

# 实现多轴电机驱动系统内的多片 C2000 PWM 同步

Brian Wang

North and West China

## ABSTRACT

在工厂自动化设备、机器人等多轴电机驱动系统当中，经常使用多片 C2000 MCU 各自独立控制一个或电机协同工作。但由于各控制环路独立运行，电机及功率级开关管的开关噪声可能对系统内其它控制回路的采样电路造成干扰，影响控制效果。为解决此问题，必须合理规划采样窗口，以规避开关噪声，提升采样电路的精度。而达到这一效果的前提是进行多片 C2000 及单片 C2000 内部的 PWM 同步，使 PWM 开关时刻可控。

下文将以 F28004x 为例，介绍实现多片 C2000 之间 PWM 同步的方法。

## Contents

1	多轴电机驱动的干扰问题 .....	2
2	C2000 片内 PWM 的同步方法 .....	2
3	C2000 片间 PWM 的同步方法 .....	4
4	主要参考代码 .....	7
5	测试结果 .....	10
6	参考文献 .....	11

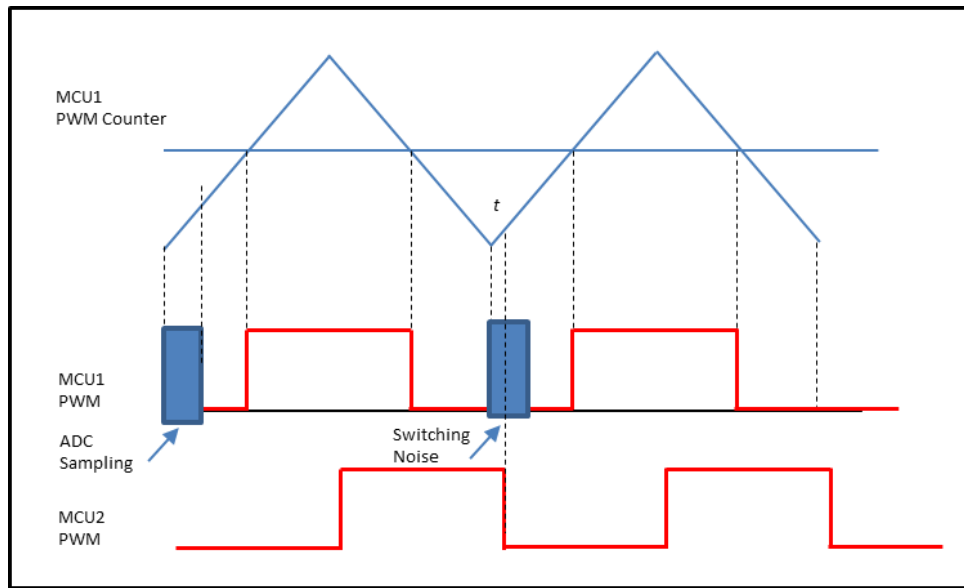
## Figures

Figure 1.	多轴电机驱动的干扰问题示意图 .....	2
Figure 2.	F28004x PWM Time-base 模块示意图 .....	3
Figure 3.	F28004x PWM 模块同步树 .....	3
Figure 4.	PWM 片内同步示意图 .....	4
Figure 5.	MCU 片内各 PWM 同步配置 .....	4
Figure 6.	F28004x PWM 模块外部同步信号 .....	5
Figure 7.	F28004x 输入 XBAR 及 EXTSYNCIN 信号走向 .....	5
Figure 8.	Input Xbar MUX 映射表 .....	6
Figure 9.	F28004x 输出 XBAR 及 EXTSYNOUT 信号走向 .....	6
Figure 10.	PWM 片间同步示意图 .....	7
Figure 11.	主机 (MCU1) 及从机 (MCU2/3) PWM 片内同步配置 .....	7
Figure 12.	同步逻辑分析仪测试结果 .....	10
Figure 13.	同步逻辑分析仪测试结果 .....	10

**Figure 14. PWM 片间同步示波器测试结果 ..... 11**

## 1 多轴电机驱动的干扰问题

以双轴系统单次采样为例说明该问题：其中 MCU1 的 PWM Counter 工作在 Up-down 计数模式。系统在 PWM Counter=0 起始时刻进行 ADC 采样，通过最大占空比及 PWM Counter 比较值的正确设置，可以配置 PWM 开关时刻尽量避开 PWM 采样窗口，从而尽量减少 ADC 采样窗口内的开关噪声，提高采样精度。



