

# AN14470

## 如何使用FlexIO状态模式生成中心对齐的PWM

第1.0版—2025年2月6日

应用笔记

### 文档信息

信息	内容
关键词	AN14470、MCXN、FlexIO、PWM
摘要	本应用笔记描述了如何在MCX N系列MCU上使用FlexIO的状态模式生成中心对齐的PWM波形。



## 1 介绍

本应用笔记描述了如何在MCX N系列MCU上使用FlexIO状态模式生成中心对齐的PWM波形。

MCX N系列MCU有八个FlexIO移位器，可以实现多达八种状态。在每种状态下，可以在多达八个FlexIO数据引脚上输出不同的逻辑电平。而状态之间的转换则可以通过多达八个FlexIO定时器来控制。

使用外部输入引脚控制状态转换是FlexIO的一种常见演示方法。然而，在很多情况下，用户只希望使用由定时器序列控制的状态机，而不考虑外部输入状态。本应用笔记正是基于这样的出发点，使用定时器来控制由移位器组成的状态机，以生成中心对齐的PWM波形。

恩智浦提供的配置工具能够快速生成FlexIO的功能代码。本应用的一个目的就是演示如何使用这一配置工具来实现FlexIO的状态机功能，使得这个实现过程简单又快捷。

## 2 FlexIO模块的状态模式

FlexIO模块的状态模式使您能够创建一个具有多达八种状态的硬件状态机。每种状态使用三个可选择的输入和八个专用输出。与软件方法相比，这种状态机能够显著减轻CPU的负担。

### 2.1 FlexIO状态的I/O分配

图1展示了一般情况下某一给定状态的I/O分配。

- 某个状态的三个输入可以由一组三个FlexIO引脚（从SHIFTCTL<sub>n</sub>[PINSEL]值开始）或由下一个(<sub>n+1</sub>)移位器的三个LSB位表示。使用SHIFTCFG<sub>n</sub>[INSRC]移位器配置寄存器中的输入源选择字段INSRC来选择。
- 当选择FlexIO引脚作为输入时，在SHIFTCTL<sub>n</sub>[PINPOL]移位器控制寄存器中配置输入的PINPOL极性。
- 状态的八个输出是固定的，并分配给FlexIO引脚FXIO\_D0-FXIO\_D7（所有状态共享相同的输出）。通过在SHIFTCFG<sub>n</sub>移位器配置寄存器中的PWDTH[3:0]、SSTOP[1:0]和SSTART[1:0]字段中设置适当的位，可以部分禁用FXIO\_D0-FXIO\_D7的输出。

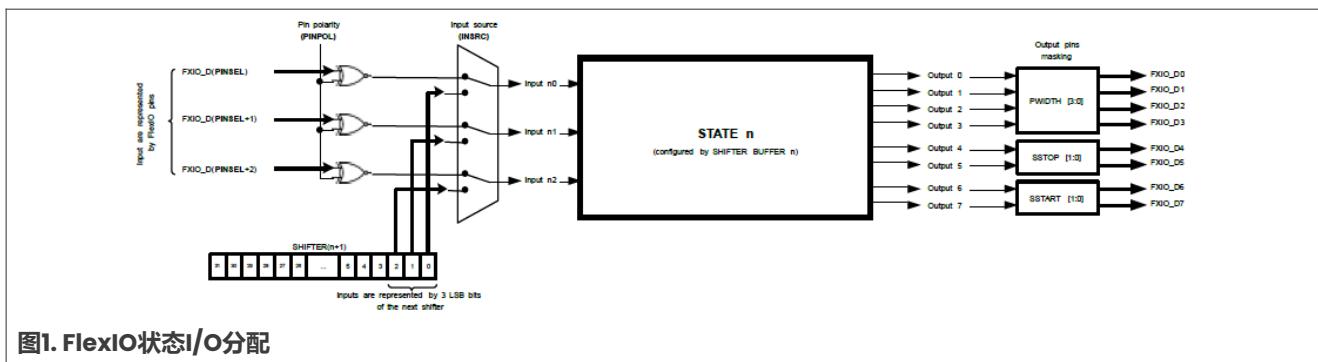


图1. FlexIO状态I/O分配

### 2.2 FlexIO状态配置

当SHIFTCTL<sub>n</sub>[SMOD]移位器控制寄存器中的SMOD字段设置为0x6时，即选择了状态模式。

表1. 状态模式

字段	功能
2-0 SMOD	<p>移位器模式 配置移位器的模式</p> <p>000b - 禁用</p> <p>001b - 接收模式；在定时器到期时，将当前移位器内容捕获到SHIFTBUF中</p> <p>010b - 发送模式；在定时器到期时，将SHIFTBUF内容加载到移位器中</p> <p>011b - 保留</p> <p>100b - 匹配存储模式；在定时器到期时，将移位器数据与SHIFTBUF内容进行比较</p> <p>101b - 连续匹配模式；移位器数据持续与SHIFTBUF内容进行比较</p> <p>110b - 状态模式；SHIFTBUF内容存储可编程状态属性</p> <p>111b - 逻辑模式；SHIFTBUF内容实现可编程逻辑查找表</p>

SHIFTSTATE移位器状态寄存器定义了在选择状态模式并启用FlexIO模块时的当前状态。默认情况下（复位后），其值为0x0。如果移位器0未配置为状态模式，应对其进行初始化，以避免不正确的状态机启动。

表2. 移位器状态

字段	功能
2-0 STATE	当前状态指针。维护一个指针以跟踪当前启用的移位器（配置为状态模式），负责驱动输出并计算下一个状态。在状态指针正在更新时读取此寄存器，可能会导致返回不正确的状态。写入此字段的值，会覆盖当前状态。

当正确启动时，SHIFTSTATE指向当前状态，该状态由移位器缓冲区n寄存器（SHIFTERBUFFn）的值定义。其32位值包含当前状态输出的配置（SHIFTBUFF[31:24]）和下一状态的选择（SHIFTBUFF[23:0]）。

移位器缓冲区寄存器的值的24个最低有效位（LSB）表示八组，每组三位。每组中的三位定义下一个状态的值。根据输入组合选择对应的组（即下一个状态）。如果输入组合为000，则SHIFTBUFF[2:0]位定义的值表示下一状态。如果输入组合为011，则SHIFTBUFF[11:9]位定义的值表示下一状态。

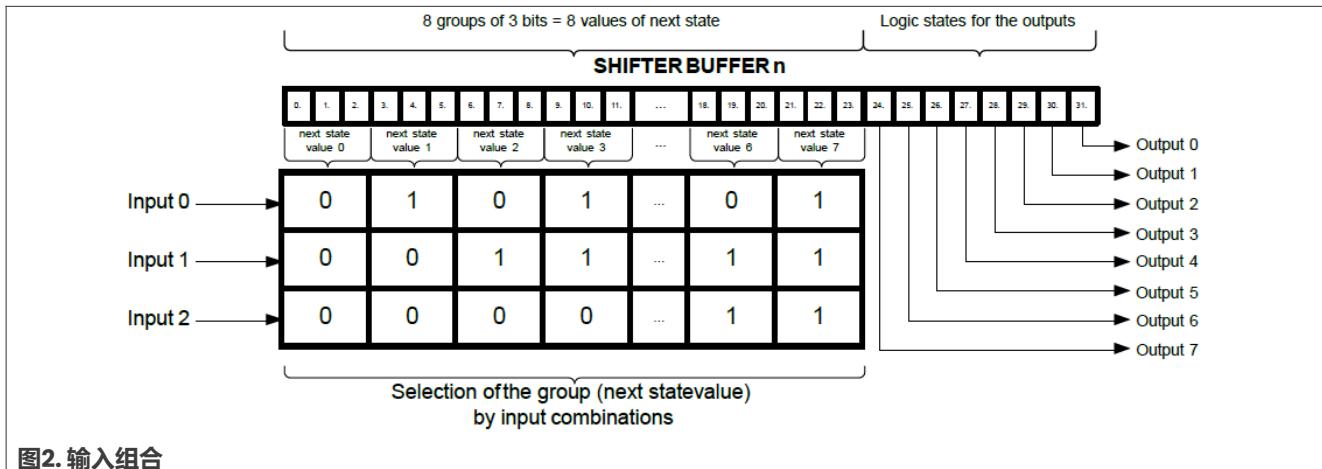


图2. 输入组合

表3. 移位器缓冲区

字段	功能
31-0 SHIFTBUF	<p>移位器缓冲区 包含要与移位器内容匹配的数据，并根据SHIFTCTLn[SMOD]的设置用于各种其他功能：</p> <ul style="list-style-type: none"> <li>如果SHIFTCTL0[SMOD]为1b（接收模式），在定时器到期时，移位器数据会被传输到SHIFTBUF中。必须仅在对应的SHIFTSTAT[SSF]标志被置位（表示有新的移位器数据可用）时读取此寄存器。</li> <li>如果SHIFTCTL0[SMOD]为10b（发送模式），在定时器启动前，SHIFTBUF中的数据会被传输到移位器中。</li> <li>如果SHIFTCTL0[SMOD]为100b（匹配存储模式），SHIFTBUF[31:16]包含要与移位器内容匹配的数据，SHIFTBUF[15:0]可用于屏蔽匹配结果（1表示屏蔽，0表示不屏蔽）。在定时器到期时检查匹配情况。每当发生匹配事件时，移位器数据[31:16]会被写入SHIFTBUF[31:16]。必须仅在对应的移位器状态标志被置位（表示有新的移位器数据可用）时读取此寄存器。</li> <li>如果SHIFTCTL0[SMOD]为101b（连续匹配模式），SHIFTBUF[31:16]包含要与移位器内容匹配的数据，SHIFTBUF[15:0]可用于屏蔽匹配结果（1表示屏蔽，0表示不屏蔽）。</li> <li>如果SHIFTCTL0[SMOD]为111b（逻辑模式），SHIFTBUF[31:0]实现一个5输入、32位的可编程逻辑查找表。</li> <li>如果SHIFTCTL0[SMOD]为110b（状态模式），当此移位器被当前状态指针选中时，使用SHIFTBUF[31:24]来驱动输出值，并使用SHIFTBUF[23:0]来配置下一状态转换的值。</li> </ul>

