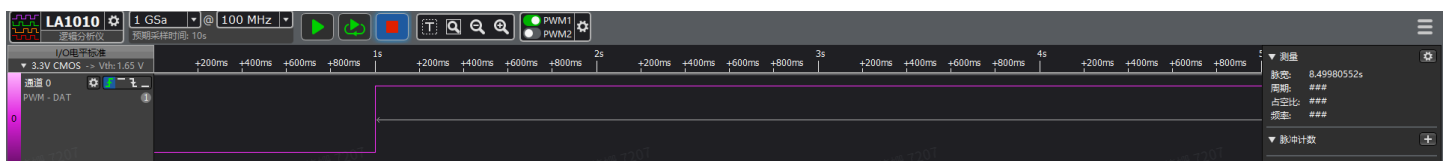


- 使用 MOD+1 实现 100%, 0% 继续使用 SDK 接口:

```
eTMR_DRV_ClearLdok(instance: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 2, dutyCycle: 0x0, offset: 0);
eTMR_DRV_UpdatePwmChannel(instance: 0, channel: 3, dutyCycle: 0x0, offset: 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CH[2].VAL0 = 0;
g_etmrBase[0]->CH[2].VAL1 = g_etmrBase[0]->MOD + 1;
g_etmrBase[0]->CH[3].VAL0 = 0;
g_etmrBase[0]->CH[3].VAL1 = g_etmrBase[0]->MOD + 1;
//eTMR_DRV_UpdatePwmChannel(0, 2, 0x8000, 0);
//eTMR_DRV_UpdatePwmChannel(0, 3, 0x8000, 0);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

现象：0%占空比第一次输出正常，后续就无法正常输出

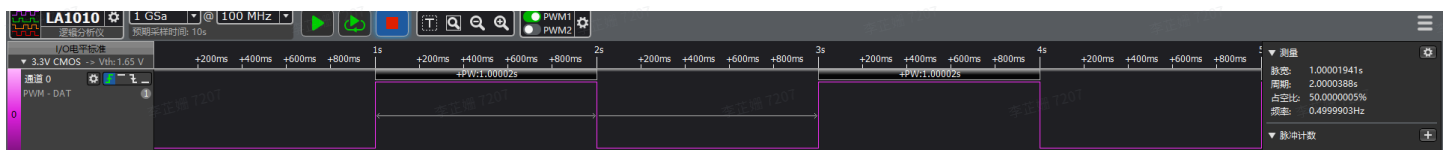


- 使用 CHMASK 强制输出高、低电平。

```
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
(0 << eTMR_CHMASK_CH2MV_SHIFT) | (0 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);

eTMR_DRV_ClearLdok(instance: 0);
g_etmrBase[0]->CHMASK = (1 << eTMR_CHMASK_CH2M_SHIFT) | (1 << eTMR_CHMASK_CH3M_SHIFT) |
(1 << eTMR_CHMASK_CH2MV_SHIFT) | (1 << eTMR_CHMASK_CH3MV_SHIFT);
eTMR_DRV_SetLdok(instance: 0);
OSIF_TimeDelay(delay: 1000);
```

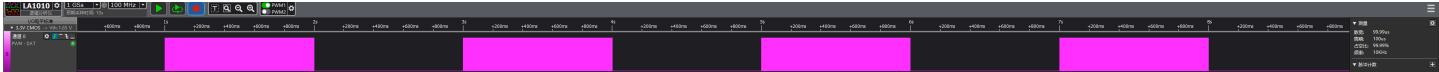
现象：输出符合预期



SDK1_4_1:

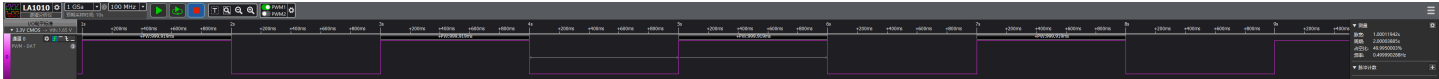
• eTMR_DRV_UpdatePwmChannel :

现象：占空比设置为 100% 时，实际输出约为 99.9%，无法达到 100%。



• 使用 MOD+1 实现 100%，0% 继续使用 SDK 接口:

现象：输出符合预期



• 小结:

HA01: 使用 SDK 接口设置 0% 占空比时，首次输出正常，但后续循环切换时无法输出 0% 占空比，因此不能视为稳定支持 0% 占空比。采用 MOD+1 实现 100% 后，0% 输出问题仍然存在。需要反复切换 0% 和 100% 时，建议统一采用 CHMASK。

SDK1_4_0:

- 1. SDK 接口无法正常输出100%占空比；
- 2. 通过 MOD+1 也无法实现100%占空比；
- 3. CHMASK 也可稳定实现 0% 和 100%；

SDK1_4_1_HF001:

- 1. SDK 接口无法正常输出100%占空比；
- 2. 100% 可通过 MOD+1 实现，但是第一个周期无法100%，后续周期可以实现；不建议使用；
- 3. CHMASK 也可稳定实现 0% 和 100%；

2.4 测试代码

📄 eTMR_PWM_0pct_100pct_SDK_v1.4.0.zip

📄 eTMR_PWM_0pct_100pct_SDK_v1.4.1.zip

3. 结论

芯片系列	SDK1.4.0	SDK1.4.0 API实现	SDK1.4.1 API实现	SDK1.4.1 API实现	SDK1.4.0 MOD + 1	SDK1.4.1 MOD + 1	CHMASK 实现0%	CHMASK 实现100%
	API实现0%	100%	0%	100%				

					实现 100%	实现 100%		
LE04/L E05	支持	不支持	支持	不支持	支持	支持	支持	支持
LE14/L E15	支持	支持	支持	不支持	不支持	不支持	支持	支持
MC03	不支持	不支持	支持	不支持	不支持	支持	支持	支持
MD14	支持	不支持	支持	不支持	支持	支持	支持	支持
MD24	支持	支持	支持	支持	不支持	不支持	支持	支持
ME05	支持	不支持	支持	不支持	支持	支持	支持	支持
HA01	不支持	不支持	支持	不支持	不支持	支持	支持	支持

Note:

1. 所有系列均支持通过 **CHMASK** 实现PWM 0%和100%占空比输出，**强烈建议 CHMASK 可作为唯一方案**；
2. LE04/LE05、MD14、ME05可以通过Mod+1实现100%输出，但是个别工况第一个周期只能99.9X%（存在极短时间的低脉冲），无法实现100%输出，后续周期可以正常输出100%，**为方便代码管理及维护，强烈建议后续不要使用MOD+1实现**；
3. LE14/LE15、MC03、MD24、HA01无法通过MOD+1实现100%输出；
4. 由于不同版本的SDK对于实现PWM边界处理方式不同，因此为了一致性，强烈建议使用CHMASK实现100%；