**Figure 1. 多轴电机驱动的干扰问题示意图**

但在由多片 C2000 独立控制的多轴电机驱动系统中，如果对于两个控制系统的 PWM 关系不加以任何限制，某一路 PWM 的开关动作很可能发生在其它控制器的采样窗口期间，开关噪声会对另外一个芯片的采样造成干扰（如 Figure 1 中 t 时刻），因此需要多片独立 C2000 内部的 PWM 相互同步。

## 2 C2000 片内 PWM 的同步方法

对于一个典型的三相电机驱动系统，首先要进行片内 6 路 PWM（3 个 PWM 模块）的同步。下面以 F28004x 为例介绍片内 PWM 同步的方法。

Figure 2 为 F28004x PWM 的 Time-base 模块，红色标注的 EPWMxSYNCl 和 EPWMxSYNCO 分别为同步输入及同步输出脉冲。其中 EPWMxSYNCl 为同步输入脉冲，用于同步本 PWM 模块计数器；而 EPWMxSYNCO 为同步输出脉冲，用于产生一个同步信号以同步链路上的其它 PWM 模块计数器。特别注意，EPWMxSYNCO 除了可以由当前 PWM 模块本身产生以外还可以选择为 EPWMxSYNCl（通过 TBCTL[SYNCOSEL]配置），即脉冲输入信号直接 bypass 当前模块作为输出。这种配置在使用同一个信号同步多路 PWM 时广泛使用。

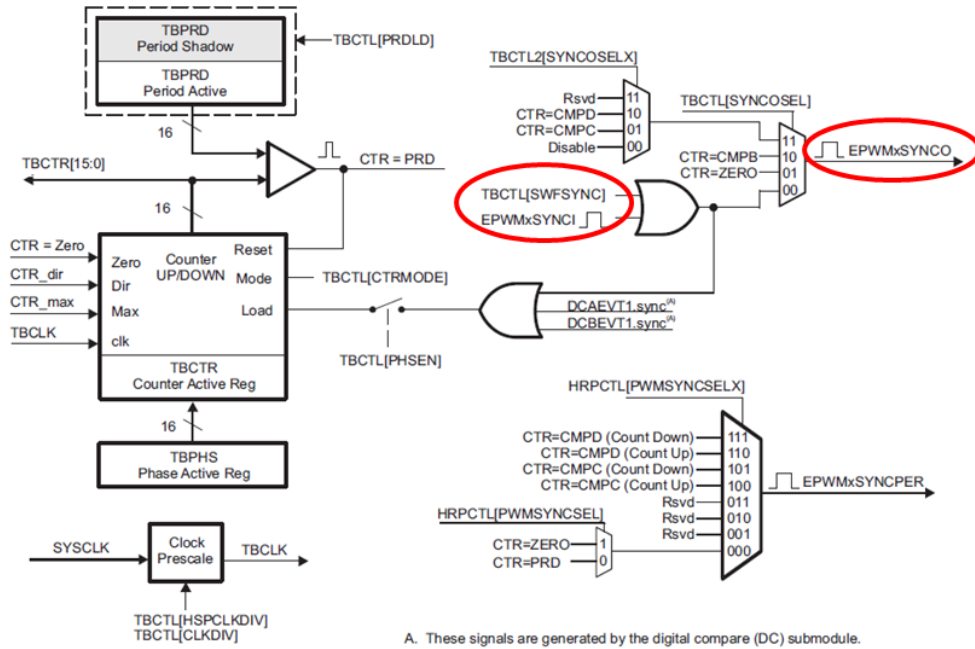


Figure 2. F28004x PWM Time-base 模块示意图

Figure3 为 F28004x 同步树，箭头代表同步信号方向（输入/输出）。对于 PWM2/3/5/6/8，其同步输入信号唯一。对于 PWM1/4/7 其同步信号有多个选择，其中 EXTSYNClN1/2 为外部同步输入，可通过 XBAR 映射到芯片的一个 Pin 脚上。