## 2.3 FlexIO状态转换

下一个状态转换会在当前状态所选的定时器到期时生成。这取决于移位器控制寄存器SHIFTCTLn[TIMSEL]中的TIMSEL定时器选择值。

当移位器状态寄存器SHIFTSTATE中的当前状态指针STATE指向对应的移位器时，会设置一个适当的移位器状态标志。

使用这个状态标志来触发中断/DMA。切换到特定状态以触发CPU/DMA，甚至强制某个定时器激活FlexIO输出触发（例如触发ADC转换的启动）。当选择一个不同的状态时，该标志会被清除。

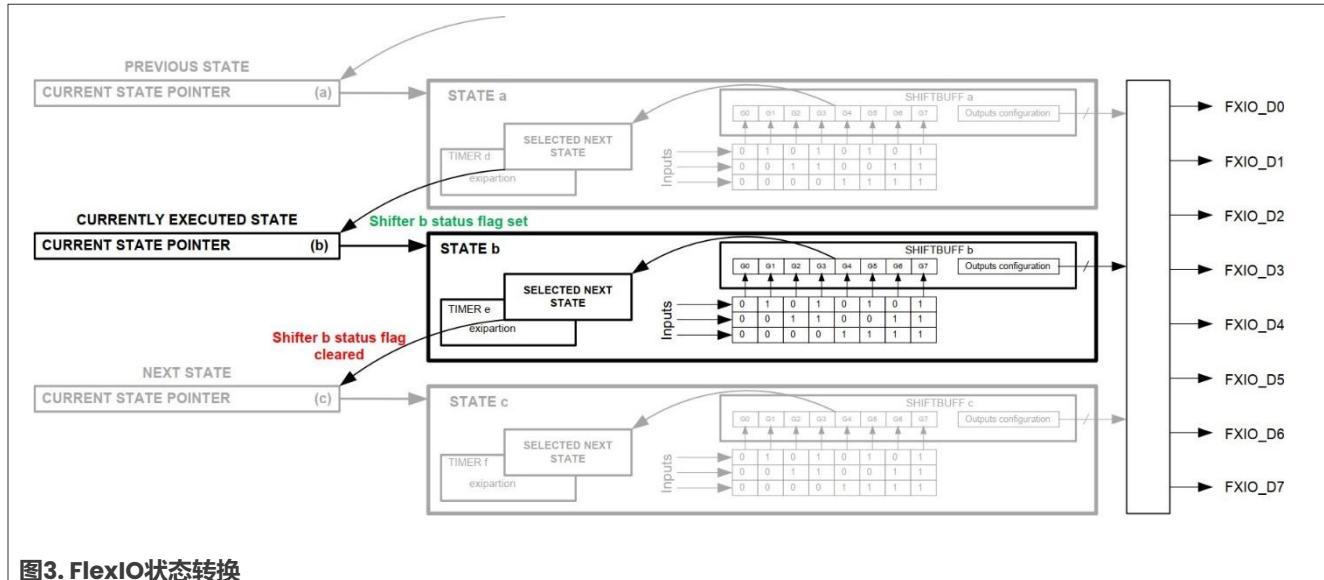


图3. FlexIO状态转换

### 3 如何在状态模式下实现中心对齐PWM

在电机控制中，将上桥臂和下桥臂视为一对互补的PWM波形，这对波形是中心对齐的，并且带有死区时间。一个周期的互补PWM波形信号可以分为五个状态：状态0、状态1、状态2、状态3和状态4。

FlexIO最多有八个状态，每个状态可以控制多达八个IO输出。这足以生成一对互补的PWM信号。

#### 3.1 使用状态机实现PWM

[图4](#)定义了PWM信号每次变化对应的状态，每个状态的持续时间由FlexIO定时器控制。

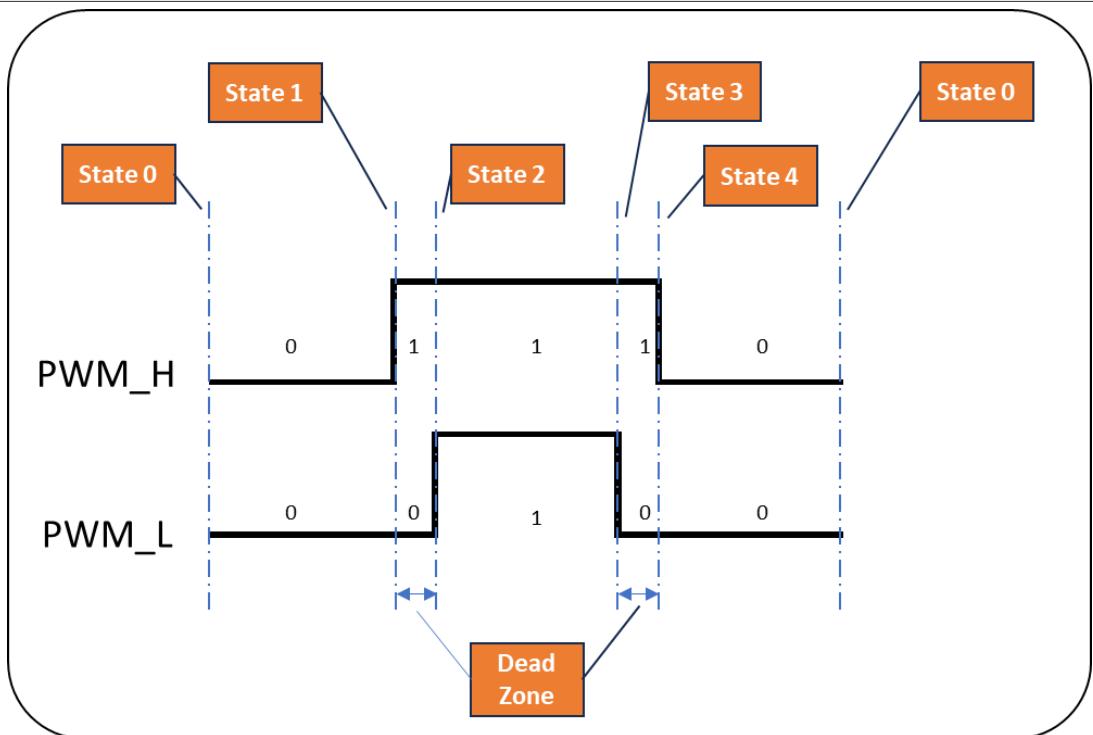


图4. 使用状态机实现的PWM

### 3.2 状态机上的GPIO配置

P0\_8引脚用于输出PWM\_H信号的波形，而P0\_9引脚则被选择用于输出PWM\_L信号的波形。

表4. 状态机上的GPIO配置

状态	PWM_H(P0_8)值	PWM_L(P0_9)值
状态0	0	0
状态1	1	0
状态2	1	1
状态3	1	0
状态4	0	0

### 3.3 内部定时器波形的输出观察

内部定时器(0/1/2/3/4)被用作状态转换的计时基准。为了更好地描述状态机的运行情况，此应用中输出了内部定时器的实际波形。

表5. 内部定时器波形的输出观察

定时器	GPIO引脚
定时器0	P4_16
定时器1	P4_17
定时器2	P4_18

表5. 内部定时器波形的输出观察 (续)

