

Abgabe 3: Timer, NVIC

Abzugeben sind bis Sonntag nach der mündlichen Abnahme, um 23:59 Uhr:

- eine in der entsprechenden Moodle-Abgabe hinterlegte PDF-Datei mit dem Namen `abgabe3.pdf`, in der die unten *kursiv* angegebenen Fragen beantwortet werden. Sie haben hier die Freiheit, ein beliebiges Tool zur Erstellung der PDF zu benutzen, wobei wir Ihnen natürlich LaTeX empfehlen würden.
- der in Ihrem Git Repository mit dem Tag `abgabe_3` eingecheckte Code in den CubeIDE-Projekten **06_Blinky_Dot**, **07_Dimming_Dot**, und **08_Stopwatch** zu den praktischen Aufgaben (nicht *kursiv*)

Timer

1. Lesen Sie sich anhand der zur Verfügung stehenden Literatur in das Thema "Cortex-M4 Timer" ein und versuchen Sie die dahinter stehenden Konzepte zu verstehen.
2. Initialisieren Sie das Zählwerk (TimeBase) von Timer 1. Stellen sie dabei den Prescaler so ein, dass der Timer mit einem Takt von 10 kHz zählt und nach einer Periodendauer von 1 Sekunde von vorne beginnt. Printen Sie den aktuellen Counter-Wert des Timers (verwenden Sie für den zugriff das HAL-Makro `__HAL_TIM_GET_COUNTER(__HANDLE__)` innerhalb der `while(1)`-Schleife auf den LCD und schätzen Sie, ob der Überlauf tatsächlich nach einer Sekunde stattfindet. Wechseln Sie nun von Timer 1 auf Timer 2, behalten den Rest der Konfiguration allerdings gleich. *Fällt Ihnen bei dem Zeitverhalten bis zum Überlauf etwas auf? Woran liegt das?*
3. Lassen Sie das "Dot"-Segment des 8-Segment-Boards (PE-11) im Sekundentakt blinken (Hälfte des Blinkzyklus lang an, die andere Hälfte lang aus) ohne dabei die CPU zu belasten. Verwenden Sie dazu eine Output-Compare-Einheit ihres Timers (PWM-Erzeugung). Das Modul *esd* dürfen Sie in dieser Aufgabe nicht verwenden.



Erstellen Sie auch hier ein neues Modul (z.B. *dot*)

4. Erweitern Sie ihre Anwendung so, dass über das Potentiometer des Evaluationsboards die Blinkfrequenz stufenlos zwischen 1Hz und 10Hz eingestellt werden kann. Das Tastverhältnis bleibt weiterhin bei 50%, nur die Anzahl der An-Aus-Zyklen pro Sekunde wird angepasst! Sie können natürlich ihr Potentiometer-Modul aus dem ADC-Kapitel wiederverwenden.



Zumindest für das Auge sieht das Einstellen stufenlos aus, durch die Verwendung einer bestimmten Auflösung (hier wäre mHz beispielsweise eine sinnvollere Einheit) ist echt stufenlos nämlich gar nicht möglich.

5. Erweitern sie ihr Modul so, dass das "Dot"-Segment alternativ prozentual über den Potentiometer in seiner Helligkeit eingestellt (gedimmt) werden kann.



Das menschliche Auge ist träge und kann nur maximal 25 Bilder pro Sekunde zeitlich auflösen. Bei Lichtblitzen ausreichend hoher Frequenz empfindet das Auge die mittlere Helligkeit der Lichtquelle als Seheindruck!

6. Beantworten Sie folgende Fragen inklusive Rechenweg:

- Welchen Prescaler müssen sie bei einem Timer einstellen, um bei einem Systemtakt von 50 MHz auf einen Zähltakt von 1 kHz zu kommen?
 - Welche Periodendauer müssen sie einstellen, wenn der Zähler des Timers alle 2 Minuten (wieder) den Wert 0 erreichen soll?
 - Sind diese errechneten Einstellungen in der Praxis möglich und erlaubt? Falls nein, warum nicht? Bietet die Timer-Konfigurationsstruktur eine Einstellung, um das gewünschte Ziel dennoch zu erreichen?
 - Zusatzaufgabe: Markieren Sie sich in Abbildung 16 des Reference Manuals den eingestellten Taktbaum von Ihrem Projekt und notieren Sie die dazugehörigen Frequenzen.
7. Passen Sie die für die erste Abgabe erstellte Funktion `utils_delay_ms(uint32_t t)` so an, dass Sie die gewünschte Zeit mithilfe eines Timers abwarten.
8. Zusatzaufgabe: Implementieren sie einen Treppenhautautomaten der ihre LED auf Tastendruck (oder Druck auf den Reset-Knopf) für 59 Sekunden aktiviert und anschließend wieder ausschaltet. Könnte hier der One-Pulse-Mode hilfreich sein?

NVIC

1. Lesen Sie das Kapitel zum NVIC und EXTI im Reference Manual.
2. Erklären Sie, welche Auswirkung ein Define von `USE_HAL_TIM_REGISTER_CALLBACKS` auf 1 hat.
3. Beschreiben Sie auf welche Signalwechsel der EXTI eingestellt werden kann. Wie wird die Einstellung vorgenommen und was bedeuten diese Einstellungen?
4. Beschreiben Sie, wie man das EXTI Modul (`stm32f4xx_hal_exti`) nutzen würde, um den EXTI separat von der GPIO Konfiguration zu konfigurieren.
5. Was ist beim EXTI der Unterschied zwischen einem Interrupt und einem Event?
6. Wieso hat der EXTI 23 Flankendetektoren, wenn es nur maximal 16 Pins pro GPIO Peripherie gibt? Welche Funktionen erfüllen die restlichen 7 konkret?
7. Sie bekommen einen Auftrag vom örtlichen Cart-Club. Für die heimische Rennstrecke soll ein Runden-Zeitmesser entwickelt werden. Dazu wurde auf der Ziellinie eine Laserlichtschranke installiert. Wenn die Lichtschranke zum ersten mal aktiviert wird, soll eine Uhr gestartet werden. Bei jeder weiteren Aktivierung soll unter der Gesamtzeit eine Rundenzeit gestoppt werden. Da die Rundenzeit der Kontrahenten oft nahe beieinander liegt, ist eine sehr genaue Zeitmessung mit einer Auflösung von 1/10.000 Sekunde gefordert. Entwickeln Sie ein entsprechendes Modul `stopwatch`. Nutzen Sie dazu Ihre Kenntnisse über Timer und Interrupts. Die Lichtschranke können Sie durch den User-Button des Discovery Boars simulieren. Orientieren Sie sich bei der Darstellung auf dem LCD an folgender Online-Stoppuhr im Web ^[1].

[1] <https://webuhr.de/stoppuhr/>