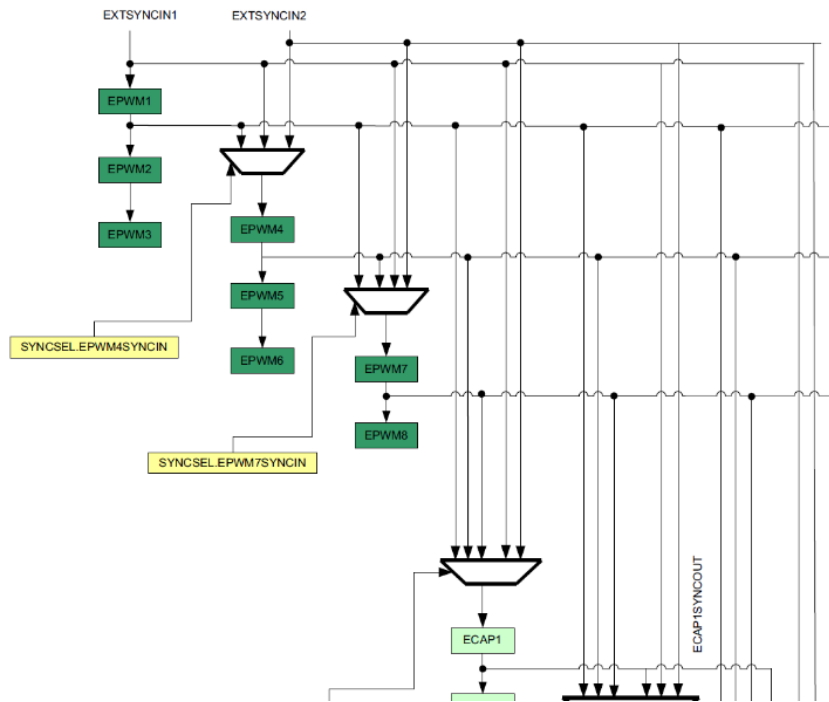
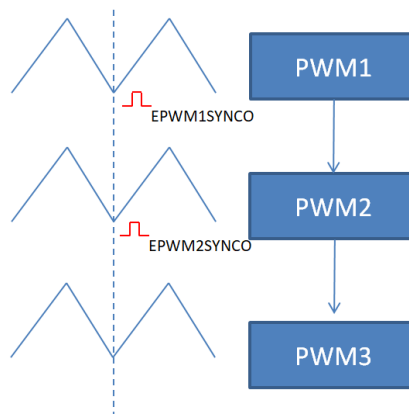


Figure 3. F28004x PWM 模块同步树

以同步 PWM1/2/3 为例，介绍如何同步一片 F28004x 内的多路 PWM。根据同步信号流向，用 PWM1 同步 2，2 同步 3，即可完成同一个 CPU 内多个 PWM 模块的同步，示意如 Figure 4。



**Figure 4. PWM 片内同步示意图**

PWM 模块同步的配置如下表：

	EPWMxSYNCI 信号选择	EPWMxSYNCO 信号选择
EPWM1	N/A	CTR = ZERO(即 CTR=0 时 EPWM 的同步输出产生一个同步信号)
EPWM2	EPWM1SYNCO (把 EPWM1 输出的同步信号作为 EPWM2 的同步信号)	EPWM2SYNCO (把 EPWM2 同步信号输入作为输出，相当于同步信号通过 EPWM2 进行了传递)
EPWM3	EPWM2SYNCO	N/A

**Figure 5. MCU 片内各 PWM 同步配置**

### 3 C2000 片间 PWM 的同步方法

前文已经介绍了实现单个片内多个 PWM Module 同步的方法，下面来介绍如何进行多个芯片之间的 PWM 同步。

再次回到时钟 F28004x 系列的 PWM 同步树，除了每个 PWM 模块的 EPWMxSYNCI/SYNCO 信号以外，整个系统还有三个对外的同步信号 EXTSYNIN1/2 和 EXTSYNOUT。