定时器	GPIO引脚
定时器3	P4_19
定时器4	P4_20

## 4 FlexIO移位器配置

此应用将FlexIO移位器配置为状态机模式，在该模式下，不同的状态机控制不同FlexIO数据线的输出电平。

### 4.1 本应用中使用的FlexIO状态模式

在本应用中，输入并未连接到芯片的引脚焊盘。因此，所有三个输入都可以视为逻辑低电平，并且在所有移位器中都选择了状态组0。

每个移位器只有一个状态组 (Group0) 可供使用。本应用使用了五个移位器，因此共有五种状态。

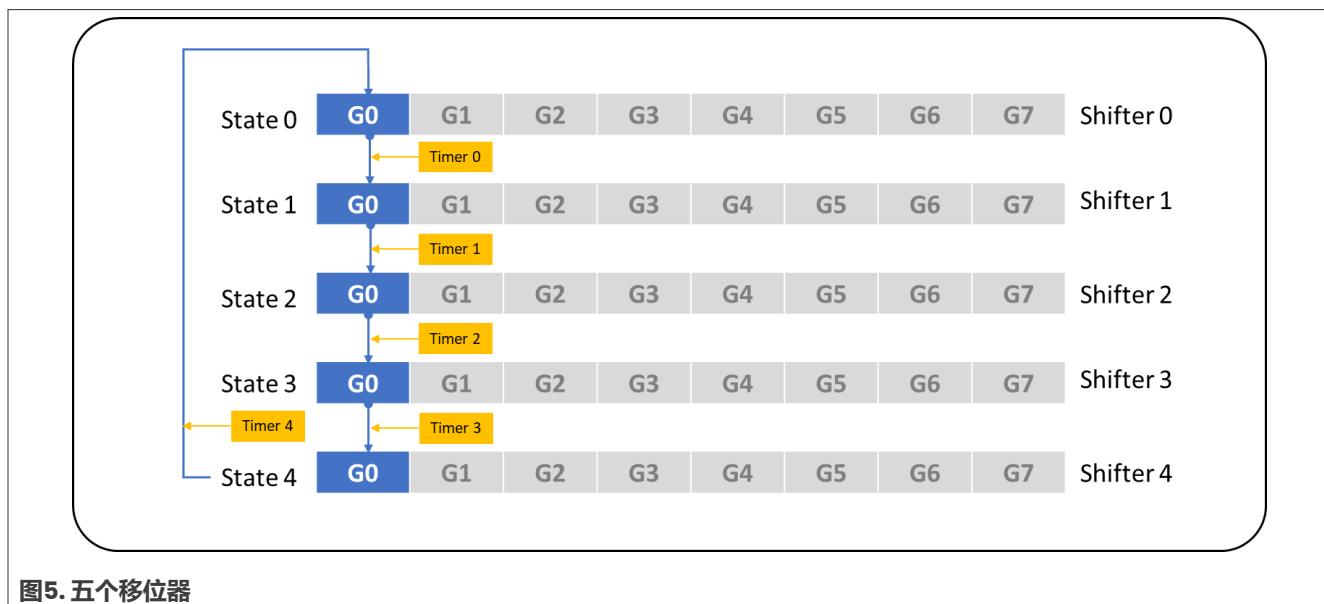


图5. 五个移位器

### 4.2 FlexIO状态模式配置

当移位器处于状态模式时，主要考虑以下几点：

1. 选择哪个定时器来触发状态转换。
2. 定时器触发电平的极性选择。
3. 使用三个引脚作为外部输入，以确定选择哪种状态。
4. 是否启用输出以及输出的极性。
5. 输出引脚的选择以及是否屏蔽某些引脚。
6. 下一状态的选择。

[表6](#)列出了本应用的相关配置。

表6. FlexIO状态模式配置

-	移位器0	移位器1	移位器2	移位器3	移位器4
模式	状态模式	状态模式	状态模式	状态模式	状态模式
定时器选择	定时器0	定时器1	定时器2	定时器3	定时器4
定时器极性	正极性	正极性	正极性	正极性	正极性
引脚选择	FlexIO_D16	FlexIO_D16	FlexIO_D16	FlexIO_D16	FlexIO_D16
引脚输出	启用	启用	启用	启用	启用
引脚极性	高电平有效	高电平有效	高电平有效	高电平有效	高电平有效
输出引脚	D0_LOW、 D1_HIGH	D0_LOW、 D1_LOW	D0_HIGH、 D1_LOW	D0_LOW、 D1_LOW	D0_LOW、 D1_HIGH
屏蔽引脚	D2~D7	D2~D7	D2~D7	D2~D7	D2~D7
下一状态移位器	移位器1	移位器2	移位器3	移位器4	移位器0

## 5 FlexIO定时器配置

FlexIO定时器具有以下模式：

1. 定时器8位波特计数器模式：低8位用于配置移位时钟的波特率，高8位用于配置传输中移位时钟的边沿数量。  
*注：在某些电机应用中，8位精度可能不够，因此需要使用16位来生成移位时钟的波特率。*
2. 定时器8位高电平PWM模式：低8位用于配置定时器输出高电平的周期，高8位用于配置定时器输出低电平的周期。
3. 定时器16位计数器模式：使用16位计数器来配置移位时钟的波特率。  
*注：本应用使用的是定时器16位计数器模式。由于定时器数量有限，PWM的数量不再通过额外的定时器来控制，而是采用在特定条件下启用或禁用定时器的方法。*
4. 其余定时器模式也不适用。详情请参阅参考手册。

### 5.1 本应用中的定时器配置

当定时器用作移位器状态机的时间触发信号时，请考虑以下配置：

1. 选择定时器的触发源和极性。
2. 要观察定时器内部的波形，可选择其输出引脚和极性。
3. 选择启用/禁用/重置定时器的条件。
4. 设置初始输出电平，以及是否由重置信号控制。
5. 配置比较寄存器并设置波特率。

[表7](#)是定时器0/1/2/3/4的配置表。

表7. 定时器配置

-	定时器0	定时器1	定时器2	定时器3	定时器4
<b>模式</b>	<b>单一16位计数器模式</b>				
触发源	定时器4触发	FlexIO_D1引脚	定时器4触发	FlexIO_D0引脚	-
触发极性	高电平有效	低电平有效	低电平有效	低电平有效	-
引脚选择	FlexIO_D24	FlexIO_D25	FlexIO_D26	FlexIO_D27	FlexIO_D28

表7. 定时器配置 (续)

-	定时器0	定时器1	定时器2	定时器3	定时器4
<b>模式</b>	<b>单一16位计数器模式</b>				
引脚输出	启用	启用	启用	启用	启用
引脚极性	高电平有效	高电平有效	高电平有效	高电平有效	高电平有效
启用条件	触发上升沿	触发上升沿	触发上升沿	触发上升沿	始终启用
禁用条件	触发下降沿	触发下降沿	触发下降沿	触发下降沿	永不禁用
重置条件	永不重置	永不重置	永不重置	永不重置	永不重置
初始输出	0	0	0	0	1
CMP寄存器值	49	4	149	4	199
波特率	1.5Mbit/s	15Mbit/s	500kbit/s	15Mbit/s	375kbps

## 5.2 定时器触发和状态转换时序

如果仅使用一个定时器来为移位器生成触发波形，则无法控制该定时器产生的PWM波形的数量。在本应用中，采用启用和禁用定时器的方法，让定时器在特定时间范围内生成PWM波形。然后，基于触发源仅在特定条件下有效的原则，确保准确的触发时间，并且所生成的多余的PWM波形周期不会影响其状态。

图6显示了状态转换的所有时间点以及对应的定时器波形。

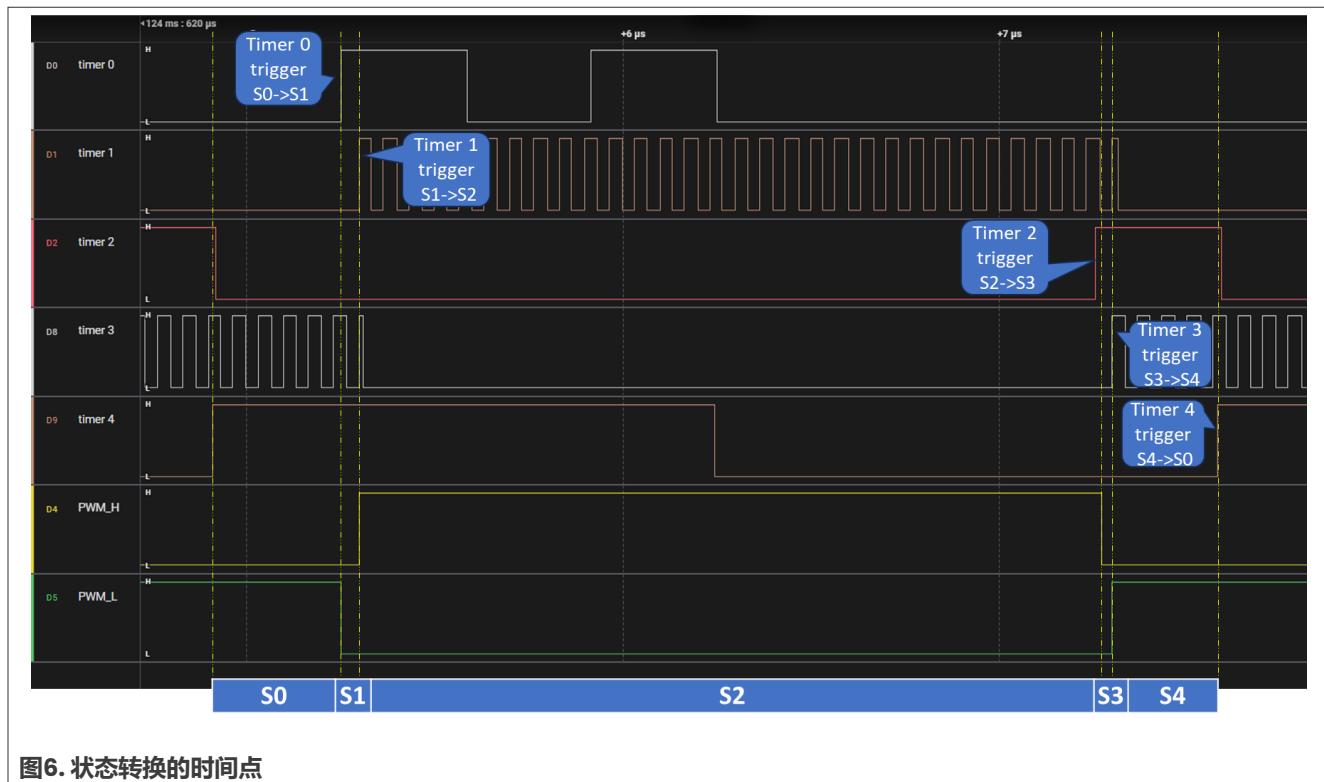


图6. 状态转换的时间点

- 图7显示了从状态0到状态1的状态转换过程，由定时器0触发。定时器0生成的PWM波形由定时器4控制，以启用和禁用。定时器0的第一个上升沿触发状态转换。当进入状态1后，定时器0的后续上升沿不会触发状态转换。

