

AN14254

使用QDC/ENC/Quad Timer等外设计算正交编码器的角度和速度

第1.0版—2024年4月2日

应用笔记

文档信息

信息	内容
关键词	AN14254、增强型M/T方法、Qtimer、ENC、QDC
摘要	本应用笔记介绍了一种测量角度的方法及一种适用于高低速场景的增强型M/T速度测量方法。



1 介绍

在电机控制中，正交编码器是检测电机转子位置的常用元件。它通过调理电路将三个通道的脉冲信号传递给控制芯片，以解码信号并完成位置与速度的计算。角度和速度的测量精度及其实时性对电机控制系统的性能有着直接的影响。因此，选择适当的角度和速度测量方法至关重要。

本应用笔记不仅提供了一种测量角度的方法，还提出了一种增强型M/T速度测量方法，该方法不仅能够适应高速，也能适应低速的测量需求。文档还阐述了如何配置QDC/ENC/QTIMER这三种常用于解码编码器信号的外设，来有效地解码编码器信号，并提取必要的信息进行分析。

2 正交编码器概述

正交编码器（双通道增量式编码器）将线性位移转化为脉冲信号。通过使用脉冲数及两信号间的相位差，用户可以准确追踪旋转位置、方向和速度。此外，第三通道被称为索引信号，可用于重置位置计数器，以确定绝对位置。

2.1 正交编码器的基本概念

原理

正交编码器通过两根信号线的脉冲输出进行数据处理。每个输出脉冲信号对应一定的增量位移。通过读取单位时间内的脉冲信号数，可有效测量速度。同时，通过累积脉冲信号数与编码器代码轮的周长（即一圈对应的距离），便可计算出移动距离。

信号

- A相脉冲输出：数据输出线，每个脉冲代表编码器旋转的固定位移量。
- B相脉冲输出：与A相脉冲输出相同，但两者输出脉冲间存在90°的相位差。
- Z相信号：当编码器旋转至零点时，Z相信号发出一个脉冲，表示编码器已完成一整圈的旋转。

编码器线数

编码器线数指的是A/B相线每旋转一圈输出的脉冲总数。如果一个编码器有1000线，表示其转子每旋转一圈输出1000个脉冲。A/B相线每圈旋转均发出等量脉冲，并保持90°的相位差。线数越多，编码器所能反映的位置精度也就越高。

2.2 测量编码器速度的方法

测量编码器的速度时，通常采用三种方法：M法、T法和M/T法。

- M法：也称频率测量法。此方法计算固定时间间隔内编码器脉冲数并计算速度值。如果每周期编码器的总脉冲数为 $(4 \times C)$ ，在时间 T_0 内计数的编码器脉冲数为 M_0 ，则速度 n 的计算公式为：

$$n = \frac{M_0}{(4 \times C \times T_0)} \text{ Hz} \quad (1)$$

M法在高速测量状态下能提供较高的测量精度和稳定性。然而，如果速度低，由于每个T0内的脉冲数很少，计算出的速度误差相对较大且不稳定。

- T法：也称周期测量法。该方法利用已知频率的一个高频脉冲进行计数。高频时钟计算两个相邻编码器脉冲，得到间隔时间 T_e ，计数值为 $M1$ 。如果每周期编码器的总脉冲数为 $(4 \times C)$ ，脉冲频率为 $F0$ ，则计算速度 n 的公式为：

$$n = \frac{1}{(4 \times C \times T_e)} = \frac{F0}{(4 \times C \times M1)} \text{ Hz} \quad (2)$$

$$T_e = \frac{M1}{F0} \quad (3)$$

T法使用一个参考脉冲来测定两个编码器脉冲之间的时间差，从而计算出速度。电机高速运转时，编码器脉冲间隔时间 T_e 非常短。这使得测量周期内的高频脉冲计数值 $M1$ 变少，导致测量误差增大。而电机低速运转时， T_e 足够长，测量周期内的 $M1$ 值也足够大。因此，T法与M法相反，更适合测量低速。

- M/T法：这种方法结合了M法和T法的优势，测量某段时间内的编码器脉冲数和高频脉冲数。在相对固定的时间周期内，首先计算编码器脉冲数 $M0$ ，然后计算已知频率 $F0$ 的高频脉冲，得到计数值 $M1$ ，以此来计算速度值。如果每周期编码器的总脉冲数为 $4 \times C$ ，则计算速度 n 的公式为：

$$n = \frac{F0 \times M0}{(4 \times C \times M1)} \text{ Hz} \quad (4)$$

由于每周期编码器的总脉冲数 $(4 \times C)$ 和高频脉冲频率 $F0$ 是固定值， $M0$ 和 $M1$ 影响速度 n 。电机高速运转时， $M0$ 增加而 $M1$ 减少，与M法相似。电机低速运转时， $M1$ 增加而 $M0$ 保持不变，与T法相似。

3 角度测量

正交编码器的每个输出脉冲均代表转子旋转的固定距离。通过配置正交解码外设，将编码器输出的A/B脉冲进行正交解码。A信号和B信号的每个边沿都会触发计数器的计数。当A相位先于B相位时，计数器数值增加；反之，当B相位先于A相位时，计数器数值减少，参见图1。

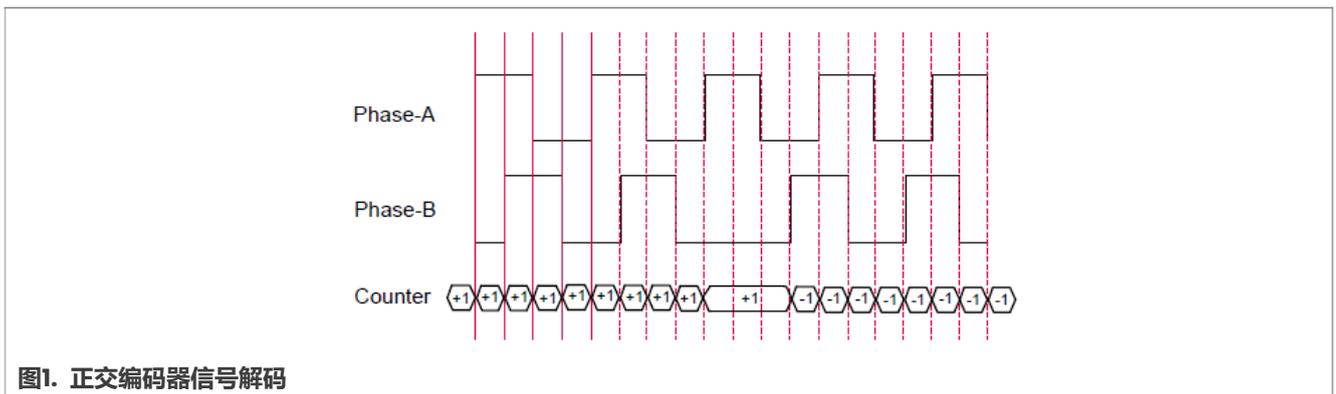


图1. 正交编码器信号解码

编码器在每次旋转中输出固定数量的A/B相脉冲，这一数量由编码器的线数决定。计数器的读数间接反映了转子的相对位置。该相对位置的机械角度为：

$$\text{Mechanical angle} = \frac{CNT \times 2 \times \pi}{4 \times C} \text{ rad} \quad (5)$$

其中，C代表编码器线数，CNT为计数器的计数值。

注：编码器脉冲信号仅能提供转子的相对位置信息，并不能直接反映转子的绝对位置。因此，需要对电机转子的初始位置进行探测。

确定转子位置的方法通常包括：

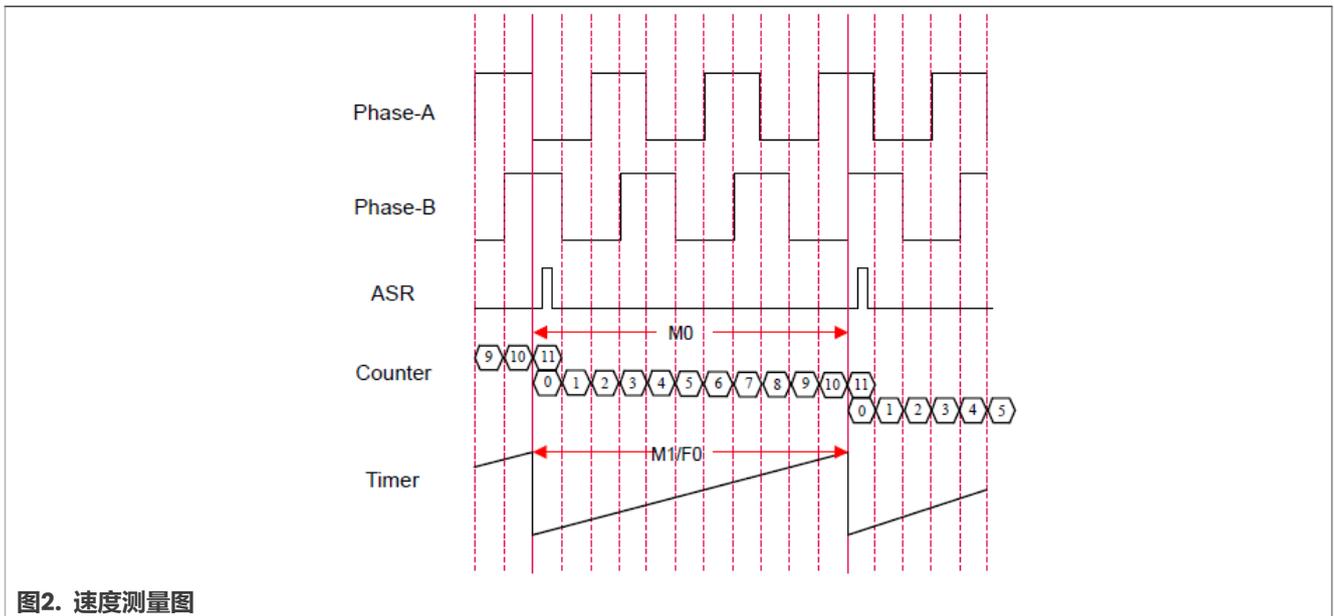
- 在操作开始前，对转子进行预定位。
- 设置一个固定角度的电流/电压矢量，使电机转子转动至预设位置。
- 然后，减去这个角度，得出转子的实际位置。

4 增强型M/T方法介绍

本节介绍增强型M/T方法的基本原理，并在后续部分详细说明在不同速度状态下的处理方法。最后，算法的流程图在[第4.3节](#)中展示。

4.1 正常操作下速度测量的方法

[图2](#)显示了测量速度的基本策略，其理念与M/T方法相同。能够被解码的外设（QDC/ENC/Timer）捕获编码器在固定时间内输出的脉冲数M0（参见[图2](#)中的ASR）。结合编码器的线数，我们可以计算出在该时间段内行进的距离，即 $M0/4C$ 。这里的C代表编码器的线数。同时，一个由高频时钟驱动的计时器也会被捕获。M0的第一个和最后一个边沿之间的计时器计数值为M1。结合高频时钟频率F0，可以得出行进该距离所用的时间，即 $M1/F0$ 。



[图3](#)展示在此速度测量方法下，正交解码模块计数器和高频时钟计时器的捕获时间示意图。捕获的编码器脉冲边沿数M0是指两个ASR之间的脉冲边沿数。而高频时钟测量的时间并非两个ASR之间的时间，而是两个ASR开始前编码器脉冲之间的时间。通过捕获这段时间内的高频时钟计数，可以得到M1。速度的计算公式参见[方程4](#)。

图4显示了在这三种状态下，两个ASR之间没有脉冲。例如，在启动状态中，在T4时刻测量速度时，编码器解码模块在T3至T4的时间段内未能捕获任何编码器脉冲（M0_T4）。由于T3和T4之间没有脉冲边沿，无法获取这段时间内的高频时钟计时器计数M1_T4。因此，速度测量无法完成。

为了解决这一问题，可以使用高频时钟计时器来计算速度测量时刻与此前最近的编码器脉冲边沿之间的时间段。高频时钟计时器的计数值为M1_edge，而F0是高频时钟的频率，因此这段时间可以表示为M1_edge/F0。假设编码器解码模块捕获的编码器脉冲边沿数M0为1（实际上为0），可以据此进行速度测量。具体的速度测量方法在后文中进一步分析。

M0指的是两个相邻速度测量时刻之间的编码器脉冲边沿数。M1是在两个相邻速度测量时刻之前最新脉冲边沿之间的高频时钟计时器的捕获计数。M1_edge是最后一个编码器脉冲边沿与速度测量时刻之间的高频时钟计时器计数。suffix_Tn表示对应的时间点。