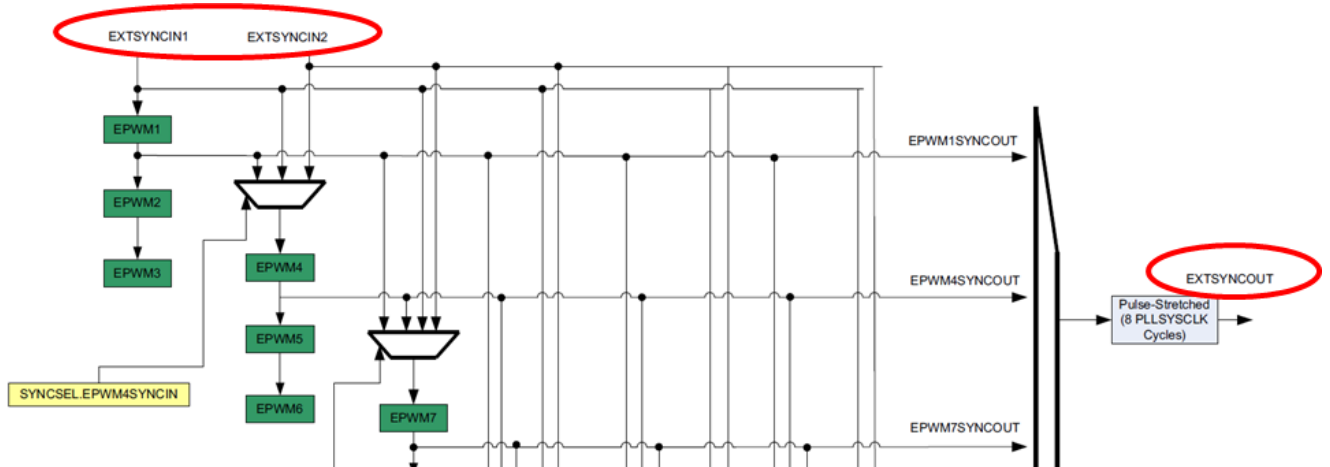


Figure 6. F28004x PWM 模块外部同步信号

EXTSYNIN1/2 为外部同步输入信号，如 Figure 6，该信号可以通过 input X-BAR 的 input5/6（如图 7）映射到芯片的 GPIO 上，从 PIN 脚上接收同步信号作为 EPWM1/4/7 的同步源。

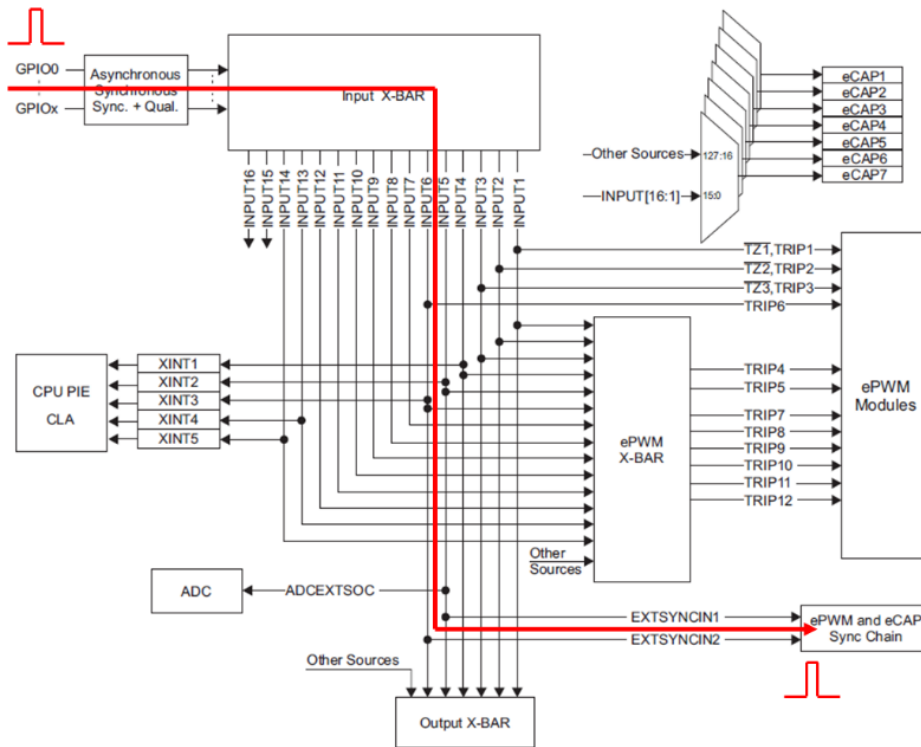


Figure 7. F28004x 输入 XBAR 及 EXTSYNIN 信号走向

Table 9-1. Input X-BAR Destinations

INPUT	DESTINATIONS
INPUT1	eCAPx, ePWM X-BAR, ePWM[TZ1,TRIP1], Output X-BAR
INPUT2	eCAPx, ePWM X-BAR, ePWM[TZ2,TRIP2], Output X-BAR
INPUT3	eCAPx, ePWM X-BAR, ePWM[TZ3,TRIP3], Output X-BAR
INPUT4	eCAPx, ePWM X-BAR, XINT1, Output X-BAR
INPUT5	eCAPx, ePWM X-BAR, XINT2, ADCEXTSOC, EXTSYNIN1, Output X-BAR
INPUT6	eCAPx, ePWM X-BAR, XINT3, ePWM[TRIP6], EXTSYNIN2, Output X-BAR