但为了避免产生更多干扰，定时器0会在定时器4的上升沿被禁用。

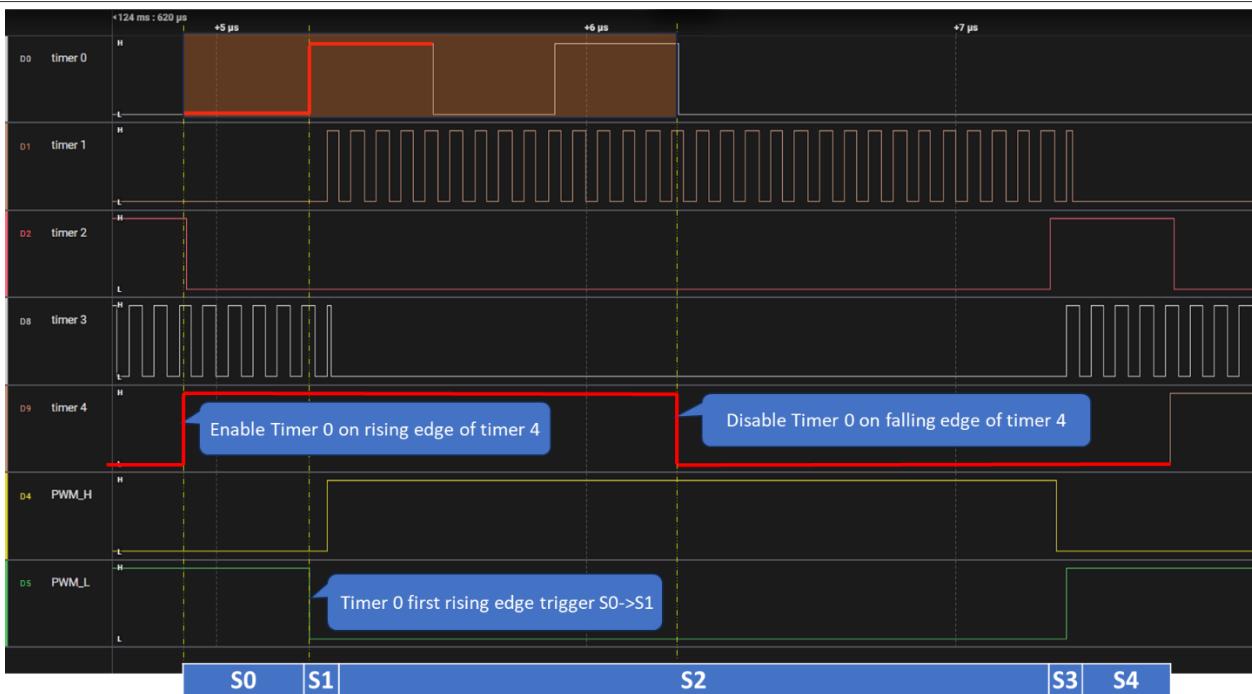


图7. 从状态0到状态1的转换

2. 图8显示了定时器1的波形和状态触发转换。在PWM\_L的下降沿启用定时器1，在上升沿禁用定时器1。定时器1的第一个上升沿触发从状态1到状态2的转换。后续上升沿不会产生影响。

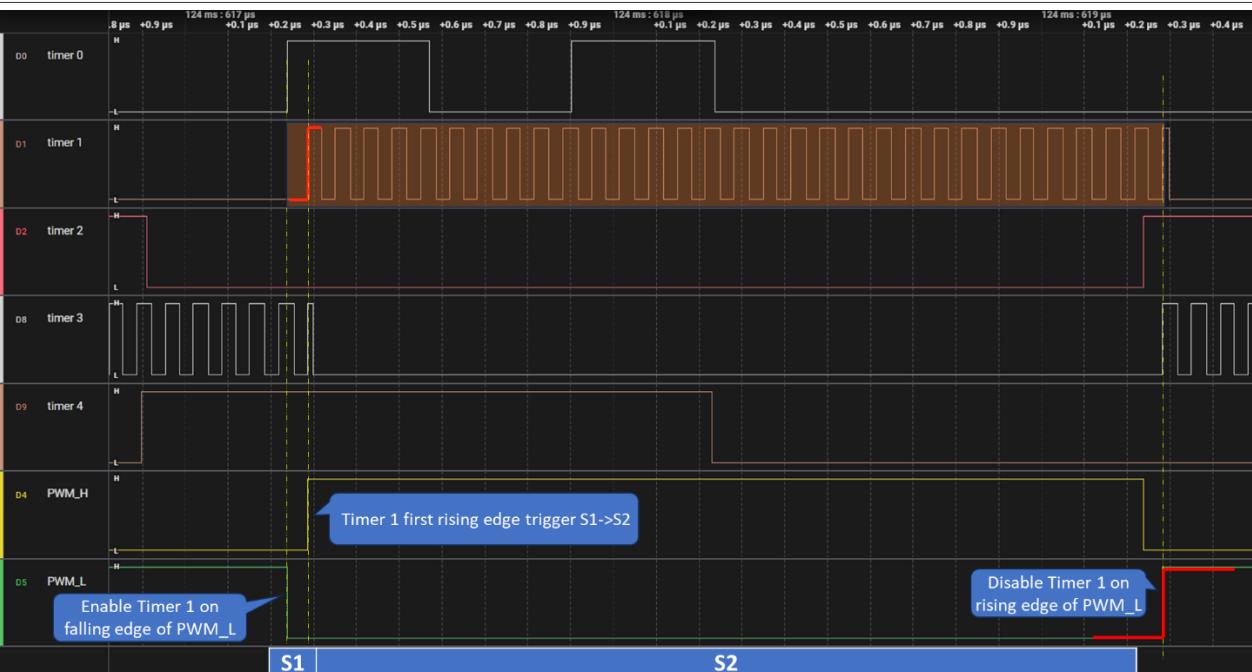


图8. 定时器1和状态触发转换

3. 图9显示了定时器2的波形和从状态2到状态3的转换。定时器2的周期小于定时器4周期的一半。定时器4的下降沿启用定时器2，而定时器4的上升沿禁用定时器2，确保定时器2的PWM位于定时器4周期的下半部分。定时器2没有完成一个完整周期，因为其下半部分被定时器4部分“截断”。定时器2仅生成一个上升沿，该上升沿触发从状态2到状态3的转换。