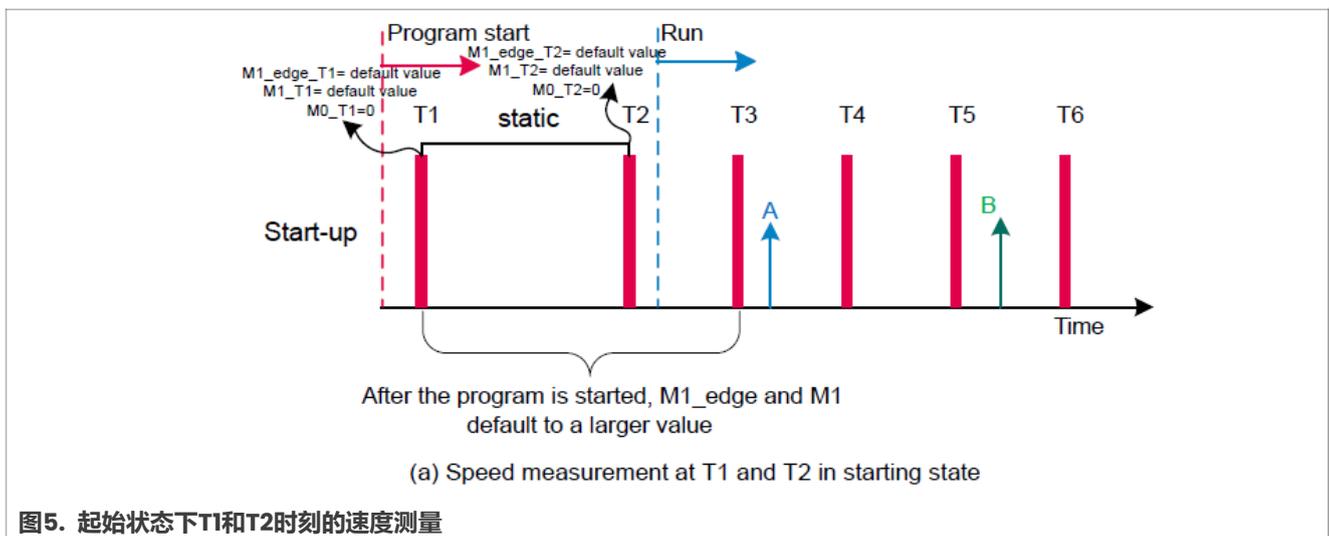


图5. 起始状态下T1和T2时刻的速度测量

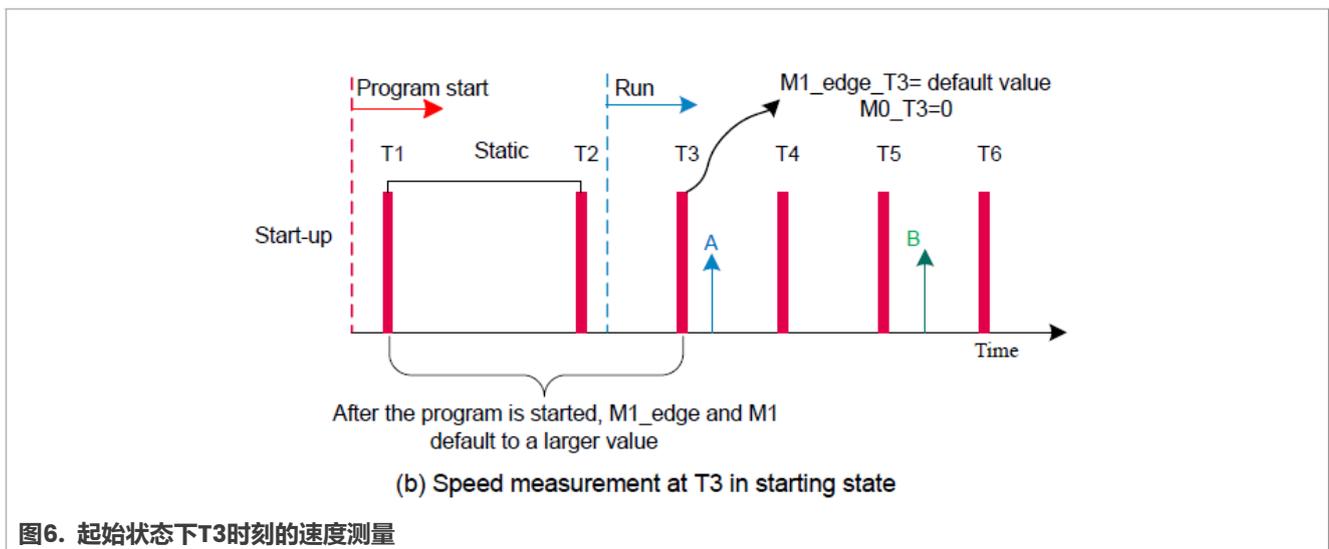


图6. 起始状态下T3时刻的速度测量

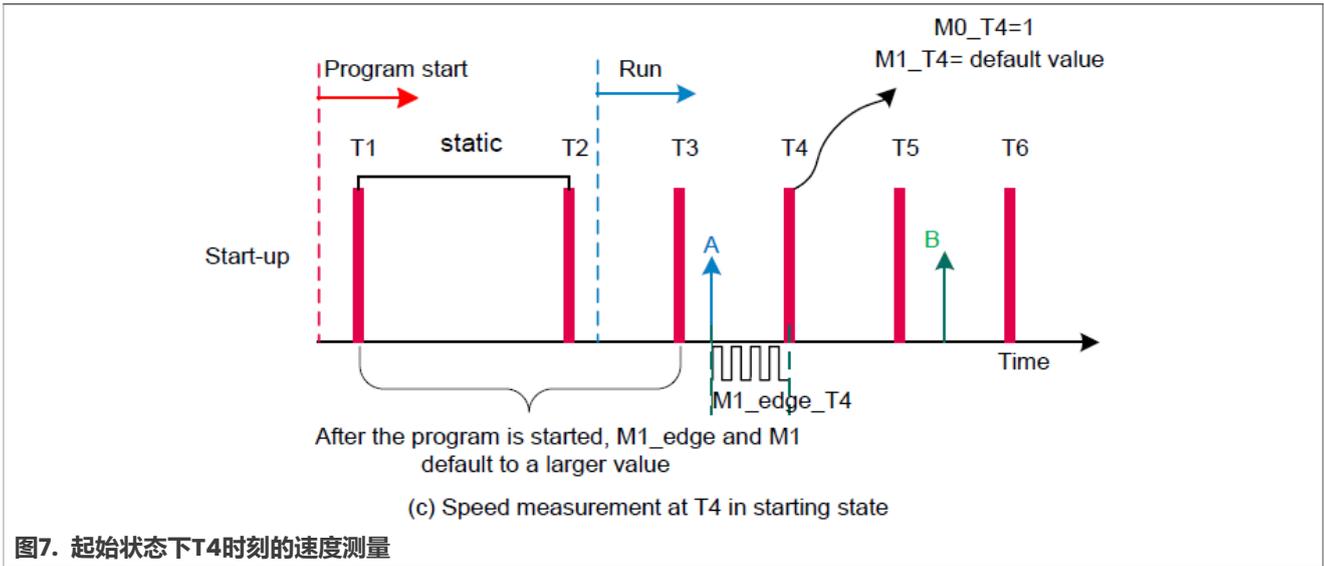


图7. 起始状态下T4时刻的速度测量

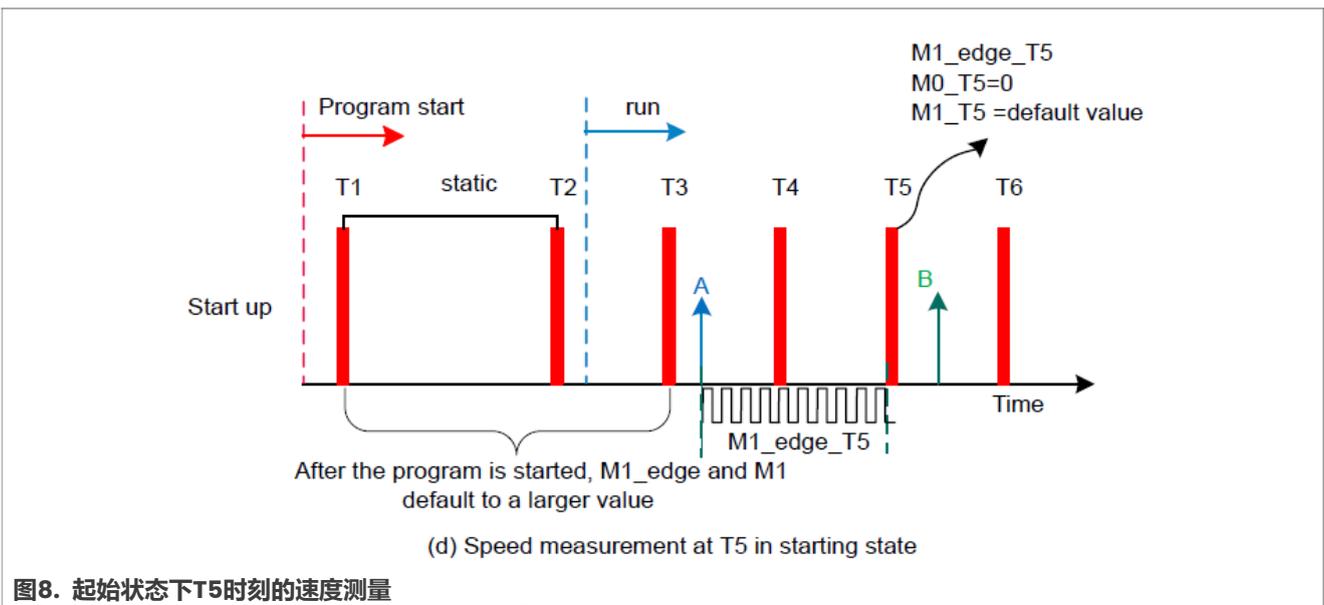


图8. 起始状态下T5时刻的速度测量

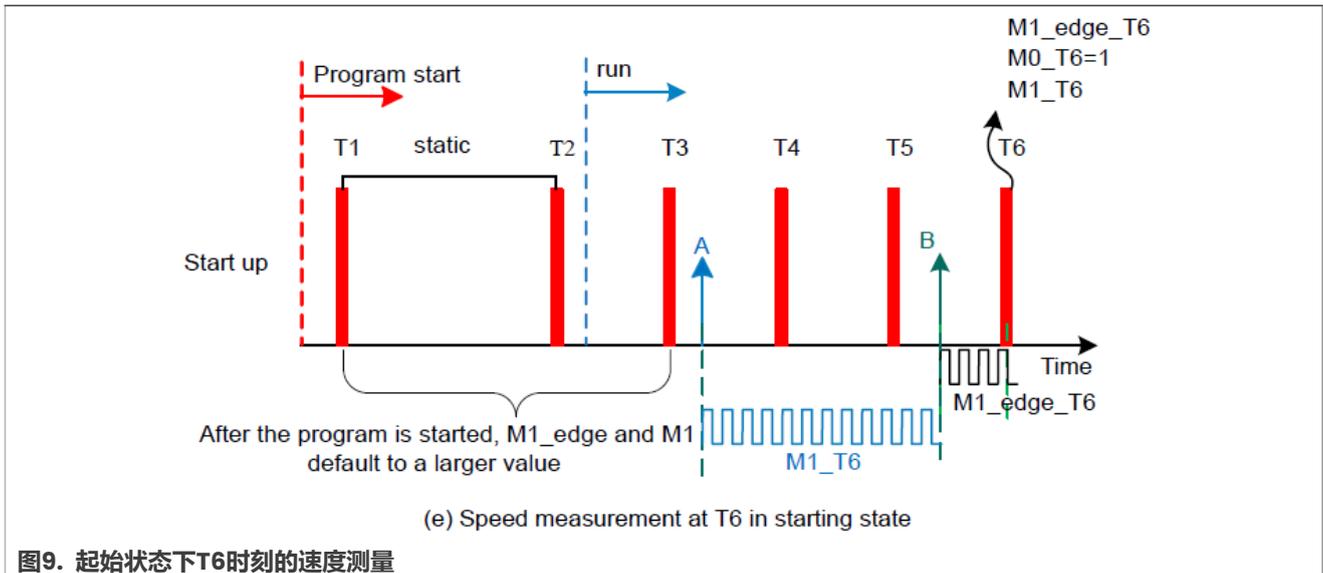


图9. 起始状态下T6时刻的速度测量

图5至图9中所示的启动状态，是指芯片上电后电机首次启动，并且在捕获到任何编码器脉冲之前的情况。解码模块未能捕获到任何脉冲边沿，导致计时器无法正确计算脉冲边沿之间以及脉冲与速度测量时刻之间的时间。