Figure 8. Input Xbar MUX 映射表

EXTSYNCOUT 为外部同步输出信号，如 Figure 6，该信号可以通过 MUX 选择模块 1/4/7 的同步输出信号作为触发源。一旦触发，将对外产生一个脉宽为 8 个 SYSCLK 的信号，并可以通过 Output X-BAR 映射到 PIN 脚上作为其它芯片的同步信号（EXTSYNCOUT 信号走向如 Figure 8）。

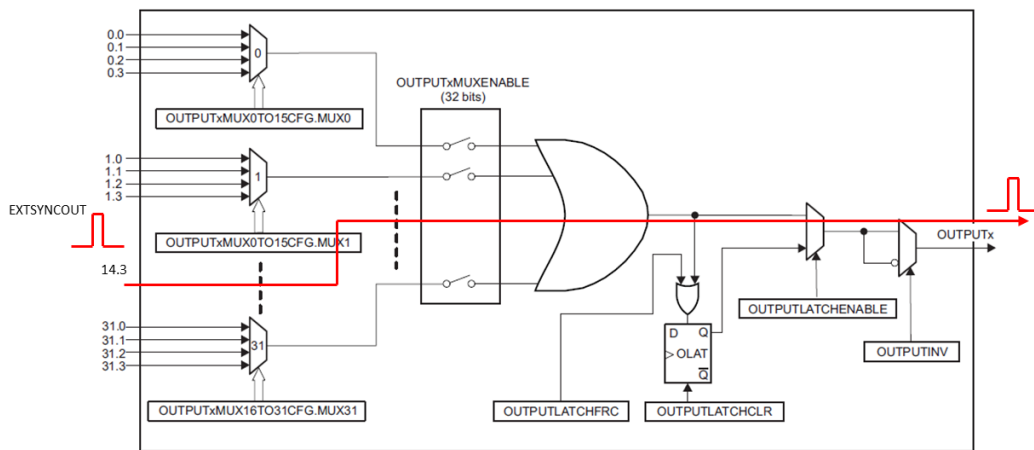


Figure 9. F28004x 输出 XBAR 及 EXTSYNOUT 信号走向

利用这两个对外同步信号，可以在系统内选取一个 C2000 作为同步的主机，通过 EXTSYNOUT 将 PWM1 的同步信号输出，其它芯片作为从机，通过 XBAR 将 GPIO 口上的同步信号映射到内部的 EXTSYNIN 作为 EPWM1 的同步信号，完成不同芯片之间 PWM1 的同步（如 figure 10）。再结合前文配置方法配置各内部的 PWM1/2/3 同步，即完成了系统内所有 PWM 的同步。

