

# 基于 C28x 系列 MCU Type 1 EPWM 模块实现 0% - 100% 占空比输出控制

Ricky Zhang

C2000 FAE / China

## 摘要

使用 EPWM 模块实现 0% - 100% 占空比即零脉宽和全脉宽的输出控制在电机控制和数字电源应用中是非常常见的需求。传统的 LF240xA 系列的 EV 模块可以实现这样的功能，但 C28x 系列 MCU 新的 EPWM 模块在原始版本 Type 0 上不能直接通过硬件支持，需要通过软件的方式进行处理，而 Type 1 EPWM 模块虽然修正了这个功能的实现，仍然需要进行额外的配置。本文基于实际应用的情况，给出了 Type 1 及以上版本 EPWM 模块的解决方法和注意事项，并提供了实现的参考代码和测试波形。参考代码可以在以下链接进行下载：<https://www.ti.com.cn/cn/lit/zip/ZHCAAL3>。

## Contents

1	EPWM 模块实现 0% - 100% 占空比输出控制.....	2
2	EPWM 模块的 Type 介绍.....	2
	2.1 EPWM Type 描述.....	2
	2.2 EPWM Type 0 升级到 Type 1 的增强.....	3
3	原因分析.....	3
4	解决方法和示例代码.....	5
5	测试参考代码和测试波形.....	6
6	参考文献.....	10

## Figures

Figure 1.	EPWM 模块 Type 描述.....	<b>Error! Bookmark not defined.</b>
Figure 2.	默认的 EPWM1A 输出.....	3
Figure 3.	期望的 EPWM1A 输出.....	4
Figure 4.	实际的 EPWM1A 输出.....	4
Figure 5.	非零占空比切换到全脉宽即 100% 高电平输出.....	7
Figure 6.	全脉宽即 100% 高电平恢复到非零占空比输出.....	7
Figure 7.	非零占空比切换到零脉宽即 0% 高电平输出.....	8
Figure 8.	零脉宽即 0% 高电平恢复到非零占空比输出.....	8
Figure 9.	全脉宽即 100% 高电平切换到零脉宽即 0% 高电平输出.....	9
Figure 10.	零脉宽即 0% 高电平切换到全脉宽即 100% 高电平输出.....	9

## 1 EPWM 模块实现 0% - 100%占空比输出控制

使用 EPWM 模块实现 0% - 100%占空比即零脉宽和全脉宽的输出控制在电机控制和数字电源应用中是非常常见的需求，因为这样可以在系统设计上最大化母线电压利用率，从而提高控制系统效率。当输出占空比由非零脉宽变成零脉宽或者全脉宽，或者相反时，如果不能按预期输出，而是延迟一个周期，则会造成电流冲击，从而损坏控制系统。传统的 LF240xA 系列的 EV 模块可以实现这样的功能，但 C28x 系列 MCU 新的 EPWM 模块在原始版本 Type 0 上不能直接通过硬件支持，需要通过软件的方式进行处理，带来了系统软件的复杂性和实时控制的性能变差。本文要解决的是如何在 Type 1 及以上版本 EPWM 模块上通过硬件配置实现这个功能。

## 2 EPWM 模块的 Type 介绍

在分析根本原因和提供解决方案之前，需要先了解 C28x 系列 MCU 新的 EPWM 模块不同版本是如何升级的，以及不同版本之间有什么区别。

### 2.1 EPWM Type 描述

在 C2000 Real-Time Control Peripheral Reference Guide (Rev. N)应用文档(spru566n)中，3.3.7 Enhanced Pulse Width Modulator (ePWM) Module 章节详细描述了 Type 0~4 不同版本的 EPWM 模块的基本功能和差别，并根据时间顺序对整个 C28x 系列 MCU 不同家族产品系列进行了映射，如下图 1:

**Table 42. Enhanced Pulse Width Modulator (ePWM) Module Type Description**

Type	Description	Devices Covered	Device-Specific Options
0	Original ePWM Module Type	280x <sup>(1)</sup> , 2801x	Time-base synchronization scheme 1: ePWM1 SYNC out is fed to eCAP1 and ePWM2. All other ePWM modules have their SYNCIN signals fed from the ePWM module numerically preceding them.
		2804x, 2809	Time-base synchronization scheme 2: Two ePWM pinouts are possible: A-channel only or 280x compatible. If the ePWM pinout is configured for A-channel only mode, ePWM1 SYNC out is fed to eCAP1, ePWM2, ePWM5, ePWM9, and ePWM13. All other ePWM modules have their SYNCIN signals fed from the ePWM module numerically preceding them. If the ePWM pinout is configured for 280x-compatible mode, synchronization scheme 1 is used.
		2823x, 2833x, 28234	Time-base synchronization scheme 3: ePWM1 SYNC out is fed to eCAP1, ePWM2, and ePWM4. All other ePWM modules have their SYNCINs fed from the ePWM module numerically preceding them.
1	Doubled deadband resolution, interrupts and ADC SOC can be generated on both CTR = 0 or CTR = PRD, added digital compare submodule, added hooks for high-resolution period.	2802x, 2803x, 2805x, 2806x	-