以下是图5至图9在T1至T6时刻的速度测量计算过程：

- T1时刻的速度测量：如果 $M1_T0 \geq M1_edge_T1$ ，则 $n_T1 = n_T0$ ；
如果 $M1_T0 < M1_edge_T1$ ，则 $n_T1 = F0 \times 1 / (4 \times C \times M1_edge_T1)$ ；
如果 $M1_edge_T1$ 过大，则 $n_T1 = 0$ ；
($M1_T0 = M1_edge_T1 =$ 默认值)
- T2时刻的速度测量：如果 $M1_T1 \geq M1_edge_T2$ ，则 $n_T2 = n_T1$ ；
如果 $M1_T1 < M1_edge_T2$ ，则 $n_T2 = F0 \times 1 / (4 \times C \times M1_edge_T2)$ ；
如果 $M1_edge_T2$ 过大，则 $n_T2 = 0$ ；
($M1_T1 = M1_edge_T2 =$ 默认值)
- T3时刻的速度测量：如果 $M1_T2 \geq M1_edge_T3$ ，则 $n_T3 = n_T2$ ；
如果 $M1_T2 < M1_edge_T3$ ，则 $n_T3 = F0 \times 1 / (4 \times C \times M1_edge_T3)$ ；
如果 $M1_edge_T3$ 过大，则 $n_T3 = 0$ ；
($M1_T2 = M1_edge_T3 =$ 默认值)
- T4时刻的速度测量： $n_T4 = F0 \times M0_T4 / (C \times M1_T4)$ ；
($M1_T4 =$ 默认值)
- T5时刻的速度测量：如果 $M1_T4 > M1_edge_T5$ ，则 $n_T5 = n_T4$ ；
如果 $M1_T4 < M1_edge_T5$ ，则 $n_T5 = F0 \times 1 / (C \times M1_edge_T5)$ ；
如果 $M1_edge_T5$ 过大，则 $n_T5 = 0$ ；
- T6时刻的速度测量： $n_T6 = F0 \times 1 / (4 \times C \times M1_T6)$

上述方法说明，在两次速度测量时刻之间如果未捕获到编码器边沿，将使用计时器捕获的 $M1_edge$ 参与速度测量。当 $M1_edge$ 小于上一时刻计时器捕获的 $M1_T$ 时，认为该时刻的速度不变。当 $M1_edge$ 大于等于上次计时器捕获的 $M1_T$ 时， $n = F0 \times 1 / (4 \times C \times M1_edge)$ 。如果 $M1_edge$ 过大，认为此时 $n = 0$ 。

上述三种速度测量方法涵盖同一方向下所有的速度测量情况。在某些情况下，如果旋转方向发生了从正向到反向的变化，将无法确定确切的行进距离。如图10所示，在T4与T5之间方向发生了变化。在这段时间内，虽然有1个脉冲边沿，但实际行进的距离是从前一刻B相上升沿脉冲的正方向旋转到B相上升沿脉冲的负方向，这个距离是无法确定的。因此，如果存在方向变化，则认为那一刻的速度为0。在图10中， $n_{T5} = 0$ 。

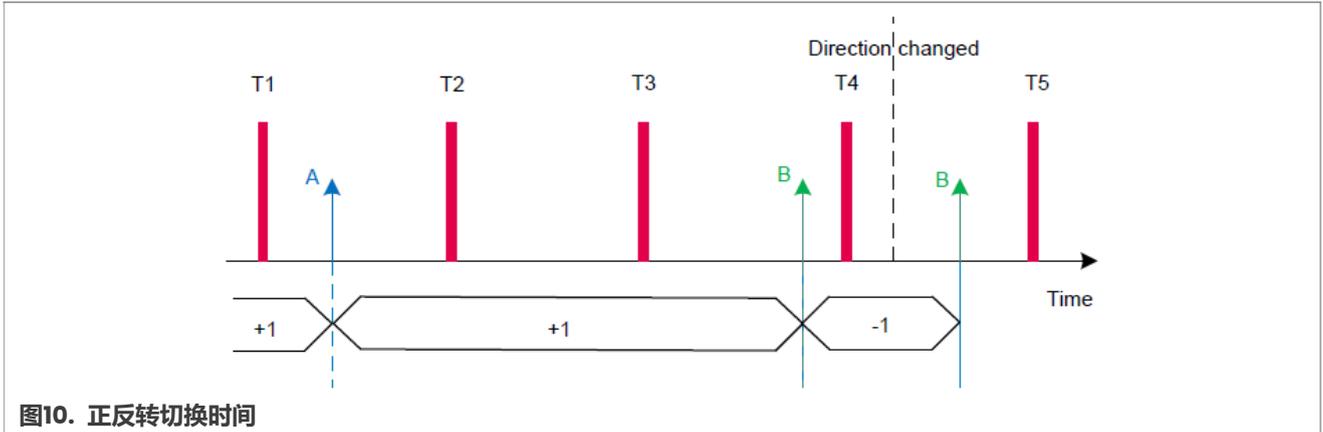


图10. 正反转切换时间

4.3 速度测量算法流程图

图11展示了速度测量算法的流程图。

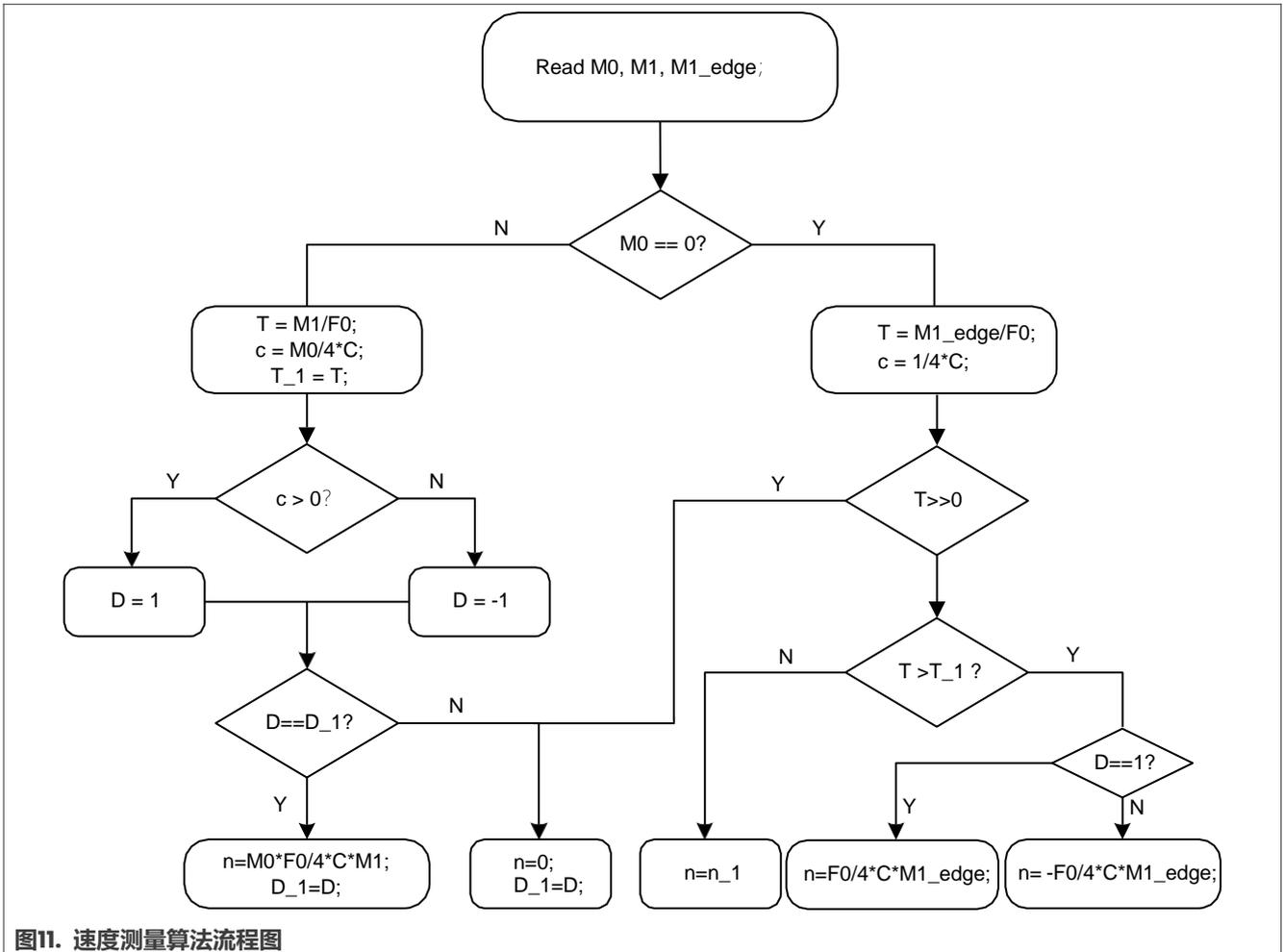


图11. 速度测量算法流程图

5 使用解码外设进行速度测量的三种方法

恩智浦针对正交解码提供了多种定时器外设。为了使用不同的正交解码外设来进行速度测量，本文介绍了三种方法。

在具体介绍这些方法之前，需要注意的是，由于不同系列产品的外设命名并不完全统一，因此具有同样功能的某些外设在不同系列中可能会有不同的命名。下文提供了一个映射表。您可以根据所选芯片找到可以使用的相应方法。

表1. 根据不同芯片选用的方法

方法	特性	适用芯片
方法1QDC ^[1]	专为正交解码设计的外设	i.MX RT : RT1170 QDC, RT1180 eQDC ^[2] , RT1160 QDC MCXN : MCXN54x QDC ^[4] , MCXN94x QDC ^[3] , MCXN23x QDC ^[3] MCXA : MCXA14x QDC ^[4] , MCXA15x QDC ^[4] LPC : LPC55(S)3x ENC ^[3] DSC : DSC81xxx QDC, DSC80xxx eQDC

表1. 根据不同芯片选用的方法 (续)

方法	特性	适用芯片
方法2 QTimer	具备解码功能的多功能定时器 (在芯片不含QDC/ENC外设时使用)	i.MX RT : I.MX RT四位数全系列QTimer外设, RT1010系列除外 DSC : DSC全系列 QTimer外设
方法3 ^[1] ENC +QTimer	ENC是QDC的前身版本, 不包含M/T速度测量所需的相关计数器。因此, 需使用QTimer外设与ENC协同工作来完成速度测量。与方法2相比, ENC具有32位计数器, 适用于高线数编码器。	DSC : DSC84xxx ENC、RT1060 QDC、RT1050 QDC、RT1040_QDC、RT1024 QDC、RT1060 QDC、RT1020 QDC

[1] "方法"列中提及的QDC或ENC用于区分各方法的特性, 并非指任何芯片的具体外设。

[2] 在速度测量应用中, eQDC外设与QDC外设类似, 但增加了针对其他应用的新特性。

[3] LPC55(S)3x上的ENC具备QDC功能, 应用可参照方法1。

[4] MCXA14x QDC、MCXA15x QDC上的QDC具有eQDC特性, 应用可参照方法1。

5.1 方法1: 使用QDC外设测量编码器信号的速度

本节阐述如何使用QDC外设实现增强型M/T速度测量方法。

5.1.1 与编码器速度测量相关的QDC外设介绍

QDC是专为编码器信号正交解码而设计的外设模块。它内置一个用于正交解码的32位计数器, 非常适合用于线数较高的编码器。此外, 它还包括为速度检测专门设计的位置差计数器、位置差时间计数器和边沿时间计数器, 以便测量关键的速度信息。

图12展示了正交解码模块的内部框图。从内部框图可以看出, 正交解码模块由五个独立的计数器组成。其中, 位置差计数器、最后边沿时间计数器和位置差周期计数器直接用于确定两次速度测量时刻之间的脉冲计数 (M0)。同时, 这些计数器还能够提取脉冲与速度测量时刻之间的高频时钟计数 (MI_edge) 以及脉冲之间的高频时钟计数 (M1)。下面将介绍三个用于速度测量的寄存器:

- **位置差计数器**: POSD 寄存器, 用于捕获位置寄存器 (UPOS/LPOS) 的变化值。当用于编码器速度测量时, 它记录的是两次速度测量时刻之间的脉冲数 M0。当读取位置计数器、位置差计数器和圈数计数器时, POSD 的值会自动加载到 POSDH 寄存器中。如果 CTRL3 [PMEN]=1, 仅读取位置差计数器时, POSD 的值也会加载到 POSDH 寄存器中。
- **最后边沿时间计数器**: LASTEDGE寄存器, 是一个16位最后边沿时间计数器, 专门记录从PHASEA或PHASEB最后一个边沿发生以来经过的时间。每个预分频时钟脉冲都会使最后边沿时间计数器递增, 而CTRL3 [PRSC]决定了分频倍数。任何A相/B相脉冲都会将LASTEDGE计数器清零, 并在计数器计数到0xffff后停止计数。当读取POSD寄存器时, LASTEDGE的值自动加载到LASTEDGEH寄存器中。
- **位置差周期计数器**: POSDPER寄存器, 用于获取上次读取POSD寄存器以来的时间。POSDPER计数器与LASTEDGE计数器共享一个高频时钟。A相/B相脉冲不会清除POSDPER, 但会将POSDPER的值发送到POSDPERBFR寄存器。当读取POSD寄存器时, POSDPERBFR的值会加载到POSDPERH寄存器中, 代表两个速度测量点之前脉冲之间的高频时钟计数M1。同时, 将LASTEDGE的值加载到POSDPER寄存器中, 使POSDPER从此时的LASTEDGE值开始计数。