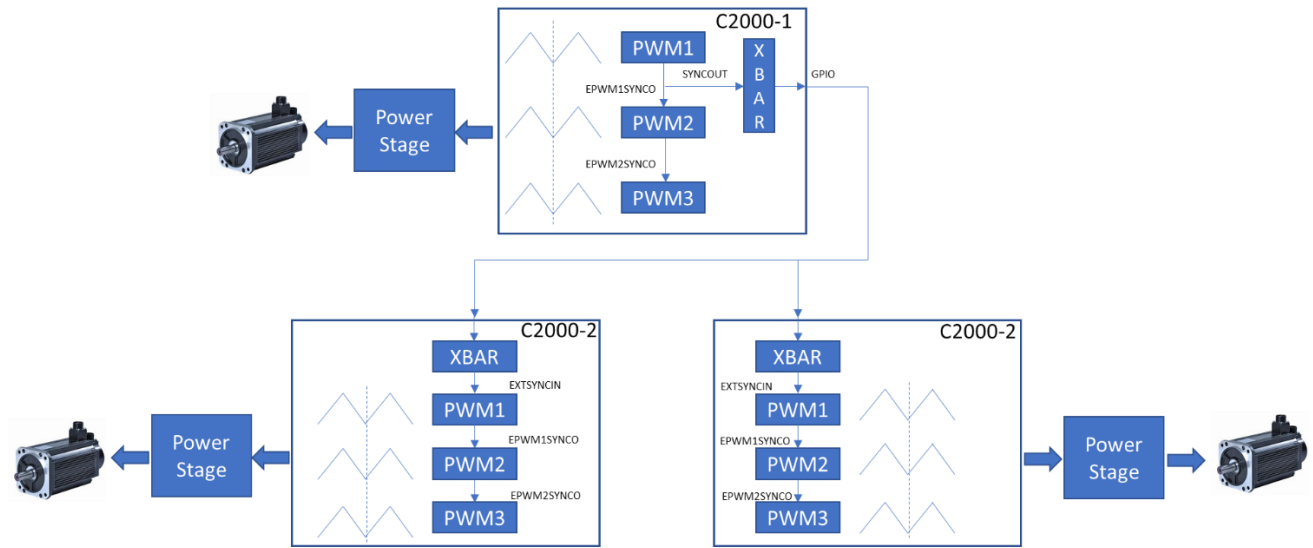


Figure 10. PWM 片间同步示意图

CPU1	EPWMxSYNCI	EPWMxSYNCO
EPWM1	N/A	CTR = ZERO SYNCOOUT = EPWM1SYNCO
EPWM2	EPWM1SYNCO	EPWM2SYNCI
EPWM3	EPWM2SYNCO	N/A

CPU2/3	EPWMxSYNCI	EPWMxSYNCO
EPWM1	EXTSYNINx	CTR = ZERO 或 EPWM1SYNCI
EPWM2	EPWM1SYNCO	EPWM2SYNCI
EPWM3	EPWM2SYNCO	N/A

Figure 11. 主机（MCU1）及从机（MCU2/3）PWM 片内同步配置

#### 4 主要参考代码

主机侧配置代码示例如下

//-----//

//主机侧SyncOUT输出配置:

```
//选取SyncOUT信号为PWM1同步输出信号
SysCtl_setSyncOutputConfig(SYSCTL_SYNC_OUT_SRC_EPWM1SYNCOUT);
//设置XBAR_OUTPUT3的信号源为MUX14的EXTSYNCOUT
XBAR_setOutputMuxConfig(XBAR_OUTPUT3, XBAR_OUT_MUX14_EXTSYNCOUT);
//设置GPIO5为标准推挽输出
GPIO_setPadConfig(5, GPIO_PIN_TYPE_STD);
//设置GPIO5为XBAR_OUTPUT3
GPIO_setPinConfig(GPIO_5_OUTPUTXBAR3);
```

//-----//

//PWM配置（省略周期、占空比等配置）:

```
void initEPWM1()
{
    // 设置PWM1生成同步输出信号
    EPWM_setSyncOutPulseMode(EPWM1_BASE, EPWM_SYNC_OUT_PULSE_ON_COUNTER_ZERO);
}
```

```
void initEPWM2()
{
    // 设置PWM2移相值，当PWM1产生的同步信号到来，计数器Load相移值
    EPWM_setPhaseShift(EPWM2_BASE, 0);

    EPWM_enablePhaseShiftLoad(EPWM2_BASE);
    // 设置SyncOut为SyncIn，即PWM1产生的同步信号bypassP PWM2直接输出；
    EPWM_setSyncOutPulseMode(EPWM2_BASE, EPWM_SYNC_OUT_PULSE_ON_EPWMxSYNCIN);
    // 设置同步后的Counter方向；
    EPWM_setCountModeAfterSync(EPWM2_BASE, EPWM_COUNT_MODE_UP_AFTER_SYNC);
}
```

```
void initEPWM3(void)
{
    // 设置PWM3移相值
    EPWM_setPhaseShift(EPWM3_BASE, 0U);
    // 使能移相装载
    EPWM_enablePhaseShiftLoad(EPWM3_BASE);
    // 设置同步后的Counter方向；
    EPWM_setCountModeAfterSync(EPWM3_BASE, EPWM_COUNT_MODE_UP_AFTER_SYNC);
}
```

从机侧配置代码示例如下:

//-----//

//从机侧SyncIN输出配置:

```
// 设置GPIO5 通过XBAR input5接收脉冲作为EXTSYNCIN信号;
```



```

GPIO_setPadConfig(5, GPIO_PIN_TYPE_PULLUP);
GPIO_setPinConfig(GPIO_5_GPIO5);
GPIO_setDirectionMode(5, GPIO_DIR_MODE_IN);
GPIO_setQualificationMode(5, GPIO_QUAL_ASYNC);
XBAR_setInputPin(XBAR_INPUT5, 5);

// 设置EXTSYNCIN1信号作为EPWM1的同步信号;
SysCtl_setSyncInputConfig(SYSCTL_SYNC_IN_EPWM1, SYSCTL_SYNC_IN_SRC_EXTSYNCIN1);

//-----//
//PWM配置（省略周期、占空比等配置）：
void initEPWM1()
{
    // 设置同步后Counter方向
    EPWM_setCountModeAfterSync(EPWM1_BASE, EPWM_COUNT_MODE_UP_AFTER_SYNC);
    // 设置SyncOut为SyncIn, 即接受的外部同步信号bypass PWM1直接输出;
    EPWM_setSyncOutPulseMode(EPWM1_BASE, EPWM_SYNC_OUT_PULSE_ON_EPWMxSYNCIN);
    // 使能相位装载;
    EPWM_enablePhaseShiftLoad(EPWM1_BASE);
}

void initEPWM2()
{
    // 设置PWM2移相值, 当PWM1产生的同步信号到来, 计数器Load相移值
    EPWM_setPhaseShift(EPWM2_BASE, 0);
    EPWM_enablePhaseShiftLoad(EPWM2_BASE);
    // 设置SyncOut为SyncIn, 即PWM1产生的同步信号bypass PWM2直接输出;
    EPWM_setSyncOutPulseMode(EPWM2_BASE, EPWM_SYNC_OUT_PULSE_ON_EPWMxSYNCIN);
    // 设置同步后的Counter方向;
    EPWM_setCountModeAfterSync(EPWM2_BASE, EPWM_COUNT_MODE_UP_AFTER_SYNC);
}

void initEPWM3(void)
{
    // 设置PWM3移相值
    EPWM_setPhaseShift(EPWM3_BASE, 0U);
    // 使能移相装载
    EPWM_enablePhaseShiftLoad(EPWM3_BASE);
    // 设置同步后的Counter方向;
    EPWM_setCountModeAfterSync(EPWM3_BASE, EPWM_COUNT_MODE_UP_AFTER_SYNC);
}

```

## 5 测试结果

基于以上实现方式，用两片 F280049 Launchpad 进行 PWM 同步效果验证。其中芯片#1 作为主机产生同步的 PWM1-4，芯片#2 作为从机产生 PWM1-2，同步信号通过信号线在板间连接，通过逻辑分析仪测试结果如 figure 12。

起始时刻芯片 1、2 独立运行，PWM 相位差随机不可控，在采样窗口期间容易被另一芯片所产生的 PWM 信号及功率级的开关噪声干扰，造成多轴系统采样精度及控制性能的下降；

在图示 Sync Signal 时刻，使能芯片 1 的对外同步信号输出，可以看到在 PWM 上升沿产生了一个同步信号。该同步信号通过信号线输入芯片 2，芯片 2 PWM1/2 立刻得到了同步，从而实现了系统内所有 PWM Counter 的同步。