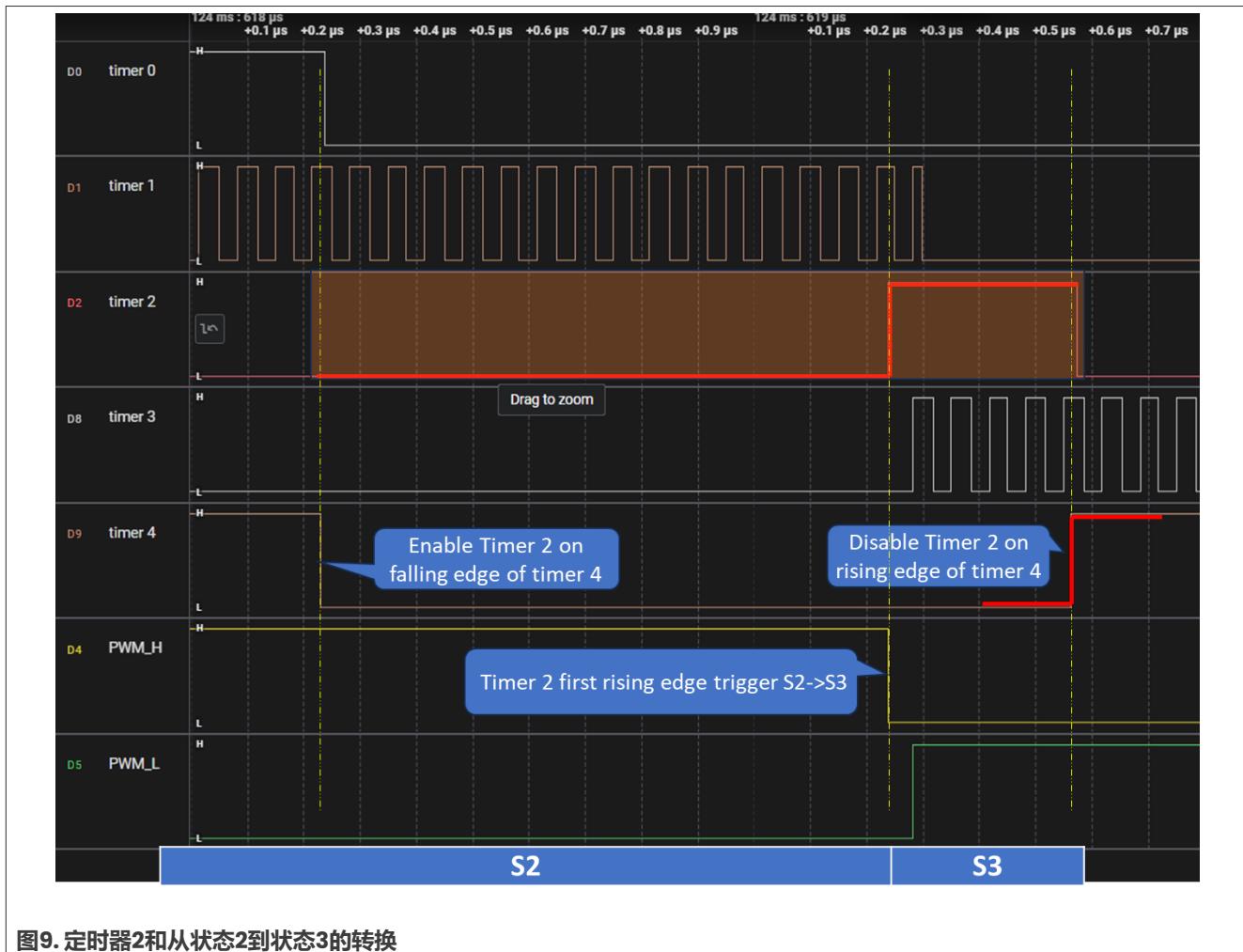


图9. 定时器2和从状态2到状态3的转换

4. 图10显示了定时器3的波形和从状态3到状态4的转换。PWM\_H的下降沿启用定时器3，而上升沿禁用定时器3。定时器3的第一个上升沿触发从状态3到状态4的转换。后续上升沿不会产生影响。

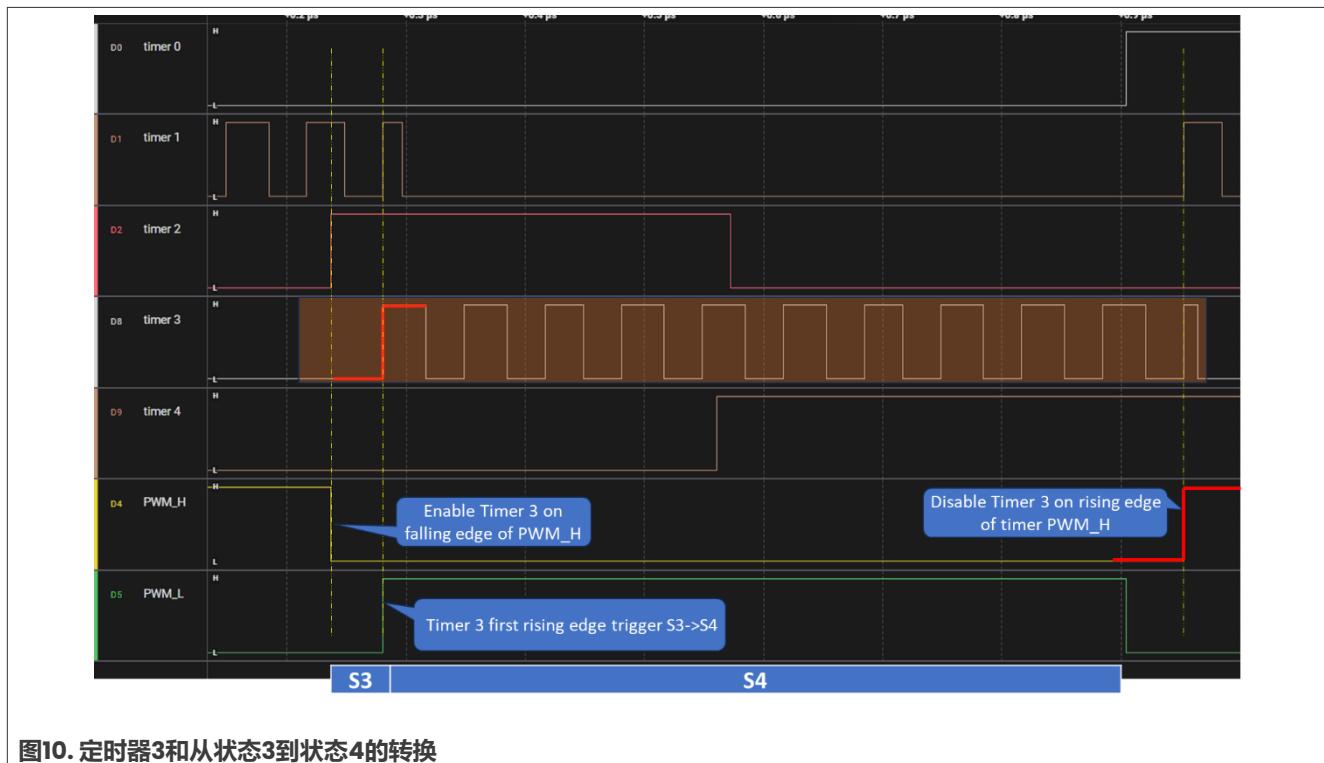


图10. 定时器3和从状态3到状态4的转换

5. 图11显示了定时器4的波形和从状态4到状态0的转换。定时器4的下降沿触发从状态4到状态0的转换。定时器4的周期与PWM周期同步。

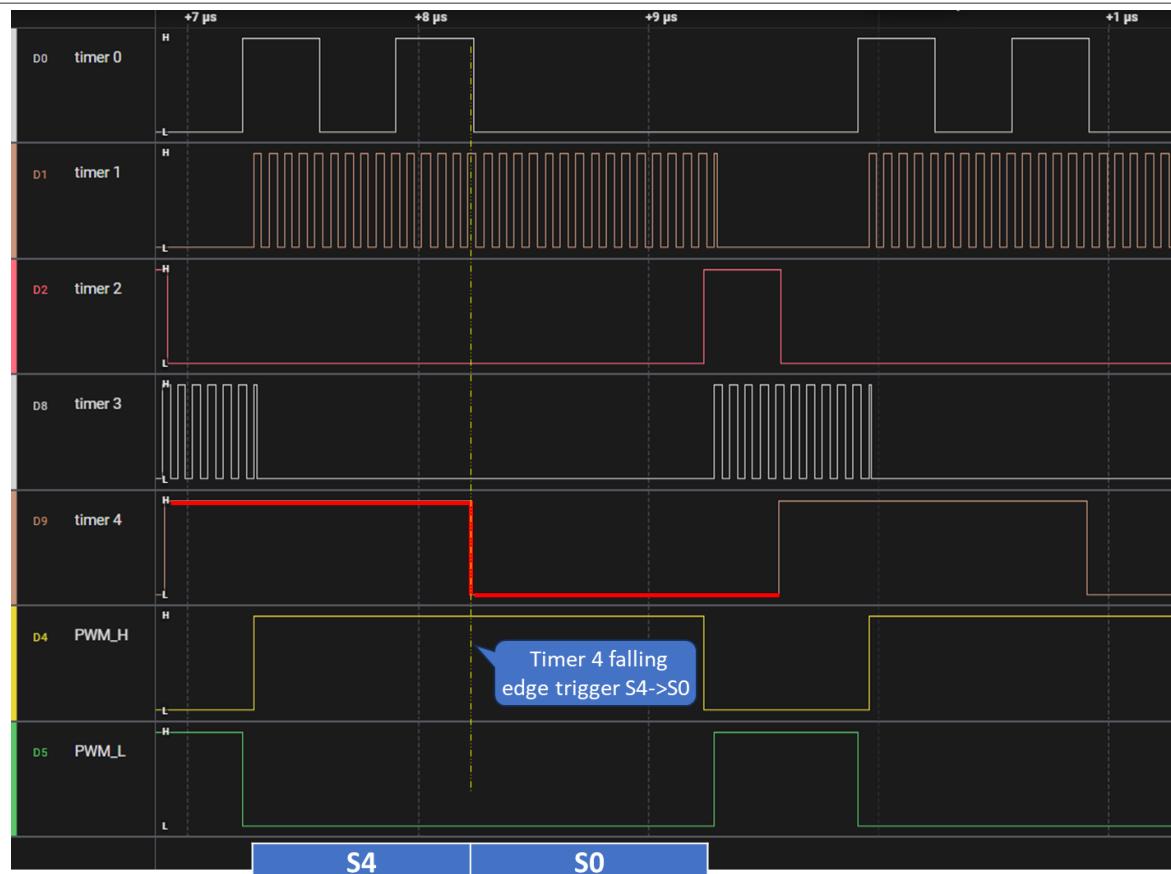


图11. 定时器4和从状态4到状态0的转换

总之，每个定时器生成的波形都是PWM波形的片段。它们的“生命周期”由定时器4、PWM\_H和PWM\_L的波形控制。状态转换主要由每个定时器的第一个上升沿触发。后续上升沿不会产生显著影响。