**Figure 1. EPWM 模块 Type 描述**

## 2.2 EPWM Type 0 升级到 Type 1 的增强

根据上图中的表格描述，EPWM Type 0 升级到 Type 1 时主要的增强包括：

- 死区时间的精度翻倍
- 中断和 ADC SOC 可以配置在 CTR=0 或 CTR=PRD 时产生
- 增加数字比较器子模块
- 增加高精度周期模式的钩子

即 F2802x、F2803x、F2805x 和 F2806x 及之后支持 Type 2 或 Type 4 的所有 C28x 系列 MCU 都有以上增强，从而可以直接硬件实现零脉宽和全脉宽的输出，因为在过零点 CTR=0 或周期点 CTR=PRD 时可以支持更多灵活的配置，包括产生中断和 ADC SOC，以及 CC 比较值寄存器的加载和 AQ 动作寄存器的生效。

## 3 原因分析

EPWM Type 0 不能直接硬件支持零脉宽和全脉宽的输出控制的根本原因在于过零点 CTR=0 或周期点 CTR=PRD 时需要执行的动作很多，但它们存在不确定的优先级，导致无法按预期工作，而 EPWM Type 1 会存在一个周期的延迟，是因为两个动作同时执行，导致 AQ 来不及基于 CMPA=0 进行生效。下面我们通过例子以 EPWM1 来进行 Type 1 输出延迟的说明。

在计数对称模式即 up-down count mode 下，我们假设 CC 比较值寄存器使能影子模式即 shadow mode，EPWM1A 通道通过 CMPA 寄存器产生的事件进行 AQ 触发动作输出特定的波形，CMPA 设置在过零点 CTR=0 时进行 shadow 到 active 寄存器的加载，而 AQ 设置在 CAU 即计数上升时置高电平输出，CAD 即计数下降时拉低电平输出，如下图 2 所示：

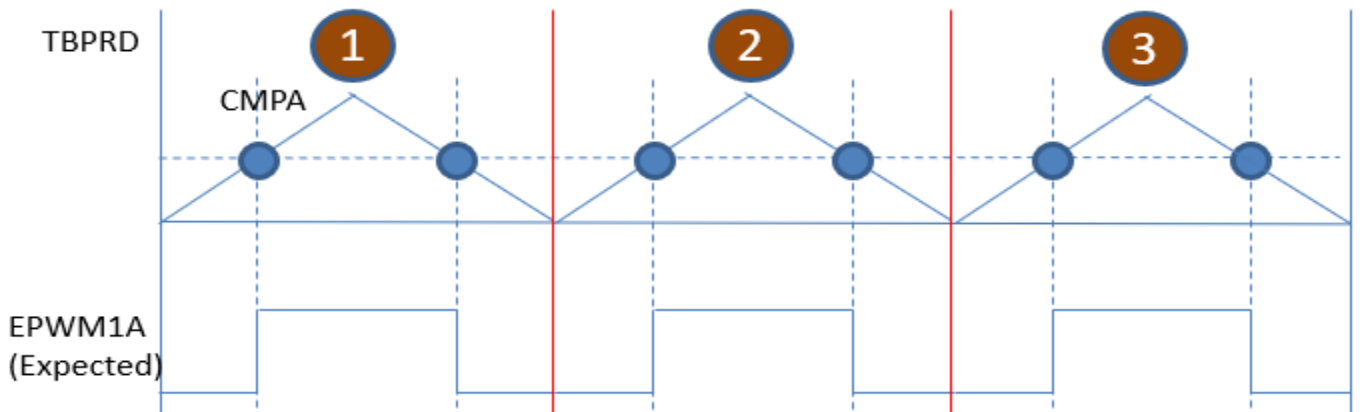


Figure 2. 默认的 EPWM1A 输出

对应的配置代码为：

```
// Setup shadow register load on ZERO
EPwm1Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
EPwm1Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
```

```
// Set Actions
EPwm1Regs.AQCTLA.bit.CAU = AQ_SET;           // Set EPWM1A on CAU
EPwm1Regs.AQCTLA.bit.CAD = AQ_CLEAR;        // Clear EPWM1A on CAD
```

现在，假定我们需要在第 2 个周期开始输出全脉宽即 100% 占空比的高电平，显然只要在第 1 个周期结束前更新 CMPA 寄存器的值为 0 即可，这可以通过第 1 个周期对应的过零点中断进行操作，那么此时期望看到的输出波形应该是如下图 3：