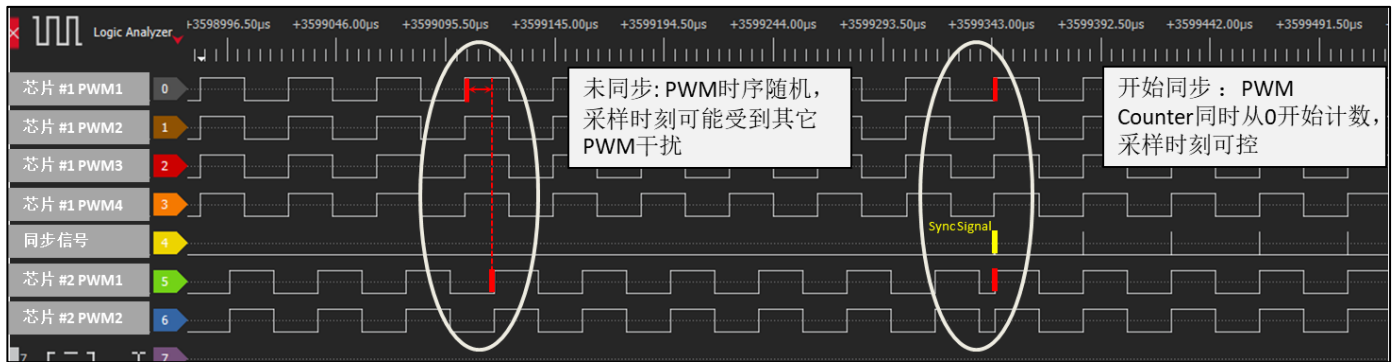


Figure 12. 同步逻辑分析仪测试结果

通过示波器，图 13 中 CH1/2 为芯片 1 PWM 信号，CH3 为芯片 2 PWM 信号，CH4 为同步信号，可见开始 3 个 PWM 信号不同步。在图中圈出的时刻使能同步输出，CH3 信号立刻得到了同步。

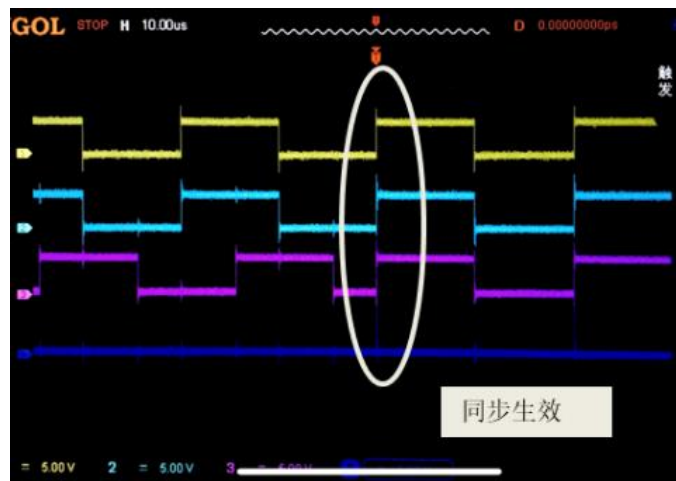


Figure 13. 同步逻辑分析仪测试结果

通过示波器，图 14 中 CH1/2 为芯片 1 PWM 信号，CH3 为芯片 2 PWM 信号，CH4 为同步信号，可见开始 3 个 PWM 信号不同步。在图中圈出的时刻使能同步输出，CH3 信号立刻得到了同步。

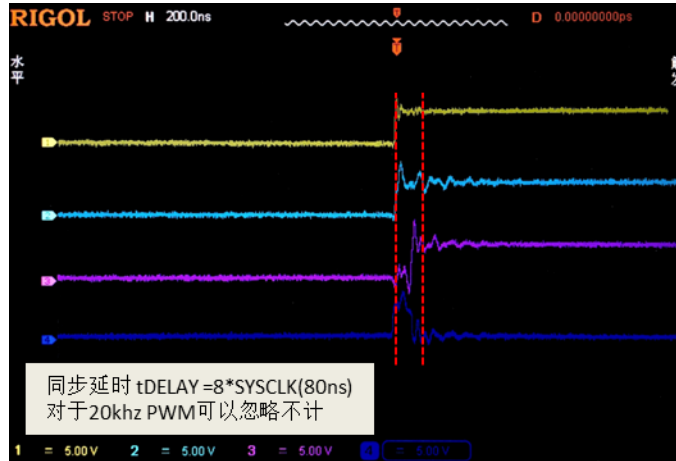


Figure 14. PWM 片间同步示波器测试结果

C2000 作为专业的实时控制 MCU，其外设在设计之初就特别针对电机驱动、数字电源等电力电子应用的控制需求进行了特殊考虑。以本需求为例，PWM 模块灵活的同步机制以及片间同步功能极大地方便了多轴电机驱动、多级电源系统中多片 MCU PWM 模块相互同步，有助于广大工程师优化此类产品性能。

## 6 参考文献

1. *TMS320F28004x Real-Time Microcontrollers Technical Reference Manual (Rev. F) (SPRUI33F)*
2. *TMS320F28003x Real-Time Microcontrollers Technical Reference Manual (Rev. A) (SPRUIW9A)*

## 重要声明和免责声明

TI“按原样”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功能安全、信息安全、监管或其他要求。

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](#) 或 [ti.com](#) 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。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2023，德州仪器 (TI) 公司