## 6 MCUXpresso Config Tools在本应用中的使用

MCUXpresso Config Tools套件是一套评估和配置工具，能够帮助用户从初步评估到生产软件开发的全过程。

使用MCUXpresso Config Tools，可以配置恩智浦Cortex-M处理器，并生成SDK驱动程序的初始化代码。

MCUXpresso Config Tools用途广泛，旨在帮助硬件设计人员、软件工程师、嵌入式工程师和现场应用工程师（FAE）。

该工具免费分发。Windows、Linux或Mac的安装程序可以从<http://www.nxp.com.cn>下载。

要编译生成的代码，需要MCUXpresso SDK软件包。可以从<http://mcuxpresso.nxp.com>下载。SDK软件包包含许多示例工程，可以帮助您快速入门。

### 6.1 FlexIO状态机

FlexIO状态机是一个复杂的组合系统。使用配置工具非常简单方便。生成的代码基于寄存器配置，更加直观且高效。

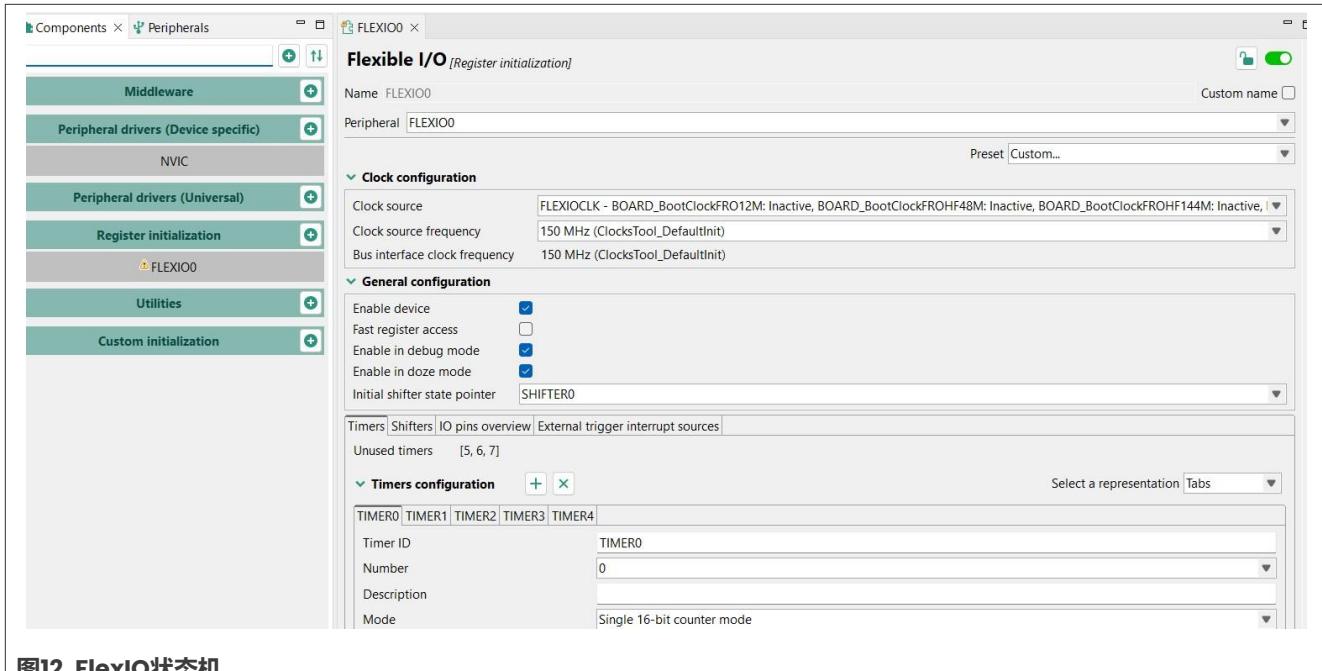


图12. FlexIO状态机

```

/* Pin output register initialization */
#ifndef FLEXIO0_PINOUTD_INIT
    FLEXIO0->PINOUTD = FLEXIO0_PINOUTD_INIT;
#else
    FLEXIO0->PINOUTD = 0x0U;
#endif /* FLEXIO0_PINOUTD_INIT */
#ifndef FLEXIO0_PINOUTD_OUTD_MASK
#endif /* FLEXIO0_PINOUTD_OUTD_MASK */

#ifndef FLEXIO0_CTRL_INIT
    FLEXIO0->CTRL = (FLEXIO0_CTRL_INIT & ~FLEXIO_CTRL_SWRST_MASK);
#else
    FLEXIO0->CTRL = 0x0U;
#endif /* FLEXIO0_CTRL_INIT */
}

/*****************
 * Initialization functions
 *****************/
static void BOARD_InitPeripherals_CommonPreInit(void)
{
    /* Enable clock gate of the FLEXIO0 peripheral. */
    SYSCON0->AHBCLKCTRL2 |= SYSCON_AHBCLKCTRL2_FLEXIO_MASK;
}

void BOARD_InitPeripherals(void)
{
    /* Common pre-initialization */
    BOARD_InitPeripherals_CommonPreInit();
    /* Initialize components */
    FLEXIO0_init();
}

/*****************
 * BOARD_InitBootPeripherals function
 *****************/

```

```
void BOARD_InitBootPeripherals(void)
{
    BOARD_InitPeripherals();
}
```