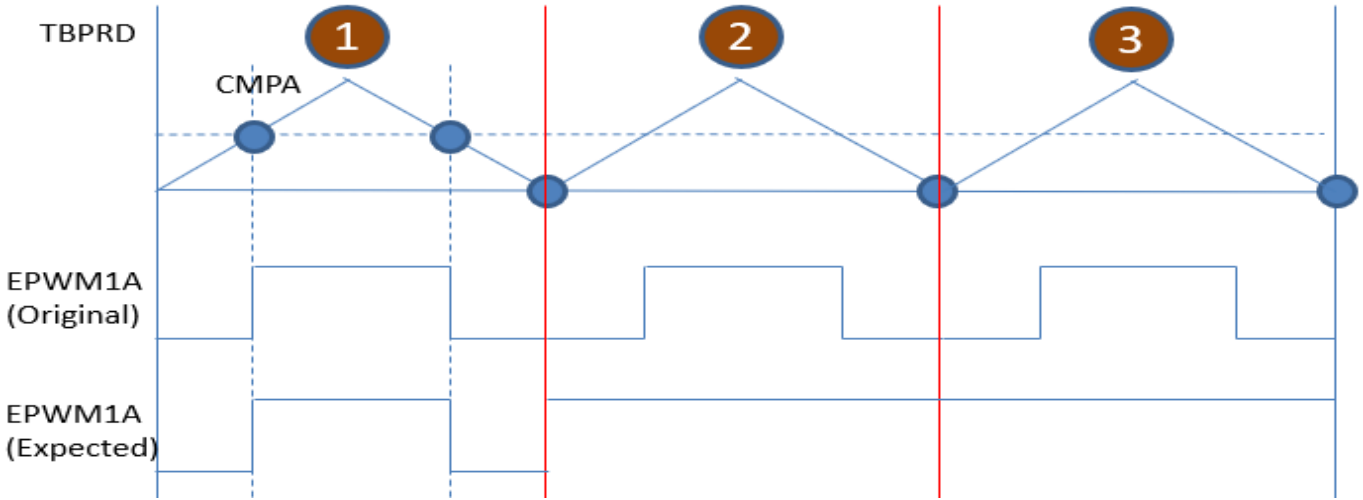


Figure 3. 期望的 EPWM1A 输出

然而实际情况是，第 2 个周期将不会有任何基于 CMPA 寄存器产生事件的 AQ 触发动作，因而会一直维持第 1 个周期结束时的低电平输出，直到第 3 个周期才正式生效，输出期望的波形，如下图 4：