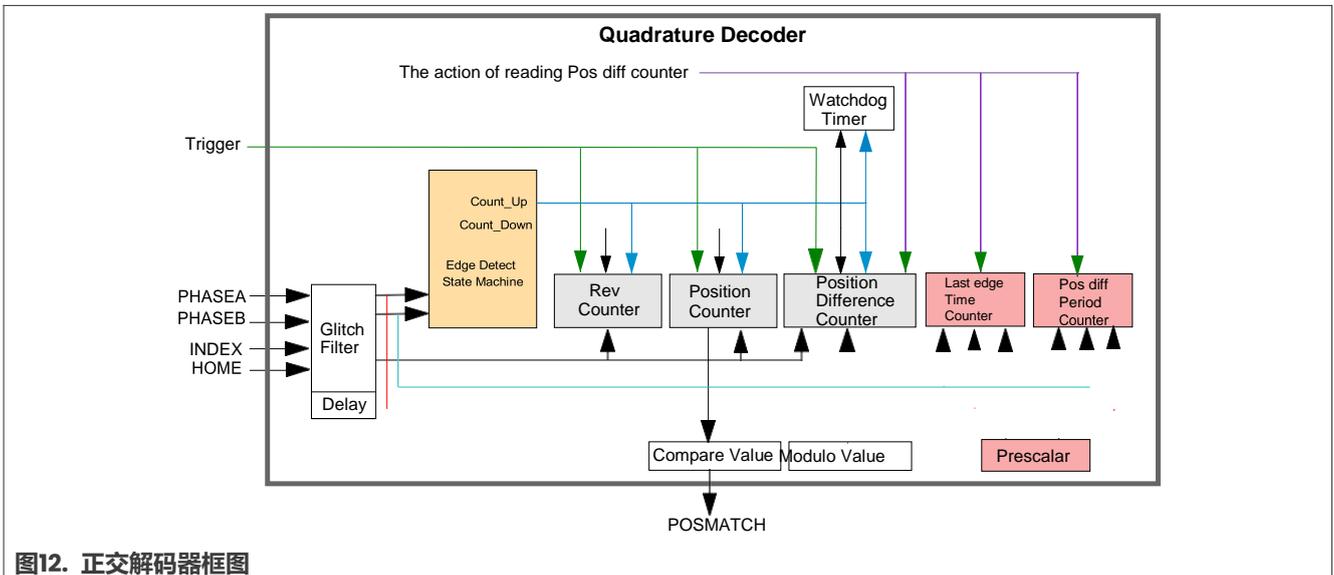


图12. 正交解码器框图

5.1.2 使用方法1提取速度测量信息M0/M1/M1_edge

图13展示了在进行速度测量时，三个寄存器的计数捕获方式。具体的寄存器配置如下：

- QDCx_LMOD=4*C-1: 此处设置计数器的比较值为4*C-1。
- QDCx_CTRL2[MOD]=1: 此配置启用比较功能，并在计数器达到比较值时初始化。
- QDCx_CTRL2[PMEN]=1: 在此配置下，在读取当前循环中的POS值时，已加载的POSD值不会再次更新，这对于速度测量来说非常方便。其余寄存器保持默认值。

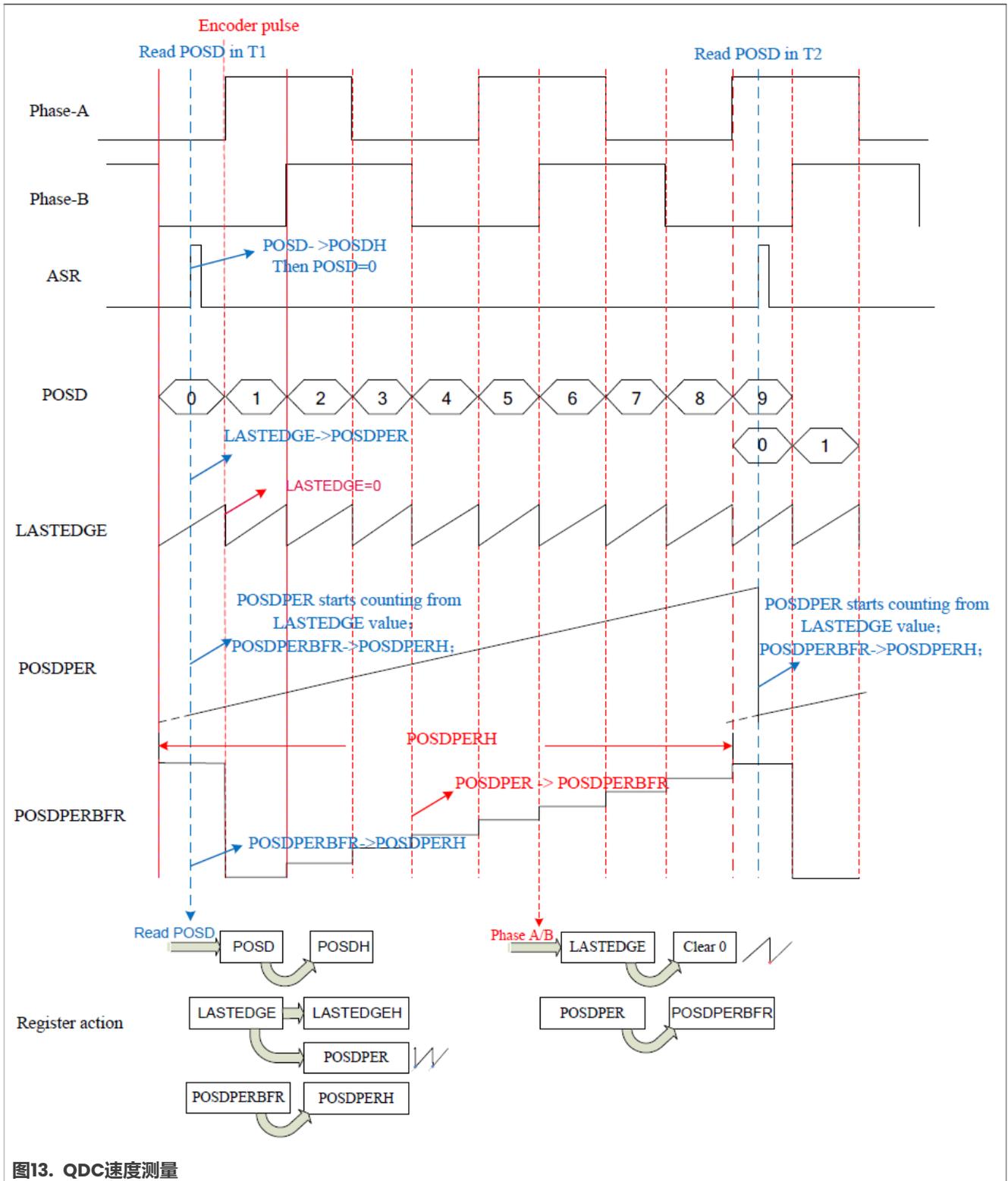


图13. QDC速度测量

如图13所示，在T1时刻读取POSD时，硬件自动使用LASTEDGE的值作为POSDPER的初始值重新计数。随后，每当编码器脉冲边沿发生，POSDPER的值就会自动被加载到POSDPERBFR寄存器中。直至下一次读取POSD的T2时刻，POSDPERBFR寄存器的值便会自动被加载到POSDPERH寄存器中。以上步骤都是为了提取第5节中提及的用于速度测量的三个关键信息M0、M1和M1_edge。其中，POSDH寄存器的值表示当前速度测量时刻与上一次速度测量时刻之间的脉冲边沿数M0。在速度测量时刻捕获的POSDPERH寄存器的值是第5节中提到的高频时钟计数M1。而LASTEDGEH寄存器则记录了脉冲边沿与速度测量时刻之间的高频时钟计数M1_edge。

需要特别注意的是，LASTEDGE和POSDPER共享同一个高频计数器，其高频时钟频率为F0。当电机处于低速状态、启动/停止状态，并且高频时钟频率F0设置过高时，LASTEDGE和POSDPER寄存器会溢出，如图14所示。此时，可以使用CTRL3 [PRSC]寄存器来分频高频时钟，确保 $F0 < 65535 \times 4C \times n_{\min}/60$ 。这里的 n_{\min} 是预期的最小速度，65535是16位计数器的最大计数值，C是编码器线数。

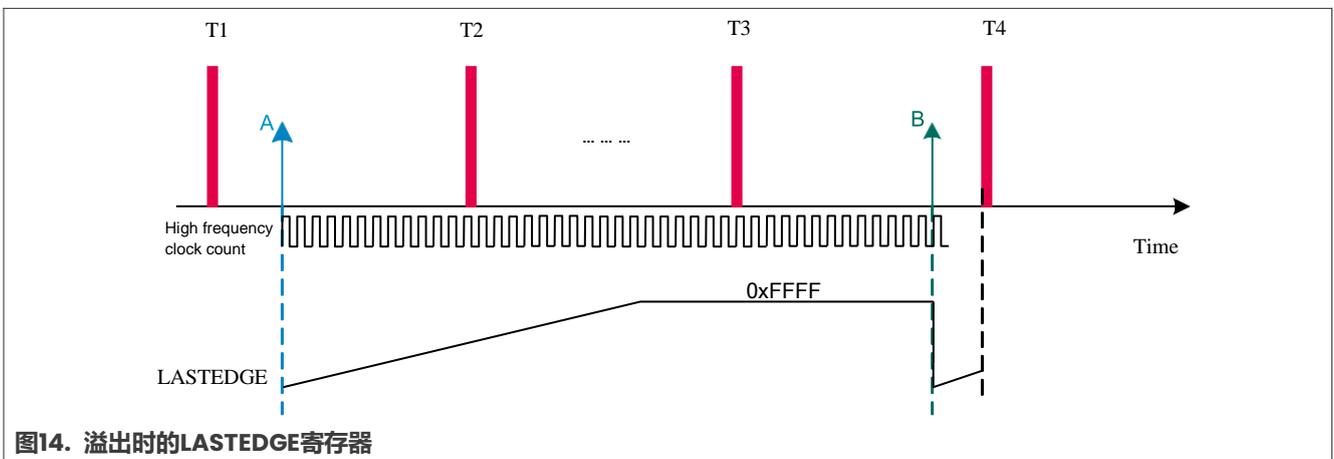


图14. 溢出时的LASTEDGE寄存器

5.2 方法2：使用QTimer外设测量编码器信号的速度

本节介绍如何使用QTimer外设实现增强型M/T速度测量方法。

5.2.1 用于编码器速度测量的QTimer外设介绍

QTimer是一个集成的外设模块，内含四个多功能的16位计数器。它支持多种计数模式，经常用于PWM生成、信号捕获、编码器正交解码等多种功能。

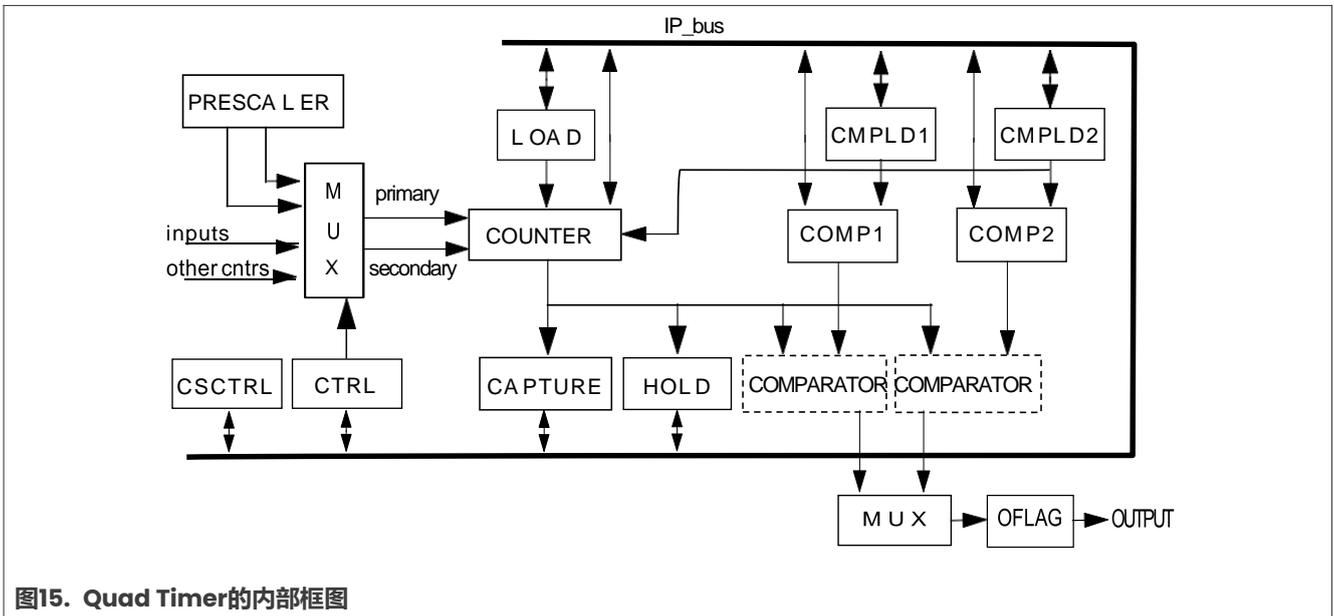


图15. Quad Timer的内部框图

图15展示了其中一个计数器的内部框图。每个计数器都有两个输入引脚——主输入和次输入。通过CTRL [PCS]和CTRL [SCS]寄存器可以分别进行通道选择。输入源可以是外部输入、QTimer模块内其他计数器的输出，或是外部时钟的分频信号。CTRL[CM]寄存器允许用户在七种不同的计数模式中进行选择，以适应各种计数需求。