## 6.2 MCUXpresso Config Tools中的定时器设置

定时器的设置主要包括何时启用、何时禁用以及何时重置。还包括输出来自哪个引脚、输出的极性是什么、触发源是什么，以及如何设置定时器的波特率。

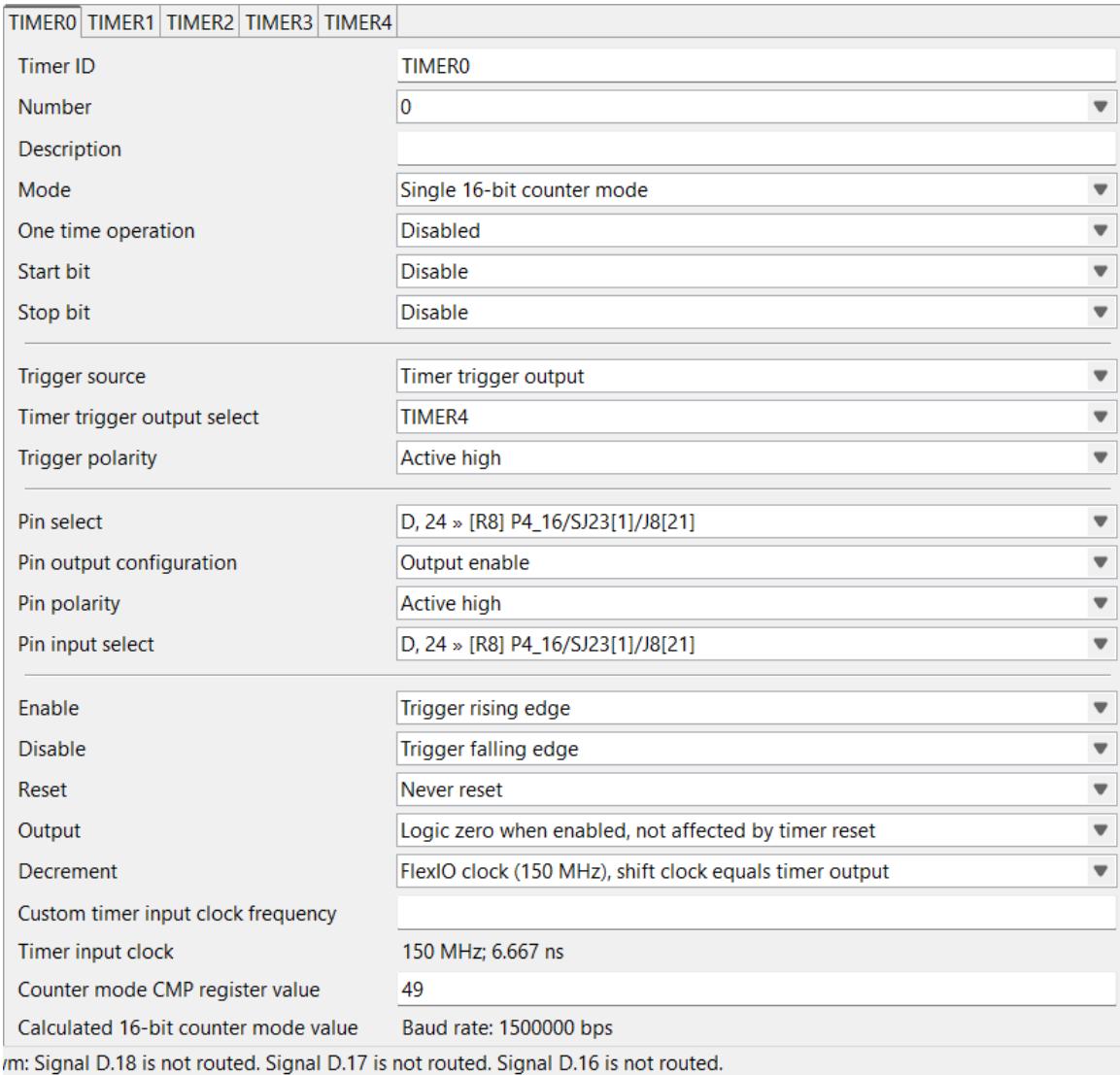


图13. MCUXpresso Config Tools中的定时器设置

## 6.3 MCUXpresso Config Tools中的移位器设置

移位器的设置主要包括选择哪个定时器作为触发源、选择当前状态的IO输出波形，以及设置哪三个引脚用于状态选择。

SHIFTER0 SHIFTER1 SHIFTER2 SHIFTER3 SHIFTER4

Shifter ID: SHIFTER0

Number: 0

Description: S0

Mode: State mode

Shifter size: 32-bit

Late store: Pre-shift register state

Timer select: TIMER0

Timer polarity: Positive edge

Input source: Pin

Pin select: D, 16

Pin output configuration: Output enable

Pin polarity: Active high

State

**Pins state shifter selection**

#	State 0	State 1	State 2	State 3	State 4	State 5	State 6	State 7
D, 18	0	0	0	0	1	1	1	1
D, 17	0	0	1	1	0	0	1	1
D, 16	0	1	0	1	0	1	0	0
Shifter select	SHIFTER1	SHIFTER0						

**Output pins**

#	D, 0 » [...]	D, 1 » [...]	D, 2 » [...]	D, 3	D, 4	D, 5	D, 6	D, 7
Output mask	Used	Used	Unused	Unused	Unused	Unused	Unused	Unused
Output state	Low	High	Low	Low	Low	Low	Low	Low

m: Signal D.18 is not routed. Signal D.17 is not routed. Signal D.16 is not routed.

图14. MCUXpresso Config Tools中的移位器设置

总结：使用配置工具来配置状态机确实非常直观简单。在使用过程中，请始终将其与寄存器描述进行比较，这有助于更好地理解其配置内容。

## 7 演示

此应用附带了一个配套程序，可以在其中查看参考工程中配置工具的相关配置。

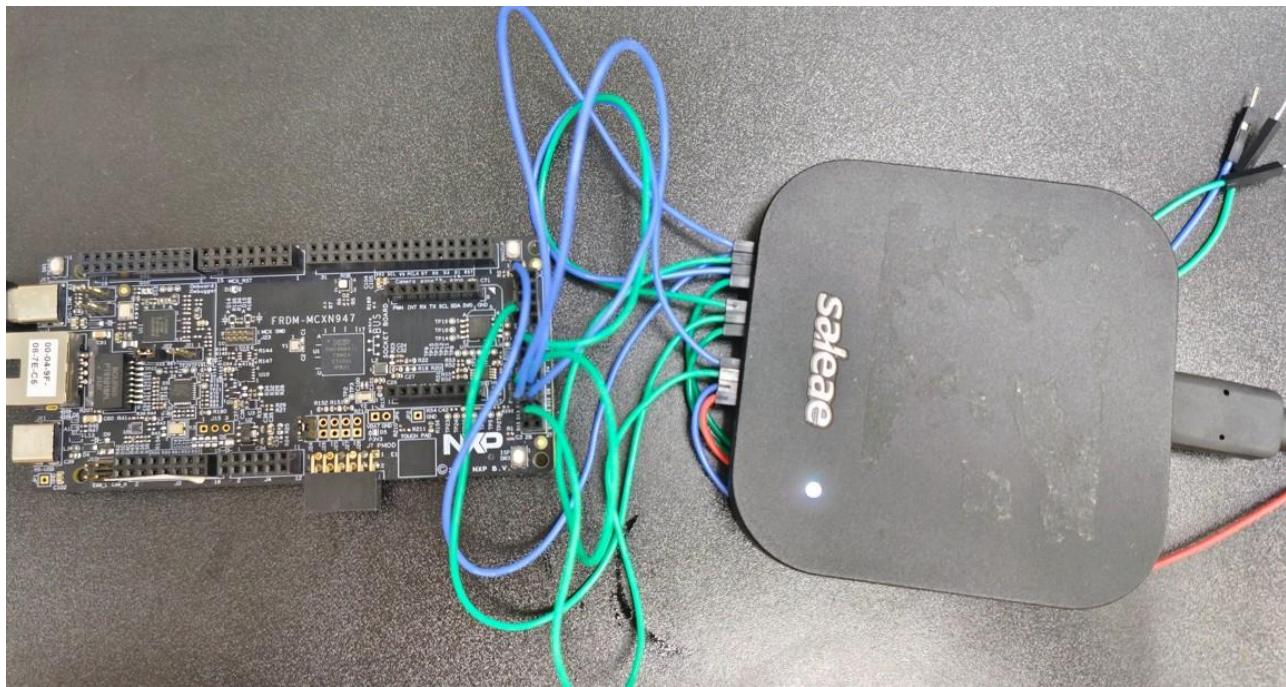
## 7.1 硬件准备

在此应用中，演示是基于FRDM-MCXN947开发板设计和开发的。[表8](#)是引脚分配表。

**表8. 引脚分配表**

功能	IO引脚	在FRDM-MCXN947上的位置
PWM_H	P0_8	J8-11
PWM_L	P0_9	J8-10
定时器0	P4_16	J8-21
定时器1	P4_17	J8-22
定时器2	P4_18	J8-23
定时器3	P4_19	J8-24
定时器4	P4_20	J8-25