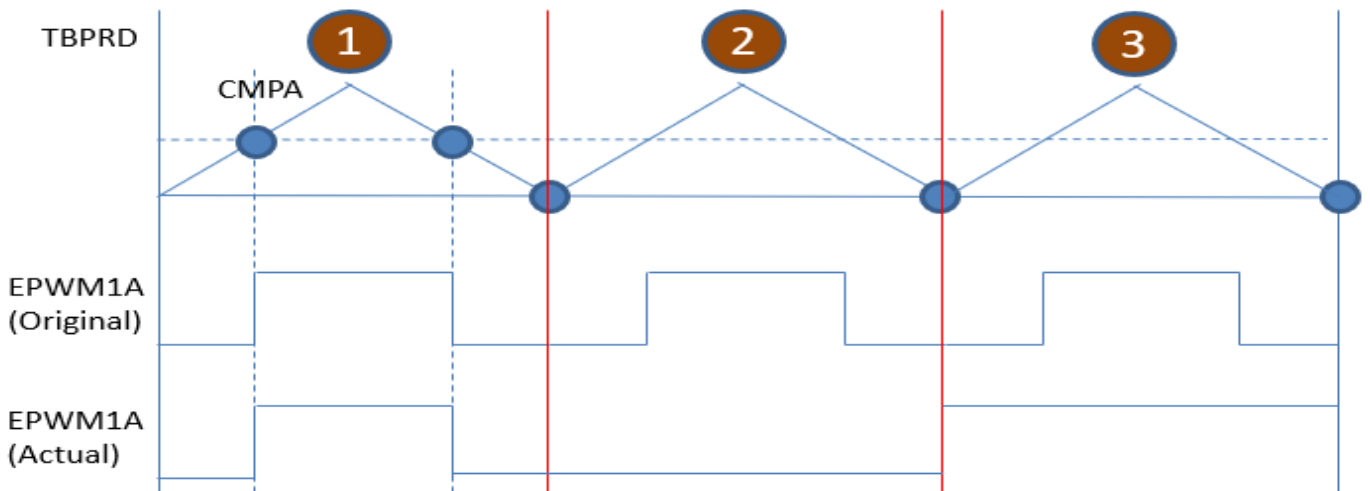


Figure 4. 实际的 EPWM1A 输出

导致这个结果的根本原因就在于，第 1 个周期结束到达过零点 CTR=0 时，EPWM 模块需要同时做两件事情，即 1). 进行 CMPA 的 shadow 到 active 寄存器的加载，2). 执行 AQ 基于 CAU 对应 CMPA=0 的触发动作，而 CMPA=0 生效和执行 CAU 的触发动作同时发生，就会导致后者实际上无法完成，因而并不会输出期望的高电平。当一个 EPWM 时钟过去，CTR=1 并逐渐增大，CMPA=0 已生效，但再没有匹配 CTR 的 CMPA 值，从而在整个第 2 个周期都不会再产生基于 CAU 或者 CAD 的事件，即无法改变输出电平，而是一直维持第 1 个周期最后时刻的输出。第 2 个周期结束到达过零点 CTR=0 时，此时只需要执行 AQ 基于 CAU 对应 CMPA=0 的触发动作，则可以直接输出高电平，从而实现全脉宽即 100% 占空比的输出，但此时相当于丢失一个周期的输出，或者产生了一个周期的延迟。

## 4 解决方法和示例代码

对于 EPWM Type 0 的解决办法，由于硬件上无法实现，只能通过软件进行处理，具体方法可以参考 Using the ePWM Module for 0% - 100% Duty Cycle Control 应用文档(SPRAAI1)。

对于 EPWM Type 1 的解决办法，根据上面的分析，只要通过其它方式在 CTR=0 时使输出状态发生变化，然后由 AQ 基于 CAU 对应 CMPA=0 在稍后生效后立即去触发新动作，就可以避免丢失第 2 个周期或者产生一个周期的延迟，从而解决问题。

由于 AQ 基于事件执行触发动作的优先级高于其它动作，且 AQ 事件也可以基于 ZRO 产生，则加上一个在计数过零点 CTR=0 时的 AQ 动作，使输出信号直接拉低，然后等新的 CAU 事件生效执行，就可以满足以上要求，同时考虑正常情况下输出时，这个配置本身并不会与现有配置冲突带来新的问题，因此可以兼容正常占空比输出和零脉宽及全脉宽的切换。

因此，对应的配置代码示例为：

```
// Setup shadow register load on ZERO
EPwm1Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
EPwm1Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
// Set Actions
EPwm1Regs.AQCTLA.bit.ZRO = AQ_CLEAR;           // Clear EPWM1A on ZRO
EPwm1Regs.AQCTLA.bit.CAU = AQ_SET;             // Set EPWM1A on CAU
EPwm1Regs.AQCTLA.bit.CAD = AQ_CLEAR;          // Clear EPWM1A on CAD
```

针对不同的 AQ 配置以及是否需要在计数周期点 CTR=PRD 进行 shadow 寄存器到 active 的加载和执行 AQ 动作，需要修改这两种情形的相关配置。

一是如果 AQ 设置是相反的，则相应的配置也要修改，比如以上例子在 CAU 即计数上升时置高电平输出和 CAD 即计数下降时拉低电平输出，在实际应用系统里是在 CAU 即计数上升时拉低电平输出和 CAD 即计数下降时置高电平输出，则以上的配置应该如下：

```
// Setup shadow register load on ZERO
EPwm1Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
EPwm1Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
// Set Actions
EPwm1Regs.AQCTLA.bit.ZRO = AQ_SET;             // Set EPWM1A on ZRO
EPwm1Regs.AQCTLA.bit.CAU = AQ_CLEAR;          // Clear EPWM1A on CAU
EPwm1Regs.AQCTLA.bit.CAD = AQ_SET;            // Set EPWM1A on CAD
```

二是如果需要在计数周期点  $CTR=PRD$  进行 shadow 寄存器到 active 的加载和执行 AQ 动作，则需要修改 CMPCTL 寄存器的 LOADAMODE 位域设置，同时加上 AQ 寄存器的 PRD 的设置。

## 5 测试参考代码和测试波形

测试参考代码加入了 GPIO 测试信号和进行 CMPA 更新的中断代码，以 EPWM3 为例，通过辅助 EPWM1 产生中断进行测试，以验证 6 种情况的切换完全没有问题：

- 从非零占空比切换到全脉宽即 100%高电平输出，然后再恢复到非零占空比；
- 从非零占空比切换到零脉宽即 0%高电平或 100%低电平输出，再恢复到非零占空比；
- 从全脉宽到零脉宽和从零脉宽到全零脉宽；

具体的测试信号说明如下：

- 通道 1：5us 的 EPWM1 过零中断里更新 EPWM3 CMPA 的值；
- 通道 2：更新 CMPA 值时对应的 GPIO9 输出，用于触发查看转换瞬间的占空比变化；
- 通道 3：EPWM3B 输出，在每次过零时翻转，主要目的是确定 EPWM3 的  $CTR=0$  过零点；
- 通道 4：EPWM3A 即需要的波形，在 CAU 时置高，CAD 时拉低，通过 5us 中断修改 CMPA 为 0 或者非 0 或者周期值，达到输出全脉宽，零脉宽或非零占空比波形。