5.2.2 使用方法2提取速度测量信息M0/M1/M1_edge

通过将QTimer内的四个计数器设置为不同的计数模式，它们可以协同工作，依据编码器的信号来完成速度的测量。图16则展示了QTimer在进行速度测量时的配置框图。

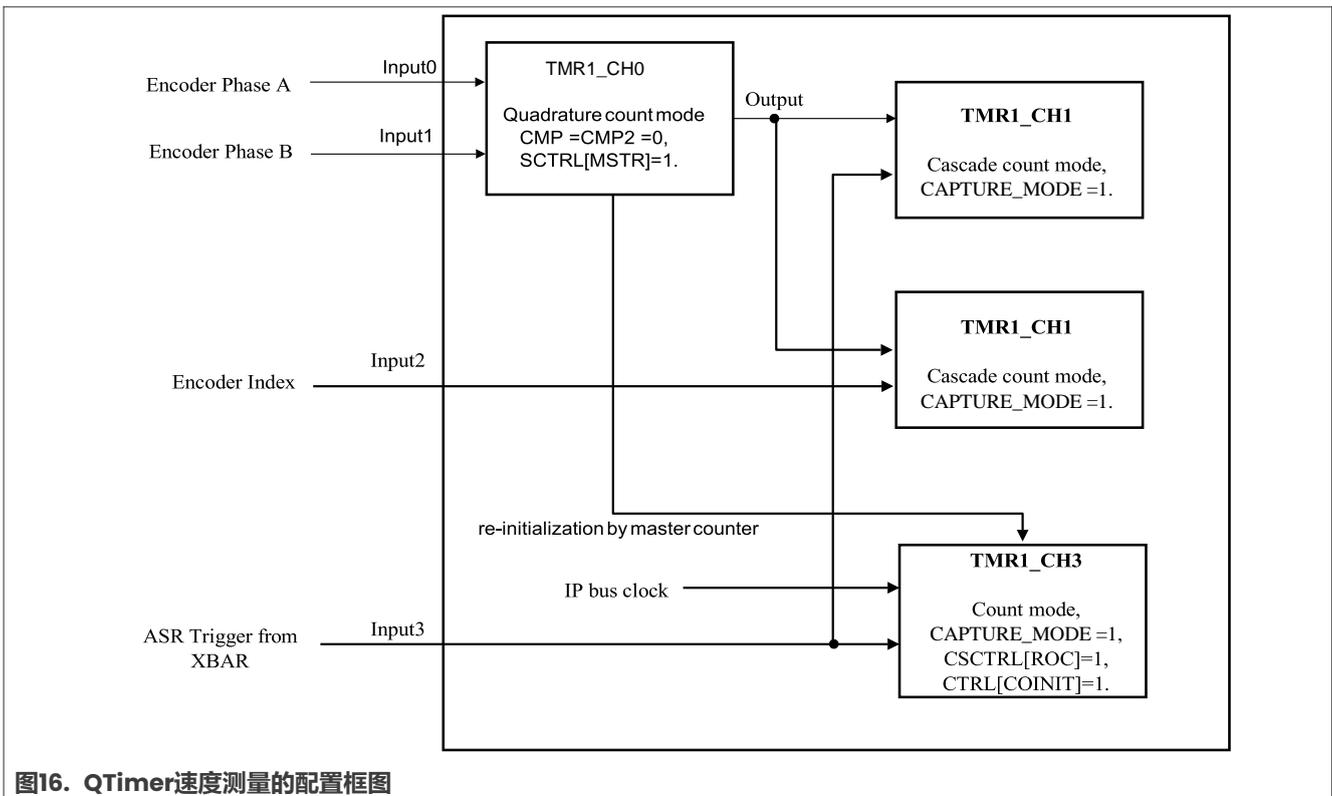


图16. QTimer速度测量的配置框图

在速度测量中，QTimer外设的具体配置如下：

- **计数器0**：选择正交计数模式作为计数模式：TMRx_CTRL [CM=4]，负责解码编码器的A相和B相脉冲信号，每次正向计数增加1，反向则减少1。A/B相编码器脉冲信号通过引脚和Xbar连接至计数器0的输入端，使用TMRx_CTRL [PCS]和TMRx_CTRL [SCS]来选择通道。将比较寄存器TMRx_COMP1和TMRx_COMP2设置为0，这样在A/B相产生脉冲时便会触发比较事件，以便在级联模式（TMRx_CTRL [CM=7]）下为其他计数器提供服务。配置为主模式（TMRx_SCTRL [MSTR]=1），主模式将计数器比较器的输出连接至模块中的其他计数器，以便重新初始化它们或将它们强制输出到TMRx_CSCTRL [OFLAG]信号。当用于速度测量时，此配置用于与其他计数器协作，实现MI_edge的提取。
- **计数器1**：计数模式设置为级联模式（TMRx_CTRL [CM=7]）。该模式将模块中其他计数器比较模块的输出作为此计数器的输入进行级联。此处，主计数源（TMR0_CTRL [PCS]）被设定为计数器0的输出。当计数器0触发比较事件时，计数器1的CNT将相应地进行增量或减量计数。次级计数源为外部固定频率触发器，通常是速度环触发器。配置输入捕获模式，以捕获次级计数源的上升沿（TMRx_SCTRL [CAPTURE_MODE]=1）。当触发捕获事件时，计数器1的CNT（TMRx_CNTRn [COUNTER]）值被存储在CAPTURE寄存器（TMRx_CAPTn [CAPTURE]）中。两次捕获值之间的差值即为所需的M0值，代表两次速度测量时刻之间经过的脉冲数量。
- **计数器2**：计数模式选择级联模式（TMR0_CTRL [CM=7]），配置与计数器1一致。此外，次级计数源被设定为编码器索引信号，配置输入捕获模式，以捕获次级计数源的上升沿（TMRx_SCTRL [CAPTURE_MODE]=1）。寄存器比较值TMRx_COMP1和TMRx_COMP2分别设置为 $4 \times C - 1$ 和 $-4 \times C + 1$ ，其中C为编码器线数。LOAD寄存器值TMRx_LOAD设为0。当计数器计数达到比较值时，它被初始化，初始值从LOAD寄存器加载。计数器的值能够反映电机转子的相对机械位置。

- 计数器3**: 计数模式选择为主计数源上升沿计数模式 (TMRx_CTRL [CM=1])，配置输入捕获模式，捕获次级计数源的上升沿 (TMRx_SCTRL [CAPTURE_MODE]=1)，并在触发捕获后重新加载计数器 (TMRx_CSCTRL [ROC]=1)。主计数源 (TMR0_CTRL [PCS]) 被选为高频时钟信号，通过IPBUS分频获取。寄存器CTRL0 [PCS]用于确定具体的分频倍数。次级计数源被选为由Xbar引入的外部固定频率触发器，通常是ASR触发器。将计数器0比较模块的输出连接至此计数器，并设置TMRx_CTRL[COINT=1]。这样，当计数器0比较事件发生时，计数器3将被重置。计数器1和计数器2的配置相互协作，共同实现MI_edge和MI的提取。

TMRx是所使用计数器的名称，例如在DSC系列中，通常是TMRA/B。在i.MX RT系列中，通常是TMR1/2。

通过这四个计数器的协同工作，能够实现编码器角度和速度测量关键信息M0、MI和MI_edge的精确提取。计数器0和计数器2负责捕获角度信息，而计数器1和计数器3则用于间接获取编码器脉冲边沿计数M0。它们还能够获取从脉冲边沿到速度测量时刻的高频时钟计数MI_edge，以及脉冲边沿之间的时钟计数MI，为速度测量提供服务。[图17](#)展示了QTimer中每个计数器的捕获计数逻辑图。使用计数器1捕获CNT值，我们可以间接获得两个ASR触发器之间通过的脉冲边沿数M0。当进入ASR_Trigger时，计数器3捕获CNT值，且可以间接获得速度测量所需的MI和MI_edge值。

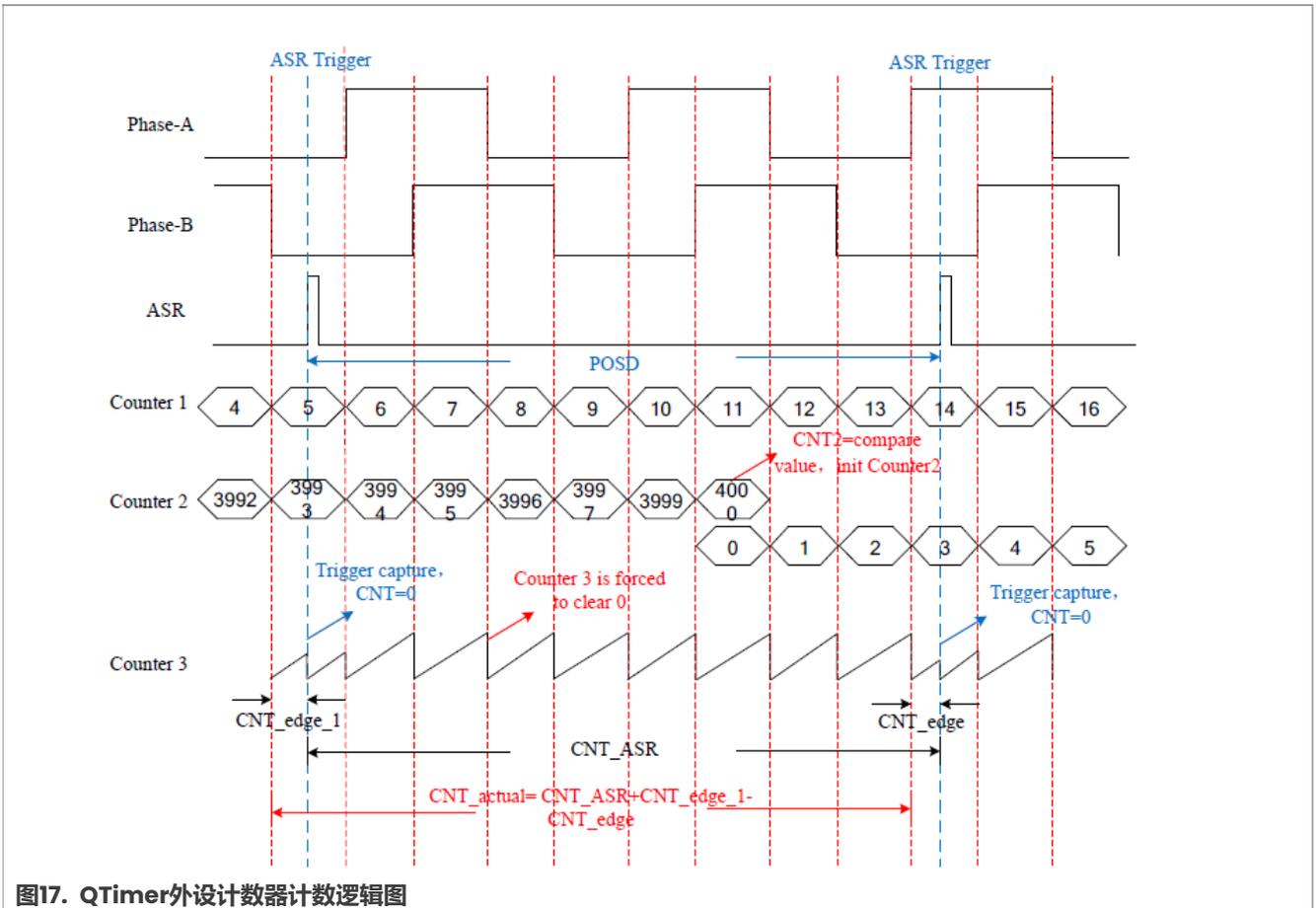


图17. QTimer外设计数器计数逻辑图

如图17所示，ASR触发器触发计数器1，以捕获CNT值。通过比较两次连续触发所捕获的CNT值，能够得到两次测速时刻之间的编码器脉冲边沿数M0。编码器脉冲边沿和ASR触发器初始化计数器3。同时，ASR触发器也触发计数器3进行CNT捕获。此时记录下的CNT值被称作CNT_edge。如果在两次测速的时间点之间存在编码器脉冲边沿，ASR触发器捕获到的CNT_edge值就是从编码器脉冲边沿到测速时刻之间的高频时钟计数M1_edge。若两次测速时刻之间没有编码器边沿出现，则计数器3在此期间不会被初始化。这样，通过ASR触发器捕获到的CNT_edge值就间接反映了速度环测速的时间间隔，但无法提供脉冲边沿的具体信息。图17展示了如何利用计数器3间接得到M1值。计数器3可以用来获取CNT_edge_1和CNT_edge值，其中CNT_edge_1表示从图17中第一次测速时刻到其前一个脉冲边沿之间的高频时钟计数，CNT_edge则表示从第二次测速点到前一个脉冲边沿之间的高频时钟计数。因此，可以得出 $M1 = CNT_edge_1 + CNT_ASR - CNT_edge$ ，这里的CNT_ASR是计数器3在一个ASR周期内的高频时钟计数值。