[图15](#)是硬件连接示意图。



**图15. 硬件连接示意图**

开发板：FRDM-MCXN947 rev. B

逻辑分析仪：Saleae logic pro 16

1. 使用USB线将逻辑分析仪连接到PC。
2. 使用USB-Type C线将FRDM-MCXN947连接到PC。
3. 使用杜邦线将捕获通道连接到接头信号。

## 7.2 软件准备

解压附带的软件工程，并用MCUXpresso IDE打开。

1. 将工程导入IDE。
2. 构建工程代码。
3. 下载固件。

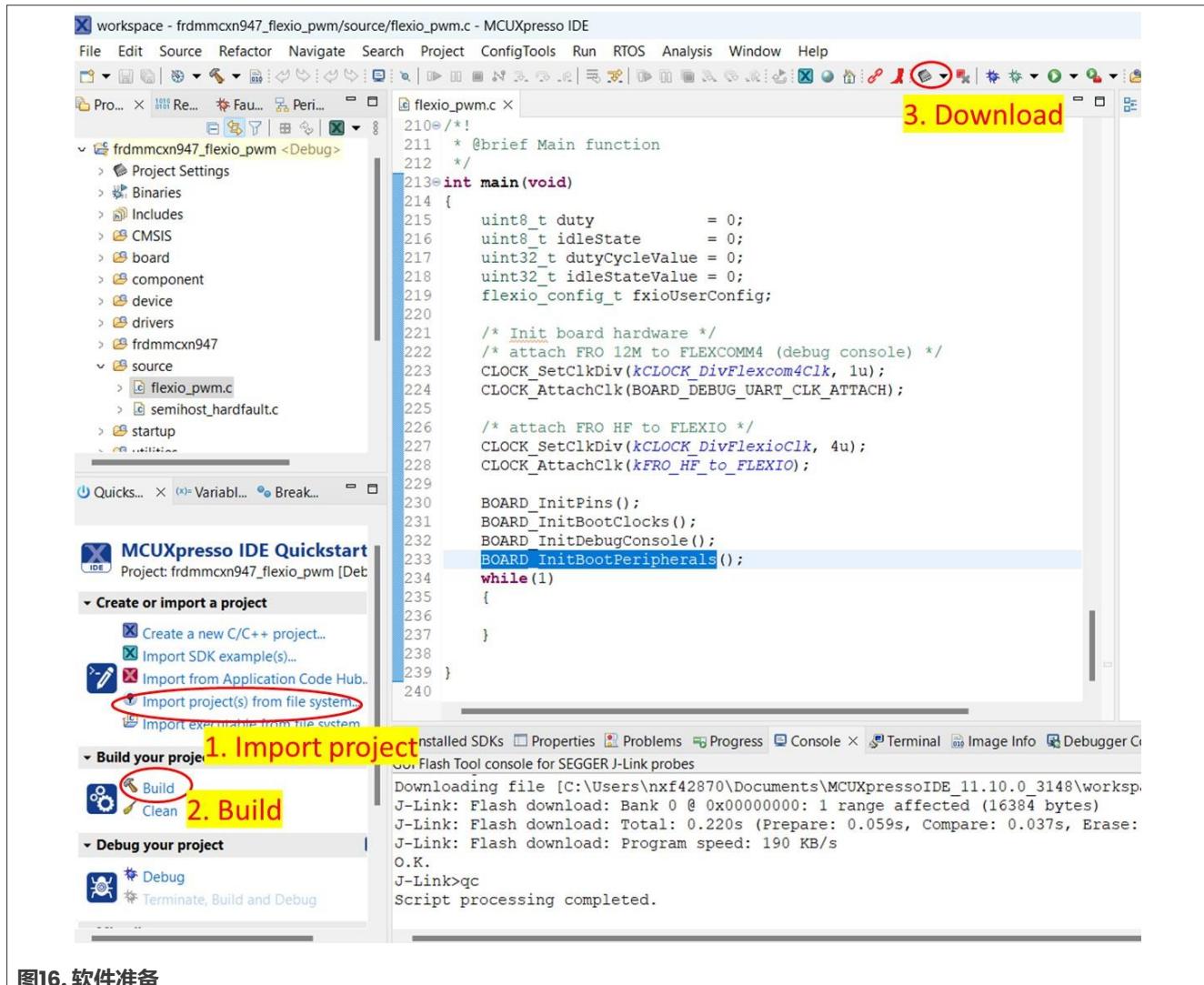


图16. 软件准备

### 7.3 结果

复位开发板，在PC上打开逻辑分析仪应用，并捕获时序，如图17所示。

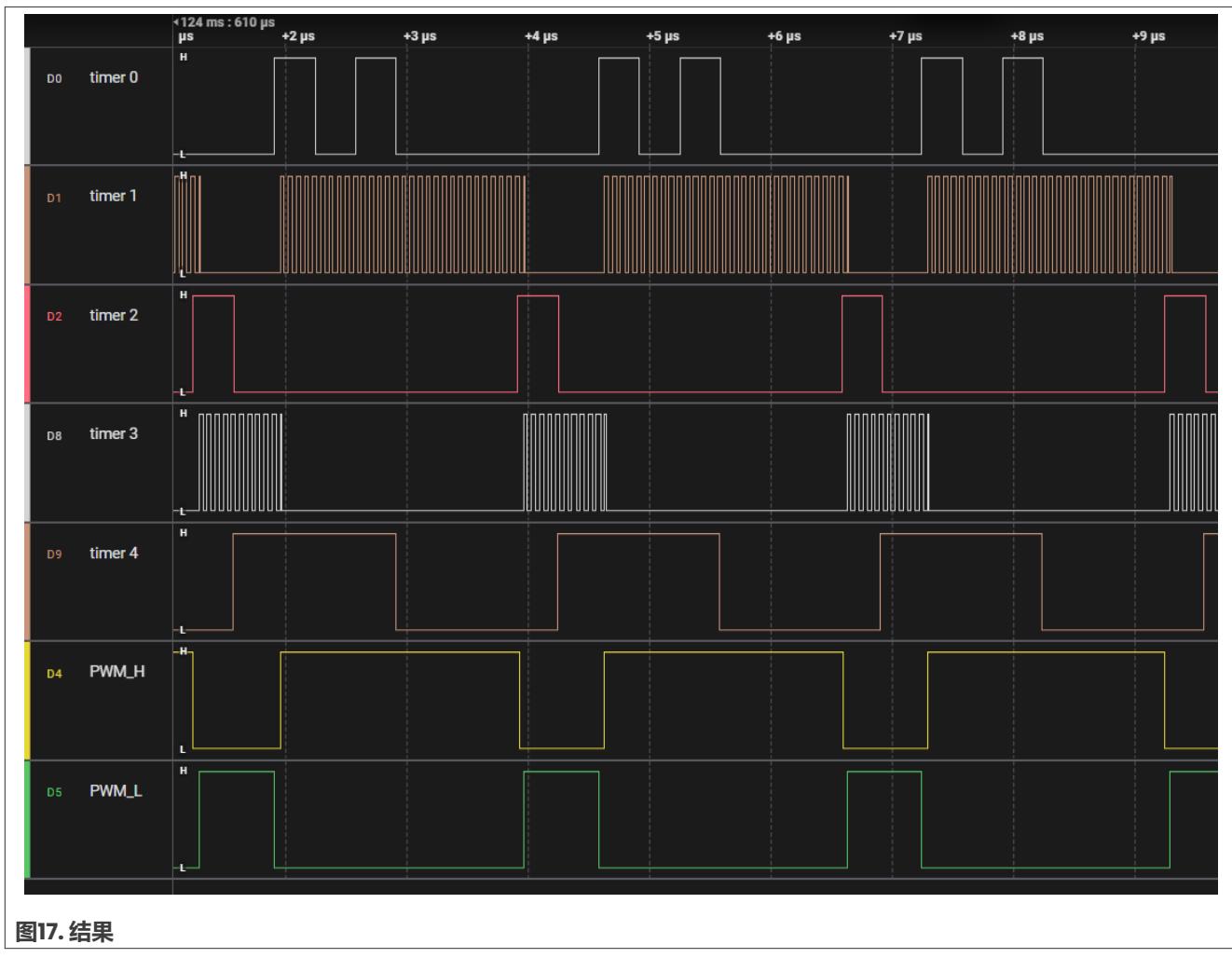


图17. 结果

## 8 参考文献

- 《使用FlexIO模块模拟硬件状态机》 (文档[AN5239](#))

## 9 关于本文中源代码的说明

本文中所示的示例代码具有以下版权和BSD-3-Clause许可：

2025年恩智浦版权所有；在满足以下条件的情况下，可以源代码和二进制文件的形式重新分发和使用本源代码（无论是否经过修改）：

- 重新分发源代码必须保留上述版权声明、这些条件和以下免责声明。
- 以二进制文件形式重新分发时，必须在文档和/或随分发提供的其他材料中复制上述版权声明、这些条件和以下免责声明。
- 未经事先书面许可，不得使用版权所有者的姓名或参与者的姓名为本软件的衍生产品进行背书或推广。

本软件由版权所有者和参与者“按原样”提供，不承担任何明示或暗示的担保责任，包括但不限于对适销性和特定用途适用性的暗示保证。在任何情况下，无论因何种原因或根据何种法律条例，版权所有者或参与者均不对因使用本软件而导致的任何直接、间接、偶然、特殊、惩戒性或后果性损害（包括但不限于采购替代商品或服务；使用损失、数据损失或利润损失或业务中断）承担责任，无论是因合同、严格责任还是侵权行为（包括疏忽或其他原因）造成的，即使事先被告知有此类损害的可能性也不例外。

## 10 修订历史

表9. 修订历史

文档ID	发布日期	说明
AN14470 v.1.0	2025年2月6日	• 初始版本

## Legal information

### Definitions

**Draft** — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

### Disclaimers

**Limited warranty and liability** — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

**Right to make changes** — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

**Suitability for use** — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

**Applications** — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

**Terms and conditions of commercial sale** — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at <https://www.nxp.com.cn/profile/terms>, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

**Export control** — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

**Suitability for use in non-automotive qualified products** — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

**HTML publications** — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

**Translations** — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

**Security** — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at [PSIRT@nxp.com](mailto:PSIRT@nxp.com)) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

**NXP B.V.** — NXP B.V. is not an operating company and it does not distribute or sell products.

### Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

**NXP** — wordmark and logo are trademarks of NXP B.V.

Microsoft, Azure, and ThreadX — are trademarks of the Microsoft group of companies.

## 目录

<b>1</b>	<b>介绍</b>	<b>2</b>
<b>2</b>	<b>FlexIO模块的状态模式</b>	<b>2</b>
2.1	FlexIO状态的I/O分配	2
2.2	FlexIO状态配置	2
2.3	FlexIO状态转换	4
<b>3</b>	<b>如何在状态模式下实现中心对齐PWM</b>	<b>5</b>
3.1	使用状态机实现PWM	5
3.2	状态机上的GPIO配置	6
3.3	内部定时器波形的输出观察	6
<b>4</b>	<b>FlexIO移位器配置</b>	<b>7</b>
4.1	本应用中使用的FlexIO状态模式	7
4.2	FlexIO状态模式配置	7
<b>5</b>	<b>FlexIO定时器配置</b>	<b>8</b>
5.1	本应用中的定时器配置	8
5.2	定时器触发和状态转换时序	9
<b>6</b>	<b>MCUXpresso Config Tools在本应用中的使用</b>	<b>13</b>
6.1	FlexIO状态机	13
6.2	MCUXpresso Config Tools中的定时器设置	15
6.3	MCUXpresso Config Tools中的移位器设置	15
<b>7</b>	<b>演示</b>	<b>16</b>
7.1	硬件准备	17
7.2	软件准备	17
7.3	结果	18
<b>8</b>	<b>参考文献</b>	<b>19</b>
<b>9</b>	<b>关于本文中源代码的说明</b>	<b>19</b>
<b>10</b>	<b>修订历史</b>	<b>20</b>
	<b>法律声明</b>	<b>21</b>

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.