该测试代码基于支持 EPWM Type 4 的 F28377D 芯片，可以直接放在例程路径下替换任意一个源文件测试，比如以下路径下(版本号可能需要调整)的 epwm\_updown\_aq\_cpu01.c 文件被替换直接编译运行：  
[C:\ti\controlSUITE\device\\_support\F2837xD\v210\F2837xD\\_examples\Cpu1\epwm\\_updown\\_aq\cpu01](C:\ti\controlSUITE\device_support\F2837xD\v210\F2837xD_examples\Cpu1\epwm_updown_aq\cpu01)

测试说明：

1. 在调试窗口里手动更改 DutyCycleChange 的值，中断程序里可以更新对应的 CMPA 值，从而得到不同的输出波形；
2. 需要注意，由于 EPWM1 的中断频率与 EPWM3 的输出频率并不一致(特意为了测试时方便查看波形)，该程序不能直接验证到在周期点更新比较值的情况，可以将中断频率改为 EPWM3 的两倍频率验证；
3. 针对上面提到的需要修改输出配置的两种情形，通过宏定义的方式进行了处理：
  - a. 默认的输出 CAU 置高而 CAD 拉低，SET\_CAU\_CLEAR\_CAD 设置为 1 即可，反之为 0 则代表 CAU 拉低而 CAD 置高；
  - b. 默认只在过零点更新比较寄存器 CMPA 的值，即 LOAD\_CMP\_ON\_ZRO\_TBPRD 设置为 0，如果需要同时在周期点也更新，则 LOAD\_CMP\_ON\_ZRO\_TBPRD 需要设置为 1，对应的配置代码会自动打开，此时只是增加了 CMPCTL 和 AQCTLA 在 TBPRD 的配置。