对于在低速、启动和停止状态下得到的M1_edge和M1值，同样也需要使用上述的CNT_edge和CNT_ASR值来进行转换。

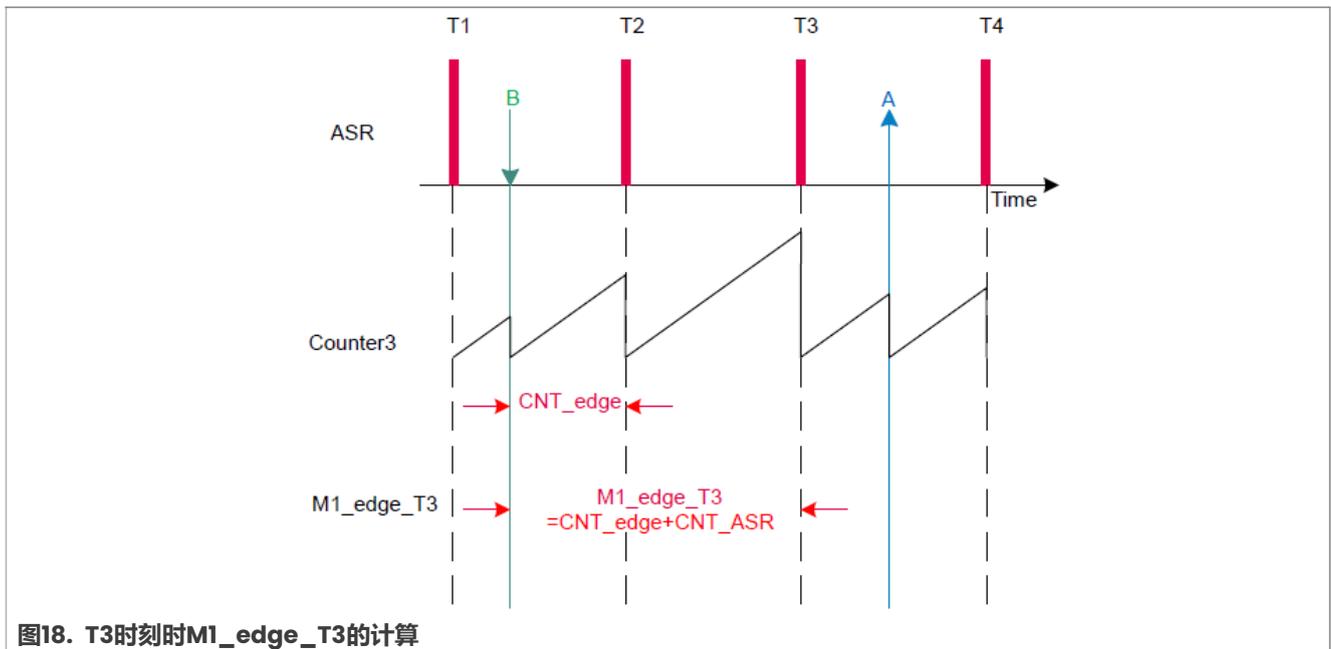
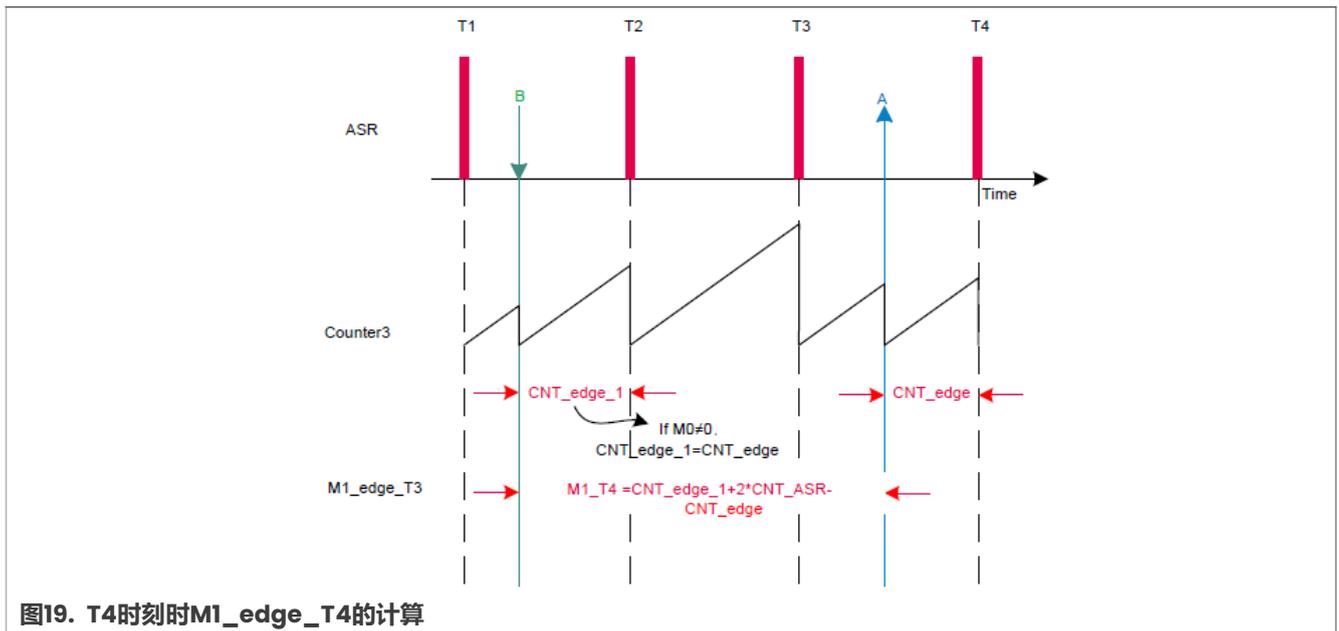
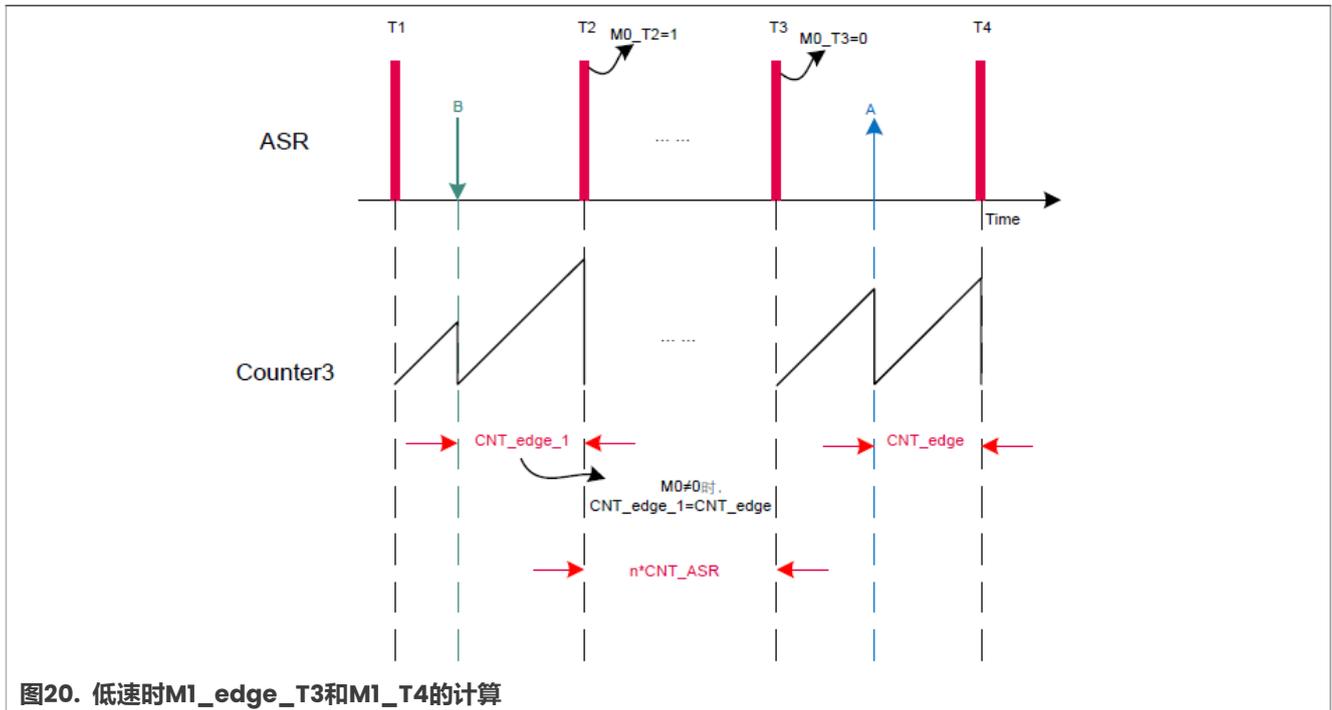


图18. T3时刻时M1_edge_T3的计算

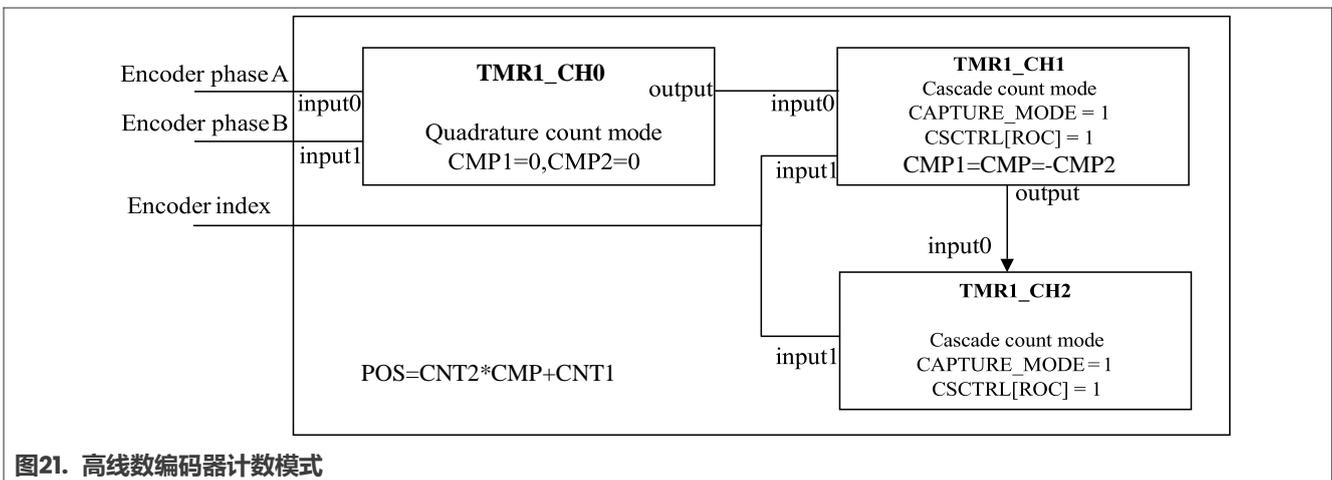


如图18所示，T2和T3时刻之间并没有编码器脉冲边沿出现。因此，在计算T3时刻的速度时，需要使用从上一个编码器脉冲边沿到T3时刻的高频时钟计数M1_edge_T3。此时，M1_edge_T3由两部分构成：一部分是B下降沿脉冲边沿到T2时刻的高频时钟计数CNT_edge，另一部分则是从T2到T3的ASR周期内的高频时钟计数CNT_ASR。因此， $M1_edge_T3 = CNT_edge + CNT_ASR$ 。如图19所示，T3至T4时刻之间出现了A相脉冲边沿。在计算T4时刻的速度时，需要使用M1_T4，此时它由四部分组成：

- 第一部分是从T1至T2时刻之间B下降沿脉冲边沿到T2时刻的高频时钟计数CNT_edge_1。
- 第二部分是从T2至T3的ASR周期内的高频时钟计数CNT_ASR。
- 第三部分是从T3至T4的ASR周期内的高频时钟计数CNT_ASR。
- 第四部分是从T3至T4时刻之间B上升沿脉冲边沿到T4时刻的高频时钟计数CNT_edge。因此， $M1_T4 = CNT_edge_1 + 2 \times CNT_ASR - CNT_edge$ 。



如图20所示，在低速运行时，两个脉冲边沿之间出现了更多M0值为0的速度测量时刻。图18中仅展示了一个M0值为0的速度测量时刻。如果存在n个M0值为0的速度测量时刻： $MI_edge_T3 = CNT_edge + n \times CNT_ASR$ ； $MI_T4 = CNT_edge_1 + (n+1) \times CNT_ASR - CNT_edge$ 。



使用QDC外设来获取MI和MI_edge时，可以直接通过POSDPERBFR/LSATEDGEH寄存器进行捕获。为了获取MI和MI_edge，QTimer外设需要结合CNT_ASR和CNT_edge。但这种方法只需捕获CNT_edge，由于CNT_ASR是固定值，这种计算方法几乎不会出现计数器溢出。QTimer在编码器速度测量方面的优点是，它能够避免高频时钟计数溢出的问题。然而，QTimer外设也有一些明显的缺点。例如，QTimer没有32位的计数器，无法直接获取超高线数编码器的位置信息。解决这一问题的方法是，级联两个计数器，实现相当于32位计数器的功能，从而完成对高线数编码器的正交解码。QTimer模块仅配备了4个计数器，这限制了其资源的使用范围，因此，如果只使用一个QTimer模块来同时进行角度采集和增强型M/T方法速度计算，是不可能的。

