

CT Übungsaufgaben

Timer/Counter, PWM

Aufgabe 1

- a) Erklären Sie in Stichworten die einzelnen Funktionseinheiten eines Timers anhand der Tabelle.

Register	Inhalt	Funktion(en)
Prescaler	Divisor für Eingangssignal	zählt vor jedem neuen Clock
Counter	Zähler von Ticks	Inkrementiert/dekrementiert bei jedem Tick
Reload	Wert für Time-out	Upcounter: zählt bis zu diesem Wert danach Overflow Downcounter: Startwert für Timer
Capture/Compare	Event trigger	Capture: Bei Event wird Wert des Counters hier gespeichert Compare: Sobald Wert von Counter erreicht wird, wird ein Event ausgelöst

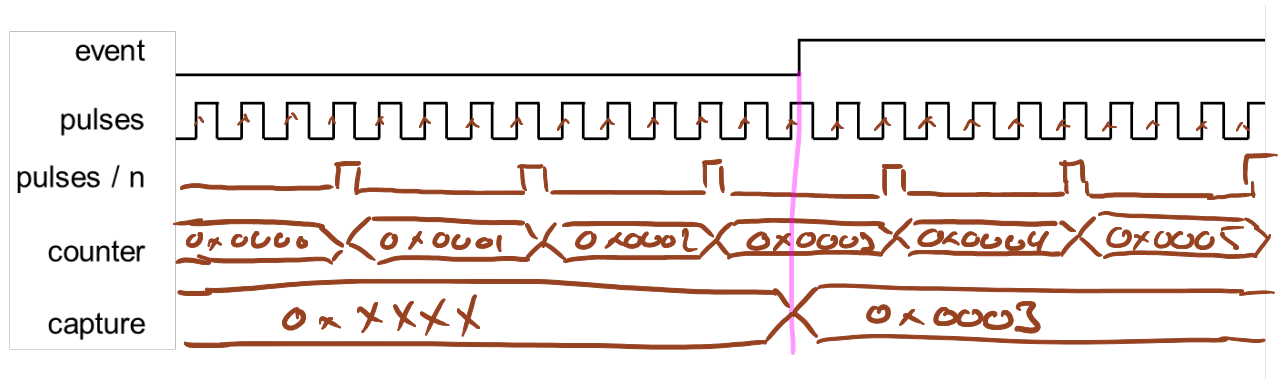
- b) Erläutern Sie die Funktion Capture.

Wenn ein Event ausgelöst wird, dann wird der aktuelle Wert des Counters in das Capture gespeichert

- c) Erläutern Sie die Funktion Compare.

Wenn der Counter den Wert erreicht, welcher im Compare steht, dann wird ein Event ausgelöst

- d) Ergänzen Sie das gegebene Timing Diagram. Die Funktion Capture wird bei steigender Flanke des Signals „event“ ausgelöst. Der Timer ist als Upcounter konfiguriert, und der Prescaler ist auf 4 (im Register steht 0x03) eingestellt. Die Startwerte entsprechen sonst dem Zustand nach einem Reset.



Aufgabe 2

Es soll Timer 3 des STM32F429 konfiguriert werden. Verwenden Sie das Reference Manual (zu finden in OLAT) zur Lösung der Aufgabe. Geben Sie die entsprechende Codezeilen in C an.

- a) Als Source soll die interne Clock CK_INT mit 84 MHz verwendet werden. Setzen Sie die Bits im entsprechenden Register auf die notwendigen Werte. Hinweis: andere Bits des Registers sollen nicht verändert werden.

TIM3_SMCR &= 0xFFFF8;

- b) Der Timer soll als Upcounter konfiguriert werden. Hinweis: andere Bits des Registers sollen nicht verändert werden.

TIM3_CR1 &= 0xFF8F

- c) Die Zeit für den Timerüberlauf soll 200 ms betragen. Welche Werte müssen Sie in die Register PSC und ARR schreiben (Angabe hexadezimaler Werte)? Hinweis: Es sind verschiedene richtige Lösungen möglich.

TIM3_PSC = 0x112F
TIM3_ARR = 0x07CF

Den Prescaler (PSC) setzen wir auf 8400, da wir mit 10kHz rechnen wollen. wir haben 84MHz gegeben, durch 8400 erhalten 10kHz. Nun wissen wir, pro Sekunde 10'000 Signale erhalten. Nun müssen wir denn automated Reload Register (ARR) auf 200ms setzen, dies entspricht bei unserer Rechnung 2000.

Aufgabe 3

Timer 4 des STM32F429 ist bereits als Upcounter konfiguriert und läuft. Das Reload Register enthält folgenden Wert:

$$\text{TIM4_ARR} = 0x9C3F \rightarrow \approx (40000 - 1)$$

- a) Geben Sie den Wert für das CCR-Register an, damit ein Duty Cycle von 25% mittels PWM Mode 1 erzeugt wird.

$$\text{TIM4_CCR} = 0x710$$

ARR = 40000 demnach periodisch durch ab 15x
in 40000 - 10000

- b) Nun ist Timer 4 als Downcounter im PWM Mode 2 statt als Upcounter im PWM Mode 1 konfiguriert. Was müssen Sie ändern, um ein identisches elektrisches Signal zu erhalten (Werte)?

$$\text{TIM4_CCR} = 0x752F$$

Aufgabe 4

Der Timer ist als Upcounter konfiguriert. Bestimmen Sie das generierte PWM-Signal am Ausgang (Zahlen + Skizze). Die Source liefert ein Signal der Frequenz 0,5 MHz.

Die relevanten Konfigurationsregister sind wie folgt initialisiert:

TIM3_PSC	=	0x01F3	500
TIM3_ARR	=	0x752F	30'000-1
TIM3_CCR1	=	0x2710	10'000
TIM3_CCMR1	=	0x0070	112 -> Sagt aus dass es in PWM2 konfiguriert ist

CCMR = Output compare mode

Welches Signal wird erzeugt? Zeichnen Sie das Signal und geben Sie die Werte für Periode und Duty Cycle an (Zeitangaben).