对应的 6 个切换波形如下图 5-10：

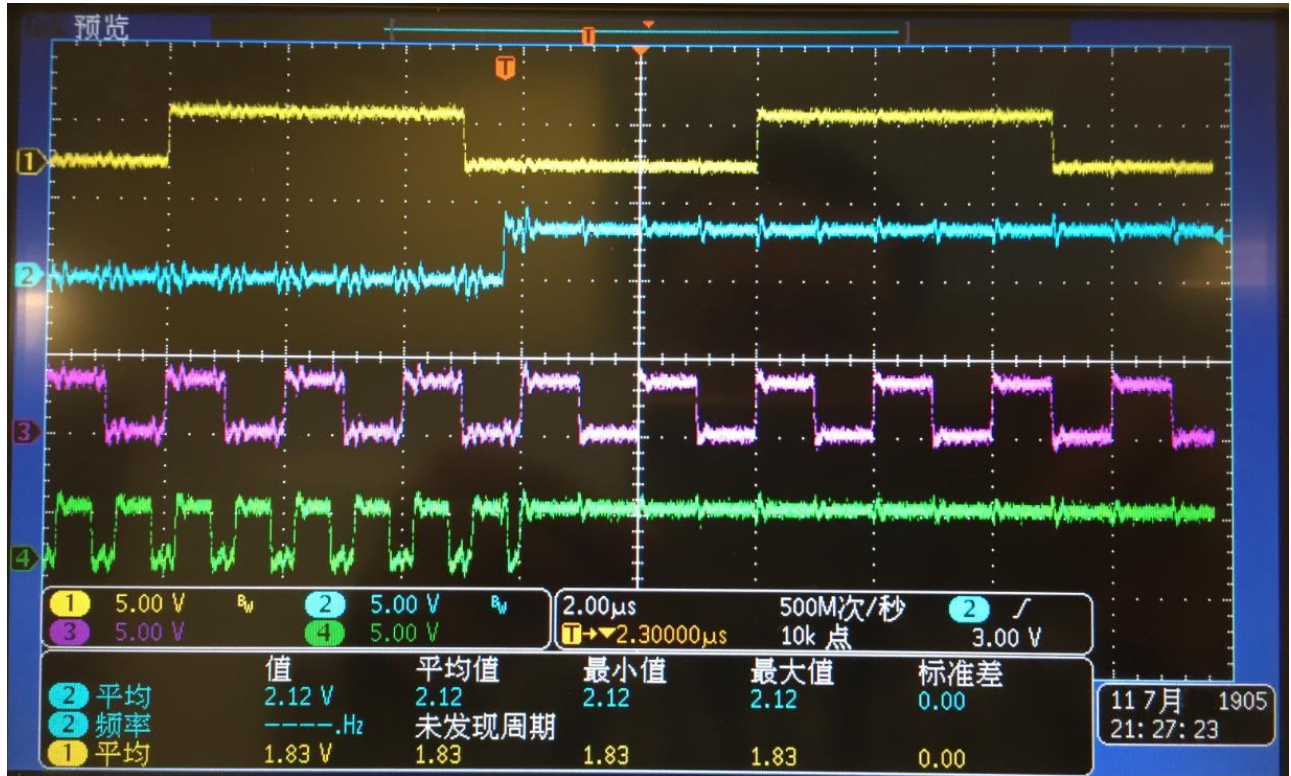


Figure 5. 非零占空比切换到全脉宽即 100%高电平输出

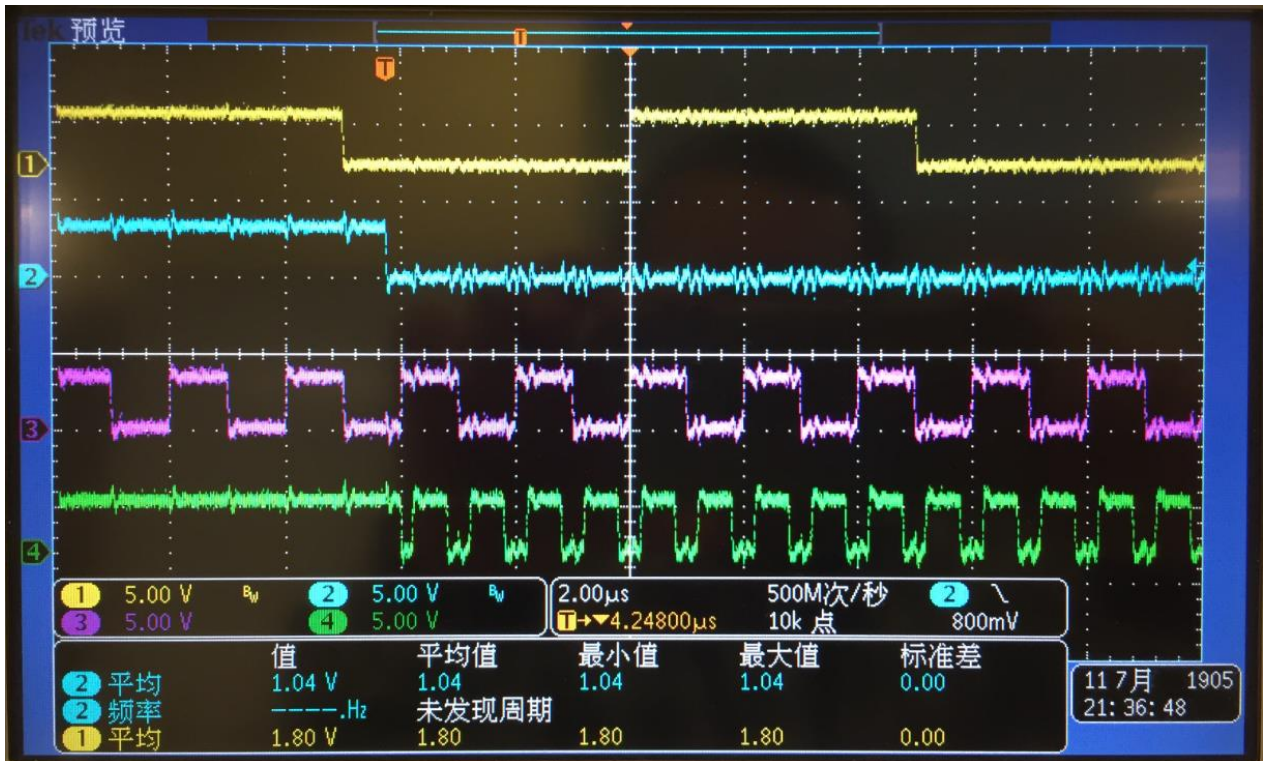


Figure 6. 全脉宽即 100%高电平恢复到非零占空比输出

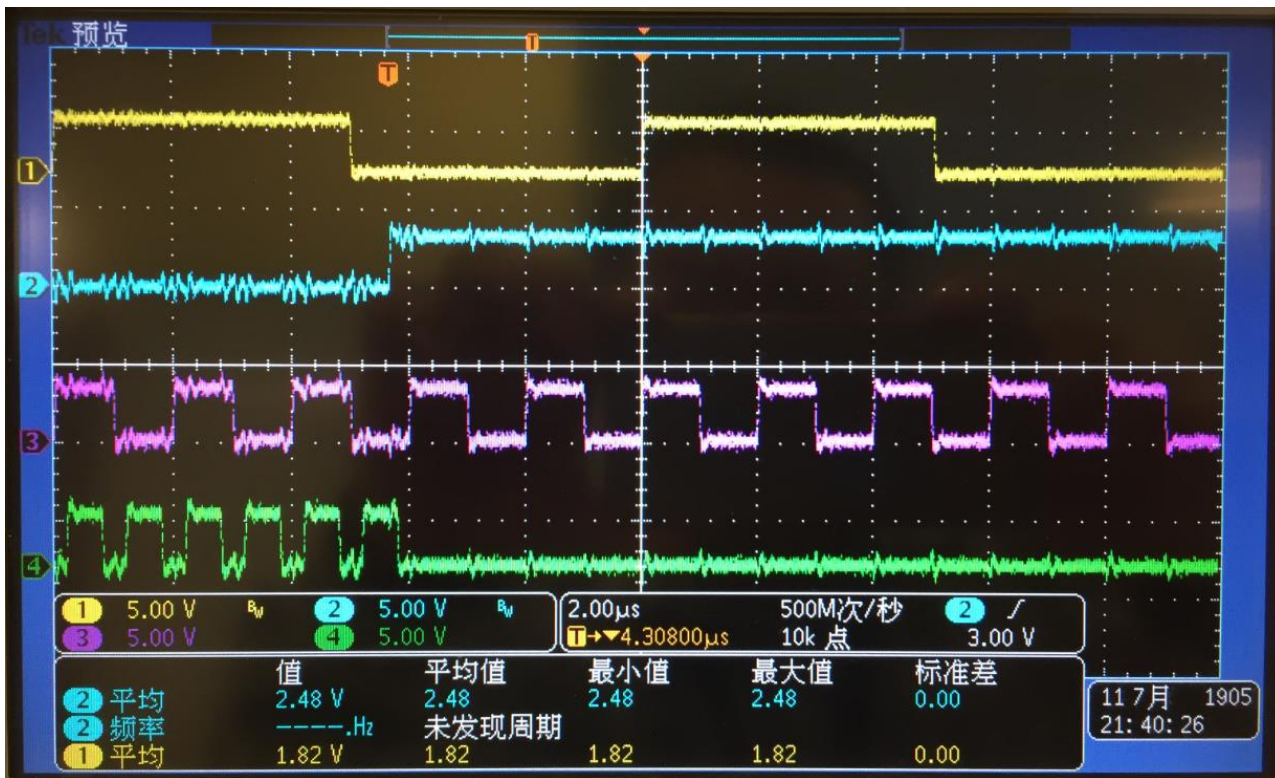


Figure 7. 非零占空比切换到零脉宽即 0%高电平输出

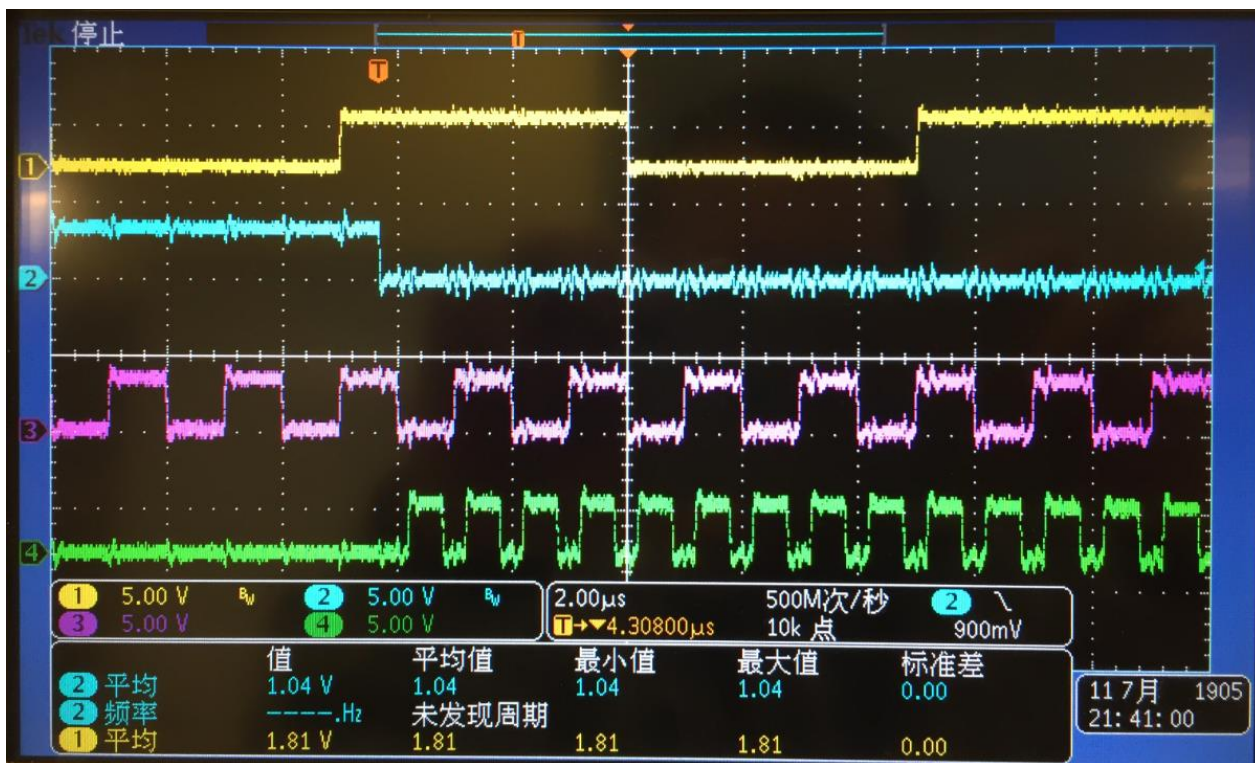


Figure 8. 零脉宽即 0%高电平恢复到非零占空比输出



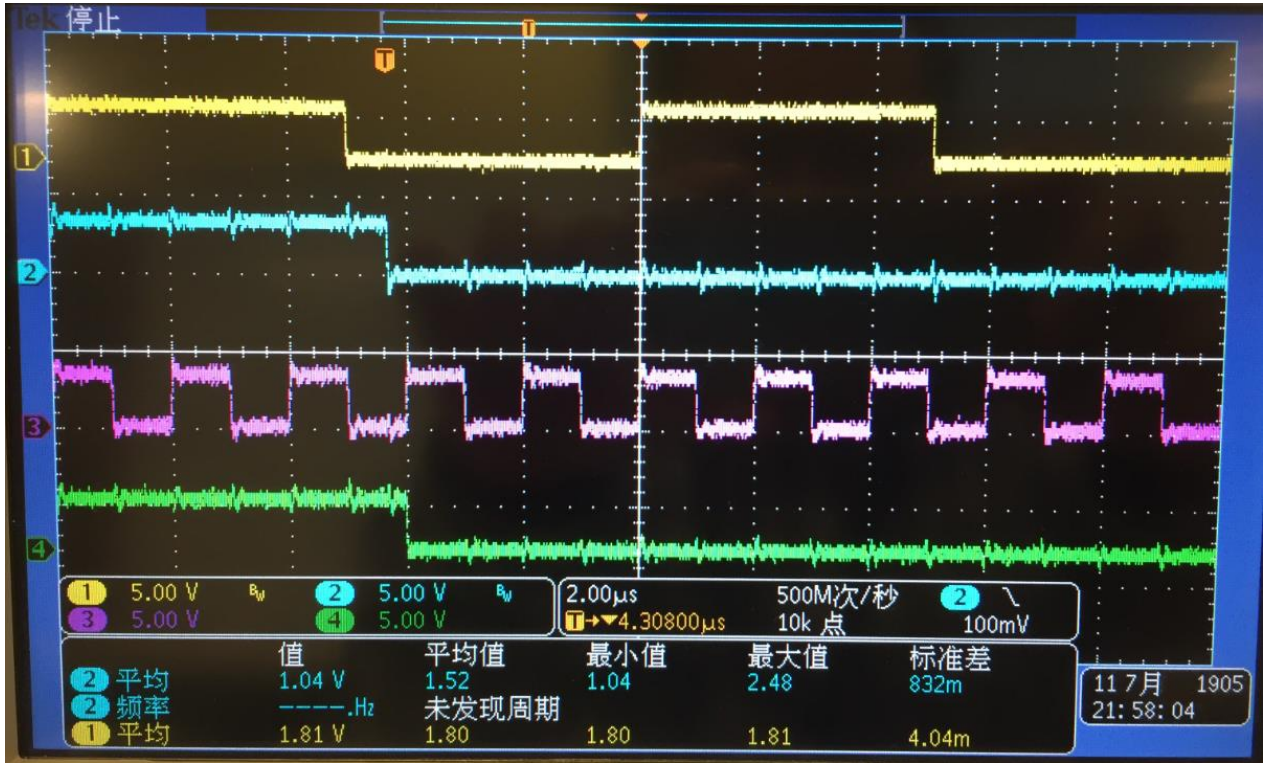


Figure 9. 全脉宽即 100%高电平切换到零脉宽即 0%高电平输出



Figure 10. 零脉宽即 0%高电平切换到全脉宽即 100%高电平输出

## 6 参考文献

1. *C2000 Real-Time Control Peripheral Reference Guide (Rev. N) (SPRU566N)*
2. *Using the ePWM Module for 0% - 100% Duty Cycle Control (SPRAAI1)*
3. *TMS320x2802x, 2803x Piccolo Enhanced Pulse Width Modulator (ePWM) Module Reference Guide (SPRUGE9F)*
4. *TMS320F2837xD Dual-Core Microcontrollers Technical Reference Manual (SPRUHM8I)*

## 重要声明和免责声明

TI 提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他安全、安保或其他要求。这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 TI 的销售条款 (<https://www.ti.com.cn/zh-cn/legal/termsofsale.html>) 或 [ti.com.cn](https://www.ti.com.cn) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

邮寄地址：上海市浦东新区世纪大道 1568 号中建大厦 32 楼，邮政编码：200122

Copyright © 2021 德州仪器半导体技术（上海）有限公司