5.3 方法3：使用ENC+QTimer外设测量编码器信号的速度

本节介绍如何使用ENC+QTimer外设来测量编码器信号的速度。

5.3.1 用于编码器速度测量的ENC外设介绍

ENC是一个专门用于解码编码器的外设模块，而QDC模块则可以被视作它的升级版。

图22展示了ENC外设的内部框图。如图22所示，ENC外设的内部结构与QDC外设颇有几分相似，由一个32位的编码器位置计数器POS、一个16位的圈数计数器REV和一个16位的位置差计数器POSD组成。但它并不包含测量速度信息的计数器，例如POSDPER和LASTEDGE。因此，仅使用ENC外设无法独立完成增强型M/T速度计算。它需要与QTimer外设配合，使用高频时钟来计算行进的距离。

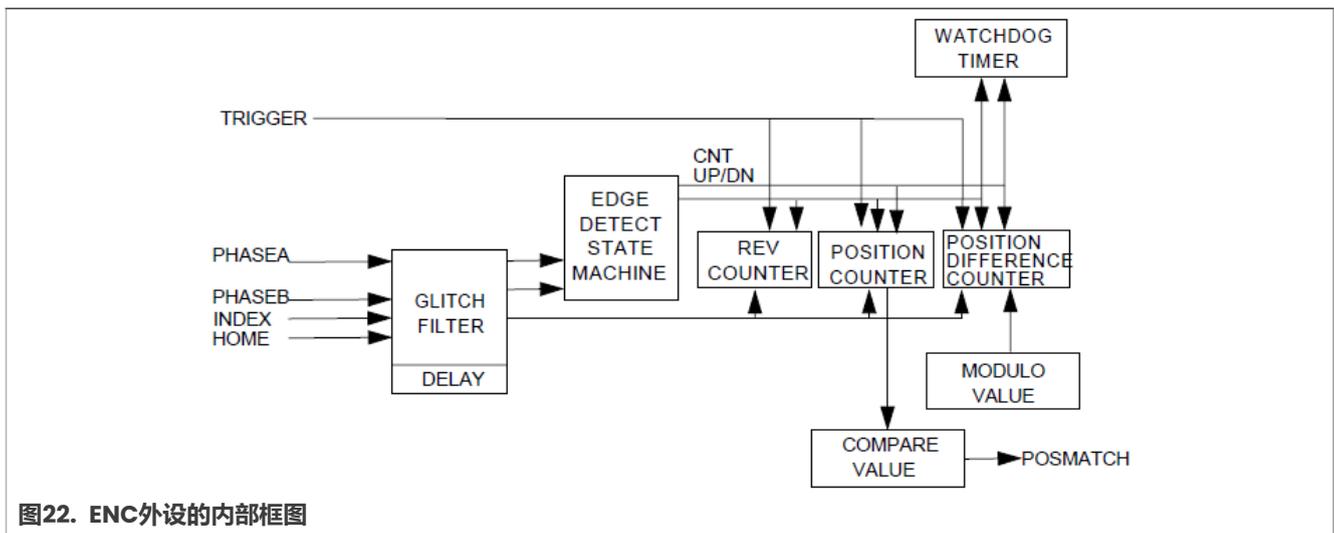


图22. ENC外设的内部框图

在ENC中，只要读取POS、REV和POSD中的任意三个计数器，POSD寄存器的值就会被载入到POSDH中，并随即清零。这一点与QDC的POSD寄存器不同。ENC模块没有CTRLn[PMEN]寄存器，所以仅读取POSD计数器时，无法将其值载入POSDH并清零。如果当前循环的频率超过了速度测量的频率，在每个当前循环中都必须读取POS寄存器。这样一来，每个当前循环中的POSD值都会被清零。因此，在下次速度测量时读取的POSDH值就无法反映两次速度测量时刻之间的脉冲边沿计数。有两种处理方法：

- 第一种是直接使用POSDH寄存器，通过软件计算两次速度测量时刻之间的差值，从而得出该时间段内的脉冲计数'M0'。
- 另一种是在上一次速度测量时刻之后的每个当前循环中读取POSDH的值，并将其累加至下一次速度测量时刻。这样累加起来的POSDH值，就是两次速度测量时刻之间的脉冲边沿计数'M0'。

5.3.2 使用方法3获取速度测量信息M0/M1/M1_edge

进行速度测量时，需要用QTimer外设来捕获准确的时间点。这个过程与前一章节中提到的QTimer外设的工作原理是一致的。CNT_edge是通过捕获高频时钟计数器得到的。接着，将这个计数器与速度测量周期相结合，计算出两次速度测量时刻之前的编码器脉冲边沿之间的高频时钟计数M1，以及速度测量时刻之间的高频时钟计数M1_edge。图23展示了这一过程的图解。

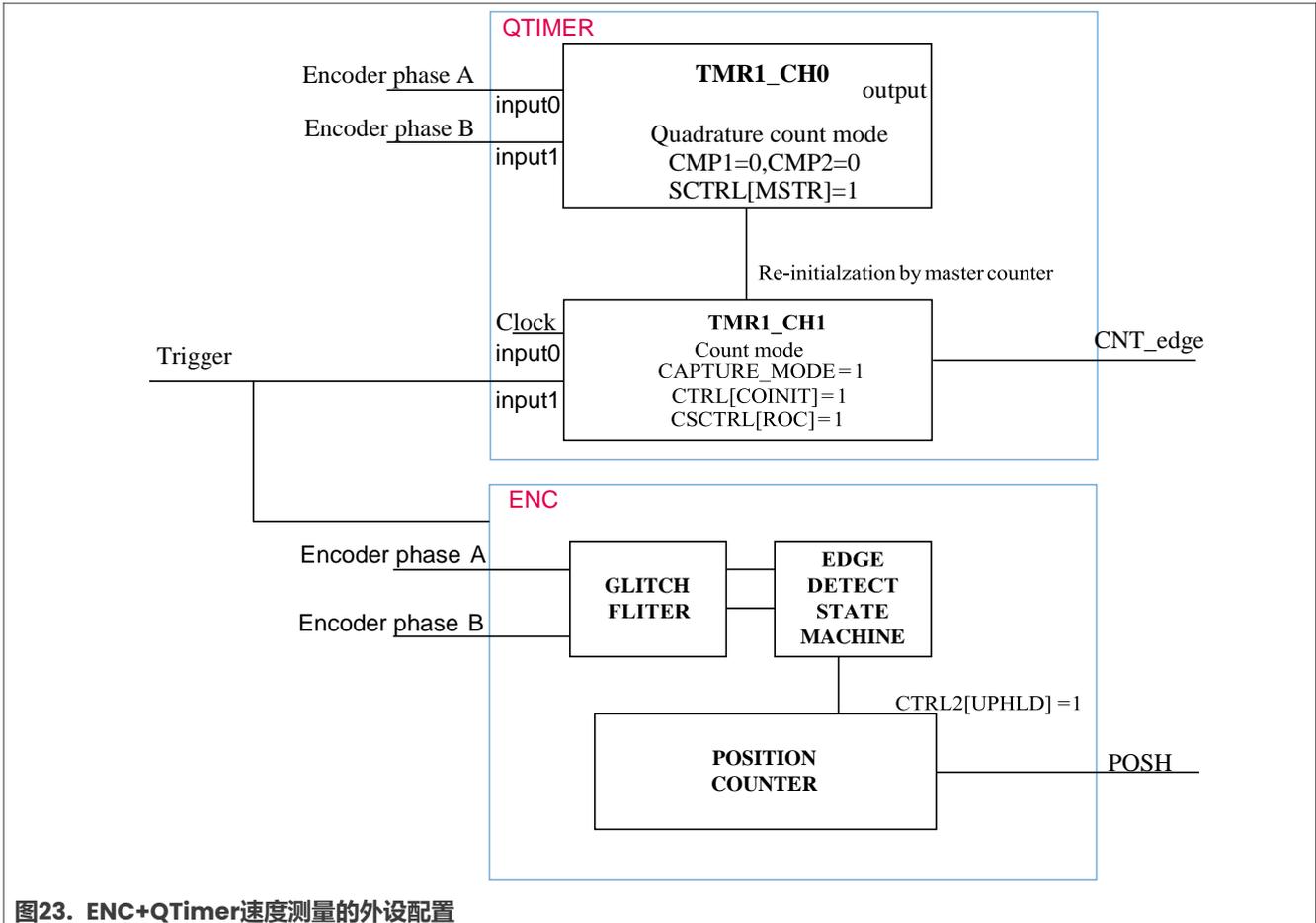


图23. ENC+QTimer速度测量的外设配置

QTimer外设的配置如下：

- **计数器0**：选择正交解码模式作为计数器模式 (TMRx_CTRL0[CM=4])，将比较值 (TMRx_COMP10和 TMR0_COMP20) 设为0。当检测到A/B相的上升沿时，它会触发比较事件，作为计数器1的级联模式。将 TMRx_SCTRL0[MSTR=1]配置为主模式。
- **计数器1**：选择主计数源上升沿计数模式 (TMRx_CTRL3[CM=1])。配置输入捕获模式来捕获次级计数源的上升沿 (TMRx_SCTRL3 [CAPTURE_MODE]=1)。触发捕获后，重新加载计数器 (TMRx_CSCTRL3 [ROC]=1)。当计数器0触发比较事件时，TMRx_CTRL3[COINIT=1]强制初始化计数器。

ENC外设的配置如下： ENCx_LMOD = 4 × C-1，其中计数器的比较值设置为4*C-1。ENCx_CTRL2[MOD]=1用于启用比较功能，当计数器达到比较值时进行初始化。LINIT寄存器用于设定初始值，在这里ENCx_LINIT寄存器被设置为0，而ENCx_CTRL2[UPDHLDD]=1。这样，触发信号就能够使POS和POSD寄存器的值被保存在它们的保持寄存器中。

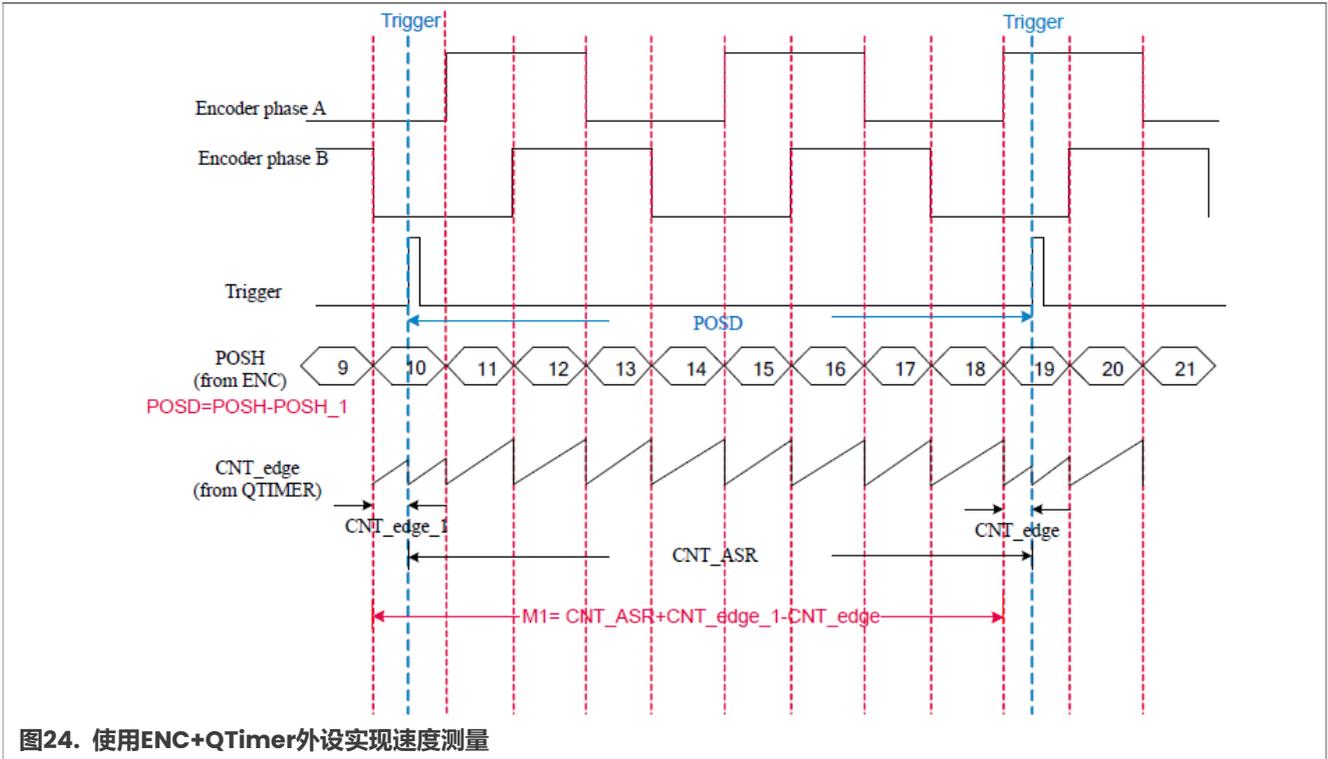
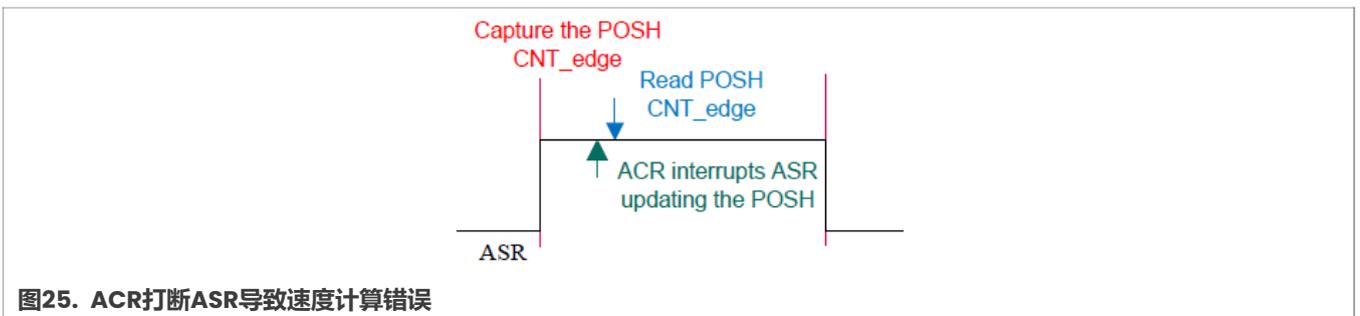


图24所示的逻辑图阐明了使用ENC+QTimer进行速度测量时，不同计数器是如何操作的。这个方法与使用QTimer测量速度相似，但M0的提取是使用ENC模块完成的，而非QTimer模块。这种方法的优势是可以直接使用高速编码器，而不必借助QTimer计数器来构建一个额外的32位计数器。ENC模块不包含CTRLn[PMEN]寄存器，而且计算速度的频率往往低于电流环的频率。因此，通过POSH的变化来获取POSD值，如图24所示。在此，POSH_1代表图中第一次触发时的POSH值，而POSH则是第二次触发时的POSH值。

在本例中，用于锁存和捕获ENC与QTimer外设的触发器并非ASR触发器，而是频率相同的触发器。这是因为速度环的中断优先级低于电流环的中断优先级。如果使用速度环触发器，ENC和QTimer模块能够精确地测量POSH值和CNT_edge值。在提取两个速度测量变量之前，它们被电流环中断。这导致POSH值被电流环中的读取POS指令更新，从而导致POSH值出现错误，速度测量误差也随之增大。这一逻辑关系在图25中展示。可以使用另一个触发器来产生中断，以便提取速度测量数据。它还将这个中断的优先级设置为最高，以避免电流环的读取指令更新POSH寄存器时带来的问题。



5.4 应用示例

这个应用示例使用了一个带有1000线正交编码器的永磁同步电机 (Teknic-2310p) 作为测试对象, 并选择FRDM-LVP-PMSM作为驱动板。

- 主控板采用DSC56F83000-EVK (主控芯片为56F83789), 并使用Quad Timer模块进行速度测量。
- 主控板采用DSC56F81000-EVK (主控芯片为56F81868), 并使用QDC模块进行速度测量。
- 主控板采用DSC56F84000-EVK (主控芯片为56F84789), 并使用ENC+ Quad Timer模块进行速度测量。

QTimer外设的配置如下, 与图16一致。TMRx_input0设置为编码器脉冲PHASEA, TMRx_input1设置为编码器脉冲PHASEB。TMRx_input2设置为编码器脉冲INDEX。TMRx_input3设置为ASR的触发信号。

示例1

```

/* TMRx_channel [0] */
/* TMRx_CTRL0: CM=0b100, PCS=0, SCS=0b01, ONCE=0b0, LENGTH=1, DIR=0, COINIT=0,
  OUTMODE=0 */
TMRx->CTRL0 = 0x80a0;
/* TMRx_SCTRL0: TCF=0, TCFIE=0, TOF=0, TOFIE=0, IEF=0, IEFIE=0, IPS=0, INPUT=0,
  CAPTURE_MODE=0, MSTR=1, EEOF=0, VAL=0, FORCE=0, OPS=0, OEN=0 */TMRx->SCTRL0
=0x0020;
/* TMRx_channel [1] */
/* TMRx_COMP11 = 4*C-1 */
TMRx->COMP11 = 0xf9f;
/* TMRx_COMP21 = -4*C+1 */
TMRx->COMP21 = 0xf061;
/* TMRx_CTRL1: CM=0b111, PCS=0b0100, SCS=0b10, ONCE=0b0, LENGTH=1, DIR=0,
  COINIT=0, OUTMODE=0 */
TMRx->CTRL1 = 0xe920;
/* TMRx_SCTRL1: TCF=0, TCFIE=0, TOF=0, TOFIE=0, IEF=0, IEFIE=0, IPS=0, INPUT=0,
  CAPTURE_MODE=1, MSTR=0, EEOF=0, VAL=0, FORCE=0, OPS=0, OEN=0 */TMRx->SCTRL1
=0x0040;
/* TMRx_channel [2] */
/* TMRx_COMP12 = 4*C-1 */
TMRx->COMP12 = 0xf9f;
/* TMRx_COMP22 = -4*C+1 */
TMRx->COMP22 = 0xf061;
/* TMRx_CTRL2: CM=0b111, PCS=0b0100, SCS=0b11, ONCE=0b0, LENGTH=1, DIR=0,
  COINIT=0, OUTMODE=0 */
TMRx->CTRL2 = 0xe9a0;
/* TMRx_SCTRL2: TCF=0, TCFIE=0, TOF=0, TOFIE=0, IEF=0, IEFIE=0, IPS=0, INPUT=0,
  CAPTURE_MODE=1, MSTR=0, EEOF=0, VAL=0, FORCE=0, OPS=0, OEN=0 */
TMRx->SCTRL2 = 0x0040;
/* TMRx_channel [3]*/
/* TMRx_CSCTRL3: DBG_EN=0, FAULT=0, ALT_LOAD=0, ROC=1, TCI=0, UP=0, OFLAG=0,
  TCF2EN=0, TCF1EN=0, TCF2=1, TCF1=0, CL2=0, CL1=0*/
TMRx->CSCTRL3 = 0x0800;
/* TMRx_CTRL3: CM=0b001, PCS=0b1101, SCS=0b11, ONCE=0b0, LENGTH=1, DIR=0,
  COINIT=0, OUTMODE=0 */
TMRx->CTRL3 = 0x3b88;
/* TMRx_SCTRL3: TCF=0, TCFIE=0, TOF=0, TOFIE=0, IEF=0, IEFIE=0, IPS=0, INPUT=0,
  CAPTURE_MODE=1, MSTR=0, EEOF=0, VAL=0, FORCE=0, OPS=0, OEN=0 */
TMRx->SCTRL3 = 0x0040;

```

QDC外设配置:

示例 2

```
/* QDC_CTRL: HIRQ=0, HIE=0, HIP=0, HNE=0, SWIP=0, REV=0, PH1=0, XIRQ=0, XIE=0,
  XIP=1, XNE=0, DIRQ=0, DIE=0, WDE=0, CMPIRQ=0, CMPIE=0 */
QDC->CTRL = 0X0040;
/* QDC_CTRL3: CMP1IRQ=0, CMP1IE=0, PRSC=0b0100, PMEN=1 */
QDC->CTRL3 = 0X0041;
```

ENC+QTimer外设配置:

TMRx_input0负责接收编码器的PHASEA脉冲信号, TMRx_input1负责接收编码器的PHASEB脉冲信号, TMRx_input2则用于接收触发信号。

示例 3

```
/* ENC_LMOD: 4*C-1 */
ENC1->LMOD = (1000 * 4U) - 1U;
/* ENC_CTRL2: OUTCTL=0, REVMOD=0, ROIRQ=0, RUIRQ=0, RUIE=0, DIR=0, MOD=1,
  UPDPOS=0, UPDHL=1 */
ENC1->CTRL2 = 0x0003;
/* TMRx_channel [0] */
/* TMRx_CTRL0: CM=0b001, PCS=0b1100, SCS=0b10, ONCE=0b0, LENGTH=1, DIR=0,
  COINIT=0, OUTMODE=0x101 */
TMRx->CTRL0 = 0x3725;
/* TMRx_CSCTRL0: DBG_EN=0, FAULT=0, ALT_LOAD=0, ROC=0, TCI=0, UP=0, OFLAG=0,
  TCF2EN=0, TCF1EN=1, TCF2=0, TCF1=0, CL2=0, CL1=01 */
TMRx->CSCTRL0 = 0x41;
TMRx->COMP10 = 0x249f;
TMRx->CPLD10 = 0x249f;
/* TMRx_channel [1] */
/* TMRx_CTRL1: CM=0b100, PCS=0b00, SCS=0b01, ONCE=0b0, LENGTH=1, DIR=0,
  COINIT=0, OUTMODE=0x0 */
TMRx->CTRL1 = 0x80a0;
/* TMRx_SCTRL1: TCF=0, TCFIE=0, TOF=0, TOFIE=0, IEF=0, IEFIE=0, IPS=0, INPUT=0,
  CAPTURE_MODE=0, MSTR=1, EEOF=0, VAL=0, FORCE=0, OPS=0, OEN=0 */
TMRx->SCTRL1 = 0x0020;
/* TMRx_channel [2] */
/* TMRx_CTRL2: CM=0b01, PCS=0b1101, SCS=0b11, ONCE=0b0, LENGTH=0, DIR=0,
  COINIT=1, OUTMODE=0x0 */
TMRx->CTRL2 = 0x3b88;
/* TMRx_CSCTRL2: DBG_EN=0, FAULT=0, ALT_LOAD=0, ROC=1, TCI=0, UP=0, OFLAG=0,
  TCF2EN=0, TCF1EN=0, TCF2=1, TCF1=0, CL2=0, CL1=0 */
TMRx->CSCTRL2 = 0x0800;
/* TMRx_SCTRL2: TCF=0, TCFIE=0, TOF=0, TOFIE=0, IEF=0, IEFIE=0, IPS=0, INPUT=0,
  CAPTURE_MODE=1, MSTR=0, EEOF=0, VAL=0, FORCE=0, OPS=0, OEN=0 */
TMRx->SCTRL2 = 0x0040;
/* TMRx_DMA2: CPLD2DE=0, CPLD1DE=0, IEFDE=1 */
TMRx->DMA2 = 0x0001;
```

6 结论

在电机控制系统中, 测量角度和速度至关重要。本应用笔记详细介绍了一种针对编码器的增强型M/T速度测量方法。首先, 文档阐述了正常匀速运行时的计算方法, 并分析了启动和停止、低速运行以及正反转操作过程中遇到的问题。接着, 它展示了改进后M/T速度测量方法的算法流程图。文档还介绍了如何从三个外设模块QDC、ENC和QTime获取速度测量值M0、M1和M1_edge。最后, 文档按不同外设的速度测量需求提供了寄存器配置示例。

7 参考资料

- MC56F81xxx参考手册 (文档[MC56F81XXXLRM](#))
- MC56F83xxx参考手册 (文档[MC56F83XXXRM](#))
- MC56F847xx参考手册 (文档[MC56F847XXRM](#))
- 支持 MCXN54x 和 MCXN94x 的 MCX Nx4x 参考手册 (文档[MCXNX4XRM](#))
- i.MX RT1170处理器参考手册 (文档[IMXRT1170RM](#))

8 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：

2024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）：

1. 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。
2. 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。
3. 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。

9 修订历史

[表2](#)汇总了本文档所做的更改

表2. 修订历史

文档ID	发布日期	描述
AN14254 v.1.0	2024年4月2日	首次公开发布

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.

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) 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.

i.MX — is a trademark of NXP B.V.

目录

1	介绍	2
2	正交编码器概述	2
2.1	正交编码器的基本概念	2
2.2	测量编码器速度的方法	2
3	角度测量	3
4	增强型M/T方法介绍	4
4.1	正常操作下速度测量的方法	4
4.2	在启动/停止及低速状态下的速度测量	5
4.3	速度测量算法流程图	9
5	使用解码外设进行速度测量的三种方法	10
5.1	方法1: 使用QDC外设测量编码器信号的速度	11
5.1.1	与编码器速度测量相关的QDC外设介绍	11
5.1.2	使用方法1提取速度测量信息M0/M1/M1_edge	12
5.2	方法2: 使用QTimer外设测量编码器信号的速度 ..	14
5.2.1	用于编码器速度测量的QTimer外设介绍	14
5.2.2	使用方法2提取速度测量信息M0/M1/M1_edge	15
5.3	方法3: 使用ENC+QTimer外设测量编码器信号的 速度	21
5.3.1	用于编码器速度测量的ENC外设介绍	21
5.3.2	使用方法3获取速度测量信息M0/M1/M1_edge	21
5.4	应用示例	24
6	结论	25
7	参考资料	26
8	关于本文中源代码的说明	26
9	修订历史	26
	法律声明	27

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

© 2024 NXP B.V.

All rights reserved.

For more information, please visit: <https://www.nxp.com.cn>

Date of release: 2 April 2024
Document identifier: AN14254